Stereo and Structure from Motion

CS143, Brown
James Hays

Many slides by Kristen Grauman, Robert Collins, Derek Hoiem, Alyosha Efros, and Svetlana Lazebnik
Why Stereo Vision?

Fundamental Ambiguity:
Any point on the ray OP has image p

\[ P = (X, Y, Z) \]

\[ p = (x, y, f) \]

\[ x = f \frac{X}{Z} = f \frac{kX}{kZ} \]

\[ y = f \frac{Y}{Z} = f \frac{kY}{kZ} \]
Why Stereo Vision?

A second camera can resolve the ambiguity, enabling measurement of depth via triangulation.
Depth from disparity

\[ \frac{X - X'}{f} = \frac{\text{baseline}}{z} \]

\[ X - X' = \frac{\text{baseline} \cdot f}{z} \]

\[ z = \frac{\text{baseline} \cdot f}{X - X'} \]
Outline

- Human stereopsis
- Stereograms
- Epipolar geometry and the epipolar constraint
  - Case example with parallel optical axes
  - General case with calibrated cameras
General case, with calibrated cameras

- The two cameras need not have parallel optical axes.
Stereo correspondence constraints

- Given p in left image, where can corresponding point p’ be?
Stereo correspondence constraints
Geometry of two views constrains where the corresponding pixel for some image point in the first view must occur in the second view.

- It must be on the line carved out by a plane connecting the world point and optical centers.
Epipolar geometry

http://www.ai.sri.com/~luong/research/Meta3DViewer/EpipolarGeo.html
Epipolar geometry: terms

- **Baseline**: line joining the camera centers
- **Epipole**: point of intersection of baseline with image plane
- **Epipolar plane**: plane containing baseline and world point
- **Epipolar line**: intersection of epipolar plane with the image plane

- All epipolar lines intersect at the epipole
- An epipolar plane intersects the left and right image planes in epipolar lines

*Why is the epipolar constraint useful?*
Epipolar constraint

This is useful because it reduces the correspondence problem to a 1D search along an epipolar line.

Image from Andrew Zisserman
Example
What do the epipolar lines look like?

1. 

2.
Example: converging cameras

Figure from Hartley & Zisserman
Example: parallel cameras

Where are the epipoles?

Figure from Hartley & Zisserman
Example: Forward motion

What would the epipolar lines look like if the camera moves directly forward?
Example: Forward motion

Epipole has same coordinates in both images.
Points move along lines radiating from e: “Focus of expansion”
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 \( F \)

\[
\begin{align*}
l' &= Fp \\
l &= p'F
\end{align*}
\]

It follows that

\[
p'Fp = 0
\]
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 $F$

$$p' F p = 0$$

- $F$ has 9 entries, but really only 7 or 8 degrees of freedom.
- With 8 points it is simple to solve for $F$, but it is also possible with 7. See Marc Pollefeys’ notes for a nice tutorial
Stereo image rectification
Stereo image rectification

- Reproject image planes onto a common plane parallel to the line between camera centers

- Pixel motion is horizontal after this transformation

- Two homographies (3x3 transform), one for each input image reprojection

Rectification example
The correspondence problem

• Epipolar geometry constrains our search, but we still have a difficult correspondence problem.
Basic stereo matching algorithm

- If necessary, rectify the two stereo images to transform epipolar lines into scanlines.
- For each pixel \( x \) in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match \( x' \)
  - Compute disparity \( x - x' \) and set \( \text{depth}(x) = \frac{f_B}{x - x'} \)
Correspondence search

- Slide a window along the right scanline and compare contents of that window with the reference window in the left image.
- Matching cost: SSD or normalized correlation.
Correspondence search

Left

Right

scanline

SSD
Correspondence search

Left

Right

scanline

Norm. corr
Effect of window size

- Smaller window
  - More detail
  - More noise

- Larger window
  - Smoother disparity maps
  - Less detail
Failures of correspondence search

Textureless surfaces

Occlusions, repetition

Non-Lambertian surfaces, specularities
Results with window search

Data

Window-based matching

Ground truth
How can we improve window-based matching?

