Multiple View Geometry
in computer vision

Chapter 8: More Single View Geometry

Olaf Booij

Intelligent Systems Lab Amsterdam
University of Amsterdam, The Netherlands

HZClub 29-02-2008

Overview clubje

- Part 0, The Background
- Part 1, Single View Geometry
  - Chap 6, Defined camera matrix P, internal, external camera parameters
  - Chap 7, Estimated P using $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$
  - Chap 8, Estimate P using $\mathbf{x}_i$ and various pieces of information about $\mathbf{X}_i$
- Part 2, Two View Geometry
  - Chap 9, Define Fundamental matrix F using P and P'
  - Chap 11, Estimate F using $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$

Overview clubje

- Part 0, The Background
- Part 1, Single View Geometry
  - Chap 6, Defined camera matrix P, internal, external camera parameters
  - Chap 7, Estimated P using $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$
  - Chap 8, Estimate P using $\mathbf{x}_i$ and various pieces of information about $\mathbf{X}_i$
- Part 2, Two View Geometry
  - Chap 9, Define Fundamental matrix F using P and P'
  - Chap 11, Estimate F using $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$

Overview clubje

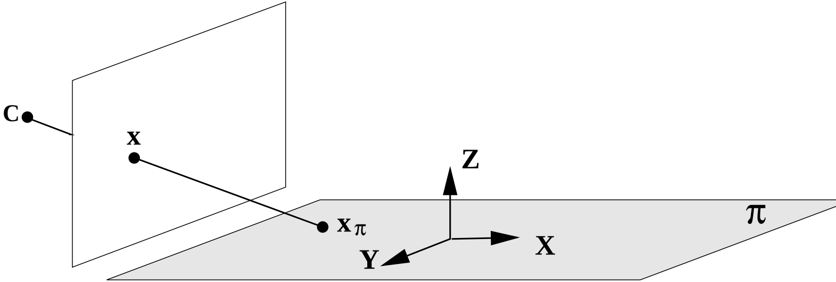
- Part 0, The Background
- Part 1, Single View Geometry
  - Chap 6, Defined camera matrix P, internal, external camera parameters
  - Chap 7, Estimated P using $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$
  - Chap 8, Estimate P using $\mathbf{x}_i$ and various pieces of information about $\mathbf{X}_i$
- Part 2, Two View Geometry
  - Chap 9, Define Fundamental matrix F using P and P'
  - Chap 11, Estimate F using $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$

Overview Chap 8

- If internal parameters K are (partially) known, euclidian properties of the scene can be measured in the image.
- K can be computed from absolute conic ω
- ω can be estimated from lines/points in the image with known geometric properties in the scene
- (geometric properties as in coplanarity/orthogonality)
- Finally…some clear applications (see Fig. 8.21)

Basics

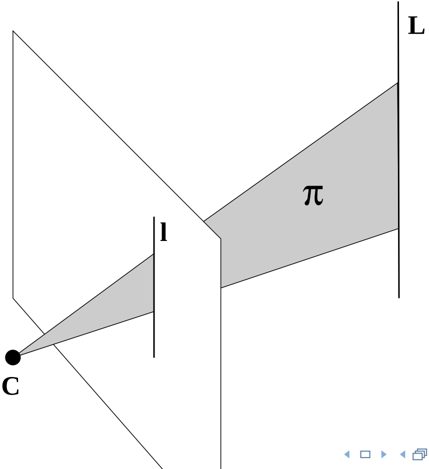
- The mapping
  from points on a plane to the image plane is a homography: $\mathbf{x} = H\mathbf{x}_\pi$.



- *extra: → P has 11 dof, pH has nine dof, thus P can not be computed from planar points only.*

Basics

- (Points on) A line $\mathbf{L}$ in the scene maps to (points on) a line $\mathbf{l}$ in the image.
- (Points on) A line $\mathbf{l}$ in the image maps to (points on) a plane $\boldsymbol{\pi}$ in the scene: $\mathbf{x}_\pi \in P^T\mathbf{l}$.



Basics

Plücker lines anyone?

