Estimation of Fundamental Matrix, Linear

Computer Vision II
CSE 252B
Lecture 12
Announcements

• Assignment 3 is due today, 11:59 PM
• Assignment 4 will be released today
  – Due Mar 6, 11:59 PM
• Reading
  – Sections 9.1, 9.2, 11.1, 12.4, 11.2, and 11.3
Two view geometry
Two view geometry

• Rotation about the same camera center
Two view geometry

• Imaging a plane
Two view geometry

• Imaging a 3D scene
Imaging geometry models

• Single view
  – Uncalibrated: camera projection matrix
  – Calibrated: normalized camera projection matrix

• Two views
  – Rotation about the same camera center
    • Uncalibrated: 2D projective transformation matrix
    • Calibrated: 3D rotation matrix
  – Imaging a plane
    • Uncalibrated: 2D projective transformation matrix
    • Calibrated: 2D projective transformation matrix
  – Imaging a 3D scene
    • Uncalibrated: fundamental matrix
Fundamental matrix

Given the canonical cameras $P = [I \mid 0]$ and $P' = [M \mid v]$, backproject the point $x = (x, y, w)^T$ in the first image to a line in 3D.

First point on the 3D line

$$C = \begin{bmatrix} \tilde{C} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Center of first camera

Second point on the 3D line

$$X = P^+X = \begin{bmatrix} I \\ 0^T \end{bmatrix} x = \begin{bmatrix} x \\ 0 \end{bmatrix},$$

where $P^+ = P^T(PP^T)^{-1} = \begin{bmatrix} I \\ 0^T \end{bmatrix}$ is the pseudoinverse of $P$.

3D line $L$ joining the two 3D points

$$L = XC^T - CX^T = \begin{bmatrix} 0 & x \\ -x^T & 0 \end{bmatrix}$$

The projection of the 3D line $L$ to a 2D line $l'$ in the second image is given by

$$[l']_x = P'LP'^T = \begin{bmatrix} M & v \end{bmatrix} \begin{bmatrix} 0 & x \\ -x^T & 0 \end{bmatrix} \begin{bmatrix} M^T \\ v^T \end{bmatrix}$$

$$l' = [v]_xMx = Fx,$$

where the fundamental matrix $F = [v]_xM$.

$3 \times 3$ skew symmetric matrices are rank 2, so $F$ is rank.
Fundamental matrix

\[ P = \begin{bmatrix} I & 0 \end{bmatrix} \]

\[ P' = \begin{bmatrix} M & v \end{bmatrix} \]

\[ F = v \times M \]
Fundamental matrix

Epipolar lines

Epipoles
Two view geometry

• All epipolar lines pass through the epipole

\[ F \text{ is a rank 2 homogeneous matrix} \]
Fundamental matrix

\[ \ell' = Fx \quad \text{and} \quad x'^\top \ell' = 0 \]
\[ x'^\top Fx = 0 \quad \text{Epipolar constraint} \]

\[
\begin{align*}
\ell' &= Fx & \text{The epipolar line corresponding to the point } x \\
\ell &= F^\top x' & \text{The epipolar line corresponding to the point } x' \\
Fe &= 0 & \text{The epipole } e \text{ is the null space of } F \\
e'^\top F = 0^\top & \text{The epipole } e' \text{ is the left null space of } F \\
F^\top e' &= 0 & \text{The epipole } e' \text{ is the null space of } F^\top
\end{align*}
\]

The fundamental matrix \( F \) from general cameras \( P \) and \( P' \)

\[ F = [e']_\times P'P^+, \text{ where } e' = P'C \text{ and } PC = 0 \quad (\text{i.e., } C \text{ is the null space of } P) \]

The fundamental matrix \( F \) from canonical cameras \( P = [I \mid 0] \) and \( P' = [M \mid v] \)

\[ F = [v]_\times M \]
\[ F = [e']_\times M = M^{-\top} [e]_\times, \text{ where } e' = v \text{ and } e = M^{-1}v \]
Fundamental matrix

