Lecture 11: Two-view geometry
Readings

• Szeliski, Chapter 7.2
• “Fundamental matrix song”
Back to stereo

- Where do epipolar lines come from?
Two-view geometry

- Where do epipolar lines come from?

3d point lies somewhere along \( r \)

epipolar plane

epipolar line

Image 1

Image 2

epipolar line (projection of \( r \))
Fundamental matrix

- This *epipolar geometry* of two views is described by a Very Special 3x3 matrix \( F \), called the *fundamental matrix*
- \( F \) maps (homogeneous) *points* in image 1 to *lines* in image 2!
- The epipolar line (in image 2) of point \( p \) is: \( Fp \)

*Epipolar constraint* on corresponding points: \( q^T Fp = 0 \)
Fundamental matrix

- Two special points: \( e_1 \) and \( e_2 \) (the *epipoles*): projection of one camera into the other
Fundamental matrix

- Two special points: $e_1$ and $e_2$ (the *epipoles*): projection of one camera into the other
- All of the epipolar lines in an image pass through the epipole
Rectified case

- Images have the same orientation, \( t \) parallel to image planes
- Where are the epipoles?
Epipolar geometry demo
Relationship with homography?

Images taken from the same center of projection? Use a homography!
Fundamental matrix – uncalibrated case

$K_1$ : intrinsics of camera 1
$K_2$ : intrinsics of camera 2
$R$ : rotation of image 2 w.r.t. camera 1

$q^T K_2^{-T} R [t] \times K_1^{-1} p = 0$

$F$ the Fundamental matrix
Cross-product as linear operator

**Useful fact:** Cross product with a vector $\mathbf{t}$ can be represented as multiplication with a \((\text{skew-symmetric})\) 3x3 matrix

$$
[\mathbf{t}]_\times = \begin{bmatrix}
0 & -t_z & t_y \\
t_z & 0 & -t_x \\
-t_y & t_x & 0 \\
\end{bmatrix}
$$

$$
\mathbf{t} \times \tilde{\mathbf{p}} = [\mathbf{t}]_\times \tilde{\mathbf{p}}
$$
Fundamental matrix – calibrated case

\[ \tilde{p} = K_1^{-1}p \quad : \text{ray through } p \text{ in camera 1’s (and world) coordinate system} \]

\[ \tilde{q} = K_2^{-1}q \quad : \text{ray through } q \text{ in camera 2’s coordinate system} \]

\[ \tilde{q}^T R \begin{bmatrix} t \end{bmatrix} \times \tilde{p} = 0 \]

\[ \tilde{q}^T E \tilde{p} = 0 \]

\( E \leftarrow \text{the Essential matrix} \)
Properties of the Fundamental Matrix

• \( Fp \) is the epipolar line associated with \( p \)

• \( F^T q \) is the epipolar line associated with \( q \)

• \( Fe_1 = 0 \) and \( F^T e_2 = 0 \)

• \( F \) is rank 2

• How many parameters does \( F \) have?
$$R = I_{3 \times 3}$$
$$t = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$$
$$E = \begin{bmatrix} 0 & 0 & 0 & 0 \\
0 & 0 & -1 \\
0 & 1 & 0 \end{bmatrix}$$
Stereo image rectification

- reproject image planes onto a common plane parallel to the line between optical centers
- pixel motion is horizontal after this transformation
- two homographies (3x3 transform), one for each input image reprojection

Questions?
Estimating $F$

- If we don’t know $K_1$, $K_2$, $R$, or $t$, can we estimate $F$ for two images?
- Yes, given enough correspondences
Estimating F – 8-point algorithm

• The fundamental matrix F is defined by

\[
x' \, ^T \, F \, x = 0
\]

for any pair of matches x and x' in two images.

• Let \( x=(u,v,1)^T \) and \( x'=(u',v',1)^T \),

\[
F = \begin{bmatrix}
    f_{11} & f_{12} & f_{13} \\
    f_{21} & f_{22} & f_{23} \\
    f_{31} & f_{32} & f_{33}
\end{bmatrix}
\]

each match gives a linear equation

\[
uu' \, f_{11} + vu' \, f_{12} + u' \, f_{13} + uv' \, f_{21} + vv' \, f_{22} + v' \, f_{23} + uf_{31} + vf_{32} + f_{33} = 0
\]
8-point algorithm

\[
\begin{bmatrix}
  u_1u_1' & v_1u_1' & u_1' & u_1v_1' & v_1v_1' & v_1' & u_1 & v_1 & 1 \\
  u_2u_2' & v_2u_2' & u_2' & u_2v_2' & v_2v_2' & v_2' & u_2 & v_2 & 1 \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
  u_nu_n' & v_nu_n' & u_n' & u_nv_n' & v_nv_n' & v_n' & u_n & v_n & 1 \\
\end{bmatrix}
\begin{bmatrix}
  f_{11} \\
  f_{12} \\
  f_{13} \\
  f_{21} \\
  f_{22} \\
  f_{23} \\
  f_{31} \\
  f_{32} \\
  f_{33}
\end{bmatrix} = 0
\]

