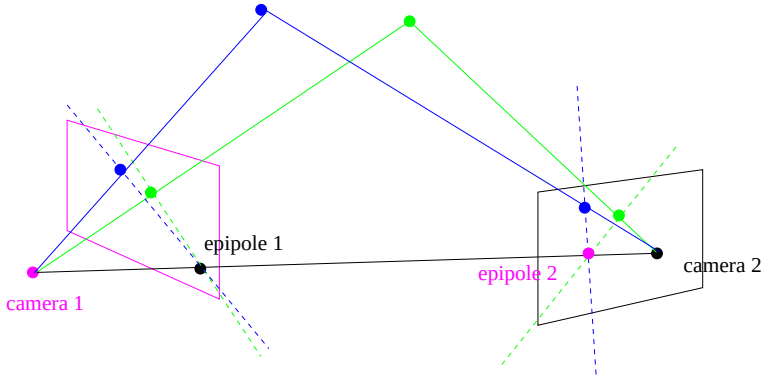
When we examined the egomotion problem, we considered two neighboring frames of a video that were close in time so that the rotation and translation were very small. In the remaining lectures, we address a related problem, namely binocular stereo. Here, we do *not* assume that the translation and rotation between cameras is small, nor do we assume that the two cameras have the same internal parameters.

Despite these differences, similarities between the problems remain. Recall that in the egomotion problem, the velocity vector at a pixel was the sum of a rotation component and a translation component, such that the translation component depended on inverse depth. The velocity vector at a point (x, y) was thus constrained to a line in (v_x, v_y) space, such that the direction of the line was in the direction of heading i.e. the direction of translation from one camera position to the next. (Recall Eqns. 1,2,4 from lecture 20.) As we see in this lecture, there is a similar constraint in binocular stereo – namely that constrains corresponding points to lie on a line.)

Stereo and Epipolar Geometry

The figure below shows the basic setup of stereo geometry. We have two cameras, 1 and 2, with centers shown respectively in pink and black. Each camera has a position in 3D. Each camera also has a coordinate system with orthonormal axes, and projection plane (illustrated using a rectangle in the figure).



For any 3D point (blue or green, for example), we project that point into the image projection planes of the two cameras. What can we say about this projection? Since the two cameras and the chosen 3D point define a plane Π, the image projection of the 3D point must lie on the intersection of this plane Π with the image projection plane, namely along a line. The dotted lines in the figure show these plane intersections for the two cameras and for two 3D points (blue and green). These dotted lines are called *epipolar* lines.

The epipolar lines intersect at a single point. To see this, note that the line joining the two cameras lies on every plane Π defined above, hence the point of intersection of this line with the image plane also lies on every epipolar line. We call this point the *epipole*. (This point could be at infinity, for example, if the cameras are facing the same direction.)

Given two cameras, if you know the epipolar geometry (namely the epipoles and epipolar lines) then you can narrow down the possible correspondences between points in the two images. All points on the blue plane project to the blue epipolar lines. So given a blue point (which lies on a

blue epipolar line) in the first image, you only need to search for the corresponding point on the blue epipolar line in the second image.

The Essential Matrix

Let us now translate the argument on the previous page from geometry to algebra. Suppose we have a 3D point that is written as $\mathbf{X}_1 = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}$ in camera 1's coordinates and the same 3D point is written as $\mathbf{X}_2 = \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix}$ in camera 2's coordinates. Let the position of camera 2 be (T_X, T_Y, T_Z) when written in camera 1's coordinates. Of course, the position of camera 2 in camera 2's coordinates is $(0,0,0)^T$ and the position of camera 1 in camera 1's coordinates is $(0,0,0)^T$. Let $\mathbf{R}_{2\leftarrow 1}$ be an orthonormal matrix that rotates/reflects camera 1 coordinate axes to camera 2 coordinate axes. (That is, the columns of $\mathbf{R}_{2\leftarrow 1}$ are the coordinates of camera 1's axes, expressed in camera 2's coordinates. Or equivalently, the rows of $\mathbf{R}_{2\leftarrow 1}$ are the camera 2 axes expressed in camera 1's coordinates.)

Then we have:

$$\mathbf{X}_2 = \mathbf{R}_{2\leftarrow 1}(\mathbf{X}_1 - \mathbf{T})$$

This is just the usual change of coordinates, where we are starting with camera 1 coordinates and mapping to camera 2 coordinates. Bringing the $\mathbf{R}$ matrix to the other side gives

$$\mathbf{R}_{1\leftarrow 2}\mathbf{X}_2 = \mathbf{X}_1 - \mathbf{T}. \tag{1}$$

The epipolar constraint is hidden in there (implicitly). To make it more explicit, we write the relationship between $\mathbf{X}_1$ and $\mathbf{X}_2$ in a different way. The vectors $\mathbf{X}_1$, $\mathbf{T}$ and $\mathbf{X}_1 - \mathbf{T}$ are linear dependent and, in particular,

$$(\mathbf{X}_1 - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{X}_1) = 0. \tag{2}$$