• Error in image point measurements
Estimation of fundamental matrix

\[ x_i = P X_i \forall i \text{ and } x'_i = P' X_i \forall i, \text{ set } P = [I \mid 0] \text{ and solve for } P' \]

- Given point correspondences \( x_i \leftrightarrow x'_i \)
- Minimize geometric error \( \sum_i \left( d(x_i, PX_i)^2 + d(x'_i, P'X_i)^2 \right) \)

- A nonlinear mapping requires a nonlinear optimization problem solver
  - Use linear estimation for initial estimate (to mitigate converging to local optimum)
  - Iterative process to determine global optimum
Estimation of fundamental matrix, linear
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

- Minimize algebraic error

\[ x'^T F x = 0 \quad \text{Epipolar constraint} \]
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

\[ x'^{\top}Fx = 0 \]

\[ x'^{\top} \begin{bmatrix} f_1^{\top} \\ f_2^{\top} \\ f_3^{\top} \end{bmatrix} x = 0, \text{ where } F = \begin{bmatrix} f_1^{\top} \\ f_2^{\top} \\ f_3^{\top} \end{bmatrix} \]

\[ \begin{bmatrix} x' & y' & w' \end{bmatrix} \begin{bmatrix} f_1^{\top} x \\ f_2^{\top} x \\ f_3^{\top} x \end{bmatrix} = 0 \]

\[ x'f_1^{\top}x + y'f_2^{\top}x + w'f_3^{\top}x = 0 \]

\[ \begin{bmatrix} x'x^{\top} & y'x^{\top} & w'x^{\top} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = 0 \]

\[ (x'^{\top} \otimes x^{\top}) f = 0, \text{ where } \otimes \text{ denotes the Kronecker product} \]

and \( f = \text{vec}(F^{\top}) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \)
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

Given $n \geq 8$ point correspondences $x_i \leftrightarrow x_i'$

$$x_i'^T F x_i = 0 \ \forall \ i,$$

solve for $F$

$$(x_i'^T \otimes x_i^T) f = 0 \ \forall \ i,$$

solve for $f = \text{vec}(F^T)$

$$\begin{bmatrix}
    x_1'^T \otimes x_1^T \\
    x_2'^T \otimes x_2^T \\
    \vdots \\
    x_n'^T \otimes x_n^T
\end{bmatrix} f = 0$$

$$A f = 0,$$ where $A = \begin{bmatrix}
    x_1'^T \otimes x_1^T \\
    x_2'^T \otimes x_2^T \\
    \vdots \\
    x_n'^T \otimes x_n^T
\end{bmatrix}$

Then, enforce rank 2 constraint (simple and convenient method using SVD)

$$F = U \Sigma V^T,$$ where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3)$

$$F = U \Sigma' V^T,$$ where $\Sigma' = \text{diag}(\sigma_1, \sigma_2, 0)$

See textbook for alternative method

But, data normalization must be used
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

Given \( n \geq 9 \) point correspondences \( x_i \leftrightarrow x'_i \)

\[
Af = 0, \text{ where } A = 
\begin{bmatrix}
  x'_1^T \otimes x_1^T \\
  x'_2^T \otimes x_2^T \\
  \vdots \\
  x'_n^T \otimes x_n^T
\end{bmatrix}
\]

In general, due to error in measurements, the (right) null space of the matrix \( A \) will be empty (i.e., \( A \) will be full rank). Instead, solve for the vector \( f \) such that \( \|Af\| \) is minimized.

\[
A = U \Sigma V^T
\]

The solution vector \( f \) is the column of \( V \) corresponding to the smallest singular value (i.e., the last column of \( V \) (or last row of \( V^T \))).

Then, enforce rank 2 constraint (simple and convenient method using SVD)

See textbook for alternative method

But, data normalization must be used
Given \( n = 8 \) point correspondences \( x_i \leftrightarrow x'_i \)

\[
\begin{bmatrix}
x'_1^T \\ x'_2^T \\ x'_3^T \\ x'_4^T \\ x'_5^T \\ x'_6^T \\ x'_7^T \\ x'_8^T
\end{bmatrix} \otimes \begin{bmatrix}
x_1^T \\ x_2^T \\ x_3^T \\ x_4^T \\ x_5^T \\ x_6^T \\ x_7^T \\ x_8^T
\end{bmatrix}
\]

\[ Af = 0, \text{ where } A = \]

The (right) null space of the matrix \( A \) is the vector \( f \).

\[ A = U \Sigma V^T \]

The solution vector \( f \) is the column of \( V \) not corresponding to a singular value (i.e., the last column of \( V \) (or last row of \( V^T \))).

