Today

- Review of Assignment 0.
- Transformations between images
- Structure from Motion
- The Essential Matrix
- The Fundamental Matrix
Assignment 1 is now released!!!

- Slight delay in the release.
- Due date is now midnight **Monday October 6th**.
Assignment 0

• Everyone successfully uploaded their assignments!!!!
• Thanks for your patience :).
• Reminder of late policy.
  • Each student is allotted a total of five late-day points for the semester. Late-day points are for use on assignments only (They cannot be used for midterm or final projects). Late-day points work as follows:
  • A person can extend an assignment deadline by one day using one point.
  • If a person does not have remaining late day points, late hand-ins will incur a 10% penalty per day (up to three days per assignment).
  • No assignments will be accepted more than three days after the deadline.
Assignment 0

• Question 1a) - Scaling of Lena

```c
// Ensure aspect ratio looks correct
imageView_.contentMode = UIViewContentModeScaleAspectFit;
```

• Question 2a) - Accessing .xml files

```c
NSString *str = [[NSBundle mainBundle]
    pathForResource:@"haarcascade_frontalface_alt" ofType:@"xml"];
const char *CascadeName = [str UTF8String]; // Convert to const char *
if(!CascadeName)
```

• Question 2a) - Color conversion

```c
// Switch colors to account for how UIImage and cv::Mat lay out their
color channels differently
cvtColor(display_im, display_im, CV_BGR2RGB);
```

More detailed solutions will be placed in your AFS dropbox!!
AFS Dropbox

- Your AFS dropbox is here,

/afs/cs.cmu.edu/academic/class/16423-f15-users/andrew-id

- Current list of drop boxes is,

<table>
<thead>
<tr>
<th>adityas2</th>
<th>eluzhnic</th>
<th>johnnyw</th>
<th>letianz</th>
<th>swadhwa</th>
<th>wku</th>
<th>yujunwan</th>
</tr>
</thead>
<tbody>
<tr>
<td>alvincho</td>
<td>grock</td>
<td>joonwhep</td>
<td>nanxuanh</td>
<td>tfu1</td>
<td>wwillsey</td>
<td>yutec</td>
</tr>
<tr>
<td>atdupuis</td>
<td>jchen4</td>
<td>juedoul</td>
<td>pganadas</td>
<td>vkasired</td>
<td>yangminc</td>
<td>zerens</td>
</tr>
<tr>
<td>chengyal</td>
<td>jdalal</td>
<td>kainingh</td>
<td>sahils</td>
<td>wendaw</td>
<td>yicongw</td>
<td>zhuhanq</td>
</tr>
<tr>
<td>chenm</td>
<td>jiayilin</td>
<td>kunfungl</td>
<td>sibowang</td>
<td>wenqianz</td>
<td>yuhanlon</td>
<td>ziqig</td>
</tr>
</tbody>
</table>

- If you do not see your andrew ID and you are enrolled in the class, please contact Chen-Hsuan through Piazza.
Today

- Review of Assignment 0.
- Transformations between images
- Structure from Motion
- The Essential Matrix
- The Fundamental Matrix
Transformations between images

• So far we have considered transformations between the image and a plane in the world

• Now consider two cameras viewing the same plane

• There is a homography between camera 1 and the plane and a second homography between camera 2 and the plane

• It follows that the relation between the two images is also a homography
Camera under pure rotation

Special case is camera under pure rotation. Homography can be showed to be \( \Phi = \Lambda \Omega_2 \Lambda^{-1} \)

Why is this?
Figure 15.22 Computing visual panoramas. a-c) Three images of the same scene where the camera has rotated but not translated. Five matching points have been identified by hand between each pair. d) A panorama can be created by mapping the first and third images into the frame of reference of the second image.

This process can be completely automated by finding features and fitting a homography to each pair of images using a robust technique such as RANSAC. In a real system, the final result is typically projected onto a cylinder and unrolled to give a more visually pleasing result.

Discussion

This chapter has presented a number of important ideas. First, we discussed a family of transformations and how each can be related to a camera viewing the scene under special conditions. These transformations are used widely within machine vision and we will see them exploited in chapters 17, 18, and 19.