Basics

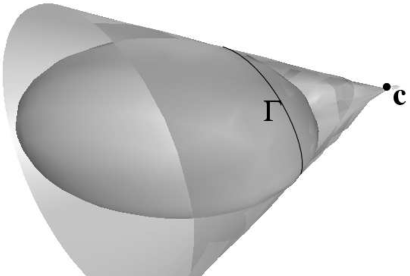
- A conic C in the image maps to a cone Q_{cone} in the scene, with: $Q_{cone} = P^T C P$
- $\leftarrow$ proof:

$$\mathbf{x}^T C \mathbf{x} = 0$$

$$\mathbf{x} = P\mathbf{X}$$

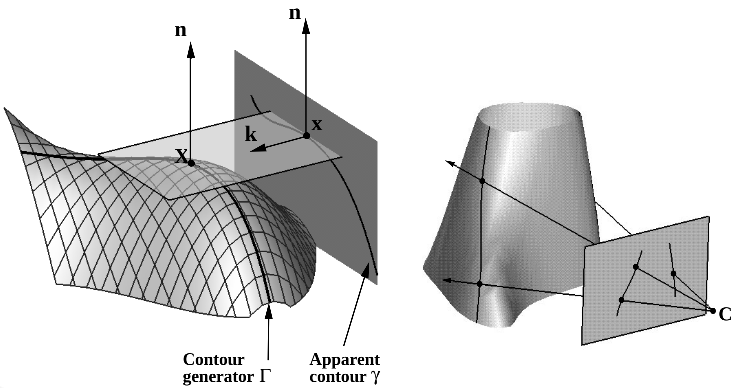
gives

$$\mathbf{X}^T \underbrace{P^T C P}_{Q_{cone}} \mathbf{X} = 0$$



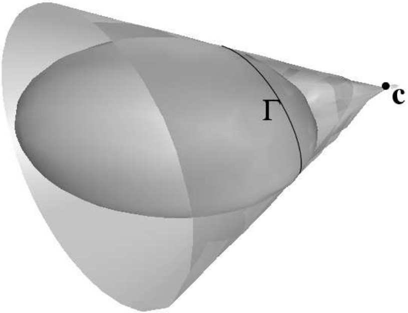
Smooth surfaces, general

- “contour-generator“ Γ results in ”profile“ γ on image-plane.
- Γ is defined by smooth surface and camera center **C**
- lines tangent to γ map to planes tangent to Γ



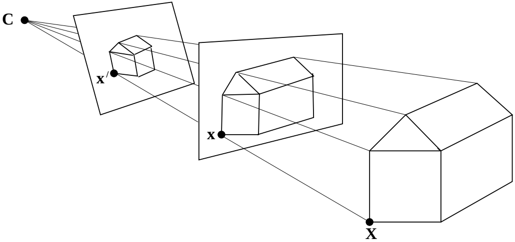
Smooth surfaces, quadrics

- A general quadric in the scene maps to a conic in the image.

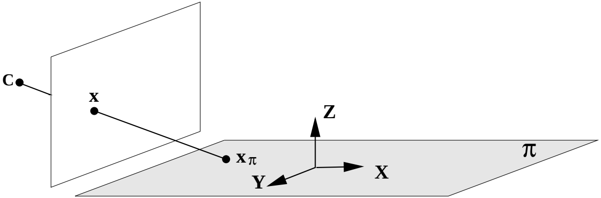


- ← spheres map to circles:
- General quadrics are 3-space projective transformations of spheres.
- Intersection and tangency is preserved, thus “contour-generator“ remains a plane conic.
- More on this, anyone?

Two cameras with equal centers



- Images taken by cameras with the same center are related by a homography: $\mathbf{x}' = P'\mathbf{X} = (K'R')(KR)^{-1}P\mathbf{X} = (K'R')(KR)^{-1}\mathbf{x} = H\mathbf{x}$
- ... we already new this:



Two cameras with equal centers

- Zooming
- K again:

$$K = \begin{bmatrix} fm_x & s & x_0 \\ 0 & fm_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} fbmI & \tilde{\mathbf{x}}_0 \\ 0^T & 1 \end{bmatrix}$$

- Given K and K' with $f'/f = k$:

$$K'K^{-1} = \begin{bmatrix} kI & (1-k)\tilde{\mathbf{x}}_0 \\ 0^T & 1 \end{bmatrix} \rightarrow K' = K\begin{bmatrix} kI & 0 \\ 0^T & 1 \end{bmatrix}$$

... Uh, yes, figures

- Does this hold for $s \neq 0$

Two cameras with equal centers

- Rotating.

$$\mathbf{x} = K[I|0]\mathbf{X} \rightarrow \mathbf{X} = K^{-1}\mathbf{x}$$

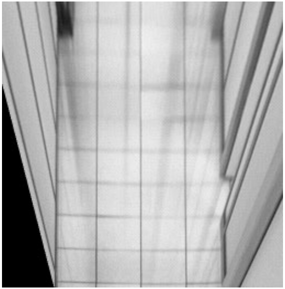
$$\mathbf{x}' = K[R|0]\mathbf{X} = KRK^{-1}\mathbf{x}$$

- $H = KRK^{-1}$ is and example of an infinite homography (?)

