Outline

Projective geometry

Vanishing points
  Application: camera calibration
  Application: single-view metrology

Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Outline

Projective geometry
Vanishing points
  Application: camera calibration
  Application: single-view metrology
Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Review: Camera projection matrix

\[ \lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathbf{P} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \]

\[ \lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} \alpha_x & \beta_x \\ \alpha_y & \beta_y \\ 1 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \]

\text{intrinsic parameters} \quad \text{extrinsic parameters}
Review: Camera parameters

- Intrinsic parameters
  - Image center \((p_x, p_y)\)
  - Focal length \((f)\)
  - Pixel magnification \((m_x, m_y)\)
  - Skew (non-rectangular pixels)
  - Radial distortion

\[
P = K[R \quad t]
\]

\[
K = \begin{bmatrix}
m_x & m_y & f & \alpha_x \\
& & f & \beta_x \\
1 & & 1 & \alpha_y \\
& & & 1
\end{bmatrix}
\]
Review: Camera parameters

- Intrinsic parameters
  - Principal point coordinates
  - Focal length
  - Pixel magnification
  - Skew (non-rectangular pixels)
  - Radial distortion

- Extrinsic parameters
  - Rotation (R) and translation (t) relative to world coordinate system

\[ P = K [R \ t] \]
Review: Camera calibration

\[ \lambda x = K [R \ t] X \]

\[ \lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \]

Source: D. Hoiem
Review: Camera calibration

- Given $n$ points with known 3D coordinates $X_i$ and known image projections $x_i$, estimate the camera parameters
Camera calibration

• What if don’t know correspondences?
• Can you determine the camera intrinsic parameters (focal length, center) or extrinsic parameters (rotation, translation)?
Camera calibration

- Let’s see what we can get from vanishing points

Vertical vanishing point (at infinity)

Vanishing line

Vanishing point
Outline

Projective geometry

Vanishing points
  Application: camera calibration
  Application: single-view metrology

Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Review: Vanishing Points

- Any set of parallel lines on a plane define a vanishing point
- The union of all of these vanishing points is the *horizon line*
  - also called *vanishing line*
- Different planes (can) define different vanishing lines
Review: Vanishing points

- All lines having the same direction share the same vanishing point
Computing Vanishing Points

How can we find lines in an image?
Computing Vanishing Points

Edge detection

Edges
Computing Vanishing Points

Edge detection + Hough transform

Strong Lines

Hough Transform

Papusha & Ho
Computing Vanishing Points

For a set of parallel lines, how can we find where they intersect?
Computing Vanishing Points

Intersect $p_1q_1$ with $p_2q_2$

$$v = (p_1 \times q_1) \times (p_2 \times q_2)$$
Computing Vanishing Points

Intersect $p_1q_1$ with $p_2q_2$

$v = (p_1 \times q_1) \times (p_2 \times q_2)$
Computing Vanishing Points

Intersect $p_1q_1$ with $p_2q_2$

$v = (p_1 \times q_1) \times (p_2 \times q_2)$
Computing Vanishing Points

Intersect $p_1q_1$ with $p_2q_2$

$$v = (p_1 \times q_1) \times (p_2 \times q_2)$$

Least squares version

• Better to use more than two lines and compute the “closest” point of intersection
Computing Vanishing Points

Alternative: vanishing points can be extracted directly from Hough transform (fit sine curves)
Computing Vanishing Points

Vanishing points can be extracted directly from Hough transform (fit sine curves)

Strong Lines

Hough Transform

Papusha & Ho
Outline

Projective geometry
Vanishing points
  Application: camera calibration
  Application: single-view metrology
Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Calibration from vanishing points

• What camera parameters can we calibrate using three orthogonal vanishing directions (points)?
Calibration from vanishing points

• Let us align the world coordinate system with three orthogonal vanishing directions in the scene:

\[
e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\]

\[
\lambda v_i = K [R | t] \begin{bmatrix} e_i \\ 0 \end{bmatrix} = KRe_i
\]

\[
e_i = \lambda R^T K^{-1} v_i, \quad e_i^T e_j = 0
\]

\[
v_i^T K^{-T} RR^T K^{-1} v_j = v_i^T K^{-T} K^{-1} v_j = 0
\]