Define the cross product with $\mathbf{T}$ operation as matrix,

$$[\mathbf{T}]_\times \equiv \begin{bmatrix} 0 & T_Z & -T_Y \\ -T_Z & 0 & T_X \\ T_Y & -T_X & 0 \end{bmatrix}$$

and so combining Eqns. (1) and (2) gives

$$\mathbf{X}_2^T \, \mathbf{R}_{2\leftarrow 1}[\mathbf{T}]_\times \mathbf{X}_1 = 0.$$

This says: you take $\mathbf{X}_1$, cross it with $\mathbf{T}$ (which gives a vector perpendicular to the plane containing the two camera centers and the 3D point), write this vector in terms of the camera 2 axes, and you get a vector that is perpendicular to $\mathbf{X}_2$, i.e. the scene point written in terms of the camera 2 coordinates.

The matrix

$$\mathbf{E} \equiv \mathbf{R}_{2\leftarrow 1}[\mathbf{T}]_\times \tag{3}$$

is called the *essential matrix*, and so one writes:

$$\mathbf{X}_2^T \, \mathbf{E}\mathbf{X}_1 = 0. \tag{4}$$

Now we are ready to define epipolar lines and epipoles. From Eq. (4), note that we can scale the vectors $\mathbf{X}_1$ and $\mathbf{X}_2$ to $\mathbf{x}_1 = (x_1, y_1, f_1)$ and $\mathbf{x}_2 = (x_2, y_2, f_2)$, respectively, so that they lie in the image planes of their respective cameras. So, for any $\mathbf{X}_1$, we get an *epipolar* line in the camera 2 projection plane, as follows. Define a 3-vector $\mathbf{l}_2 \equiv \mathbf{E}\mathbf{x}_1$. Then

$$\mathbf{x}_2^T \mathbf{E}\mathbf{x}_1 = \mathbf{x}_2^T \mathbf{l}_2 = 0$$

which is the *epipolar line* in the second image.

Similarly, for any $\mathbf{X}_2$, define $\mathbf{l}_1^T \equiv \mathbf{x}_2^T \mathbf{E}$, then

$$\mathbf{x}_2 \mathbf{E}\mathbf{x}_1 = \mathbf{l}_1^T \mathbf{x}_1 = 0$$

which is the epipolar line in the first image.

What about the intersections of the epipolar lines? Note that the essential matrix is of rank 2, since $[\mathbf{T}]_\times$ is of rank 2. In particular, $[\mathbf{T}]_\times \mathbf{T} = \mathbf{0}$ and so $\mathbf{E}\mathbf{T} = \mathbf{0}$. Thus, $\mathbf{x}_2^T \mathbf{E}\mathbf{T} = \mathbf{0}$ for any $\mathbf{x}_2$. Thus, $f_1 \frac{\mathbf{T}}{|\mathbf{T}}$ lies on all epipolar lines in camera 1's projection plane. Thus it is the epipole.

For the epipole in camera 2, we need $\mathbf{x}_2^T \mathbf{E} = \mathbf{0}$, or equivalently, $\mathbf{E}^T \mathbf{x}_2 = \mathbf{0}$. But

$$\mathbf{E}^T \mathbf{x}_2 = -[\mathbf{T}]_\times \mathbf{R}_{1 \leftarrow 2} \mathbf{x}_2,$$

since $[\mathbf{T}]_\times$ is anti-symmetric. So, for the camera 2 epipole, we want $\mathbf{R}_{1 \leftarrow 2} \mathbf{x}_2 = \mathbf{T}$. So the epipole is $\mathbf{x}_2 = \mathbf{R}_{2 \leftarrow 1} \mathbf{T}$, that is, the translation vector written in camera 2's coordinates.

The Fundamental Matrix

The above arguments relied on points on image projection planes. We next write them in terms of pixel coordinates.

Suppose the two cameras have calibration matrices $\mathbf{K}_1$ and $\mathbf{K}_2$. If a 3D point (X, Y, Z) is written in the the first camera's coordinates, then its pixel position is (x_1, y_1) where

$$\begin{bmatrix} w_1 x_1 \\ w_1 y_1 \\ w_1 \end{bmatrix} = \mathbf{K}_1 \mathbf{X}_1$$

and its pixel position in the second camera is (x_2, y_2) where:

$$\begin{bmatrix} w_2 x_2 \\ w_2 y_2 \\ w_2 \end{bmatrix} = \mathbf{K}_2 \mathbf{X}_2$$

Note the (x_1, y_1) and (x_2, y_2) values are different from what we saw at the top of this page, where they referred to points on the projection plane (not pixels).

We can rewrite Eq. (4) in terms of the pixel coordinates and ignore the scalers w_1 and w_2,

$$\left[\begin{array}{ccc} x_2 & y_2 & 1 \end{array} \right] \mathbf{K}_2^{-T} \mathbf{E} \mathbf{K}_1^{-1} \left[\begin{array}{c} x_1 \\ y_1 \\ 1 \end{array} \right] = 0.$$

The matrix

$$\mathbf{F} \equiv \mathbf{K}_2^{-T} \mathbf{E} \mathbf{K}_1^{-1}$$

is called the *fundamental matrix*. It relates corresponding pixels (x_1, y_1) and (x_2, y_2) in the two cameras.

