Two-view geometry
Epipolar geometry

- **Baseline** – line connecting the two camera centers
- **Epipolar Plane** – plane containing baseline (1D family)
- **Epipoles**
  - intersections of baseline with image planes
  - projections of the other camera center
  - vanishing points of camera motion direction
- **Epipolar Lines** - intersections of epipolar plane with image planes (always come in corresponding pairs)
Example: Converging cameras
Example: Motion parallel to image plane
Example: Forward motion

Epipole has same coordinates in both images. Points move along lines radiating from e: “Focus of expansion”
Epipolar constraint

- If we observe a point \( x \) in one image, where can the corresponding point \( x' \) be in the other image?
Epipolar constraint

- Potential matches for $x$ have to lie on the corresponding epipolar line $l'$.

- Potential matches for $x'$ have to lie on the corresponding epipolar line $l$. 
Epipolar constraint example

[Image of epipolar constraint example with green lines]
Epipolar constraint: Calibrated case

- Assume that the intrinsic and extrinsic parameters of the cameras are known
- We can multiply the projection matrix of each camera (and the image points) by the inverse of the calibration matrix to get normalized image coordinates
- We can also set the global coordinate system to the coordinate system of the first camera
Epipolar constraint: Calibrated case

Camera matrix: $[I|0]$

$X = (u, v, w, 1)^T$

$x = (u, v, w)^T$

Camera matrix: $[R^T| -R^T t]$

Vector $x'$ in second coord. system has coordinates $Rx'$ in the first one

The vectors $x$, $t$, and $Rx'$ are coplanar
Epipolar constraint: Calibrated case

The vectors $x$, $t$, and $Rx'$ are coplanar
Epipolar constraint: Calibrated case

\[ x \cdot [t \times (Rx')] = 0 \quad \iff \quad x^T E x' = 0 \quad \text{with} \quad E = [t_x]R \]

- \( E x' \) is the epipolar line associated with \( x' \) (\( l = E x' \))
- \( E^T x \) is the epipolar line associated with \( x \) (\( l' = E^T x \))
- \( E e' = 0 \) and \( E^T e = 0 \)
- \( E \) is singular (rank two)
- \( E \) has five degrees of freedom
Epipolar constraint: Uncalibrated case

- The calibration matrices $K$ and $K'$ of the two cameras are unknown.
- We can write the epipolar constraint in terms of unknown normalized coordinates:

$$\hat{x}^T E \hat{x}' = 0$$

$$x = K \hat{x}, \quad x' = K' \hat{x}'$$
Epipolar constraint: Uncalibrated case

\[ \hat{x}^T E \hat{x}' = 0 \quad \Rightarrow \quad x^T F x' = 0 \quad \text{with} \quad F = K^{-T} E K'^{-1} \]

\[ x = K \hat{x} \]

\[ x' = K' \hat{x}' \]

**Fundamental Matrix**
(Faugeras and Luong, 1992)
Epipolar constraint: Uncalibrated case

\[ \hat{x}^T E \hat{x}' = 0 \quad \Rightarrow \quad x^T F x' = 0 \quad \text{with} \quad F = K^{-T} E K'^{-1} \]

- \( F x' \) is the epipolar line associated with \( x' (l = F x') \)
- \( F^T x \) is the epipolar line associated with \( x (l' = F^T x) \)
- \( F e' = 0 \) and \( F^T e = 0 \)
- \( F \) is singular (rank two)
- \( F \) has seven degrees of freedom
The eight-point algorithm

\[ \mathbf{x} = (u, v, 1)^T, \quad \mathbf{x}' = (u', v', 1)^T \]

\[
\begin{pmatrix}
F_{11} & F_{12} & F_{13} \\
F_{21} & F_{22} & F_{23} \\
F_{31} & F_{32} & F_{33}
\end{pmatrix}
\begin{pmatrix}
u' \\
v'
\end{pmatrix} = 0
\]

Minimize:

\[ \sum_{i=1}^{N} (\mathbf{x}_i^T F \mathbf{x}_i')^2 \]

under the constraint

\[ |F|^2 = 1 \]
The eight-point algorithm

- Meaning of error \( \sum_{i=1}^{N} (x_i^T F x'_i)^2 \):
  
  sum of Euclidean distances between points \( x_i \) and epipolar lines \( Fx'_i \) (or points \( x'_i \) and epipolar lines \( F^T x_i \)) multiplied by a scale factor

