Lecture 4
Single View Metrology

Professor Silvio Savarese
Computational Vision and Geometry Lab
Announcements:
- Remember to review the basics of linear algebra – e.g. SVD, etc...!!
- Read notes of lecture 2!
Lecture 4

Single View Metrology

- Review calibration and 2D transformations
- Vanishing points and lines
- Estimating geometry from a single image
- Extensions

Reading:

[HZ] Chapter 2 “Projective Geometry and Transformation in 2D”
[HZ] Chapter 3 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 8 “More Single View Geometry”
[Hoeim & Savarese] Chapter 2
Calibration Problem

\[ p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = M P_i \]

\[ M = K [R \quad T] \]

In pixels
World ref. system

11 unknowns
Need at least 6 correspondences
Once the camera is calibrated...

\[ M = K[R \quad T] \]

- Internal parameters K are known
- R, T are known – but these can only relate C to the calibration rig

Can I estimate P from the measurement p from a single image?

No - in general 😞 (P can be anywhere along the line defined by C and p)
Recovering structure from a single view

http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/hut/hutme.wrl
Transformation in 2D

- Isometries
- Similarities
- Affinity
- Projective
Transformation in 2D

Isometries:  
\[
\begin{bmatrix}
  x' \\
  y' \\
  1
\end{bmatrix}
= 
\begin{bmatrix}
  R & t \\
  0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
= H_e
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
\]  
[Eq. 4]

- Preserve distance (areas)
- 3 DOF
- Regulate motion of rigid object
Transformation in 2D

Similarities:

\[
\begin{bmatrix}
    x' \\
    y' \\
    1
\end{bmatrix}
= \begin{bmatrix}
    SR & t \\
    0 & 1
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
= H_s
\begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\]

[S = \begin{bmatrix}
    s & 0 \\
    0 & s
\end{bmatrix}

- Preserve
  - ratio of lengths
  - angles
- 4 DOF

[Eq. 5]
Transformation in 2D

Affinities:

\[
\begin{bmatrix}
    x' \\
    y' \\
    1
\end{bmatrix} = \begin{bmatrix}
    A & t \\
    0 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix} = H_a \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\]

[Eq. 6]

\[
A = \begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi)
\]

[Eq. 7]

\[
D = \begin{bmatrix}
    s_x & 0 \\
    0 & s_y
\end{bmatrix}
\]

\[
A = \begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix}
\]

---

Transformation in 2D

Affinities:

\[
\begin{bmatrix}
    x' \\
    y' \\
    1
\end{bmatrix} = \begin{bmatrix}
    A & t \\
    0 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix} = H_a \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\]

[Eq. 6]

\[
A = \begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi)
\]

[Eq. 7]

\[
D = \begin{bmatrix}
    s_x & 0 \\
    0 & s_y
\end{bmatrix}
\]

---

Transformation in 2D

Affinities:

\[
\begin{bmatrix}
    x' \\
    y' \\
    1
\end{bmatrix} = \begin{bmatrix}
    A & t \\
    0 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix} = H_a \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\]

[Eq. 6]

\[
A = \begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi)
\]

[Eq. 7]

\[
D = \begin{bmatrix}
    s_x & 0 \\
    0 & s_y
\end{bmatrix}
\]

---

Transformation in 2D

Affinities:

\[
\begin{bmatrix}
    x' \\
    y' \\
    1
\end{bmatrix} = \begin{bmatrix}
    A & t \\
    0 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix} = H_a \begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\]

[Eq. 6]

\[
A = \begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi)
\]

[Eq. 7]

\[
D = \begin{bmatrix}
    s_x & 0 \\
    0 & s_y
\end{bmatrix}
\]

---
Transformation in 2D

Affinities:

\[
\begin{bmatrix}
  x' \\
  y' \\
  1
\end{bmatrix} = \begin{bmatrix}
  A & t \\
  0 & 1
\end{bmatrix} \begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix} = H_a \begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
\]

\[\text{[Eq. 6]}\]

\[
A = \begin{bmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{bmatrix}
\]

\[= R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi)\]

\[\text{[Eq. 7]}\]

- Preserve:
  - Parallel lines
  - Ratio of areas
  - Ratio of lengths on collinear lines
  - others...
  - 6 DOF
Transformation in 2D

Projective:

\[
\begin{bmatrix}
  x' \\
  y' \\
  1
\end{bmatrix} =
\begin{bmatrix}
  A & t \\
  v & b
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix} = H_p
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
\]

[Eq. 8]

