7.1-7.2 3D - Motion

Outline

- Triangulation
- Two-Frame Structure from Motion

Outline

- **Triangulation**

- Two-Frame Structure from Motion

Structure from Motion

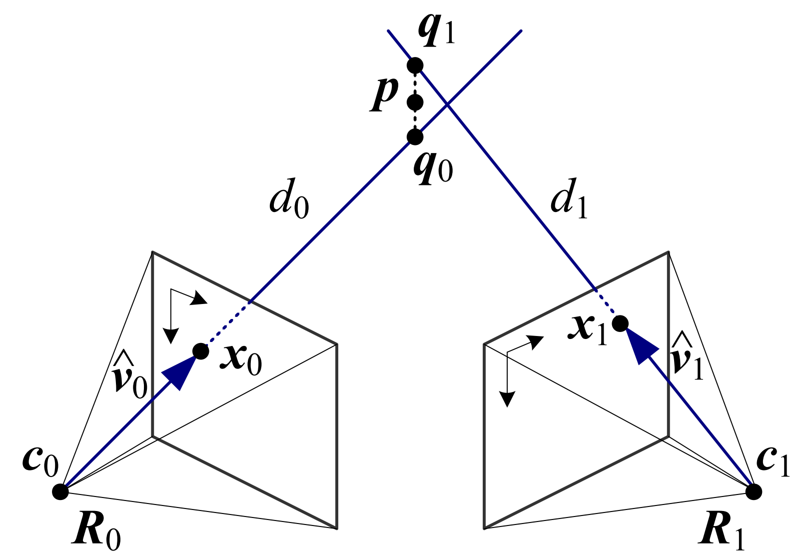
- Pose Estimation and Geometric Camera Calibration:
  - Given *known* 3D scene points and 2D correspondences in *one* image, compute the camera pose and intrinsic parameters.
- Triangulation:
  - Given 2D correspondences over *multiple* images and known camera pose, compute the *unknown* 3D scene points

Triangulation

- ***Definition***. The identification of a 3D point from a set of corresponding 2D image locations, from *known* camera poses.
- Consider multiple cameras with projection matrices $\boldsymbol{P}_j$: $\boldsymbol{P}_j = \boldsymbol{K}_j \left[\boldsymbol{R}_j \,|\, \boldsymbol{t}_j \right]$
- Let $\boldsymbol{c}_j$ represent the 3D camera centre for camera j, in world coordinates.
- Observe that $\boldsymbol{t}_j = -\boldsymbol{R}_j \boldsymbol{c}_j$
- Now consider a 3D point $\boldsymbol{p}$ that projects to 2D image points $\boldsymbol{x}_j$ in each of the cameras.
- To recover the point $\boldsymbol{p}$, we seek the 3D point that comes closest to the set of rays passing through each camera centre $\boldsymbol{c}_j$ and each 2D image projection $\boldsymbol{x}_j$.
- In other words, we seek the $\boldsymbol{p}$ that minimizes

$$\|\boldsymbol{c}_j + d_j \hat{\boldsymbol{v}}_j - \boldsymbol{p}\|^2$$

- where $\hat{\boldsymbol{v}}_j = \mathcal{N}(\boldsymbol{R}_j^{-1} \boldsymbol{K}_j^{-1} \boldsymbol{x}_j)$



3D Deviations

$\|\boldsymbol{c}_j + d_j \hat{\boldsymbol{v}}_j - \boldsymbol{p}\|^2$

- Let $\boldsymbol{q}_j$ represent the point on the jth ray lying closest to $\boldsymbol{p}$: $\boldsymbol{q}_j = \boldsymbol{c}_j + d_j \hat{\boldsymbol{v}}_j$

- Observe that at $\boldsymbol{q}_j$, $d_j = \hat{\boldsymbol{v}}_j \cdot (\boldsymbol{p} - \boldsymbol{c}_j)$.

- Thus $\boldsymbol{q}_j = \boldsymbol{c}_j + (\hat{\boldsymbol{v}}_j \hat{\boldsymbol{v}}_j^T)(\boldsymbol{p} - \boldsymbol{c}_j) = \boldsymbol{c}_j + (\boldsymbol{p} - \boldsymbol{c}_j)_\|$,