• Each pair of vanishing points gives us a constraint on the focal length and principal point
Intrinsic calibration from vanishing points

1 finite vanishing point, 2 infinite vanishing points

2 finite vanishing points, 1 infinite vanishing point

3 finite vanishing points

Cannot recover focal length, image center is the finite vanishing point

Can solve for focal length, image center
Rotation from vanishing points

\[ \lambda \mathbf{v}_i = K [R | t] \begin{bmatrix} e_i \\ 0 \end{bmatrix} = K \mathbf{R} \mathbf{e}_i \]

\[ \lambda \mathbf{K}^{-1} \mathbf{v}_1 = \mathbf{R} \mathbf{e}_1 = [\mathbf{r}_1 \ \ \mathbf{r}_2 \ \ \mathbf{r}_3] \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \mathbf{r}_1 \]

Thus, \[ \lambda \mathbf{K}^{-1} \mathbf{v}_i = \mathbf{r}_i. \]

Get \( \lambda \) by using the constraint \( ||\mathbf{r}_i||^2 = 1. \)
Calibration from vanishing points: Summary

• Solve for $K$ (focal length, principal point) using three orthogonal vanishing points
• Get rotation directly from vanishing points once $K$ is known

• Advantages
  • No need for calibration chart (2D-3D correspondences)
  • Could be completely automatic

• Disadvantages
  • Only applies to certain kinds of scenes
  • Inaccuracies in computation of vanishing points
  • Problems due to infinite vanishing points
Outline

Projective geometry

Vanishing points

  Application: camera calibration

  Application: single-view metrology

Epipolar geometry

  Application: stereo correspondence

  Application: structure from motion revisited
How Tall is the Man in this Image?
How Tall is the Man in this Image?

10 meters
How Tall is the Man in this Image?

10 meters
How Tall is the Man in this Image?

Goal: compute $Z$

Problem: depends on camera angle and distance
The cross-ratio

- A *projective invariant*: quantity that does not change under projective transformations (including perspective projection)
- The cross-ratio of four points:

\[
\frac{||P_3 - P_1|| \cdot ||P_4 - P_2||}{||P_3 - P_2|| \cdot ||P_4 - P_1||}
\]
Measuring height

\[
\frac{||T-B||}{||R-B||} \cdot \frac{||\infty - R||}{||\infty - T||} = \frac{H}{R}
\]

scene cross ratio

\[
\frac{||t-b||}{||r-b||} \cdot \frac{||v_z - r||}{||v_z - t||} = \frac{H}{R}
\]

image cross ratio

Svetlana Lazebnik
How Tall is the Man in this Image?

10 meters
vanishing line (horizon)

$$\frac{\|t - b\| \|v_Z - r\|}{\|r - b\| \|v_Z - t\|} = \frac{H}{R}$$

image cross ratio
2D lines in homogeneous coordinates

- Line equation: \( ax + by + c = 0 \)

\[
\mathbf{l}^T \mathbf{x} = 0 \quad \text{where} \quad \mathbf{l} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}
\]

- Line passing through two points: \( \mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2 \)

- Intersection of two lines: \( \mathbf{x} = \mathbf{l}_1 \times \mathbf{l}_2 \)
\[ v \approx (b \times b_0) \times (v_x \times v_y) \]

vanishing line (horizon)

\[ t \approx (v \times t_0) \times (r \times b) \]

\[
\frac{\|t - b\|\|v_z - r\|}{\|r - b\|\|v_z - t\|} = \frac{H}{R}
\]

image cross ratio
How Long is this Line Segment?

Can we measure distances between two points on same plane?
How Long is this Line Segment?