- 8 DOF
- Preserve:
  - collinearity
  - cross ratio of 4 collinear points
  - and a few others...
The cross ratio

The cross-ratio of 4 collinear points is defined as

\[
\frac{\left\| \mathbf{P}_3 - \mathbf{P}_1 \right\| \left\| \mathbf{P}_4 - \mathbf{P}_2 \right\|}{\left\| \mathbf{P}_3 - \mathbf{P}_2 \right\| \left\| \mathbf{P}_4 - \mathbf{P}_1 \right\|}
\]

[Eq. 9]
Lecture 4
Single View Metrology

• Review calibration and 2D transformations
• Vanishing points and lines
• Estimating geometry from a single image
• Extensions

Reading:
[HZ] Chapter 2 “Projective Geometry and Transformation in 2D”
[HZ] Chapter 3 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 8 “More Single View Geometry”
[Hoeim & Savarese] Chapter 2
Lines in a 2D plane

ax + by + c = 0

\[ l = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \]

If \( x = [ x_1, x_2]^T \in l \)

\[ \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix}^T \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0 \]

[Eq. 10]
Lines in a 2D plane

Intersecting lines

\[ x = l \times l' \] \text{ [Eq. 11]}  

Proof

\[ l \times l' \perp l \quad \rightarrow \quad (l \times l') \cdot l = 0 \quad \rightarrow \quad x \in l \] \text{ [Eq. 12]}  

\[ l \times l' \perp l' \quad \rightarrow \quad (l \times l') \cdot l' = 0 \quad \rightarrow \quad x \in l' \] \text{ [Eq. 13]}  

\[ \rightarrow \quad x \text{ is the intersection point} \]
2D Points at infinity (ideal points)

Let's intersect two parallel lines:

\[ l \times l' \propto \begin{bmatrix} b \\ -a \\ 0 \end{bmatrix} = x_\infty \ [\text{Eq. 13}] \]

= ideal point!

- In Euclidian coordinates this point is at infinity
- Agree with the general idea of two lines intersecting at infinity
2D Points at infinity (ideal points)

\[ x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, x_3 \neq 0 \]

\(-a/b = -a'/b'\)

Note: the line \( l = [a \ b \ c]^T \) pass trough the ideal point \( x_\infty \)

\[ l^T \ x_\infty = [a \ b \ c] \begin{bmatrix} b \\ -a \\ 0 \end{bmatrix} = 0 \quad [\text{Eq. 15}] \]

So does the line \( l' \) since \( a \ b' = a' \ b \)
Lines infinity $1_{\infty}$

Set of ideal points lies on a line called the line at infinity. How does it look like?

Indeed:

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

A line at infinity can be thought of the set of “directions” of lines in the plane.
Projective transformation of a point at infinity

\[
H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}
\]

\[
p' = H \cdot p
\]

\[
H \cdot p_\infty = ? = \begin{bmatrix} A & t \\ v & b \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} p'_x \\ p'_y \\ p'_z \end{bmatrix}
\]

...no!

An affine transformation of a point at infinity is still a point at infinity

\[
H_A \cdot p_\infty = ? = \begin{bmatrix} A & t \\ 0 & b \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} p'_x \\ p'_y \\ 0 \end{bmatrix}
\]

[Eq. 17]

[Eq. 18]
Projective transformation of a line (in 2D)

$$H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}$$

$$l' = H^{-T} l$$  
[Eq. 19]

$$H^{-T} l_\infty = ? = \begin{bmatrix} A & t \\ v & b \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ b \end{bmatrix}$$  
...no!

$$H^{-T}_A l_\infty = ? = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} A^{-T} & 0 \\ -t^T A^{-T} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$  
[Eq. 21]
Points and planes in 3D

\[ x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{bmatrix} \quad \Pi = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \]

\[ x \in \Pi \iff x^T \Pi = 0 \]  
\[ ax + by + cz + d = 0 \]

[Eq. 22]  
[Eq. 23]
Lines in 3D

• Lines have 4 degrees of freedom - hard to represent in 3D-space

• Can be defined as intersection of 2 planes

\[
d = \text{direction of the line} \\
= [a, b, c]^T
\]
Points at infinity in 3D

Points where parallel lines intersect in 3D

\[ x_\infty = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \end{bmatrix} \]
Vanishing points

The projective projection of a point at infinity into the image plane defines a vanishing point.