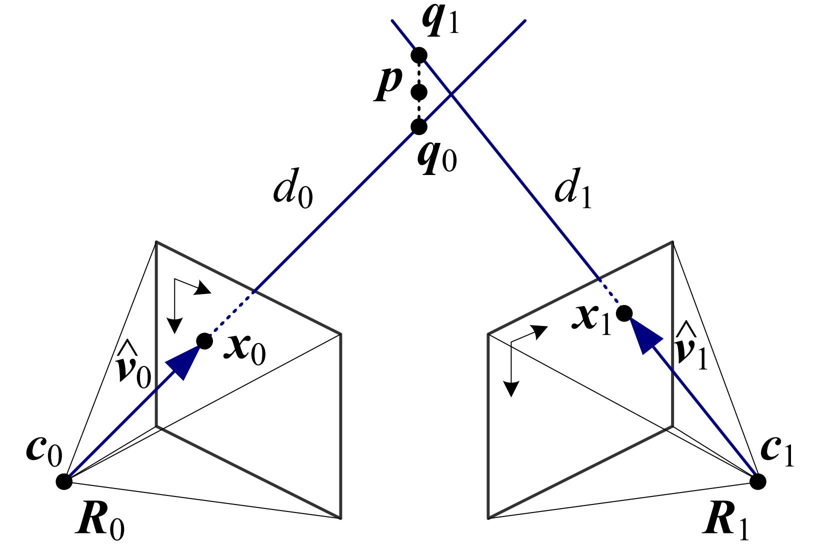
  where $\left(\boldsymbol{p} - \boldsymbol{c}_j\right)_\|$ is the projection of $\boldsymbol{p} - \boldsymbol{c}_j$ onto $\hat{\boldsymbol{v}}_j$.

- and the squared deviation between $\boldsymbol{p}$ and $\boldsymbol{q}_j$ is

  $$r_j^2 = \|(\boldsymbol{I} - \hat{\boldsymbol{v}}_j \hat{\boldsymbol{v}}_j^T)(\boldsymbol{p} - \boldsymbol{c}_j)\|^2 = \|(\boldsymbol{p} - \boldsymbol{c}_j)_\perp\|^2.$$

- Minimizing the sum of squares over all cameras yields

  $$\boldsymbol{p} = \left[\sum_j (\boldsymbol{I} - \hat{\boldsymbol{v}}_j \hat{\boldsymbol{v}}_j^T)\right]^{-1} \left[\sum_j (\boldsymbol{I} - \hat{\boldsymbol{v}}_j \hat{\boldsymbol{v}}_j^T)\boldsymbol{c}_j\right]$$



End of Lecture
Nov 28, 2018

2D Deviations

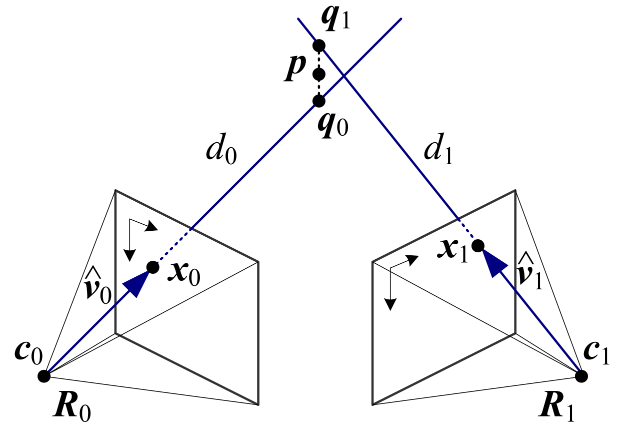
$$r_j^2 = \|(\boldsymbol{I} - \hat{\boldsymbol{v}}_j \hat{\boldsymbol{v}}_j^T)(\boldsymbol{p} - \boldsymbol{c}_j)\|^2 = \|(\boldsymbol{p} - \boldsymbol{c}_j)_\perp\|^2.$$

- Note that this solution minimizes deviation in 3D space, whereas the primary error is introduced by mislocalization of the 2D points $\boldsymbol{x}_j$ in the images.

- If this image localization error is modelled as zero-mean iid Gaussian, it is optimal to minimize the residual between the image points and the reprojections of the estimated 3D points, given by

$$x_j = \frac{p_{00}^{(j)} X + p_{01}^{(j)} Y + p_{02}^{(j)} Z + p_{03}^{(j)} W}{p_{20}^{(j)} X + p_{21}^{(j)} Y + p_{22}^{(j)} Z + p_{23}^{(j)} W}$$

$$y_j = \frac{p_{10}^{(j)} X + p_{11}^{(j)} Y + p_{12}^{(j)} Z + p_{13}^{(j)} W}{p_{20}^{(j)} X + p_{21}^{(j)} Y + p_{22}^{(j)} Z + p_{23}^{(j)} W}$$



