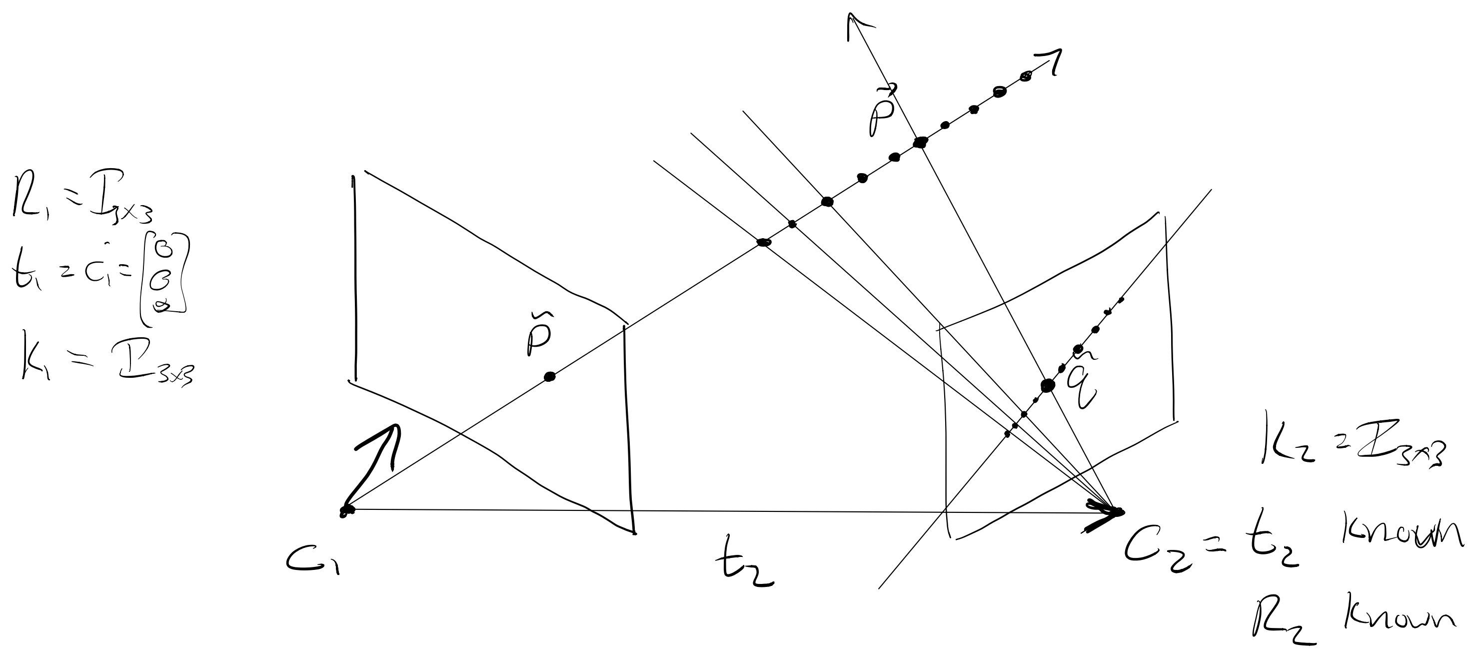
497P/597P - Projective Geometry 2: Epipolar Geometry

We've now talked about the projective representation of points and lines on an image plane. We can interpret these objects two ways:

- their "true" form (in $\mathbb{P}^2$), their projection on the 2D image plane (points and lines)
- their 3D analogues (in $\mathbb{R}^3$), before projection (rays and planes)

Where can a point in Camera 1 project to in Camera 2?



Assume that Camera 1 is in canonical position: c_1 is at the origin and its optical axis coincides with world $-z$, or in other words $t_1 = \mathbf{0}_{3\times 1}$ and $R_1 = I_{3\times 3}$. Now let's add a second camera with known extrinsics $[R_2; t_2]$ and ask: for a given 2D homogeneous point $\tilde{p}$ in Camera 1, where might it appear in Camera 2?