\[ \mathbf{x}_\infty = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \end{bmatrix} = \text{direction of corresponding parallel lines in 3D} \]

\[ \mathbf{p}_\infty = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \]
Vanishing points and directions

d = direction of the line
   = \([a, b, c]^\top\)

\[
v = K d
\]
[Eq. 24]

\[
d = \frac{K^{-1} v}{\|K^{-1} v\|}
\]
[Eq. 25]

Proof:

\[
X_\infty = \begin{bmatrix}
a \\
b \\
c \\
0
\end{bmatrix}
\]

\[
v = M X_\infty = K \begin{bmatrix}
a & b & c \\
0 & 1 & 0
\end{bmatrix} = K \begin{bmatrix}
a \\
b \\
c
\end{bmatrix}
\]
Vanishing (horizon) line

Projective transformation $M$

$$l_{hor} = H_P^{-T} l_{\infty}$$  [Eq. 26]
Example of horizon line

The orange line is the horizon!
Are these two lines parallel or not?

- Recognize the horizon line
- Measure if the 2 lines meet at the horizon
- if yes, these 2 lines are // in 3D

• Recognition helps reconstruction!
• Humans have learnt this
Vanishing points and planes

\[ n = K^T l_{\text{horiz}} \]

[Eq. 27]

See sec. 8.6.2 [HZ] for details
Planes at infinity

- Parallel planes intersect at infinity in a common line – the line at infinity
- A set of 2 or more lines at infinity defines the plane at infinity $\Pi_\infty$
Angle between 2 vanishing points

\[
\cos \theta = \frac{V_1^T \omega V_2}{\sqrt{V_1^T \omega V_1} \sqrt{V_2^T \omega V_2}}
\]

[Eq. 28]

\[
\omega = (K K^T)^{-1}
\]

[Eq. 30]

If \( \theta = 90 \) \rightarrow

\[
V_1^T \omega V_2 = 0
\]

[Eq. 29]
Projective transformation of a conic $\Omega$

$\omega_p = M^{-T} \Omega M^{-1}$

HZ page 73, eq. 3.16
Projective transformation of $\Omega_\infty$  

$\omega = M^{-T} \Omega_\infty M^{-1} = (K \ K^T)^{-1}$

HZ page 73
Properties of $\omega$

\[\omega = (K K^T)^{-1}\]

\[M = K \begin{bmatrix} R & T \end{bmatrix}\]  
[Eq. 30]

1. $\omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_4 \\ \omega_2 & \omega_3 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix}$  
symmetric and known up scale

2. $\omega_2 = 0$  
zero-skew

3. $\omega_1 = \omega_3$  
square pixel
Summary

\[
\mathbf{v} = K \mathbf{d}
\]
[Eq. 24]

\[
\mathbf{n} = K^T \mathbf{l}_{\text{horiz}}
\]
[Eq. 27]

\[
\cos \theta = \frac{\mathbf{v}_1^T \omega \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \omega \mathbf{v}_1} \sqrt{\mathbf{v}_2^T \omega \mathbf{v}_2}}
\]
[Eq. 28]

\[
\theta = 90 \quad \Rightarrow \quad \mathbf{v}_1^T \omega \mathbf{v}_2 = 0
\]
[Eq. 29]

\[
\omega = (K K^T)^{-1}
\]
[Eq. 30]

Useful to:

• To calibrate the camera
• To estimate the geometry of the 3D world
Lecture 4

Single View Metrology

- Review calibration
- Vanishing points and line
- Estimating geometry from a single image
- Extensions

Reading:
[HZ] Chapter 2 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 3 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 8 “More Single View Geometry”
[Hoeim & Savarese] Chapter 2
Single view calibration - example

\[ \cos \theta = \frac{v_1^T \omega v_2}{\sqrt{v_1^T \omega v_1} \sqrt{v_2^T \omega v_2}} \]

\[ \theta = 90^\circ \]

\[ \begin{cases} [\text{Eq. 29}] \\ v_1^T \omega v_2 = 0 \\ \omega = (K K^T)^{-1} \end{cases} \]

Do we have enough constraints to estimate \( K \)?