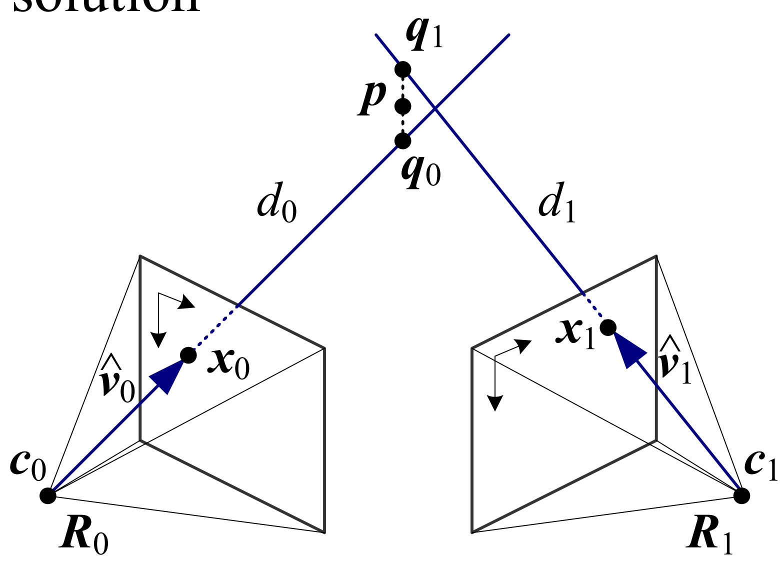
- where the p_{ij} are the parameters of the known projection matrices.

Homogenous Solution

$$x_j = \frac{p_{00}^{(j)}X + p_{01}^{(j)}Y + p_{02}^{(j)}Z + p_{03}^{(j)}W}{p_{20}^{(j)}X + p_{21}^{(j)}Y + p_{22}^{(j)}Z + p_{23}^{(j)}W}$$

$$y_j = \frac{p_{10}^{(j)}X + p_{11}^{(j)}Y + p_{12}^{(j)}Z + p_{13}^{(j)}W}{p_{20}^{(j)}X + p_{21}^{(j)}Y + p_{22}^{(j)}Z + p_{23}^{(j)}W}$$

- Note that we have used homogeneous coordinates for the 3D point here: we seek to estimate X, Y, Z, W.
- Multiplying through by the denominator, this becomes a homogeneous problem, solvable through our two-stage method:
  - DLT: Use SVD to obtain a linear algebraic solution as an initial guess
  - Non-linear least squares: Iterative minimization of squared reprojection error using Levenberg-Marquardt to obtain a maximum likelihood solution

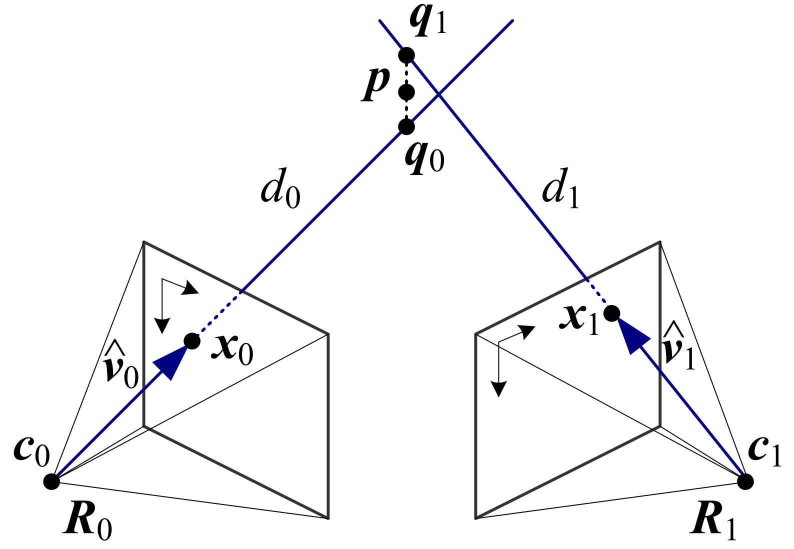


Inhomogeneous Solution

$$x_j = \frac{p_{00}^{(j)} X + p_{01}^{(j)} Y + p_{02}^{(j)} Z + p_{03}^{(j)} W}{p_{20}^{(j)} X + p_{21}^{(j)} Y + p_{22}^{(j)} Z + p_{23}^{(j)} W}$$

$$y_j = \frac{p_{10}^{(j)} X + p_{11}^{(j)} Y + p_{12}^{(j)} Z + p_{13}^{(j)} W}{p_{20}^{(j)} X + p_{21}^{(j)} Y + p_{22}^{(j)} Z + p_{23}^{(j)} W}$$

- We could instead have used augmented coordinates for the 3D world point (W = 1), thus obtaining a regular linear least squares problem ($\boldsymbol{Ap} = \boldsymbol{b}$).
- However this system becomes poorly conditioned for distant objects.



