CSE 252A: Stereo Vision I

Lecturer: David Kriegman

Scribed by Owen Jow on November 06, 2018

1 Stereo

Binocular stereopsis: the process of estimating depth from two views of a scene. Algorithmically, this means establishing correspondences between the images and using *what we know about relative camera poses* and *disparity between the corresponding points* to estimate the depth of each of these points. (If the disparity is small, the point is far away;[1] if the disparity is large, the point is close.)

1.1 Depth from Binocular Stereo

1. *Offline*:
   (a) Calibrate cameras and determine epipolar geometry.
2. *Online*:
   (a) Acquire stereo images, rectify to make the correspondence problem easier to solve.
   (b) Determine correspondences and estimate depth from the disparity signal.

1.2 Epipolar Geometry

Naively, finding correspondences would require searches over the entire image, which is extremely computationally expensive. After calibrating the stereo rig s.t. the camera setup is known, we can frame things in an epipolar geometry context and drastically reduce the correspondence search space.

Namely, potential matches for $\mathbf{p}$ have to lie on the epipolar line[2] corresponding to $\mathbf{p}$. Similarly, potential matches for $\mathbf{p}'$ have to lie on the epipolar line corresponding to $\mathbf{p}'$. Instead of looking over the entire image, we can just look along a line! This is the main power of epipolar geometry.

Vocabulary:

- **Baseline**:[3] line connecting two centers of projection
- **Epipole**: intersection points of baseline with image plane
- **Epipolar plane**: a plane containing the baseline
- **Epipolar line**: intersection of epipolar plane with image plane

[1] Zero disparity means the 3D point is infinitely far away.

[2] *Epipolar line*: the projection of the line through the pinhole and $\mathbf{p}$ onto the other image.

[3] sometimes also used to refer to the distance between the centers of projection

- **Family of epipolar planes**: a collection of epipolar planes rotated around the baseline
  - each epipolar plane gives different epipolar lines, so if we want to match over many epipolar lines, we can iterate over a family of epipolar planes for a relevant angle range

1.2.1 Epipolar Constraint: Calibrated Case

In this setup, each camera is "calibrated," i.e. focal length is 1 mm, the image origin is at the center, and image coordinates are in mm. We will also assume that we have two pinhole cameras which differ by a rotation $\mathbf{R}$ and a translation $\mathbf{t}$,[4] and a point $\mathbf{P}$ in the world which projects to $\mathbf{p}$ in camera 1 and $\mathbf{p}'$ in camera 2. We would like to derive a relationship between $\mathbf{p}$ and $\mathbf{p}'$.

Let the centers of projection be $\mathbf{O}_1$ and $\mathbf{O}_2$.

Geometrically, the vectors $\mathbf{O_1p}$, $\mathbf{O_1O_2}$, and $\mathbf{O_1p}'$ are coplanar. So

$$\begin{aligned}\mathbf{O_1p} \cdot [\mathbf{O_1O_2} \times \mathbf{O_1p}'] &= 0\\ \mathbf{p} \cdot [\mathbf{t} \times \mathbf{Rp}'] &= 0\\ \mathbf{p}^T [\mathbf{t}]_\times \mathbf{Rp}' &= 0\\ \mathbf{p}^T \mathbf{Ep}' &= 0\end{aligned}$$

Note: the determinant of a skew-symmetric matrix is zero (it has zeros along the diagonal), so the **essential matrix** $\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$ is also going to have a zero determinant.

We can obtain the essential matrix by estimating $\mathbf{R}, \mathbf{t}$ through camera calibration (photograph a calibration rig with known 3D geometry like a checkerboard and solve some equations), or using the eight-point algorithm (which requires correspondences in two images).

- We don't need to solve for all nine elements; we can set $\mathbf{E}_{33}$ to 1 for example.

1.2.2 Properties of the essential matrix and the epipolar constraint

1. The epipolar constraint equation is homogeneous in $\mathbf{p}$, $\mathbf{p}'$, and $\mathbf{E}$ (equals 0).
   (a) If you take a $\mathbf{p}$, $\mathbf{p}'$, or $\mathbf{E}$ which works and multiply it by a scale factor, it still works.
2. The epipolar constraint equation is bilinear in $\mathbf{p}$ and $\mathbf{p}'$.
   (a) For a given value of $\mathbf{p}$, it is linear in $\mathbf{p}'$ (and vice-versa).
   (b) Given $\mathbf{p}'$, the equation of its epipolar line in image 1 is $\mathbf{a}^T\mathbf{p} = 0$ where $\mathbf{a} = \mathbf{Ep}'$.
   (c) Given $\mathbf{p}$, the equation of its epipolar line in image 2 is $\mathbf{b}^T\mathbf{p}' = 0$ where $\mathbf{b} = \mathbf{E^T p}$.
3. $\mathbf{Ee}' = 0$ and $\mathbf{E^T e} = 0$. The epipole is a degenerate case; there is no epipolar line corresponding to it[5] and we can't estimate depth for it because there's no triangle formed.
   (a) The epipole $\mathbf{e}'$ is the eigenvector of $\mathbf{E}$ with eigenvalue zero.
   (b) The epipole $\mathbf{e}$ is the eigenvector of $\mathbf{E^T}$ with eigenvalue zero.
4. $\mathbf{E}$ is singular, with rank two and determinant zero. It has two equal nonzero singular values.