Then, enforce rank 2 constraint (simple and convenient method using SVD)

But, data normalization must be used

See textbook for alternative method
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

Given $n = 7$ point correspondences $x_i \leftrightarrow x'_i$

$$x_i^T F x_i = 0 \forall i = 1, \ldots, 7, \text{ solve for } F$$

$$(x_i^T \otimes x_i^T) f = 0 \forall i = 1, \ldots, 7, \text{ solve for } f = \text{vec}(F^T)$$

$$
\begin{bmatrix}
  x_1^T \otimes x_1^T \\
  x_2^T \otimes x_2^T \\
  x_3^T \otimes x_3^T \\
  x_4^T \otimes x_4^T \\
  x_5^T \otimes x_5^T \\
  x_6^T \otimes x_6^T \\
  x_7^T \otimes x_7^T
\end{bmatrix}
\begin{bmatrix}
  f_{1,1} & f_{2,1} \\
  f_{1,2} & f_{2,2} \\
  f_{1,3} & f_{2,3} \\
  f_{1,4} & f_{2,4} \\
  f_{1,5} & f_{2,5} \\
  f_{1,6} & f_{2,6} \\
  f_{1,7} & f_{2,7} \\
  f_{1,8} & f_{2,8} \\
  f_{1,9} & f_{2,9}
\end{bmatrix}
= 0
$$

$$A \begin{bmatrix} f_1 & f_2 \end{bmatrix} = 0, \text{ where } A =
\begin{bmatrix}
  x_1^T \otimes x_1^T \\
  x_2^T \otimes x_2^T \\
  \vdots \\
  x_7^T \otimes x_7^T
\end{bmatrix},$$

$$f_1 = (f_{1,1}, f_{1,2}, f_{1,3}, f_{1,4}, f_{1,5}, f_{1,6}, f_{1,7}, f_{1,8}, f_{1,9})^T, \text{ and }$$

$$f_2 = (f_{2,1}, f_{2,2}, f_{2,3}, f_{2,4}, f_{2,5}, f_{2,6}, f_{2,7}, f_{2,8}, f_{2,9})^T$$

But, data normalization must be used
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

$$A \begin{bmatrix} f_1 & f_2 \end{bmatrix} = 0,$$

where

$$A = \begin{bmatrix} x'_1 \otimes x'_1 \\ x'_2 \otimes x'_2 \\ x'_3 \otimes x'_3 \\ x'_4 \otimes x'_4 \\ x'_5 \otimes x'_5 \\ x'_6 \otimes x'_6 \\ x'_7 \otimes x'_7 \end{bmatrix}$$

The (right) null space of the matrix $A$ is the matrix $\begin{bmatrix} f_1 & f_2 \end{bmatrix}$.

$$A = U \Sigma V^\top$$

The solution matrix $\begin{bmatrix} f_1 & f_2 \end{bmatrix}$ are the columns of $V$ not corresponding to singular values (i.e., the last two columns of $V$ (or last two rows of $V^\top$)).

But, data normalization must be used.
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

\[ \mathbf{f}_1 = (f_{1,1}, f_{1,2}, f_{1,3}, f_{1,4}, f_{1,5}, f_{1,6}, f_{1,7}, f_{1,8}, f_{1,9})^\top, \text{ and} \]
\[ \mathbf{f}_2 = (f_{2,1}, f_{2,2}, f_{2,3}, f_{2,4}, f_{2,5}, f_{2,6}, f_{2,7}, f_{2,8}, f_{2,9})^\top \]

\[ \mathbf{F}_1 = \begin{bmatrix} f_{1,1} & f_{1,2} & f_{1,3} \\ f_{1,4} & f_{1,5} & f_{1,6} \\ f_{1,7} & f_{1,8} & f_{1,9} \end{bmatrix} \quad \text{and} \quad \mathbf{F}_2 = \begin{bmatrix} f_{2,1} & f_{2,2} & f_{2,3} \\ f_{2,4} & f_{2,5} & f_{2,6} \\ f_{2,7} & f_{2,8} & f_{2,9} \end{bmatrix} \]

\[ \mathbf{F} = \alpha \mathbf{F}_1 + \mathbf{F}_2 \]
\[ \det(\mathbf{F}) = 0, \text{ solve for } \alpha \]