- In reality, instead of solving \( Af = 0 \), we seek \( f \) to minimize \( \| Af \| \), least eigenvector of \( A^TA \).
8-point algorithm – Problem?

• \( \mathbf{F} \) should have rank 2
• To enforce that \( \mathbf{F} \) is of rank 2, \( \mathbf{F} \) is replaced by \( \mathbf{F}' \) that minimizes \( \| \mathbf{F} - \mathbf{F}' \| \) subject to the rank constraint.

• This is achieved by SVD. Let \( \mathbf{F} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \), where

\[
\Sigma = \begin{bmatrix}
\sigma_1 & 0 & 0 \\
0 & \sigma_2 & 0 \\
0 & 0 & \sigma_3 \\
\end{bmatrix}, \quad \text{let} \quad \Sigma' = \begin{bmatrix}
\sigma_1 & 0 & 0 \\
0 & \sigma_2 & 0 \\
0 & 0 & 0 \\
\end{bmatrix}
\]

then \( \mathbf{F}' = \mathbf{U} \Sigma' \mathbf{V}^T \) is the solution.
8-point algorithm

% Build the constraint matrix
A = [x2(1,:)'.*x1(1,:)' x2(1,:)'.*x1(2,:)' x2(1,:)'
     x2(2,:)'.*x1(1,:)' x2(2,:)'.*x1(2,:)' x2(2,:)'
     x1(1,:)'
     x1(2,:)'
     ones(npts,1) ];

[U,D,V] = svd(A);

% Extract fundamental matrix from the column of V
% corresponding to the smallest singular value.
F = reshape(V(:,9),3,3)';

% Enforce rank2 constraint
[U,D,V] = svd(F);
F = U*diag([D(1,1) D(2,2) 0])*V';
8-point algorithm

- Pros: it is linear, easy to implement and fast
- Cons: susceptible to noise
Problem with 8-point algorithm

Orders of magnitude difference between column of data matrix → least-squares yields poor results
Normalized 8-point algorithm

normalized least squares yields good results

Transform image to \([-1,1] \times [-1,1]\)
Normalized 8-point algorithm

1. Transform input by $\hat{x}_i = T x_i$, $\hat{x}'_i = T x'_i$

2. Call 8-point on $\hat{x}_i, \hat{x}'_i$ to obtain $\hat{F}$

3. $F = T'^T \hat{F} T$

\[ x'^T F x = 0 \]

\[ \hat{x}'^T T'^{-T} \hat{F} T^{-1} \hat{x} = 0 \]
Normalized 8-point algorithm

\[\begin{align*}
[x1, T1] &= \text{normalise2dpts}(x1); \\
[x2, T2] &= \text{normalise2dpts}(x2);
\end{align*}\]

\[A = [x2(1,:)' \cdot x1(1,:)' \ x2(1,:)' \cdot x1(2,:)' \ x2(1,:)' \ x2(2,:)' \cdot x1(1,:)' \ x2(2,:)' \cdot x1(2,:)' \ x2(2,:)' \ x1(1,:)' \ x1(2,:)' \ \text{ones}(npts,1) ];\]

\[[U,D,V] = \text{svd}(A);\]

\[F = \text{reshape}(V(:,9),3,3)';\]

\[[U,D,V] = \text{svd}(F);\]
\[F = U \cdot \text{diag}([D(1,1) \ D(2,2) \ 0]) \cdot V';\]

% Denormalise
\[F = T2' \cdot F \cdot T1;\]
Results (ground truth) with standard stereo calibration
Results (8-point algorithm)
Results (normalized 8-point algorithm)
What about more than two views?

• The geometry of three views is described by a $3 \times 3 \times 3$ tensor called the \textit{trifocal tensor}.

• The geometry of four views is described by a $3 \times 3 \times 3 \times 3$ tensor called the \textit{quadrifocal tensor}.

• After this it starts to get complicated...
Large-scale structure from motion

Dubrovnik, Croatia. 4,619 images (out of an initial 57,845).
Total reconstruction time: 23 hours
Number of cores: 352