What if we know distances between some pairs of points on the same plane?
Measurements on planes

Approach: unwarp then measure
What kind of warp is this?
Measurements on planes

Approach: unwarp then measure Homography!
Piero della Francesca, *Flagellation*, ca. 1455
Application: Image editing

Inserting synthetic objects into images:
http://vimeo.com/28962540

K. Karsch and V. Hedau and D. Forsyth and D. Hoiem, “Rendering Synthetic Objects into Legacy Photographs,” SIGGRAPH Asia 2011
D. Hoiem, A.A. Efros, and M. Hebert, "Putting Objects in Perspective", CVPR 2006
Outline

Projective geometry

Vanishing points
  Application: camera calibration
  Application: single-view metrology

Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
• Given p in left image, where can corresponding point p’ be?
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

- Epipolar Plane
- Epipole
- Baseline
- Epipole

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

Kristen Grauman
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.
What do the epipolar lines look like?

1. 

![Diagram of epipolar lines with points O₁ and O_r]

2. 

![Diagram with points O₁ and O_r]
Epipolar lines: converging cameras

Figure from Hartley & Zisserman
Epipolar lines: parallel cameras

Where are the epipoles?
Epipolar lines

So far, we have the explanation in terms of geometry.

Now, how to express the epipolar constraints algebraically?
Stereo geometry

Main idea
If the stereo rig is calibrated, we know:

how to **rotate** and **translate** camera reference frame 1 to get to camera reference frame 2.

Rotation: 3 x 3 matrix $\mathbf{R}$; translation: 3 vector $\mathbf{T}$. 
If the stereo rig is calibrated, we know:

how to **rotate** and **translate** camera reference frame 1 to get to camera reference frame 2.

\[
\mathbf{X}'_c = \mathbf{R} \mathbf{X}_c + \mathbf{T}
\]
An aside: cross product

\[ \vec{a} \times \vec{b} = \vec{c} \]
\[ \vec{a} \cdot \vec{c} = 0 \]
\[ \vec{b} \cdot \vec{c} = 0 \]

Vector cross product takes two vectors and returns a third vector that’s perpendicular to both inputs.

So here, \( \vec{c} \) is perpendicular to both \( \vec{a} \) and \( \vec{b} \), which means the dot product \( = 0 \).
From geometry to algebra

\[ X' = RX + T \]

\[ T \times X' = \text{Normal to the plane} \]

\[ = T \times RX \]

\[ X' \cdot (T \times X') = X' \cdot (T \times RX) = 0 \]
Another aside: Matrix form of cross product

\[ \mathbf{a} \times \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \mathbf{c} \]

\[ \mathbf{a} \cdot \mathbf{c} = 0 \]
\[ \mathbf{b} \cdot \mathbf{c} = 0 \]

Can be expressed as a matrix multiplication.

\[ [\mathbf{a}_x] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \]

\[ \mathbf{a} \times \mathbf{b} = [\mathbf{a}_x] \mathbf{b} \]
From geometry to algebra

\[ X' = RX + T \]

\[ T \times X' = T \times RX + T \times T \]

\[ \text{Normal to the plane} \]

\[ = T \times RX \]

\[ X' \cdot (T \times X') = X' \cdot (T \times RX) = 0 \]
Essential matrix

\[
X' \cdot (T \times RX) = 0
\]
\[
X' \cdot ([T_x]RX) = 0
\]

Let \( E = [T_x]R \)

\[
X'^T EX = 0
\]

\( E \) is called the **essential matrix**, and it relates corresponding image points between both cameras, given the rotation and translation.

If we observe a point in one image, its position in other image is constrained to lie on line defined by above.

Note: these points are in **camera coordinate systems**.
Fundamental matrix

Relates **pixel coordinates** in the two views

\[
p_{im, right}^T F p_{im, left} = 0
\]

More general form than essential matrix: we remove need to know intrinsic parameters

If we estimate fundamental matrix from correspondences in **pixel coordinates**, can reconstruct epipolar geometry without intrinsic or extrinsic parameters.
Fundamental Matrix

From before, the essential matrix $E$. 