3 solutions \( \alpha_1, \alpha_2, \) and \( \alpha_3 \) (1 or 3 will be real solutions, i.e., the real roots of the cubic polynomial)
Then substitute \( \alpha_i \) to solve for \( \mathbf{F}_i \)

\[ \mathbf{F}_i = \alpha_i \mathbf{F}_1 + \mathbf{F}_2 \quad (1 \text{ or } 3 \text{ real solutions}) \]

But, data normalization must be used
Data normalization

• Data normalization is required to reduce the propagated uncertainty under the nonlinear projection.

![Diagram showing measured point mapped to line with and without data normalization uncertainly.](image)
Data normalizing transformations

• 2D

Determine the similarity transformation $T$ such that the mean (i.e., centroid) of the transformed points is at the origin and their standard deviation from the origin is $\sqrt{2}$.

$$T = \begin{bmatrix} s & 0 & -s\mu_{\tilde{x}} \\ 0 & s & -s\mu_{\tilde{y}} \\ 0 & 0 & 1 \end{bmatrix}, \text{ where } s = \sqrt{\frac{2}{\sigma_{\tilde{x}}^2 + \sigma_{\tilde{y}}^2}}$$

where $\mu_{\tilde{x}}$ and $\sigma_{\tilde{x}}^2$, and $\mu_{\tilde{y}}$ and $\sigma_{\tilde{y}}^2$ are the mean and variance of the $\tilde{x}$ and $\tilde{y}$ coordinates, respectively.
Linear estimation of fundamental matrix using the direct linear transformation (DLT) algorithm

• Algorithm

1. Data normalize the points

\[ x_{DN,i} = T x_i \forall i \]
\[ x'_{DN,i} = T' x'_i \forall i \]

2. Estimate the data normalized fundamental matrix \( F_{DN} \) from the data normalized points

3. Data denormalize the data normalized fundamental matrix

\[ x'^{\top}_{DN,i} F_{DN} x_{DN,i} = 0 \forall i \]
\[ (T' x'_i)^{\top} F_{DN} T x_i = 0 \forall i \]
\[ x'_i^{\top} T'^{\top} F_{DN} T x_i = 0 \forall i \]
\[ x'_i^{\top} F x_i = 0 \forall i, \text{ where } F = T'^{\top} F_{DN} T \]
Sampson correction and Sampson error
Fundamental matrix

• Error in image point measurements

measured point $x$

mapped to line $F^T x'$

image 1

measured point $x'$

mapped to line $Fx$

image 2

measured point $x$

projected point $PX$

image 1

measured point $x'$

projected point $P'X$

image 2
Sampson cost

Sampson error uses a cost function that lies between algebraic and geometric cost functions (in terms of complexity) and gives a close approximation to geometric error. Measurement vector $\mathbf{X} = (\tilde{x}, \tilde{y}, \tilde{x}', \tilde{y}')^\top$

$C_F(\mathbf{X})$ is the cost associated with $\mathbf{X}$

$$C_F(\mathbf{X}) = 0$$

$$C_F(\mathbf{X} + \delta_\mathbf{X}) = C_F(\mathbf{X}) + \frac{dC_F(\mathbf{X})}{\partial \mathbf{X}} \delta_\mathbf{X}$$

first order Taylor expansion

if $\hat{\mathbf{X}} = \mathbf{X} + \delta_\mathbf{X}$

$\hat{\mathbf{X}} - \mathbf{X} = \delta_\mathbf{X}$ and desire $\hat{\mathbf{X}}$ such that $C_F(\hat{\mathbf{X}}) = 0$, then