• So far, matches are independent for each point

• What constraints or priors can we add?
Stereo constraints/priors

- **Uniqueness**
  - For any point in one image, there should be at most one matching point in the other image.
Stereo constraints/priors

- **Uniqueness**
  - For any point in one image, there should be at most one matching point in the other image

- **Ordering**
  - Corresponding points should be in the same order in both views
Stereo constraints/priors

• Uniqueness
  – For any point in one image, there should be at most one matching point in the other image

• Ordering
  – Corresponding points should be in the same order in both views

Ordering constraint doesn’t hold
Priors and constraints

- **Uniqueness**
  - For any point in one image, there should be at most one matching point in the other image

- **Ordering**
  - Corresponding points should be in the same order in both views

- **Smoothness**
  - We expect disparity values to change slowly (for the most part)
Scanline stereo

- Try to coherently match pixels on the entire scanline
- Different scanlines are still optimized independently
“Shortest paths” for scan-line stereo

Can be implemented with dynamic programming

Ohta & Kanade ’85, Cox et al. ‘96

Slide credit: Y. Boykov
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 (random field interpretation)

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
Many of these constraints can be encoded in an energy function and solved using graph cuts.

Before

Graph cuts

Ground truth


For the latest and greatest: http://www.middlebury.edu/stereo/
Active stereo with structured light

- Project “structured” light patterns onto the object
  - Simplifies the correspondence problem
  - Allows us to use only one camera

Kinect: Structured infrared light

Potential matches for $x$ have to lie on the corresponding line $l'$. 

Potential matches for $x'$ have to lie on the corresponding line $l$. 

Summary: Key idea: Epipolar constraint
Summary

• Epipolar geometry
  – Epipoles are intersection of baseline with image planes
  – Matching point in second image is on a line passing through its epipole
  – Fundamental matrix maps from a point in one image to a line (its epipolar line) in the other
  – Can solve for F given corresponding points (e.g., interest points)

• Stereo depth estimation
  – Estimate disparity by finding corresponding points along scanlines
  – Depth is inverse to disparity
Structure from motion

- Given a set of corresponding points in two or more images, compute the camera parameters and the 3D point coordinates.

\[
R_1, t_1, \quad R_2, t_2, \quad R_3, t_3
\]
Structure from motion ambiguity

• If we scale the entire scene by some factor $k$ and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same:

$$x = PX = \left(\frac{1}{k}P\right)(kX)$$

It is impossible to recover the absolute scale of the scene!
How do we know the scale of image content?
Structure from motion ambiguity

- If we scale the entire scene by some factor $k$ and, at the same time, scale the camera matrices by the factor of $1/k$, the projections of the scene points in the image remain exactly the same.

- More generally: if we transform the scene using a transformation $Q$ and apply the inverse transformation to the camera matrices, then the images do not change.

\[ x = PX = (PQ^{-1})(QX) \]
Projective structure from motion

- Given: $m$ images of $n$ fixed 3D points
  - $x_{ij} = P_i X_j$, $i = 1, \ldots, m$, $j = 1, \ldots, n$
- Problem: estimate $m$ projection matrices $P_i$ and $n$ 3D points $X_j$ from the $mn$ corresponding points $x_{ij}$
Projective structure from motion

• Given: \( m \) images of \( n \) fixed 3D points
  
  \[
  x_{ij} = P_i X_j, \quad i = 1, \ldots , m, \quad j = 1, \ldots , n
  \]

• Problem: estimate \( m \) projection matrices \( P_i \) and \( n \) 3D points \( X_j \) from the \( mn \) corresponding points \( x_{ij} \)

• With no calibration info, cameras and points can only be recovered up to a 4x4 projective transformation \( Q \):

  \[
  X \rightarrow QX, \quad P \rightarrow PQ^{-1}
  \]

• We can solve for structure and motion when

  \[
  2mn \geq 11m + 3n - 15
  \]

• For two cameras, at least 7 points are needed
Projective ambiguity

\[ x = PX = \left(PQ_P^{-1}\right)(Q_P X) \]

\[ Q_P = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix} \]
Projective ambiguity
Bundle adjustment

• Non-linear method for refining structure and motion
• Minimizing reprojection error

\[ E(P, X) = \sum_{i=1}^{m} \sum_{j=1}^{n} D(x_{ij}, P_i X_j)^2 \]
Photo synth


http://photosynth.net/