Outline

- Triangulation

- **Two-Frame Structure from Motion**

Structure from Motion (SLAM)

- Pose Estimation and Geometric Camera Calibration:
  - Given *known* 3D scene points and 2D correspondences in *one* image, compute the camera pose and intrinsic parameters.
- Triangulation:
  - Given 2D correspondences over *multiple* images and known camera pose, compute the *unknown* 3D scene points
- Structure from Motion, aka Simultaneous Localization & Mapping (SLAM):
  - Given 2D correspondences over *multiple* images, compute the *unknown* 3D scene points *and unknown camera pose (motion)*

Two-Frame Structure from Motion

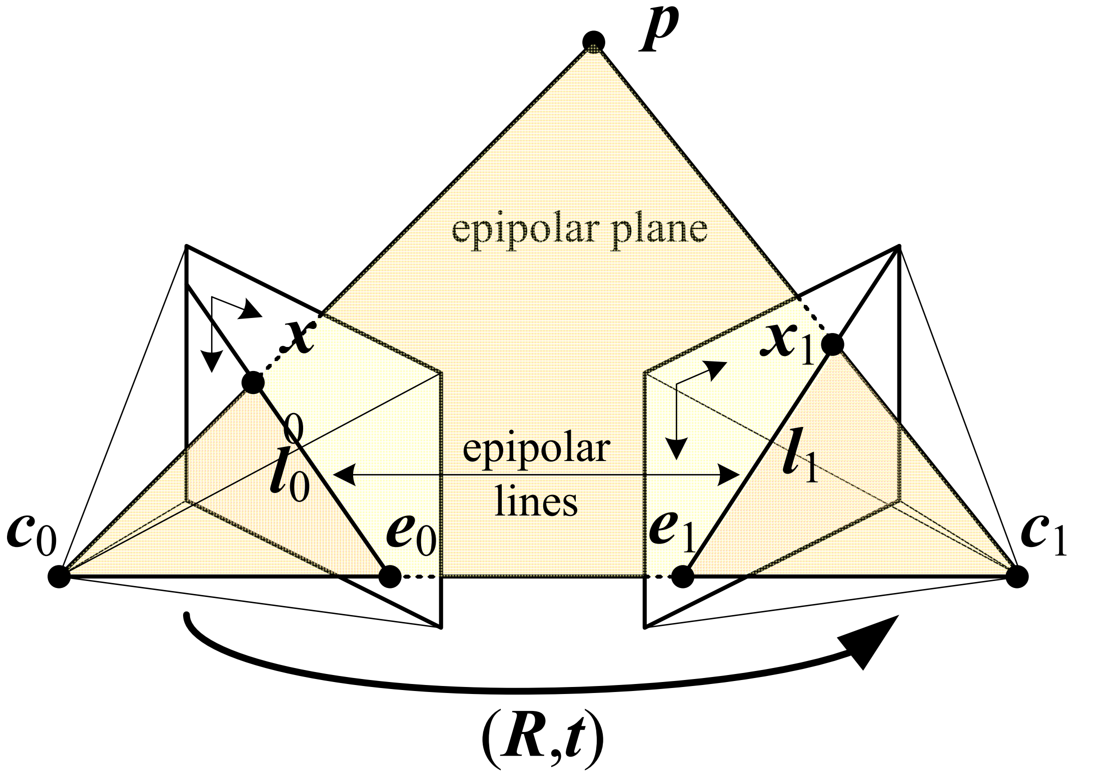
- Consider a point $\boldsymbol{p}$ seen from two cameras (Camera 0 and Camera 1), related by a rigid transformation $(\boldsymbol{R}, \boldsymbol{t})$.

- wlog, we can set $\boldsymbol{c}_0 = 0$ and $\boldsymbol{R}_0 = \boldsymbol{I}$.

- In other words, we align the world frame with Camera 0.

Let $\boldsymbol{p}_0 = d_0\hat{\boldsymbol{x}}_0$ and $\boldsymbol{p}_1 = d_1\hat{\boldsymbol{x}}_1$ represent the location of 3D world point $\boldsymbol{p}$ in the coordinate systems of Camera 0 and 1, respectively.

Here $\hat{\boldsymbol{x}}_0 = \boldsymbol{K}^{-1}\boldsymbol{x}_0$ and $\hat{\boldsymbol{x}}_1 = \boldsymbol{K}^{-1}\boldsymbol{x}_1$ are the ray direction vectors in their respective camera coordinate systems.