$$C_F(\mathbf{X}) + \frac{dC_F(\mathbf{X})}{\partial \mathbf{X}} \delta_\mathbf{X} = 0$$

$$\epsilon + J\delta_\mathbf{X} = 0, \text{ where } \epsilon = C_F(\mathbf{X}) \text{ and } J = \frac{dC_F(\mathbf{X})}{\partial \mathbf{X}}$$

$$J\delta_\mathbf{X} = -\epsilon, \text{ solve for } \delta_\mathbf{X}$$
Sampson corrected and Sampson error

\[ J \delta_X = -\epsilon, \text{ solve for } \delta_X \]

Use a Lagrange multiplier \( \lambda \) and find extrema of \( \delta_X^T \delta_X - 2\lambda (J \delta_X + \epsilon) \)

\[ JJ^T \lambda = -\epsilon, \text{ solve for } \lambda \]

then substitute \( \lambda \) to solve for \( \delta_X \)

\[ \delta_X = J^T \lambda \]

Sampson corrected

\[ \hat{X} = X + \delta_X \]

Squared sampson error

\[ \|\delta_X\|^2 = \delta_X^T \delta_X \]
Sampson corrected and Sampson error

Determine $J$ and $\epsilon$

$$x_i^TFx_i = 0 \forall i$$

$$x_i^{T} \begin{bmatrix} f_1^T \\ f_2^T \\ f_3^T \end{bmatrix} x_i = 0 \forall i, \text{ where } F = \begin{bmatrix} f_1^T \\ f_2^T \\ f_3^T \end{bmatrix}$$

$$\begin{bmatrix} x_i'x_i^T & y_i'x_i^T & w_i'x_i^T \end{bmatrix} \begin{bmatrix} f_1^T \\ f_2^T \\ f_3^T \end{bmatrix} = 0 \forall i$$

$$A_i f = 0 \forall i, \text{ where } A_i = \begin{bmatrix} x_i'x_i^T & y_i'x_i^T & w_i'x_i^T \end{bmatrix} \text{ and } f = \text{vec}(F^T) = \begin{bmatrix} f_1^T \\ f_2^T \\ f_3^T \end{bmatrix}$$
**Sampson corrected and Sampson error**

$$A_i f = \begin{bmatrix} \tilde{x}' x_i^T & \tilde{y}' x_i^T & x_i^T \end{bmatrix} \begin{bmatrix} f^1 \\ f^2 \\ f^3 \end{bmatrix} \quad \forall i$$

$$A_i f = \tilde{x}' f^{1^T} x_i + \tilde{y}' f^{2^T} x_i + f^{3^T} x_i \quad \forall i$$

$$X_i = (\tilde{x}_i, \tilde{y}_i, \tilde{x}'_i, \tilde{y}'_i)^T$$

$$\epsilon_i = C_F(X_i) = A_i f$$

$$\epsilon_i = \tilde{x}_i \tilde{x}'_i f_{11} + \tilde{x}_i \tilde{y}'_i f_{21} + \tilde{y}_i \tilde{x}'_i f_{12} + \tilde{y}_i \tilde{y}'_i f_{22} + \tilde{y}_i f_{32} + \tilde{x}'_i f_{13} + \tilde{y}'_i f_{23} + f_{33}$$

$$J_i = \frac{dC_F(X_i)}{\partial X_i} = \frac{d(A_i f)}{\partial (\tilde{x}_i, \tilde{y}_i, \tilde{x}'_i, \tilde{y}'_i)}$$

$$J_i = \begin{bmatrix} \tilde{x}' f_{11} + \tilde{y}' f_{21} + f_{31} & \tilde{x}' f_{12} + \tilde{y}' f_{22} + f_{32} & \tilde{x}' f_{13} + \tilde{y} f_{23} + f_{23} \end{bmatrix}$$

$$J_i J_i^T \lambda_i = -\epsilon_i$$, solve for $$\lambda_i = -\frac{\epsilon_i}{J_i J_i^T}$$, then $$\delta X_i = J_i^T \lambda_i$$

