House keeping

- **Hawkeye:** Tim Harley  tim.harley@hawkeyeinnovations.co.uk

- **exam information**
  http://teaching.eng.cam.ac.uk/content/exam-information-students

- **webpage:**  http://cbl.eng.cam.ac.uk/Public/Turner/Teaching
Main idea: reconstruction from multiple images
Main idea: collect images from different views
Main idea: collect images from different views
Main idea: collect images from different views
Main idea: establish correspondences and triangulate
Main idea: establish correspondences and triangulate
Main idea: establish correspondences and triangulate
Main idea: establish correspondences and triangulate

- how do we find the correspondences efficiently
- how do we triangulate
  - to what degree do the cameras need to be calibrated?
  - how do we handle errors?
Main idea: establish correspondences and triangulate

- how do we find the correspondences efficiently
- how do we triangulate
  - to what degree do the cameras need to be calibrated?
  - how do we handle errors?

⇒ first need to understand the geometry of stereo vision
Outline

• We’re going to figure out the relationship between pairs of points in stereo images

• working in cartesian coordinates (to begin with)

• operate mainly in camera centred coordinates
Rays

camera-centred coordinates
Rays

Image plane coordinates

Camera-centred coordinates

$x$ image plane coordinates

camera-centred coordinates
Rays
Rays

\[ \text{translate by } T \text{ rotate by } R \]
Rays

world point

translate by $T$ rotate by $R$

camera-centred coordinates

image plane coordinates

$p$

$X_c$

$Y_c$

$Z_c$

$O_c$

$X'_c$

$Y'_c$

$Z'_c$

$O'_c$
Rays

world point

translate by $T$ rotate by $R$
Rays

translate by $T$ rotate by $R$
Rays

$p = [x, y, f]$

world point

translate by $T$ rotate by $R$
Rays

\[ p = [x, y, f] \]

\[ p' = [x', y', f'] \]

Translate by \( T \) rotate by \( R \)

Camera-centred coordinates

Image plane coordinates

World point

\[ [0, 0, f] \]
Rays

Camera-centred coordinates

Baseline

Ray 1

Ray 2

X

X′

Y

Y′

Z

Z′

O

O′

Camera-centred coordinates
Epipolar geometry

- Camera-centred coordinates
- Left epipole
- Right epipole
- Epipolar plane
- Baseline

Diagram showing the epipolar geometry with camera-centred coordinates, left epipole, right epipole, and epipolar plane.
Epipolar geometry

- Camera-centred coordinates
- Left epipole
- Right epipole
- Epipolar plane
- Baseline
- Image of other camera’s optic centre
Epipolar geometry

camera-centred coordinates
left epipole
left epipolar line
right epipolar line
right epipole
epipolar plane
baseline
image of other camera's optic centre

camera-centred coordinates
left epipole
right epipole
e image of other camera's optic centre
Epipolar geometry

camera-centred
coordinates
left epipole
left epipolar line
right epipolar line
right epipole
epipolar plane
baseline
image of other camera’s optic centre
image of ray from other camera’s optic centre to world point

camera-centred coordinates

left epipolar line
right epipolar line

epipolar plane

left epipole
right epipole

image of other camera’s optic centre
Epipolar geometry

Camera-centred coordinates

Left epipole

Baseline

Right epipole

Camera-centred coordinates
Epipolar geometry

Camera-centred coordinates

Left epipole

Right epipole

Baseline

Camera-centred coordinates
Epipolar geometry

camera-centred coordinates
left epipole right epipole baseline
Epipolar geometry

- Camera-centred coordinates
- Left epipole
- Right epipole
- Baseline
- New epipolar plane rotated around baseline

Diagram showing epipolar geometry with camera-centred coordinates and epipoles.
Epipolar geometry

all epipolar lines intersect at the epipoles

camera-centred coordinates

new epipolar plane
rotated around baseline

left epipole
right epipole
Epipolar geometry

found feature in first image

lies along ray

camera-centred coordinates
Epipolar geometry

found feature in first image

lies along ray

camera-centred coordinates

"?"
Epipolar geometry

found feature in first image

lies along ray

camera-centred coordinates
Epipolar geometry

found feature in first image

lies along ray
Epipolar geometry

found feature in first image

? lies along ray

camera-centred coordinates

found feature in first image
Epipolar geometry

found feature in first image

lies along ray
Epipolar geometry

found feature in first image
lies along ray

feature in second lies along epipolar line

camera-centred coordinates
Epipolar geometry

found feature in first image
lies along ray

feature in second
lies along epipolar
line

feature matching = one D search

camera-centred coordinates

feature matching = one D search
What’s the form of the epipolar line?

