3D Vision: Stereo
Marc Pollefeys
Viktor Larsson
Spring 2020
http://cvg.ethz.ch/teaching/3dvision/
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 17</td>
<td>Introduction</td>
</tr>
<tr>
<td>Feb 24</td>
<td>Geometry, Camera Model, Calibration</td>
</tr>
<tr>
<td>Mar 2</td>
<td>Features, Tracking / Matching</td>
</tr>
<tr>
<td>Mar 9</td>
<td><strong>Project Proposals by Students</strong></td>
</tr>
<tr>
<td>Mar 16</td>
<td>Structure from Motion (SfM) + papers</td>
</tr>
<tr>
<td>Mar 23</td>
<td><strong>Dense Correspondence (stereo / optical flow) + papers</strong></td>
</tr>
<tr>
<td>Mar 30</td>
<td>Bundle Adjustment &amp; SLAM + papers</td>
</tr>
<tr>
<td>Apr 6</td>
<td><strong>Student Midterm Presentations</strong></td>
</tr>
<tr>
<td>Apr 13</td>
<td>Easter break</td>
</tr>
<tr>
<td>Apr 20</td>
<td>Multi-View Stereo &amp; Volumetric Modeling + papers</td>
</tr>
<tr>
<td>Apr 27</td>
<td>3D Modeling with Depth Sensors + papers</td>
</tr>
<tr>
<td>May 4</td>
<td>3D Scene Understanding + papers</td>
</tr>
<tr>
<td>May 11</td>
<td>4D Video &amp; Dynamic Scenes + papers</td>
</tr>
<tr>
<td>May 25</td>
<td><strong>Student Project Demo Day = Final Presentations</strong></td>
</tr>
</tbody>
</table>
Dense Correspondence & Stereo Matching
Dense Correspondence & Stereo Matching

Tsukuba dataset

http://cat.middlebury.edu/stereo/
Relationship Disparity - Depth

How to recover a 3D point from two corresponding image points?

- Equal triangles (only when image planes are parallel)
- Using the definition $d = x - x'$:

\[
\frac{Z - f}{B - (x - x')} = \frac{Z}{B}
\]

\[
ZB - fB = ZB - Z(x - x')
\]

\[
Z = \frac{fB}{x - x'} = \frac{fB}{d}
\]

\[
d = \frac{fB}{Z}
\]
Overview

Task

- Construct a 3D model from 2 images of a calibrated camera

Pipeline:

1. Find a set of corresponding points
2. Estimate the **epipolar geometry**
3. Rectify both images
4. Dense feature matching
5. 3D reconstruction
Disparity map

image $I(x,y)$

Disparity map $D(x,y)$

image $I'(x',y')$

$(x',y')=(x+D(x,y),y)$
Photoconsistency
Photoconsistency

- $w_L$ and $w_R$ are corresponding $m \times m$ windows of pixels
- We can write them as vectors: $w_L, w_R \in \mathbb{R}^{m^2}$
- Normalized correlation (cosine of the enclosed angle):

$$NC(x, y, d) = \frac{(w_L(x, y) - \bar{w}_L(x, y))^T (w_R(x - d, y) - \bar{w}_R(x - d, y))}{\|w_L(x, y) - \bar{w}_L(x, y)\|_2 \|w_R(x - d, y) - \bar{w}_R(x - d, y)\|_2}$$

Sum of squared differences (SSD):

$$SSD(x, y, d) = \|w_L(x, y) - w_R(x - d, y)\|_2^2$$
Photoconsistency

Block Matching:

- Choose some disparity range $[0, d_{max}]$
- For all pixels $x = (x, y)$ try all disparities and choose the one that maximizes the normalized correlation or minimizes the SSD
- This strategy is called: Winner-takes-all (WTA)
- Do this for both images, apply left-right consistency check

Challenges:

- Which window size to choose? Tradeoff: Ambiguity $\leftrightarrow$ Bleeding!
- Block matching $=$ fronto-parallel assumption (often invalid!)
Hierarchical stereo matching

Allows faster computation
Deals with large disparity ranges

Downsampling
(Gaussian pyramid)

Disparity propagation
Stereo camera configurations

Short baseline:
- Good matches
- Few occlusions
- Poor precision

Long baseline:
- Harder to match
- More occlusions
- Better precision
Occlusions

→ Consistency test
Uniqueness constraint

- In an image pair each pixel has at most one corresponding pixel
  - In general one corresponding pixel
  - In case of occlusion there is none
Disparity constraint

surface slice

bounding box

surface as a path

disparity band

constant disparity surfaces

use reconstructed features to determine bounding box
Ordering constraint

surface slice

surface as a path

occlusion left

occlusion right
Stereo matching

Consider all paths that satisfy the constraints
pick best using dynamic programming

Similarity measure (SSD or NCC)
Optimal path (dynamic programming)

Constraints
• epipolar
• ordering
• uniqueness
• disparity limit

Trade-off
• Matching cost (data)
• Discontinuities (prior)
True disparities

*2 – Dynamic progr.