**Sampson corrected points**

$$\hat{X}_i = (\hat{x}_i, \hat{y}_i, \hat{x}'_i, \hat{y}'_i)^T = X_i + \delta X_i$$

**Squared Sampson error**

$$\|\delta X_i\|^2 = \delta_{X_i}^T \delta X_i = \epsilon (J J^T)^{-1} \epsilon = \frac{\epsilon^2}{J J^T}$$
Outlier rejection
Outlier rejection

• Even the presence of a single outlier may result in an inaccurate estimate

• Linear estimate
  – Minimizes sum of squared error and, in general, outliers have large error relative to inliers
  – Resulting estimate may not be near global minimum

• Nonlinear estimate
  – Cost function minimizes sum of squared error, so may not converge to accurate solution
    • Alternatively, use a robust cost function
Random sample consensus (RANSAC) and M-estimator sample consensus (MSAC)

Objective is to determine the consensus set with the minimum cost

tol is the tolerance for establishing datum/model compatibility
τ_{cost} is the upper bound on the cost of an acceptable consensus set
maxTrials is the maximum number of attempts to find a consensus set

\[ \text{consensus}_{\text{cost}} = \infty \]
for (trials = 0; trials < maxTrials && consensus_{\text{cost}} > \tau_{\text{cost}}; ++\text{trials})

Select a random sample of unique data points
Calculate the model using the random sample
Calculate the error for each data point using model
Calculate the cost
if cost < consensus_{\text{cost}}
    \[ \text{consensus}_{\text{cost}} = \text{cost} \]
    \[ \text{consensus}_{\text{model}} = \text{model} \]
Calculate the error for each data point using consensus_{model}
Calculate the set of inliers (i.e., data points with error \( \leq \) tol)

If multiple solutions, try each one, but it is still a single trial
Outlier rejection

• **RANSAC cost**

  count is the total number of data points  
  tol is the tolerance for establishing datum/model compatibility

  \[
  \text{cost} = 0 \\
  \text{for} \ (n = 0; \ n < \text{count}; \ ++n) \\
  \quad \text{cost} += \text{error}[n] \leq \text{tol} \ ? \ 0 : 1
  \]

• **MSAC cost**

  count is the total number of data points  
  tol is the tolerance for establishing datum/model compatibility

  \[
  \text{cost} = 0 \\
  \text{for} \ (n = 0; \ n < \text{count}; \ ++n) \\
  \quad \text{cost} += \text{error}[n] \leq \text{tol} \ ? \ \text{error}[n] : \text{tol}
  \]
RANSAC and MSAC

• Maximum number of trials
  – Adaptive maximum number of trials

• The tolerance for establishing datum/model compatibility
RANSAC and MSAC

• Maximum number of trials

The smaller the sample size, the smaller maximum number of trials. Use minimum solution!

<table>
<thead>
<tr>
<th>Sample size</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>11</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>19</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>5</td>
<td>9</td>
<td>13</td>
<td>17</td>
<td>34</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6</td>
<td>12</td>
<td>17</td>
<td>26</td>
<td>57</td>
<td>146</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>7</td>
<td>16</td>
<td>24</td>
<td>37</td>
<td>97</td>
<td>293</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>8</td>
<td>20</td>
<td>33</td>
<td>54</td>
<td>163</td>
<td>588</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>9</td>
<td>26</td>
<td>44</td>
<td>78</td>
<td>272</td>
<td>1177</td>
</tr>
</tbody>
</table>

Table 4.3. The number $N$ of samples required to ensure, with a probability $p = 0.99$, that at least one sample has no outliers for a given size of sample, $s$, and proportion of outliers, $\epsilon$. 
Adaptive maximum number of trials

$s$ is the sample size
$p$ is the assumed probability that at least one of the random samples does not contain any outliers
$\text{tol}$ is the tolerance for establishing datum/model compatibility
$\tau_{\text{cost}}$ is the upper bound on the cost of an acceptable consensus set

$maxTrials = \infty$
$\text{consensus}_{\text{cost}} = \infty$

for (trials = 0; trials < maxTrials && $\text{consensus}_{\text{cost}} > \tau_{\text{cost}}$; ++trials)