\[
\begin{align*}
(p_{c,\text{right}})^T E p_{c,\text{left}} &= 0 \\
(M^{-1}_{\text{int, right}} p_{\text{im, right}})^T E (M^{-1}_{\text{int, left}} p_{\text{im, left}}) &= 0 \\
p^T_{\text{im, right}} \begin{pmatrix} M^{-T}_{\text{int, right}} & EM^{-1}_{\text{int, left}} \end{pmatrix} p_{\text{im, left}} &= 0
\end{align*}
\]

"Fundamental matrix"

\[
p^T_{\text{im, right}} F p_{\text{im, left}} = 0
\]
Properties of the Fundamental Matrix

- $F_p$ is the epipolar line associated with $p$
- $F^T q$ is the epipolar line associated with $q$
- $F e_1 = 0$ and $F^T e_2 = 0$
- $F$ is rank 2
Estimating the Fundamental Matrix

If we don’t know $K_1$, $K_2$, $R$, or $t$, can we estimate $F$ for two images?

Yes, given enough correspondences
Estimating F – 8-point algorithm

The fundamental matrix $F$ is defined by

$$x'^T F x = 0$$

for any pair of matches $x$ and $x'$ in two images.

- Let $x = (u, v, 1)^T$ and $x' = (u', v', 1)^T$, $F = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix}$

each match gives a linear equation

$$uu' f_{11} + vu' f_{12} + u' f_{13} + uv' f_{21} + vv' f_{22} + v' f_{23} + uf_{31} + vf_{32} + f_{33} = 0$$
In reality, instead of solving \( A\mathbf{f} = 0 \), we seek \( \mathbf{f} \) to minimize \( \|A\mathbf{f}\| \), least eigenvector of \( A^T A \).
Estimating $F$ – 8-point algorithm

$F$ should have rank 2

To enforce that $F$ is of rank 2, $F$ is replaced by $F'$ that minimizes $\|F - F'\|$ subject to the rank constraint.

- This is achieved by SVD. Let $F = U\Sigma V^T$, where

$$
\Sigma = \begin{bmatrix}
\sigma_1 & 0 & 0 \\
0 & \sigma_2 & 0 \\
0 & 0 & \sigma_3
\end{bmatrix}, \text{ let } \Sigma' = \begin{bmatrix}
\sigma_1 & 0 & 0 \\
0 & \sigma_2 & 0 \\
0 & 0 & 0
\end{bmatrix}
$$

then $F' = U\Sigma' V^T$ is the solution.
Estimating F – 8-point algorithm

Pros: it is linear, easy to implement and fast
Cons: susceptible to noise
Outline

Projective geometry

Vanishing points
  Application: camera calibration
  Application: single-view metrology

Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Stereo Correspondence

Goal:
• Find correspondences (pairs of points $(u', v') \leftrightarrow (u, v)$).

1) Find interest points in image
2) Compute correspondences
3) Compute epipolar geometry
4) Refine

Example from Andrew Zisserman
1) Find interest points
Stereo Correspondence

2) Match points within proximity to get putative matches
3) Compute epipolar geometry -- robustly with RANSAC

Select random sample of putative correspondences

Compute $F$ using them
- determines epipolar constraint

Evaluate amount of support
- inliers within threshold distance of epipolar line

Choose $F$ with most support (inliers)
Using window search to get putative matches: noisy, but enough to compute $F$ using RANSAC

Pruned matches: those consistent with epipolar geometry
Outline

Projective geometry

Vanishing points
  Application: camera calibration
  Application: single-view metrology

Epipolar geometry
  Application: stereo correspondence
  Application: structure from motion revisited
Structure from Motion

1. Detect features using SIFT
2. Match features between pairs of images
3. Refine matching using RANSAC to find correspondences between each pair
4. Massive bundle adjustment
Structure from Motion

1. Detect features using SIFT
2. Match features between pairs of images
3. Refine matching using RANSAC to find correspondences between each pair
4. Massive bundle adjustment

Use fundamental matrix to detect inliers during RANSAC!
Structure from Motion Results
Structure from Motion Results
Recap

Projective geometry

Vanishing points

  Application: camera calibration
  Application: single-view metrology

Epipolar geometry

  Application: stereo correspondence
  Application: structure from motion revisited