16 – Fast Correlation

(Scharstein & Szeliski, IJCV‘02)
Energy minimization

Disparity continuous in most places, except at depth discontinuities.

1. Matching pixels should have similar intensities.
2. Most nearby pixels should have similar disparities.

\[ \text{Minimize} \quad \sum [I_1(x + D(x, y), y) - I_2(x, y)]^2 \]
\[ + \lambda \sum [D(x + 1, y) - D(x, y)]^2 \]
\[ + \mu \sum [D(x, y + 1) - D(x, y)]^2 \]
Graph Cut

1. Stereo is a labeling problem
2. Graph cut corresponds to a labeling.
   → Assign edge weights cleverly so that the min-weight cut gives the minimum energy!
Simplified graph cut

\[ V = V^* \cup \{s, t\} \]
\[ E = E^* \cup \{(s, v) : v \in \text{Front}\} \cup \{(u, t) : u \in \text{Back}\} \]

(Roy and Cox ICCV‘98)

(a) initial labeling  
(b) standard move  
(c) \(\alpha\)-\(\beta\)-swap  
(d) \(\alpha\)-expansion

(Boykov et al ICCV‘99)
(Scharstein & Szeliski, IJCV‘02)
Semi-global optimization

• Optimize:
  \[ E = E_{\text{data}} + E(|D_p - D_q| = 1) + E(|D_p - D_q| > 1) \]
  • Use mutual information as cost
  • [Hirschmüller CVPR05]

• NP-hard using graph cuts or belief propagation (2-D optimization)

• Instead do dynamic programming along many directions
  • Don’t use visibility or ordering constraints
  • Add costs of all paths
More Complex Priors

(Güney & Geiger, CVPR 2015)
Stereo matching with general camera configurations
Image pair rectification
Epipolar Geometry

- Let's assume the camera parameters and geometry is known!
- Given a projection of a 3D point in the left image
- Where is it located in 3D?
- On the epipolar line defined by this point and the camera centers
- Reduces the search problem to 1D!
Epipolar Geometry

- $\overline{CC'}$: Baseline (translation between cameras)
- $e, e'$: Epipole (intersection of image plane with baseline)
- $l, l'$: Epipolar line (intersection of image plane with epipolar plane)
Planar rectification

Bring two views to standard stereo setup
(moves epipole to origin)
(not possible when in/close to image)

$H'^{-\top}FH^{-1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$

~ image size (calibrated)

Distortion minimization (uncalibrated)
Planar rectification

Source: https://en.wikipedia.org/wiki/Image_rectification
Polar rectification
(Pollefeys et al. ICCV’99)

- Polar re-parameterization around epipoles
- Requires only (oriented) epipolar geometry
- Preserve length of epipolar lines
- Choose $\Delta \theta$ so that no pixels are compressed

Works for all relative motions
Guarantees minimal image size
original image pair

planar rectification

polar rectification
Example: Béguinage of Leuven

Does not work with standard Homography-based approaches
Plane-sweep multi-view matching

- Simple algorithm for multiple cameras
- No rectification necessary
- Doesn’t deal with occlusions

Collins’ 96; Roy and Cox’ 98 (GC)
PatchMatch Stereo

fronto-parallel windows vs. slanted support windows

(Bleyer et al. BMVC’11)
PatchMatch Stereo
(Bleyer et al. BMVC’11)

• For a particular plane the disparity at a pixel is given by

\[ d_p = a_f p_x + b_f p_y + c_f \]

• The plane with the minimal cost is chosen

\[ f_p = \arg\min_{f \in \mathcal{F}} m(p, f) \]

• The dissimilarity cost is calculated as

\[
m(p, f) = \sum_{q \in W_p} w(p, q) \cdot \rho(q, q - (a_f q_x + b_f q_y + c_f))
\]

with

\[
w(p, q) = e^{-\frac{\|I_p - I_q\|}{\gamma}}
\]

\[
\rho(q, q') = (1 - \alpha) \cdot \min(||I_q - I_{q'}||, \tau_{col}) + \alpha \cdot \min(||\nabla I_q - \nabla I_{q'}||, \tau_{grad})
\]
PatchMatch Stereo
(Bleyer et al. BMVC’11)

Idea: Start with a random initialization of disparities and plane parameters for each pixel and update the estimates by propagating information from the neighboring pixels

- **Spatial propagation**: Check for each pixel the disparities and plane parameters for the left and upper (right and lower) neighbors and replace the current estimates if matching costs are smaller.
- **View propagation**: Warp the point in the other view and check the corresponding estimates in the other image. Replace if the matching costs are lower.
- **Temporal propagation**: Propagate the information analogously by considering the estimates for the same pixel at the preceding and consecutive video frame.
PatchMatch Stereo
(Bleyer et al. BMVC’11)
PatchMatch Stereo

Left to right:
- Fronto-parallel, discrete disparities
- Fronto-parallel, continuous disparities
- PatchMatch Stereo (slanted, continuous disparities)

(Bleyer et al. BMVC’11)
Next week: Bundle Adjustment & SLAM

Now: Papers!