Select a random sample of unique data points
Calculate the model using the random sample
Calculate the error for each data point using model
Calculate the cost
if cost < $\text{consensus}_{\text{cost}}$

$\text{consensus}_{\text{cost}} = \text{cost}$
$\text{consensus}_{\text{model}} = \text{model}$
Calculate the number of inliers
$w = \frac{\text{number of inliers}}{\text{total number of data points}}$
$maxTrials = \frac{\log(1-p)}{\log(1-w^s)}$

Calculate the error for each data point using consensus$_{\text{model}}$
Calculate the set of inliers (i.e., data points with error \leq \text{tol})

If multiple solutions, try each one, but it is still a single trial
RANSAC and MSAC

• The tolerance for establishing datum/model compatibility

  Probability $\alpha$ that a data point is an inlier (usually $\alpha$ is chosen to be 0.95)
  Variance $\sigma^2$ of the measurement error (if unknown, assumed to be 1)
  Codimension $m$
  Tolerance is square distance $t^2 = F_m^{-1}(\alpha)\sigma^2$, where $F_m^{-1}(\alpha)$ is the inverse chi-squared cumulative distribution function with $m$ degrees of freedom at the probability $\alpha$

• Codimension

  If $w$ is a subspace of a finite-dimensional vector space $v$, then $\text{codim}(w) = \dim(v) - \dim(w)$
  The fundamental matrix is a variety of dimension 3 in $(\tilde{x}, \tilde{y}, \tilde{x}', \tilde{y}') \in \mathbb{R}^4$, so has codimension 1

Use the square Sampson error
Feature detection and matching

Input Images
Feature detection and matching

Detected Corners
Feature detection and matching

Expect some false matches
Feature detection and matching

Simple Matching
Including Outlier Rejection

No geometric outliers
But still some false matches
Retrieve camera projection matrices from fundamental matrix

Camera projection matrices must be compatible with the fundamental matrix. General formula

\[ P = [I \mid 0] \text{ and } P' = \left[ [e']_\times F + e'v^\top \mid \lambda e' \right] \text{ up to 3D projective transformation} \]

where \( F^\top e' = 0 \) (i.e., \( e' \) is the null space of \( F^\top \)), \( v \) is any 3-vector, and \( \lambda \) is any nonzero scalar.
If \( v = 0 \) and \( \lambda = 1 \), then

\[ P = [I \mid 0] \text{ and } P' = \left[ [e']_\times F \mid e' \right] \text{ up to 3D projective transformation} \]

Up to 3D projective transformation

\[ x = PX \quad x' = P'X \]
\[ x = PH^{-1}\bar{H}X \quad x' = P'\bar{H}^{-1}H \bar{X} \text{, where } \bar{H} \text{ is any 3D projective transformation} \]

Both pairs of camera projection matrices, \( P \) and \( P' \), and \( PH^{-1} \) and \( P'\bar{H}^{-1} \) have the same fundamental matrix \( F \)
Retrieve camera projection matrices from fundamental matrix

\[ P = [I \mid 0] \text{ and } P' = [M \mid v] \] up to 3D projective transformation

\( M \) of full rank (i.e., rank 3) and as far away from rank 2 as possible is obtained as follows

\[ F = U D V^T \] singular value decomposition, where \( U = [u_1 \mid u_2 \mid u_3] \) and \( D = \text{diag}(s, t, 0) \)

\[ F = U W Z D' V^T, \text{ where } D = W Z D' \text{ and } W Z = \text{diag}(1, 1, 0) \]

\[ W = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ and } D' = \text{diag}(s, t, (s + t)/2) \]

\[ F = (U W U^T)(U Z D' V^T) \]

\[ F = S M \text{ and } P' = [M \mid v], \text{ where } S = U W U^T = -[v]_{\times} = -[u_3]_{\times} \text{ and } M = U Z D' V^T \text{ and } v = u_3 = e' \]
Up to 3D projective transformation
Next lecture

• Estimation of fundamental matrix, nonlinear
• Reading
  – Sections 10.1, 9.5, 11.4, 12.5, 10.5