Epipolar Geometry

Jordan Richards

October 22, 2021

1 Fundamental Matrix

https://web.stanford.edu/class/cs231a/course_notes/03-epipolar-geometry.pdf

1.1 typical epipolar camera setup

- baseline: the line between the camera centers
- epipole: the intersection of the baseline and the respective camera plane
  - this is a point at infinity if the cameras are parallel

1.1.1 for some 3D point, P, in both cameras' image space

- epipolar plane: the plane formed by the camera centers and P
- epipolar line: the intersection of the epipolar plane and the respective camera plane
  - for any choice of P, the epipolar line will contain the epipole (given the definition of epipolar plane)

1.2 essential matrix

- if we have some point p in one image, and want to find it's match p' in the other image, we know that it must lie on the corresponding epipolar line for P
- to find the epipolar line we must know the relative position of our cameras (rotation and translation)
- Assuming we have 2 canonical cameras (cameras with focal length of 1, so K = K' = I):
  - their projection matrices will be M = [I 0] and M' = $[\mathrm{R}^{\mathrm{T}}\ \text{-}\mathrm{R}^{\mathrm{T}} \cdot \mathrm{T}]$
  - for some point, p' on camera 2's image plane, we know that it maps to Rp' + T on camera 1's image plane
  - Since T and (Rp' + T) are on the epipolar plane for P, T $\times$ (Rp' + T) = T $\times$ Rp' is the normal of the plane
  - Since p is on the epipolar plane as well, $p^T \cdot (T \times Rp') = 0$
  - Rewriting T $\times$ Rp' as $\mathrm{T}_\times \cdot$ Rp', $\mathrm{p}^{\mathrm{T}} \cdot [\mathrm{T}_\times]\mathrm{R} \cdot$ p' = 0
  - our essential matrix, E = $[\mathrm{T}_\times]$R, so pEp' = 0

1.3 fundamental matrix

- we assumed the camera's were canonical to get the essential matrix, so we can factor K back into the equation:
  - $\mathrm{p}^{\mathrm{T}} \cdot \mathrm{K}^{\text{-T}} \cdot \mathrm{E} \cdot \mathrm{K}'^{\text{-1}} \cdot \mathrm{p}' = 0$
  - fundamental matrix, $\mathrm{F} = \mathrm{K}^{\text{-T}} \cdot \mathrm{E} \cdot \mathrm{K}'^{\text{-1}} = \mathrm{K}^{\text{-T}} \cdot \mathrm{T}_\times \cdot \mathrm{R} \cdot \mathrm{K}'^{\text{-1}}$

1.3.1 properties:

- $p^T \cdot F$: epipolar line in our first image
- $F \cdot p'$: epipolar line in our second image
- scale for lines doesn't matter, so F has only 8 DOF
- F maps points to lines so it should only have rank 2

1.3.2 estimation, 8-point algo:

- without knowing any of our camera parameters (K, K', T, R) if we have enough independent epipolar lines, we should be able to solve for F
- given 2 matching points in our images, p = (u, v, 1) and p' = (u', v', 1):
  1. $p^T \cdot F \cdot p' = 0$
  2. $[u\ v\ 1] \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = 0$
  3. $[\ (u * F_{11} + v * F_{21} + F_{31})\ (u * F_{12} + v * F_{22} + F_{32})\ (u * F_{13} + v * F_{23} + F_{33})\ ] \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = 0$
  4. $u'(u * F_{11} + v * F_{21} + F_{31}) + v'(u * F_{12} + v * F_{22} + F_{32}) + (u * F_{13} + v * F_{23} + F_{33}) = 0$
  5. $[\ u'u\ v'u\ u\ u'v\ v'v\ v\ u'\ v'\ 1\ ] \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0$
- With 8 perfect matches, we can solve for F directly
- Since our matches probably have some error, it is usually better to use more than 8 and find F with the least squared error using SVD
- Our resulting F might have rank 3, but the real F should only be rank 2, but we can fix this using SVD

1. dealing with noise

   `https://www.cs.princeton.edu/courses/archive/fall13/cos429/lectures/11-epipolar`

   - RANSAC: random simple concensus. We can take random samples of matches and use them to construct F, then take the best one (the fundamental matrix with the most inliers)
   - Least Median Squares: find the fundamental matrix that minimizes the median of the squares of our matches, instead of the sum/average of squares

1.4 image rectification

- It would be useful for future calculations if our cameras were parallel.
- From above:
  - when our cameras are parallel, their epipoles are at infinity

- all of our epipolar lines intersect at the epipole

- we can use the fundamental matrix to find the epipole, then use this to create a homography that maps these epipoles to a point at infinity
- our new camera planes (after this transformation) should be parallel

1.4.1 algorithm

`https://engineering.purdue.edu/kak/Tutorials/StereoRectification.pdf` Loop and Zhang break down the homography into 3 parts

1. projection component Maps the epipoles to some point at infinity
2. similarity component Ensures the epipoles in both images are on the X axis and that the images are aligned
3. shearing component
   - Minimizes projective distortion by making our final aspect ratio as close to 1:1 as possible
   - Implementation `https://scicomp.stackexchange.com/questions/2844/shearing-and-hartleys-rectifica`

2 Depth, Disparity with parallel cameras

`https://blog.pollithy.com/vision/epipolar-geometry`

- f: focal length in camera units (pixels)
- $x_{(l, r)}$: $(p, p')_x$ in camera units (pixels)
- b: baseline, distance between the camera centers in real units
- $b_{(l, r)}$: displacement between camera (1, 2)'s center and P along the baseline
- z: depth, distance between either camera and P, perpendicular to the baseline
- d: disparity, $x_l + x_r$
- (As the triangles are similar) x_l / d = b_l / b
  - $\rightarrow b_l = x_l \cdot b / d$
- (As the triangles are similar) x_l / f = b_l / z
  - $\rightarrow z = f \cdot b_l / x_l$
- $z = \frac{f \cdot x_l \cdot b}{x_l \cdot d}$
  - $\rightarrow z = \frac{f \cdot b}{d}$

3 TODO extra sources

3.1 `https://github.com/eddiecorrigall/Vision/blob/master/Stereo/rectify.py`

3.2 `https://stackoverflow.com/questions/36172913/opencv-depth-map-from-uncalibrated-stereo-sy`