Epipolar lines (and epipoles) are defined in exactly the same way as was done for the essential matrix. Any pixel $\mathbf{x}_1 = (x_1, y_1, 1)^T$ in the first image defines a vector

$$\mathbf{l_2} = \mathbf{K}_2^{-T} \mathbf{E} \mathbf{K}_1^{-1} \mathbf{x}_1$$

such that $\mathbf{x}_2 \mathbf{l_2} = 0$, which is an epipolar line in the second image. Similarly, any pixel $\mathbf{x}_2 = (x_2, y_2, 1)$ in the second image defines a vector

$$\mathbf{l_1} = \mathbf{x}_2 \mathbf{K}_2^{-T} \mathbf{E} \mathbf{K}_1^{-1}$$

such that $\mathbf{l}_1 \cdot \mathbf{x_1} = 0$, which is an epipolar line in the first image.

$\mathbf{F}$ has rank 2, since the essential matrix $\mathbf{E}$ has rank 2. The epipole $\mathbf{e}_1$ in the camera 1 image is in the null space of $\mathbf{F}$. The epipole $\mathbf{e}_2$ in the camera 2 image is in the null space of $\mathbf{F}^T$. Note that the epipoles positions might not lie within the image domain (which is finite). Indeed they could even be at infinity.

Finally, note that if you know the fundamental matrix $\mathbf{F}$, then the correspondence problem from points in one image to points in the other image is restricted to lines. Notice that *this is restriction holds, even if you only know* $\mathbf{F}$*, but you do not know the camera calibration matrices (internal) or the rotation and translation between cameras (external).*

Estimation of Fundamental matrix (8 point algorithm)

How could we find the fundamental matrix that relates two images? Suppose we find a pair of corresponding points (x_1, y_1) and (x_2, y_2) in the first and second camera's image, respectdively. We then would have the following constraint on the nine $\mathbf{F}_{ij}$ elements.

$$(x_1x_2,\ y_1x_2,\ x_2,\ x_1y_2,\ y_1y_2,\ y_2,\ x_1,\ y_1,\ 1) \cdot (F_{11},\ F_{12},\ F_{13},\ F_{21},\ F_{22},\ F_{23},\ F_{31},\ F_{32},\ F_{33}) = 0$$

Eight points gives a systems of 8 equations with 9 unknowns. This defines an 8×9 matrix. The null space of this system of equations would give us an exact estimate of $\mathbf{F}$. This is called the *eight point algorithm.*

The positions of the eight corresponding points will typically be noisy. To get a more accurate estimate, we can use $N \gg 8$ corresponding points and take the SVD of an $N \times 9$ matrix. This amounts to solving the least squares problem, of finding an $\mathbf{F}$ that minimizes:

$$\sum_{i=1}^{N} (\mathbf{x}^{i}_{2}{}^{T} \mathbf{F} \mathbf{x}^{i}_{1})^2$$

subject to a constraint such as $\| \mathbf{F} \| = 1$. [ASIDE: I neglected to mention this constraint in class but it is important, since you can trivially minimize the sum of squares by letting $\mathbf{F} \rightarrow [\mathbf{0}]$, that is, the zero matrix.] For the solution, we take the last column of $\mathbf{V}^T$, which corresponds to the smallest eigenvalue.

There are several final points to make. One is that the fundamental matrix is a 3×3 matrix of rank 2, and it has seven degrees of freedom. To see this, note that the rank 2 constraint implies that any column is a linear combination of the other two columns e.g. the third column is a linear combination of the first two columns. So the first two elements of the third column specify the linear combination and hence specify the third element of the third column. This suggests there ar eight degrees of freedom. However, recall that we are working with homogeneous coordinates, so you can scale $\mathbf{F}$ by any constant without changing the $\mathbf{x}_2^T \mathbf{F} \mathbf{x}_1 = 0$ constraint. This removes one of the eight degrees of freedom, leaving seven.

Another point is that, if there is any error/noise in the positions of the points then the ninth singular value will not be zero, and most likely the estimated $\mathbf{F}$ will be of rank 3 rather than rank 2. If we use this estimated $\mathbf{F}$, then the epipolar lines will not intersect exactly at the epipoles, and the epipolar constraints will not hold exactly.

If the estimated $\mathbf{F}$ is of rank 3, then often one decomposes the $\mathbf{F}$ (3×3) using the SVD,

$$\mathbf{F} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T$$

and one finds a rank 2 matrix that is close to $\mathbf{F}$. It can be shown that you can obtain the rank 2 matrix that is "closest to" $\mathbf{F}$ by setting the third singular value to 0. (That is, set Λ_{33} to 0.) This forces the $\mathbf{F}$ matrix to be rank 2. It forces the corresponding points to line on epipolar lines and all epipolar lines in an image pass through an epipole. (Of course, this epipolar lines and epipoles will be incorrect, since we are making an approximation.) But it is the best we can do in the presence of noise.