The Epipolar Constraint

- Then we have that

  $$d_1 \hat{\boldsymbol{x}}_1 = \boldsymbol{p}_1 = \boldsymbol{R}\boldsymbol{p}_0 + \boldsymbol{t} = \boldsymbol{R}(d_0 \hat{\boldsymbol{x}}_0) + \boldsymbol{t}.$$

  Taking the cross-product of both sides with $\boldsymbol{t}$ yields

  $$d_1 [\boldsymbol{t}]_\times \hat{\boldsymbol{x}}_1 = d_0 [\boldsymbol{t}]_\times \boldsymbol{R} \hat{\boldsymbol{x}}_0$$

  Now taking the dot-product of both sides with $\hat{\boldsymbol{x}}_1$ yields

  $$d_0 \hat{\boldsymbol{x}}_1^T ([\boldsymbol{t}]_\times \boldsymbol{R}) \hat{\boldsymbol{x}}_0 = d_1 \hat{\boldsymbol{x}}_1^T [\boldsymbol{t}]_\times \hat{\boldsymbol{x}}_1 = 0.$$

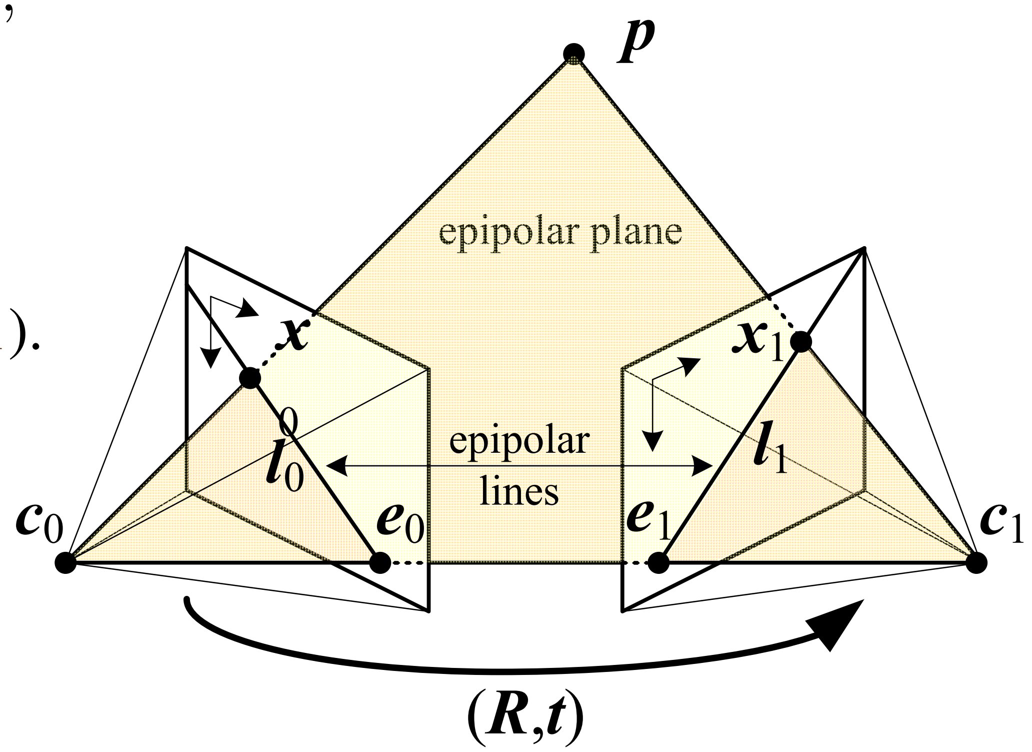
  We therefore arrive at the basic *epipolar constraint*

  $$\hat{\boldsymbol{x}}_1^T \boldsymbol{E} \, \hat{\boldsymbol{x}}_0 = 0,$$

  where

  $$\boldsymbol{E} = [\boldsymbol{t}]_\times \boldsymbol{R}$$

  is called the *essential matrix* (Longuet-Higgins 1981).