\( K \) has 5 degrees of freedom and Eq. 29 is a scalar equation.
Single view calibration - example

\[ \cos \theta = \frac{v_1^T \omega v_2}{\sqrt{v_1^T \omega v_1} \sqrt{v_2^T \omega v_2}} \]

\[
\begin{align*}
\{ V_1^T \omega V_2 &= 0 \\
\{ V_1^T \omega V_3 &= 0 \\
V_2^T \omega V_3 &= 0
\}
\end{align*}
\]
Single view calibration - example

\[ \omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_4 \\ \omega_2 & \omega_3 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix} \]

- Square pixels
- No skew

\[ \begin{align*}
\omega_2 &= 0 \\
\omega_1 &= \omega_3
\end{align*} \]

\[ \begin{cases}
V_1^T \omega V_2 = 0 \\
V_1^T \omega V_3 = 0 \\
V_2^T \omega V_3 = 0
\end{cases} \]
Single view calibration - example

\[ \omega = \begin{bmatrix} \omega_1 & 0 & \omega_4 \\ 0 & \omega_1 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix} \]

known up to scale

- Square pixels
- No skew \( \Rightarrow \omega_2 = 0 \), \( \omega_1 = \omega_3 \)

\[ \begin{cases} V_1^T \omega V_2 = 0 \\ V_1^T \omega V_3 = 0 \\ V_2^T \omega V_3 = 0 \end{cases} \]

Eqs. 31

\( \Rightarrow \) Compute \( \omega \) !
Single view calibration - example

\[ \omega = \begin{bmatrix} \omega_1 & 0 & \omega_4 \\ 0 & \omega_1 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix} \]

- Square pixels \[ \omega_2 = 0 \]
- No skew \[ \omega_1 = \omega_3 \]

\[ \begin{cases} V_1^T \omega V_2 = 0 \\ V_1^T \omega V_3 = 0 \\ V_2^T \omega V_3 = 0 \end{cases} \]

Once \( \omega \) is calculated, we get \( K \):

\[ \omega = (K \ K^T)^{-1} \rightarrow K \]

(Cholesky factorization; HZ pag 582)
Single view reconstruction - example

\[ \mathbf{K}_{\text{known}} \rightarrow \mathbf{n} = \mathbf{K}^T \mathbf{l}_{\text{horiz}} \]

= Scene plane orientation in the camera reference system

Select orientation discontinuities
Single view reconstruction - example

Recover the structure within the camera reference system
Notice: the actual scale of the scene is NOT recovered

- Recognition helps reconstruction!
- Humans have learnt this
Lecture 4

Single View Metrology

- Review calibration
- Vanishing points and lines
- Estimating geometry from a single image
- Extensions

Reading:

[HZ] Chapter 2 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 3 “Projective Geometry and Transformation in 3D”
[HZ] Chapter 8 “More Single View Geometry”
[Hoeim & Savarese] Chapter 2
http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/merton/merton.wrl
http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/merton/merton.wrl
La Trinita' (1426)
Firenze, Santa Maria Novella; by Masaccio (1401-1428)
La Trinità' (1426)
Firenze, Santa Maria Novella; by Masaccio (1401-1428)
Manually select:
• Vanishing points and lines;
• Planar surfaces;
• Occluding boundaries;
• Etc..
Automatic Photo Pop-up

Hoiem et al, 05
Automatic Photo Pop-up

Hoiem et al, 05...
Automatic Photo Pop-up

Software:

Make3D

Saxena, Sun, Ng, 05...

Training

Image

Depth

Prediction

Planar Surface Segmentation

Plane Parameter MRF

\[ P(\alpha|X, \nu, y, R; \theta) = \frac{1}{Z} \prod_i f_1(\alpha_i|X_i, \nu_i, R_i; \theta) \prod_{i,j} f_2(\alpha_i, \alpha_j|y_{ij}, R_i, R_j) \]

(a) Connectivity

(b) Co-Planarity
A software: **Make3D**

“Convert your image into 3d model”

http://make3d.stanford.edu/

http://make3d.cs.cornell.edu/
Depth map reconstruction using deep learning

Eigen et al., 2014

3D Layout estimation

Dasgupta, et al. CVPR 2016
3D Layout estimation
Coherent object detection and scene layout estimation from a single image

Y. Bao, M. Sun, S. Savarese, CVPR 2010, BMVC 2010
Next lecture:

Multi-view geometry (epipolar geometry)
Appendix
Vanishing points - example

v1, v2: measurements
K = known and constant

Can I compute R?
No rotation around z

\[ d_1 = \frac{K^{-1} v_1}{\| K^{-1} v_1 \|} \]

\[ d_2 = \frac{K^{-1} v_2}{\| K^{-1} v_2 \|} \]

\[ R d_1 = d_2 \rightarrow R \]

In 2D

\[ \theta_R = \alpha - \beta \]
\[ d_1 = \frac{K^{-1} v_1}{\|K^{-1} v_1\|} \]
\[ d_2 = \frac{K^{-1} v_2}{\|K^{-1} v_2\|} \]