Two cameras with equal centers

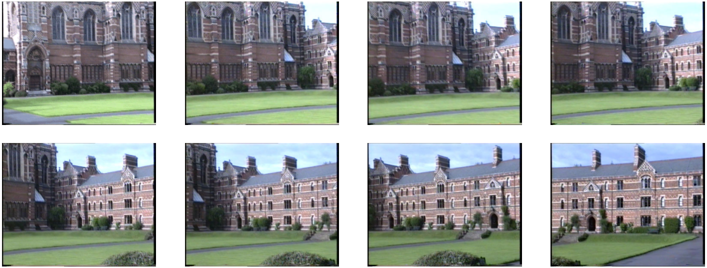
Two example applications.

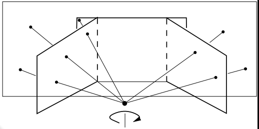
- Synthetic views:

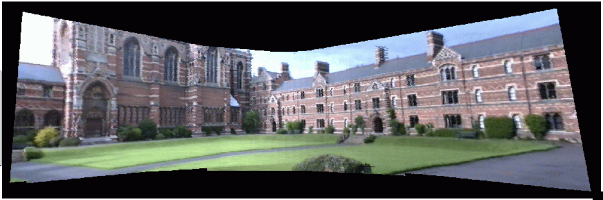


Two cameras with equal centers

- Mosaicing:

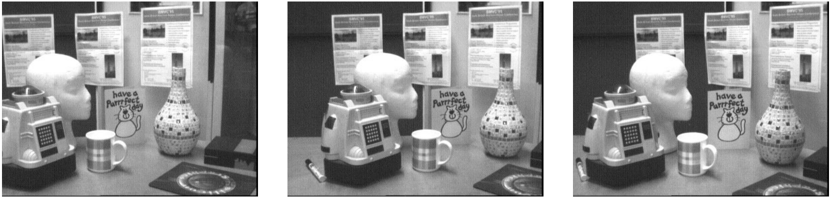






Two cameras without equal centers

Motion parallax.

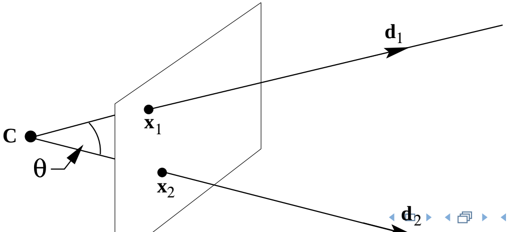


K and ω

- Calibration (i.e. determining K) relates image points $\mathbf{x}$ to the ray's *direction* $\mathbf{d}$: $\mathbf{d} = K^{-1}\mathbf{x}$.
- If K is known (i.e. the camera is calibrated) angles between rays can bejcomputed:

$$\cos(\theta) = \frac{\mathbf{d}_1^T \mathbf{d}_2}{\sqrt{\mathbf{d}_1^T \mathbf{d}_1}\sqrt{\mathbf{d}_2^T \mathbf{d}_2}} = \tag{1}$$

$$\frac{\mathbf{x}_1^T (K^{-T}K^{-1})\mathbf{x}_2}{\sqrt{\mathbf{x}_1^T (K^{-T}K^{-1})\mathbf{x}_1}\sqrt{\mathbf{x}_2^T (K^{-T}K^{-1})\mathbf{x}_2}} \tag{2}$$



K and ω

- Now finally something important: the image of the absolute conic ω is related to the calibration K.
- Points on π_∞, say $\mathbf{X}_\infty = (\mathbf{d}^T, 0)^T$ map to $KR\mathbf{d}$:

$$\mathbf{x} = P\mathbf{X}_\infty = KR[I|t](\mathbf{d}^T, 0)^T = KR\mathbf{d}.$$

- Notice: points on $\pi_\infty \equiv$ *direction*.
- Notice-2: π_∞ is really a plane and thus there is a $H = KR$ that maps it to the image plane.
- Notice-3 H does not depend on t (dC) (think of a stary night)

K and ω

- The absolute conic $\Theta_\infty = I$ on $\boldsymbol{\pi}_\infty$.
- Mapping Θ_∞ using $H = KR$ to the image plane gives us $\boldsymbol{\omega}$:

$$\boldsymbol{\omega} = H^{-T} I H^{-1} = (KR)^{-T} I (KR)^{-1} = K^{-T} R R^{-1} K^{-1} = (KK^T)^{-1}.$$

- Using Cholesky decomposition K can be recovered from $\boldsymbol{\omega}$.
- Angles between rays can now be expressed in $\boldsymbol{\omega}$:

$$\cos(\theta) = \frac{\mathbf{x}_1^T \boldsymbol{\omega} \mathbf{x}_2}{\sqrt{\mathbf{x}_1^T \boldsymbol{\omega} \mathbf{x}_1} \sqrt{\mathbf{x}_2^T \boldsymbol{\omega} \mathbf{x}_2}}.$$

- So, knowing $\boldsymbol{\omega}$ resutls in a calibrated camera!!