The Epipolar Constraint

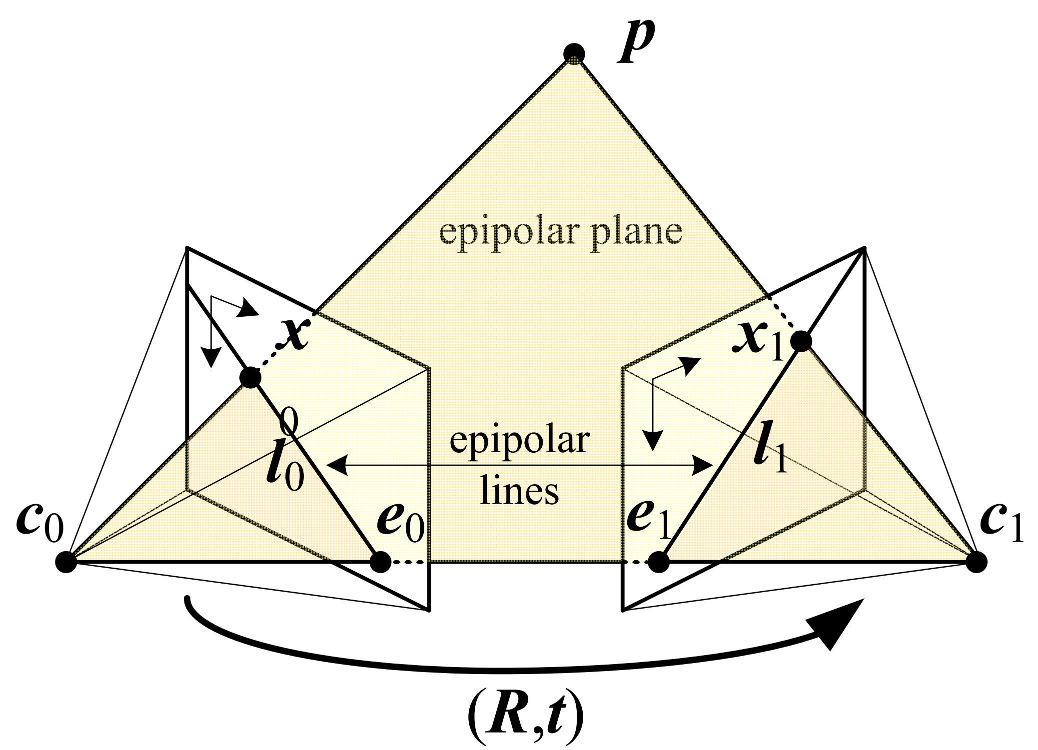
- Perhaps more intuitively, note that the vector connecting the camera centres and the rays connecting the camera centres to the observed 3D point $\boldsymbol{p}$ must be coplanar.

$$\boldsymbol{t}_j = -\boldsymbol{R}_j \boldsymbol{c}_j \rightarrow \boldsymbol{c}_j = -\boldsymbol{R}_j^{-1} \boldsymbol{t}_j$$

$$\text{Thus } \boldsymbol{c}_1 - \boldsymbol{c}_0 = -\boldsymbol{R}_1^{-1}\boldsymbol{t}_1 = -\boldsymbol{R}^{-1}\boldsymbol{t}$$

- For these three vectors to be coplanar, their triple product must be zero:

$$\begin{aligned}
&\left(\boldsymbol{c}_1 - \boldsymbol{c}_0\right)\cdot\left(\left(\boldsymbol{R}_1^{-1}\hat{\boldsymbol{x}}_1\right)\times\left(\boldsymbol{R}_0^{-1}\hat{\boldsymbol{x}}_0\right)\right)\\
&= -\left(\boldsymbol{R}^{-1}\boldsymbol{t}\right)\cdot\left(\left(\boldsymbol{R}^{-1}\hat{\boldsymbol{x}}_1\right)\times\hat{\boldsymbol{x}}_0\right)\\
&= -\boldsymbol{t}\cdot\left(\hat{\boldsymbol{x}}_1\times\boldsymbol{R}\hat{\boldsymbol{x}}_0\right)\\
&= \hat{\boldsymbol{x}}_1\cdot\left(\boldsymbol{t}\times\boldsymbol{R}\hat{\boldsymbol{x}}_0\right)\\
&= \hat{\boldsymbol{x}}_1^{\top}\left(\left[\boldsymbol{t}\right]_{\times}\boldsymbol{R}\right)\hat{\boldsymbol{x}}_0 = 0
\end{aligned}$$