Today

• Review of Assignment 0.
• Transformations between images
• Structure from Motion
• The Essential Matrix
• The Fundamental Matrix
Taken from Agarwal et al. “Building Rome in a Day”, ICCV 2009.
For simplicity, we’ll start with simpler problem

- Just $J=2$ images
- Known intrinsic matrix $\Lambda$

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Epipolar lines

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Special configurations

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Today

- Review of Assignment 0.
- Transformations between images
- Structure from Motion
- The Essential Matrix
- The Fundamental Matrix
The Essential Matrix

First camera:

\[ \lambda_1 \tilde{x}_1 = w \]

Second camera:

\[ \lambda_2 \tilde{x}_2 = \Omega w + \tau \]

Substituting:

\[ \lambda_2 \tilde{x}_2 = \lambda_1 \Omega \tilde{x}_1 + \tau \]

This is a mathematical relationship between the points in the two images, but it’s not in the most convenient form.

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
The Essential Matrix

\[ \lambda_2 \tilde{x}_2 = \lambda_1 \Omega \tilde{x}_1 + \tau \]

\[ \lambda_2 \tau \times \tilde{x}_2 = \lambda_1 \tau \times \Omega \tilde{x}_1 \]

\[ \tilde{x}_2^T \tau \times \Omega \tilde{x}_1 = 0 \]
The Essential Matrix

\[ \tilde{x}_2^T \tau \times \Omega \tilde{x}_1 = 0 \]

The cross product term can be expressed as a matrix

\[ \tau \times = \begin{bmatrix} 0 & -\tau_z & \tau_y \\ \tau_z & 0 & -\tau_x \\ -\tau_y & \tau_x & 0 \end{bmatrix} \]

Defining:

\[ E = \tau \times \Omega \]

We now have the essential matrix relation

\[ \tilde{x}_2^T E \tilde{x}_1 = 0 \]

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Properties of the Essential Matrix

\[ \tilde{x}_2^T E \tilde{x}_1 = 0 \]

- Rank 2: \( \text{det}[E] = 0 \)
- 5 degrees of freedom
- Non-linear constraints between elements

\[ 2EE^T E - \text{trace}[EE^T] E = 0 \]
Recovering Epipolar Lines

Equation of a line: \( ax + by + c = 0 \)

or

\[
\begin{bmatrix}
a & b & c
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix} = 0
\]

or

\( \mathbf{l} \mathbf{x} = 0 \)
Recovering Epipolar Lines

Equation of a line: \( \mathbf{l} \tilde{\mathbf{x}} = 0 \)

Now consider \( \tilde{\mathbf{x}}_2^T \mathbf{E} \tilde{\mathbf{x}}_1 = 0 \)

This has the form \( l_1 \tilde{\mathbf{x}}_1 = 0 \) where \( l_1 = \tilde{\mathbf{x}}_2^T \mathbf{E} \)

So the epipolar lines are

\[
\begin{align*}
l_1 &= \tilde{\mathbf{x}}_2^T \mathbf{E} \\
l_2 &= \tilde{\mathbf{x}}_1^T \mathbf{E}^T
\end{align*}
\]
Recovering Epipolar Lines

Every epipolar line in image 1 passes through the epipole $e_1$.

In other words $\tilde{x}_2^T E \tilde{e}_1 = 0$ for all $\tilde{x}_2$.

This can only be true if $e_1$ is in the nullspace of $E$.

$$\tilde{e}_1 = \text{null}[E]$$

Similarly:

$$\tilde{e}_2 = \text{null}[E^T]$$

We find the null spaces by computing $E = U L V^T$, and taking the last column of $V$ and the last row of $U$. 

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Essential matrix: \( E = \tau \times \Omega \)

To recover translation and rotation use the matrix:

\[
W = \begin{bmatrix}
0 & -1 & 0 \\
1 & 0 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

We take the SVD \( E = ULV^T \) and then we set

\[
\tau \times = ULWU^T \\
\Omega = UW^{-1}V^T
\]
To get the different solutions, we multiply $\tau$ by $-1$ and substitute $W$ for $W^{-1}$. 
Today

• Review of Assignment 0.
• Transformations between images
• Structure from Motion
• The Essential Matrix
• The Fundamental Matrix
Now consider two cameras that are not normalised

\[ \lambda_1 \tilde{x}_1 = \Lambda_1 [I, 0] \tilde{w} \]

\[ \lambda_2 \tilde{x}_2 = \Lambda_2 [\Omega, \tau] \tilde{w} \]