Estimating ω

- Let's define linear constraints on ω so we can estimate it
- If $\mathbf{x}_1$ and $\mathbf{x}_2$ resulted from perpendicular rays then:

$$\mathbf{x}_1^T \omega \mathbf{x}_2 = 0$$

- Through the pole-polar relationship we get for imagepoint $\mathbf{x}$ and imageline $\mathbf{l}$ resulting from a ray perpendicular to a scene plane:

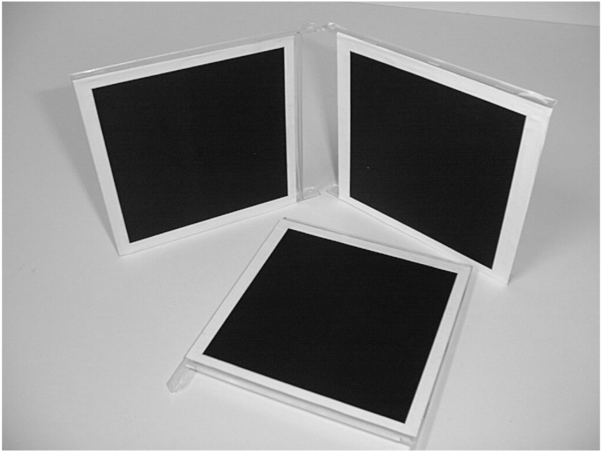
$$\mathbf{l} = \omega \mathbf{x} \rightarrow [\mathbf{l}]_\times \omega \mathbf{x} = 0$$

- Compute the imaged circular points of a mapped scene plane using an estimated H: $H(1, \pm i, 0)$. These lie on ω:

$$\mathbf{h}_1^T \omega \mathbf{h}_2 = 0 \;\text{ and }\; \mathbf{h}_1^T \omega \mathbf{h}_1 = \mathbf{h}_2^T \omega \mathbf{h}_2$$

Estimating ω

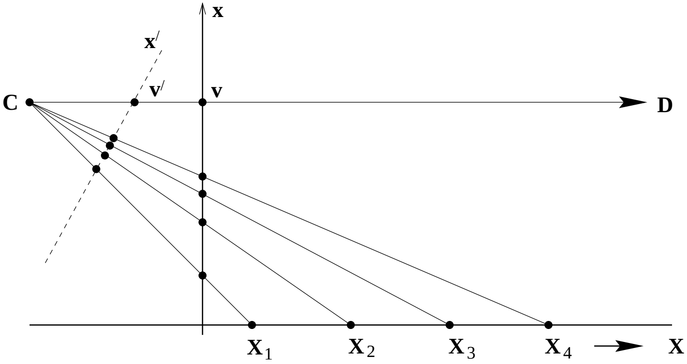
- We know how to estimate homographies.
- Three homographies provide 6 linear constraints.
- Solve by stacking constraints and apply the DLT algorithm



Estimating ω

- So how do we determine perpendicular points/lines in images:

vanishing points/lines



Estimating ω

- Especially in man-made environments there are a lot of parallel and perpendicular lines.
- Using the fact that we *know* that lines are parallel/perpendicular in the plane helps us find vanishing points/constraints on ω.

Estimating ω

- Vanishing points are just like regular image point:

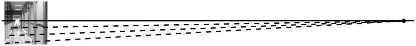
$$\cos(\theta) = \frac{\mathbf{v}_1^T \omega \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \omega \mathbf{v}_1}\sqrt{\mathbf{v}_2^T \omega \mathbf{v}_2}}.$$

- Also for vanishing points related to perpendicular lines in the scene define a linear constraint on ω:

$$\mathbf{v}_1^T \omega \mathbf{v}_2 = 0$$

Estimating ω

- Vanishing lines correpond to planes intersecting π_∞.
- They can be found by joining vanishing points resulting from rays on a the plane.
- Or, by using equally spaced parallel scene lines.



Estimating ω

ω can also be constrained by constraining certain internal camera parameters:

- Zero skew, results in
$$\omega_{12} = \omega_{21} = 0.$$
- This can also be seen as: x and y axes are orthogonal.
- zero skew and square pixels, results in
$$\omega_{11} = \omega_{22}.$$
- This can also be seen as: diagonal lines x = y and x = -y are orthogonal.

Measuring height

- The computation of K does not have to be explicit for measuring euclidian entities.
- See image 8.20 (p221)
- The ratio of parallel vertical scene lines can be determined using the vanishing line of the ground plane.
- In image 8.21 (p223) this is used to compute the length of two terrorists.

Almost done

Calibrating conic, anyone...

...coffee?