- Nonlinear approach: minimize

\[
\sum_{i=1}^{N} \left[ d(x_i, F x'_i) + d(x'_i, F^T x_i) \right]
\]
Problem with eight-point algorithm

\[
\begin{pmatrix}
u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 \\
u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 \\
u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 \\
u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 \\
u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 \\
u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 \\
u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 \\
u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8
\end{pmatrix}
\begin{pmatrix}
F_{11} \\
F_{12} \\
F_{13} \\
F_{21} \\
F_{22} \\
F_{23} \\
F_{31} \\
F_{32}
\end{pmatrix} =
\begin{pmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{pmatrix}
\]
Problem with eight-point algorithm

<table>
<thead>
<tr>
<th></th>
<th>250906.36</th>
<th>183269.57</th>
<th>921.81</th>
<th>200931.10</th>
<th>146766.13</th>
<th>738.21</th>
<th>272.19</th>
<th>198.81</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2692.28</td>
<td>131633.03</td>
<td>176.27</td>
<td>6196.73</td>
<td>302975.59</td>
<td>405.71</td>
<td>15.27</td>
<td>746.79</td>
</tr>
<tr>
<td></td>
<td>416374.23</td>
<td>871684.30</td>
<td>935.47</td>
<td>408110.89</td>
<td>854384.92</td>
<td>916.90</td>
<td>445.10</td>
<td>931.81</td>
</tr>
<tr>
<td></td>
<td>191183.60</td>
<td>171759.40</td>
<td>410.27</td>
<td>416435.62</td>
<td>374125.90</td>
<td>893.65</td>
<td>465.99</td>
<td>418.65</td>
</tr>
<tr>
<td></td>
<td>48988.86</td>
<td>30401.76</td>
<td>57.89</td>
<td>298604.57</td>
<td>185309.58</td>
<td>352.87</td>
<td>846.22</td>
<td>525.15</td>
</tr>
<tr>
<td></td>
<td>164786.04</td>
<td>546559.67</td>
<td>813.17</td>
<td>1998.37</td>
<td>6628.15</td>
<td>9.86</td>
<td>202.65</td>
<td>672.14</td>
</tr>
<tr>
<td></td>
<td>116407.01</td>
<td>2727.75</td>
<td>138.89</td>
<td>169941.27</td>
<td>3982.21</td>
<td>202.77</td>
<td>838.12</td>
<td>19.64</td>
</tr>
<tr>
<td></td>
<td>135384.58</td>
<td>75411.13</td>
<td>198.72</td>
<td>411350.03</td>
<td>229127.78</td>
<td>603.79</td>
<td>681.28</td>
<td>379.48</td>
</tr>
</tbody>
</table>

\[
\begin{pmatrix}
F_{11} \\
F_{12} \\
F_{13} \\
F_{21} \\
F_{22} \\
F_{23} \\
F_{31} \\
F_{32}
\end{pmatrix} = -
\begin{pmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{pmatrix}
\]

Poor numerical conditioning
Can be fixed by rescaling the data
The normalized eight-point algorithm

(Hartley, 1995)

• Center the image data at the origin, and scale it so the mean squared distance between the origin and the data points is 2 pixels
• Use the eight-point algorithm to compute $F$ from the normalized points
• Enforce the rank-2 constraint (for example, take SVD of $F$ and throw out the smallest singular value)
• Transform fundamental matrix back to original units: if $T$ and $T'$ are the normalizing transformations in the two images, than the fundamental matrix in original coordinates is $T^T F T'$
Comparison of estimation algorithms

<table>
<thead>
<tr>
<th></th>
<th>8-point</th>
<th>Normalized 8-point</th>
<th>Nonlinear least squares</th>
</tr>
</thead>
<tbody>
<tr>
<td>Av. Dist. 1</td>
<td>2.33 pixels</td>
<td>0.92 pixel</td>
<td>0.86 pixel</td>
</tr>
<tr>
<td>Av. Dist. 2</td>
<td>2.18 pixels</td>
<td>0.85 pixel</td>
<td>0.80 pixel</td>
</tr>
</tbody>
</table>
From epipolar geometry to camera calibration

- Estimating the fundamental matrix is known as “weak calibration”
- If we know the calibration matrices of the two cameras, we can estimate the essential matrix: $E = K^T F K'$
- The essential matrix gives us the relative rotation and translation between the cameras, or their extrinsic parameters
Assignment 3 (due March 17)

http://www.cs.unc.edu/~lazebnik/spring09/assignment3.html