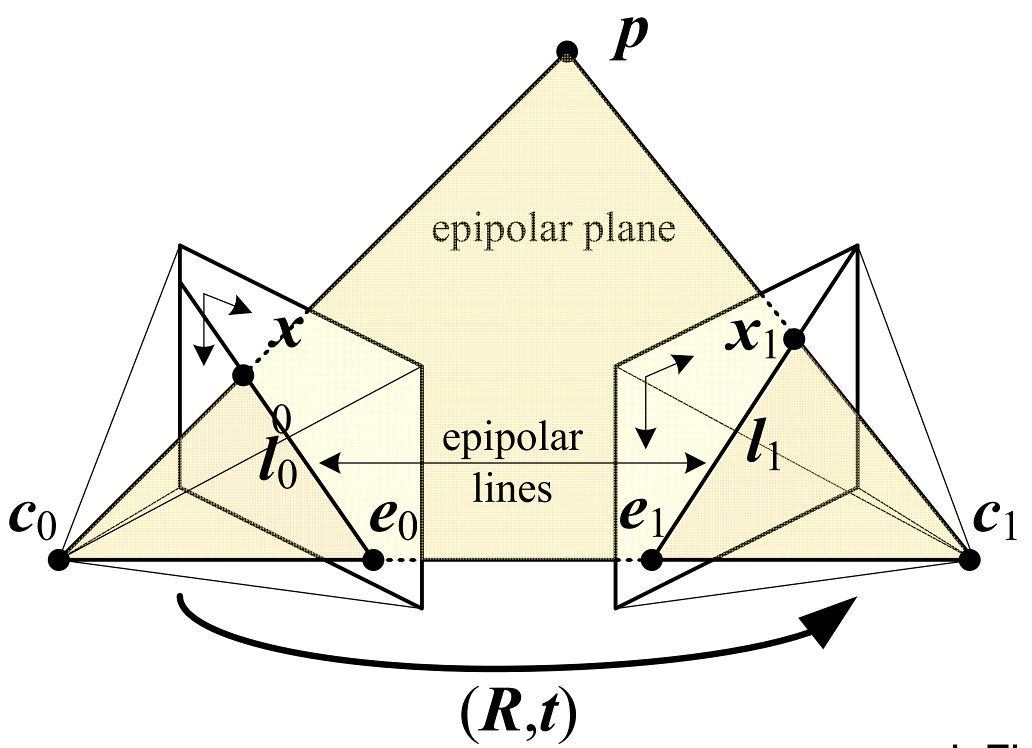
Epipolar Lines

$$\hat{\boldsymbol{x}}_1^\top \boldsymbol{E} \hat{\boldsymbol{x}}_0 = 0$$

The essential matrix $\boldsymbol{E}$ maps a point $\hat{\boldsymbol{x}}_0$ in Image 0 to a line $\boldsymbol{l}_1 = \boldsymbol{E}\hat{\boldsymbol{x}}_0$ in Image 1, since $\hat{\boldsymbol{x}}_1^\top \boldsymbol{l}_1 = 0$.

By taking the transpose, we obtain a similar line $\boldsymbol{l}_0 = \boldsymbol{E}^\top \hat{\boldsymbol{x}}_1$ in Image 0.

- These are the ***epipolar lines***, defining the 1D subspaces in which correspondences must lie.
- Note that $\boldsymbol{l}_1$ contain a point $\boldsymbol{e}_1$ which is the projection of $\boldsymbol{c}_0$ onto Image 1.
- Similarly, $\boldsymbol{l}_0$ contain a point $\boldsymbol{e}_0$ which is the projection of $\boldsymbol{c}_1$ onto Image 0.
- These are the ***epipoles***.



Estimating the Essential Matrix

$\hat{\boldsymbol{x}}_1^\top \boldsymbol{E} \hat{\boldsymbol{x}}_0 = 0 \rightarrow \bar{\boldsymbol{x}}_1^\top \boldsymbol{E} \bar{\boldsymbol{x}}_0 = 0$

where $\bar{\boldsymbol{x}}_1$ and $\bar{\boldsymbol{x}}_2$ are the augmented representations of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$.

- Thus each pair of corresponding image measurements in Image 0 and Image 1 generates a homogenous equation in the elements of $\boldsymbol{E}$:

$$
\begin{array}{rcrcrcl}
x_{i0}x_{i1}e_{00} & + & y_{i0}x_{i1}e_{01} & + & x_{i1}e_{02} & + & \\
x_{i0}y_{i1}e_{00} & + & y_{i0}y_{i1}e_{11} & + & y_{i1}e_{12} & + & \\
x_{i0}e_{20} & + & y_{i0}e_{21} & + & e_{22} & = & 0
\end{array}
$$

- Given at least 8 pairs of corresponding points, we can estimate $\boldsymbol{E}$ (up to a scale factor) using SVD.
- Generally, >8 pairs of points will lead to more accurate results due to noise averaging.
- However, some of these terms will generally be overweighted, particularly the bilinear terms, where one or both of the coordinates is large.
- Can reduce this effect by applying linear transforms $\boldsymbol{T}_0$ and $\boldsymbol{T}_1$ to shift and scale points to have zero mean and unit variance:

$$\tilde{\boldsymbol{x}}_{i0} = \boldsymbol{T}_0 \hat{\boldsymbol{x}}_{i0} \text{ and } \tilde{\boldsymbol{x}}_{i1} = \boldsymbol{T}_1 \hat{\boldsymbol{x}}_{i1} \text{ such that } \mathbb{E}\left[\tilde{\boldsymbol{x}}_{ij}\right] = \boldsymbol{0} \text{ and } \mathbb{E}\left[x_{ij}^2\right] + \mathbb{E}\left[y_{ij}^2\right] = 2$$