By a similar procedure to before, we get the relation

\[ \tilde{x}_2^T \Lambda_2^{-T} E \Lambda_1^{-1} \tilde{x}_1 = 0 \]

or

\[ \tilde{x}_2^T F \tilde{x}_1 = 0 \]

where

\[ F = \Lambda_2^{-T} E \Lambda_1^{-1} = \Lambda_2^{-T} \tau \times \Omega \Lambda_1^{-1} \]

Relation between essential and fundamental

\[ E = \Lambda_2^T F \Lambda_1 \]

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
Estimation of the Fundamental Matrix
Estimation of Fundamental Matrix

When the fundamental matrix is correct, the epipolar line induced by a point in the first image should pass through the matching point in the second image and vice-versa.

This suggests the criterion

$$\hat{F} = \arg\min_{F} \left[ \sum_{i=1}^{I} \left( (\text{dist}[x_{i1}, l_{i1}])^2 + (\text{dist}[x_{i2}, l_{i2}])^2 \right) \right]$$

If \( l = [a, b, c] \) and \( x = [x, y]^T \)

$$\text{dist}[x, l] = \frac{ax + by + c}{\sqrt{a^2 + b^2}}$$

Unfortunately, there is no closed form solution for this quantity.

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
The 8 Point Algorithm

Approach:
- solve for fundamental matrix using homogeneous coordinates
- closed form solution (but to wrong problem!)
- Known as the 8 point algorithm

Start with fundamental matrix relation \( \tilde{x}_2^T F \tilde{x}_1 = 0 \)

Writing out in full:

\[
\begin{bmatrix} x_{i2} & y_{i2} & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_{i1} \\ y_{i1} \\ 1 \end{bmatrix} = 0
\]

or

\[
x_{i2} x_{i1} f_{11} + x_{i2} y_{i1} f_{12} + x_{i2} f_{13} + y_{i2} x_{i1} f_{21} + y_{i2} y_{i1} f_{22} + y_{i2} f_{23} + x_{i1} f_{31} + y_{i1} f_{32} + f_{33} = 0.
\]

Adapted from: Computer vision: models, learning and inference. Simon J.D. Prince
The 8 Point Algorithm

Can be written as: \[ [x_{i2}x_{i1}, x_{i2}y_{i1}, x_{i2}, y_{i2}x_{i1}, y_{i2}y_{i1}, y_{i2}, x_{i1}, y_{i1}, 1]f = 0 \]

where \( f = [f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]^T \)

Stacking together constraints from at least 8 pairs of points, we get the system of equations

\[
Af = \begin{bmatrix}
  x_{12}x_{11} & x_{12}y_{11} & x_{12} & y_{12}x_{11} & y_{12}y_{11} & y_{12} & x_{11} & y_{11} & 1 \\
  x_{22}x_{21} & x_{22}y_{21} & x_{22} & y_{22}x_{21} & y_{22}y_{21} & y_{22} & x_{21} & y_{21} & 1 \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
  x_{I2}x_{I1} & x_{I2}y_{I1} & x_{I2} & y_{I2}x_{I1} & y_{I2}y_{I1} & y_{I2} & x_{I1} & y_{I1} & 1
\end{bmatrix}f = 0.
\]
Minimum direction problem of the form \( \mathbf{A} \mathbf{b} = 0 \)
Find minimum of \( \left| \mathbf{A} \mathbf{b} \right|^2 \) subject to \( \left| \mathbf{b} \right| = 1 \).

To solve, compute the SVD \( \mathbf{A} = \mathbf{U} \mathbf{L} \mathbf{V}^T \)
and then set \( \hat{\mathbf{b}} \) to the last column of \( \mathbf{V} \).
Fitting Concerns

• This procedure does not ensure that solution is rank 2. Solution: set last singular value to zero.

• Can be unreliable because of numerical problems to do with the data scaling – better to re-scale the data first.

• Needs 8 points in general positions (cannot all be planar).

• Fails if there is not sufficient translation between the views.

• Use this solution to start non-linear optimisation of true criterion (must ensure non-linear constraints obeyed).

• There is also a 7 point algorithm.
More to read...

- Prince et al.
  - Chapter 16, Sections 1-3.