Camera-centred coordinates of point in world: $X_c$ or $X'_c$

Camera-centred coordinates of image of world point: $p = [x, y, f]$

Camera-centred coordinates of image of world point: $p' = [x', y', f']$

Camera-centred coordinates of image point

Camera-centred coordinates

Translate by $T$ rotate by $R$
Epipolar geometry

The camera-centred coordinates of point in world $X_c$ or $X'_c$ lies along ray

The image plane coordinates translate by and rotate by

camera centred coordinates of image of world point $p = [x, y, f]$

camera centred coordinates of image of world point $p' = [x', y', f']$

$X'_c = RX_c + T$
Epipolar geometry

camera-centred coordinates of point in world
$X_c$ or $X'_c$

camera-centred coordinates of image of world point
$p = [x, y, f]$

camera-centred coordinates of image of world point
$p' = [x', y', f']$

$X_c$ or $X'_c$

$X'_c = R0 + T$

translate by $T$, rotate by $R$.
Epipolar geometry

Camera-centred coordinates of point in world \( X_c \) or \( X'_c \)

Camera-centred coordinates of image of world point \( p = [x, y, f] \)

Camera-centred coordinates of image of world point \( p' = [x', y', f'] \)

Translate by \( T \) rotate by \( R \) \( X'_c = T \)

\( X'_c \)
What’s left to do?

- Uncalibrated cameras: to use framework, need to convert from camera centred coordinates to (homogeneous) pixel coordinates

- that’s the focus of our next lecture

- for now some demos...

  http://www.3dflow.net/technology/samantha-structure-from-motion/
  http://www.youtube.com/watch?v=faZSoE1qPxA
  http://www.acute3d.com/videos/
Kinect Sensor: Triangulation is the key to estimating depth.
Kinect

Book vs No Book (image courtesy: http://www.futurepicture.org/?p=97)
Kinect

Book vs No Book (image courtesy: http://www.futurepicture.org/?p=97)
Kinect modifies the pattern a bit and projects a dot-ted tic-tac-toe pattern
Kinect

How fusion works

Kinect Sensor
Kinect

Depth image obtained from the camera: Note the missing values in blue.
Kinect

Another image obtained from the camera: Note the missing values in blue.
Kinect

Third image obtained from the camera.
Fuse these images to reconstruct a 3D model.
Fusion smooths out the noise and missing values.
Recap: The cross-ratio

lines in world coordinates project to lines in pixel coordinates
Recap: The cross-ratio

\[
\begin{bmatrix}
    su \\
    sv \\
    s
\end{bmatrix} =
\begin{bmatrix}
    p_{11} & p_{12} & p_{13} & p_{14} \\
    p_{21} & p_{22} & p_{23} & p_{24} \\
    p_{31} & p_{32} & p_{33} & p_{34}
\end{bmatrix}
\begin{bmatrix}
    X \\
    Y \\
    Z \\
    1
\end{bmatrix}
\]

projective camera
Recap: The cross-ratio

\[
\begin{bmatrix}
  sx \\
  sy \\
  s
\end{bmatrix}
= \begin{bmatrix}
  p_{11} & p_{12} & p_{13} & p_{14} \\
  p_{21} & p_{22} & p_{23} & p_{24} \\
  p_{31} & p_{32} & p_{33} & p_{34}
\end{bmatrix}
\begin{bmatrix}
  X \\
  Y \\
  Z \\
  1
\end{bmatrix}
\]

projective camera
Recap: The cross-ratio

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

wlg can be set to 0

can be removed

wlg align world coordinates with line
Recap: The cross-ratio

\[
\begin{bmatrix}
  su \\
  sv \\
  s
\end{bmatrix} =
\begin{bmatrix}
  p_{11} & p_{14} \\
  p_{21} & p_{24} \\
  p_{31} & p_{34}
\end{bmatrix}
\begin{bmatrix}
  X \\
  1
\end{bmatrix}
\]

viewing a line
Recap: The cross-ratio

Consider colinear points in the world/image

\[
\begin{bmatrix}
    su \\
    sv \\
    s
\end{bmatrix}
= \begin{bmatrix}
    p_{11} & p_{14} \\
    p_{21} & p_{24} \\
    p_{31} & p_{34}
\end{bmatrix}
\begin{bmatrix}
    X \\
    1
\end{bmatrix}
\]
Recap: The cross-ratio

Consider colinear points in the world/image

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{14} \\ p_{21} & p_{24} \\ p_{31} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix}$$
Recap: The cross-ratio

\[
\begin{bmatrix}
    su \\
    sv \\
    s
\end{bmatrix} =
\begin{bmatrix}
    p_{11} & p_{14} \\
    p_{21} & p_{24} \\
    p_{31} & p_{34}
\end{bmatrix}
\begin{bmatrix}
    X \\
    1
\end{bmatrix} \quad \Rightarrow \quad
\begin{bmatrix}
    sl_i \\
    s
\end{bmatrix} =
\begin{bmatrix}
    p & q \\
    r & 1
\end{bmatrix}
\begin{bmatrix}
    L_i \\
    1
\end{bmatrix}
\]

using the fact that: \( l_i \propto u \)
Recap: The cross-ratio

using the fact that: $l_i \propto u$
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates

\[
\begin{bmatrix}
  s l_i \\
  s
\end{bmatrix} = \begin{bmatrix}
  p & q \\
  r & 1
\end{bmatrix} \begin{bmatrix}
  L_i \\
  1
\end{bmatrix}
\]

\[
l_i = \frac{p L_i + q}{r L_i + 1}
\]
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates

\[ l_c - l_a = \frac{pL_c + q}{rL_c + 1} - \frac{pL_a + q}{rL_a + 1} \]

\[
\begin{bmatrix}
  sl_i \\
  s
\end{bmatrix} = \begin{bmatrix}
p & q \\
r & 1
\end{bmatrix} \begin{bmatrix}
  L_i \\
  1
\end{bmatrix}
\]

\[ l_i = \frac{pL_i + q}{rL_i + 1} \]
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates:

\[ l_c - l_a = \frac{pL_c + q}{rL_c + 1} - \frac{pL_a + q}{rL_a + 1} \]

\[ = \frac{(pL_c + q)(rL_a + 1) - (pL_a + q)(rL_c + 1)}{(rL_c + 1)(rL_a + 1)} \]

\[ l_i = \frac{pL_i + q}{rL_i + 1} \]
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates

\[ l_c - l_a = \frac{pL_c + q}{rL_c + 1} - \frac{pL_a + q}{rL_a + 1} \]

\[ = \frac{(pL_c + q)(rL_a + 1) - (pL_a + q)(rL_c + 1)}{(rL_c + 1)(rL_a + 1)} \]

\[ = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)} \]

\[
\begin{bmatrix}
sl_i \\
s
\end{bmatrix} = \begin{bmatrix}
p & q \\
r & 1
\end{bmatrix} \begin{bmatrix}
L_i \\
1
\end{bmatrix}
\]

\[ l_i = \frac{pL_i + q}{rL_i + 1} \]
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates.
Recap: The cross-ratio

$l_c - l_a = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)}$

$\begin{bmatrix} s l_i \\ s \end{bmatrix} = \begin{bmatrix} p & q \\ r & 1 \end{bmatrix} \begin{bmatrix} L_i \\ 1 \end{bmatrix}$

$l_i = \frac{pL_i + q}{rL_i + 1}$

consider ratio of lengths along the line in pixel coordinates
Recap: The cross-ratio

Consider ratio of lengths along the line in pixel coordinates.

\[
l_c - l_a = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)}
\]

\[
l_c - l_b = \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)}
\]

\[
\begin{bmatrix}
  s l_i \\
  s
\end{bmatrix} = \begin{bmatrix}
  p & q \\
  r & 1
\end{bmatrix} \begin{bmatrix}
  L_i \\
  1
\end{bmatrix}
\]

\[
l_i = \frac{pL_i + q}{rL_i + 1}
\]
Recap: The cross-ratio

Ratios of lengths are NOT invariant (unlike the affine case)

\[
\begin{align*}
l_c - l_a &= \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)} \\
l_c - l_b &= \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)} \\
l_c - l_a &\quad l_c - l_b = \frac{(L_c - L_a)(rL_b + 1)}{(L_c - L_b)(rL_a + 1)}
\end{align*}
\]

\[
\begin{bmatrix}
sl_i \\
s
\end{bmatrix} = 
\begin{bmatrix}
p & q \\
r & 1
\end{bmatrix} 
\begin{bmatrix}
L_i \\
1
\end{bmatrix}
\]

\[
l_i = \frac{pL_i + q}{rL_i + 1}
\]
Recap: The cross-ratio

World coordinates

Pixel coordinates

\[
l_c - l_a = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)}
\]

\[
l_c - l_b = \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)}
\]

\[
\frac{l_c - l_a}{l_c - l_b} = \frac{(L_c - L_a)(rL_b + 1)}{(L_c - L_b)(rL_a + 1)}
\]

depends on \( r \)

Ratios of lengths are NOT invariant (unlike the affine case)
Recap: The cross-ratio

$$l_c - l_a = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)}$$

$$l_c - l_b = \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)}$$

$$\frac{l_c - l_a}{l_c - l_b} = \frac{(L_c - L_a)(rL_b + 1)}{(L_c - L_b)(rL_a + 1)}$$

BUT now consider ratio derived from point b to points a and b
Recap: The cross-ratio

\[
\begin{align*}
L_c - L_a &= \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)} \\
L_c - L_b &= \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)} \\
\frac{L_c - L_a}{L_c - L_b} &= \frac{(L_c - L_a)(rL_b + 1)}{(L_c - L_b)(rL_a + 1)}
\end{align*}
\]

BUT now consider ratio derived from point b to points a and b
Recap: The cross-ratio

The cross-ratio is the ratio of these two quantities.

\[
\frac{L_c - L_a}{L_c - L_b} = \frac{(L_c - L_a)(p - qr)}{(rL_c + 1)(rL_a + 1)}
\]

\[
\frac{L_c - L_b}{L_c - L_a} = \frac{(L_c - L_b)(p - qr)}{(rL_c + 1)(rL_b + 1)}
\]

\[
\frac{L_c - L_a}{L_c - L_b} = \frac{(L_c - L_a)(rL_b + 1)}{(L_c - L_b)(rL_a + 1)}
\]

\[
\frac{L_c - L_b}{L_c - L_a} = \frac{(L_c - L_b)(L_c - L_b)}{(L_d - L_b)(L_c - L_a)}
\]

the CROSS-RATIO is the ratio of these two quantities