- The 3D point $\mathbf{p}$ that projected to $\tilde{p}$ could be anywhere along the 3D ray from c_1 in the direction of $\tilde{p}$.
- Unless the two camera centers coincide, this **ray** projects to a **line** in Camera 2.

What line does it project to? Once again, let's think in 3D:

- The 3D **plane** that projects to that line is spanned by the two vectors $\tilde{p}$ and $c_2 - c_1$. Recall that c_1 is the origin and c_2 is known to be t_2: the translation of the second camera's extrinsics.
- Therefore, So the equation for the line along which $\tilde{p}$'s projection lies in Camera 2 is:

$$\ell_{\tilde{p}}^2 = t_2 \times \tilde{p}$$

or, using our cross-product conversion:

$$\ell_{\tilde{p}}^2 = [t_2]_\times \cdot \tilde{p}$$

There's one problem with this: this is the equation for the line **in Camera 1** coordinates. To put it in Camera 2's coordinate system, we need to change our perspective such that c_2 is the origin and Camera 2's optical axis is looking down the $-z$ axis.

- Conveniently, we've assumed that Camera 1's coordinate system is the same as world coordinates! We have tools for this: Camera 2's extrinsics tell us how to get from world coordinates to Camera 2 coordiantes.
- Normally, we'd need to apply the translation then the rotation. However, the plane we're working with already goes through both c_1 and c_2, and it's represented by its normal vector.
- The normal vector is assumed to start at the origin, wherever that is - it's a *direction*, not a position.
- This means that we don't need to translate it in order to express it in Camera 2 coordinates: all we have to do is account for the change in **orientation**.
- So in Camera 2 coordinates, the line that $\mathbf{p}$ must project onto is:

$$\ell_{\tilde{p}}^2 = R_2 [t_2]_\times \tilde{p}$$

The Essential Matrix

One way to use the knowledge we've just derived is the following:

If a Camera 2 point $\tilde{q}$ corresponds to the same 3D point as $\tilde{p}$ in Camera 1, then $\tilde{q}$ must lie on the above line. Using our point-on-line test, we have:

$$\tilde{q}^T \ell^2_{\tilde{p}} = 0$$

Breaking it back apart, this can be written:

$$\tilde{q}^T (R_2 [t_2]_\times) \tilde{p} = 0$$

This is a pretty neat property, so it deserves a name. The penthesized quantity $R_2[t_2]_\times$ is a 3×3 matrix called the **essential matrix**:

$$\tilde{q}^T \mathbf{E} \tilde{p} = 0$$

Notice that this property holds for **any** pair of corresponding points $\tilde{p}$ and $\tilde{q}$!

The Fundamental Matrix

All of the above has assumed that $\tilde{p}$ and $\tilde{q}$ are in (respectively) Camera 1's and Camera 2's **camera coordiantes**. Another possible way to say this would be that both cameras have identity intrisnics (i.e., $K_1 = K_2 = I_{3\times3}$). If this is not the case but K_1 and K_2 are known, we can still work with this.

Let p be the point in Camera 1's *pixel* coordinates corresponding to $\tilde{p}$.

Let q be the point in Camera 2's *pixel* coordinates corresponding to $\tilde{q}$.

Then, using the known intrinsics, we have

$$p = K_1 \tilde{p}$$
$$q = K_2 \tilde{q}$$

or

$$\tilde{p} = K_1^{-1} p$$
$$\tilde{q} = K_2^{-1} q$$

If we plug that into our Essential Matrix constraint, we get:

$$
\begin{aligned}
\tilde{q}^T \mathbf{E} \tilde{p} &= 0 \\
(q^T K_2^{-T}) \mathbf{E} (K_1^{-1} p) &= 0 \\
q^T (K_2^{-T} R_2 [t_2]_\times K_1^{-1}) p &= 0
\end{aligned}
$$

Substituting the definition of $\mathbf{E}$, and rearranging parentheses:

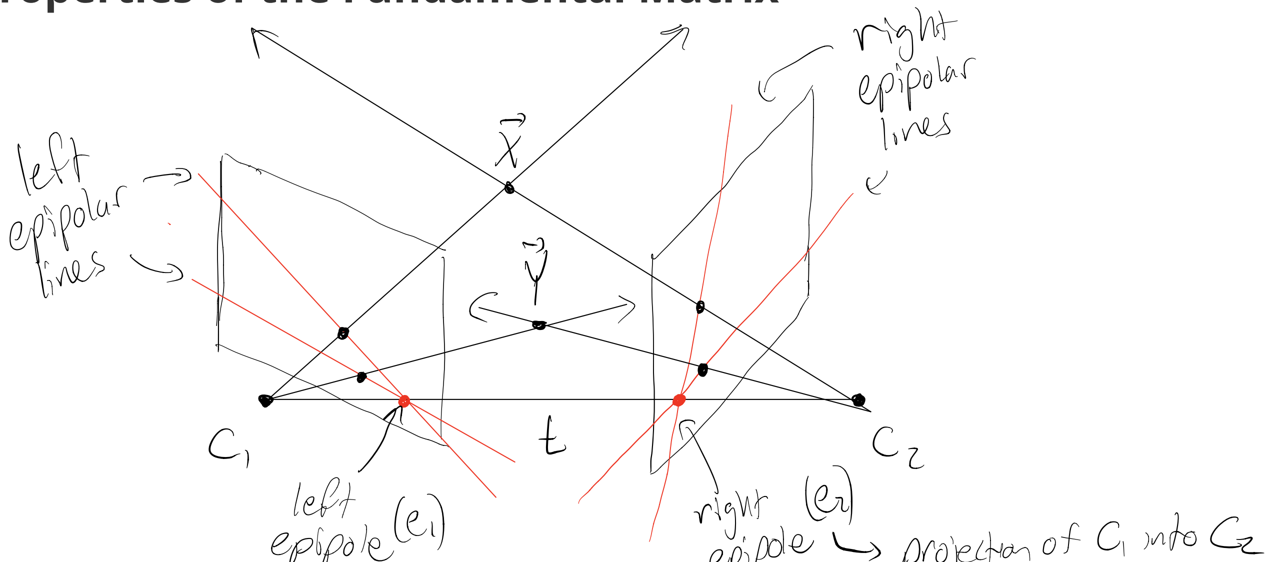
$$
\begin{aligned}
q^T K_2^{-T} R_2 [t_2]_\times K_1^{-1} p &= 0 \\
q^T (K_2^{-T} R_2 [t_2]_\times K_1^{-1}) p &= 0 \\
q^T \mathbf{F} p &= 0
\end{aligned}
$$

← "The Epipolar Constraint"

where $\mathbf{F}$ is cool enough to earn the name **Fundamental Matrix**. This is crazy: we have a matrix that imposes a constraint on the pixel coordinates of two corresponding points.

It's not quite good enough to **solve** for the location of q given p or vice versa - that's impossible, because it's geometrically ambiguous without knowing the 3D location of the 3D scene point $\mathbf{p}$ that projected t p and q. $\mathbf{F}p$ is the equation of a line, after all - but it restricts the possible solutions to a 1D search space along that line.

Properties of the Fundamental Matrix



- If $\mathbf{F}$ is the fundamental matrix that relates Camera 1 to Camera 2
- $\mathbf{F}$ has rank 2: $\mathbf{F}p$ maps to a 1-dimensional solution space (geometrically speaking, a line).
- All epipolar lines go through the **epipole**.
- The baseline vector t_2 spans all epipolar planes and passes through both epipoles.
- e_1 spans the null space of $\mathbf{F}$, i.e., $\mathbf{F}e_1 = 0$.
- e_2 spans the null space of $\mathbf{F}^T$, i.e., $\mathbf{F}^T e_2 = 0$.