[4] s.t. $\mathbf{Rp}' + \mathbf{t}$ expresses a point in image 2 coordinates as a point in image 1 coordinates

[5] $\mathbf{e}$'s matching point is $\mathbf{e}'$ and vice-versa.

1.2.3 Epipolar Constraint: Uncalibrated Case

In the uncalibrated case, we're just given two images. We don't know anything about the camera intrinsics or extrinsics. We need to determine a mapping which works for pixel coordinates.

This turns out to be the same thing, except computed on pixel coordinates.

Let $\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$ be the 3×4 matrix mapping from homogeneous 4×1 world coordinates to calibrated (camera) coordinates. Let $\mathbf{K}$ be the 3×3 intrinsic matrix mapping from calibrated coordinates to image (uncalibrated) coordinates. Then $\mathbf{q} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{P}$ is the full mapping from 3D to image coordinates, and $\mathbf{K}^{-1}\mathbf{q}$ is the mapping from image coordinates to calibrated coordinates.

The relationship between calibrated coordinates $\mathbf{p}$, $\mathbf{p}'$ and uncalibrated coordinates $\mathbf{q}$, $\mathbf{q}'$ is

$$\begin{aligned}\mathbf{p} &= \mathbf{K}^{-1}\mathbf{q} \\ \mathbf{p}' &= (\mathbf{K}')^{-1}\mathbf{q}'\end{aligned}$$

If we plug this into the epipolar constraint, we observe

$$\begin{aligned}\mathbf{p}^T\mathbf{E}\mathbf{p}' &= 0 \\ (\mathbf{K}^{-1}\mathbf{q})^T\mathbf{E}((\mathbf{K}')^{-1}\mathbf{q}') &= 0 \\ \mathbf{q}^T\left[(\mathbf{K}^{-1})^T\mathbf{E}(\mathbf{K}')^{-1}\right]\mathbf{q}' &= 0 \\ \mathbf{q}^T\mathbf{F}\mathbf{q}' &= 0\end{aligned}$$

for $\mathbf{F} = (\mathbf{K}^{-1})^T\mathbf{E}(\mathbf{K}')^{-1}$ the **fundamental matrix**.

- We can solve for this using the eight-point algorithm without calibration.
- Most properties are the same as those of the essential matrix. However, one difference is that the two singular values of the fundamental matrix are no longer necessarily identical.

1.3 Rectification

To make correspondence even easier, we usually perform rectification. Namely, we warp the images so that the epipolar lines become scan lines. (Note: if the cameras are pure translations of each other in a direction parallel to the image plane, the epipolar lines are identical rows in each image.)

A mapping from a plane to a plane through the same optical center is a homography. So we'll devise 3×3 homographies $\mathbf{H_L}$, $\mathbf{H_R}$ that map each original image to the rectified image that we want.

Basically, we send each epipole out to infinity.[6][7] The left homography $\mathbf{H_L}$ should satisfy $\mathbf{H_L}\mathbf{e} = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$ (we want the horizontal point at infinity since we want the epipolar lines to be rows).

1.4 Getting Correspondences

We're not going to talk about this today.

[6] All epipolar lines converge at the epipoles, so if we want them to be parallel, the epipoles need to be at infinity.

[7] If the epipole is inside of the image, this isn't going to work.

1.5 Estimating Depth

The hard part about stereopsis is getting the correspondences. Once we have correspondences, it becomes a triangulation problem, where we have a point in one image and the corresponding point in another image, and (assuming the cameras are calibrated[8]) we simply project rays out from the centers of projection through the image points, and take their intersection as the 3D point.

However, the 3D rays probably won't intersect unless calibration is perfect. So for triangulation, we'll actually want to either

- find the 3D point where the two rays are closest to each other **(linear, faster)**, or
- find the 3D point which minimizes reprojection error onto the two images **(more accurate)**
  - involves solving the nonlinear optimization problem $d^2(\mathbf{p}, \mathbf{M}\mathbf{Q}^*) + d^2(\mathbf{p}', \mathbf{M}'\mathbf{Q}^*)$ for $\mathbf{Q}^*$
  - equivalent to finding the $\mathbf{q}$ and $\mathbf{q}'$ closest to $\mathbf{p}$ and $\mathbf{p}'$ whose rays do intersect in 3D

If we have two 1D image planes in a 2D space, e.g. in an XZ-plane, where the image planes are a parallel X-translation by b (the baseline length) of each other and f is the focal length of both cameras, the disparity $d = x_L - x_R = \frac{fX}{Z} - \frac{f(X-b)}{Z} = \frac{fb}{Z}$ and the depth $Z = \frac{fb}{d}$.

1.6 Visualizing Depth

A disparity image (disparity plotted at every location) is one way to show depth; disparity is inversely proportional to depth so it's all in how you shade it. The other way is to directly show 1/(disparity).

[8]Meaning we know the camera calibration matrix $\mathbf{K}$ (also sometimes called the *intrinsic matrix*). Via $\mathbf{K}$, a ray in camera coordinates projects to a single point in image coordinates. Note that we can also "back-project" using $\mathbf{K}^{-1}$, which means we take a point $\mathbf{x}$ in (homogeneous) image coordinates and compute $\mathbf{d} = \mathbf{K}^{-1}\mathbf{x}$ where $\mathbf{d}$ can be thought of as the relevant ray through the pinhole or alternatively as the point in "normalized image coordinates."