Now after solving for the essential matrix $\tilde{\boldsymbol{E}}$ corresponding to these transformed points,

we can recover the essential matrix $\boldsymbol{E}$ for the original points: $\boldsymbol{E} = \boldsymbol{T}_1^\top \tilde{\boldsymbol{E}} \boldsymbol{T}_0$

Estimating the Translation

$$\boldsymbol{E} = \left(\left[\boldsymbol{t} \right]_{\times} \boldsymbol{R} \right)$$

- The absolute distance between the two cameras can never be recovered from image measurements alone.

- However, we *can* recover the direction $\hat{\boldsymbol{t}}$ of the translation.

- Observe that the essential matrix is singular:

$$\boldsymbol{t}^{\top} \boldsymbol{E} = 0$$

Thus $\hat{\boldsymbol{t}}$ is the last column of the $\boldsymbol{U}$ matrix in an SVD decomposition of $\boldsymbol{E}$:

$$\boldsymbol{E} = \boldsymbol{U} \Sigma \boldsymbol{V}^{\top}$$

Estimating the Rotation

Recall that the cross-product operator $[\hat{\boldsymbol{t}}]_\times$ (2.32) projects a vector onto a set of orthogonal basis vectors that include $\hat{\boldsymbol{t}}$, zeros out the $\hat{\boldsymbol{t}}$ component, and rotates the other two by 90°,

$$[\hat{\boldsymbol{t}}]_\times = \boldsymbol{S}\boldsymbol{Z}\boldsymbol{R}_{90^\circ}\boldsymbol{S}^T = \begin{bmatrix} \boldsymbol{s}_0 & \boldsymbol{s}_1 & \hat{\boldsymbol{t}} \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \\ & & 0 \end{bmatrix} \begin{bmatrix} 0 & -1 & \\ 1 & 0 & \\ & & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{s}_0^T \\ \boldsymbol{s}_1^T \\ \hat{\boldsymbol{t}}^T \end{bmatrix}, \quad (7.21)$$

where $\hat{\boldsymbol{t}} = \boldsymbol{s}_0 \times \boldsymbol{s}_1$

- Using this expression together with an SVD decomposition of the essential matrix $\boldsymbol{E}$ yields $\boldsymbol{E} = [\hat{\boldsymbol{t}}]_\times \boldsymbol{R} = \boldsymbol{S}\boldsymbol{Z}\boldsymbol{R}_{90^\circ}\boldsymbol{S}^T\boldsymbol{R} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$
- from which we can conclude that $\boldsymbol{S} = \boldsymbol{U}$.

  Since $\boldsymbol{E}$ is singular but in general of Rank 2, $\Sigma = \boldsymbol{Z}$, and thus

  $\boldsymbol{R}_{90^\circ}\boldsymbol{U}^T\boldsymbol{R} = \boldsymbol{V}^T \longrightarrow \boldsymbol{R} = \boldsymbol{U}\boldsymbol{R}_{90^\circ}^T\boldsymbol{V}^T$

- We only know $\boldsymbol{E}$ and $\boldsymbol{t}$ up to a sign.
- Thus we have to consider 4 possible candidates for $\boldsymbol{R}$ given by:

  $$\boldsymbol{R} = \pm\boldsymbol{U}\boldsymbol{R}_{\pm 90^\circ}^T\boldsymbol{V}^T$$

Chirality

$$\boldsymbol{R} = \pm \boldsymbol{U} \boldsymbol{R}^T_{\pm 90^\circ} \boldsymbol{V}^T$$

- First we can restrict our attention to the two solutions (*chiralities*) for which $|\boldsymbol{R}| = 1$ (and thus for which $\boldsymbol{R}$ represents a valid rotation).
- To select between these remaining two solutions, we pair with the two possible translation vectors $\pm\boldsymbol{t}$, and use triangulation to reconstruct the 3D locations of the points given the hypothesized rotation and translation.
- Now we select the hypothesized $(\boldsymbol{R}, \boldsymbol{t})$ pair that generates the largest number of 3D points lying in front of both cameras.

Building Rome in a Day

Agarwal et al, 2009

Outline

- Triangulation
- Two-Frame Structure from Motion