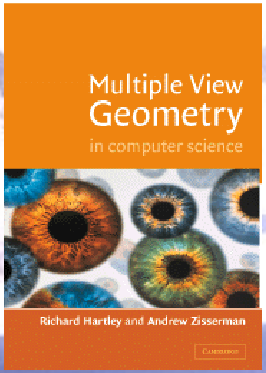
Computer Vision

The 2D projective plane and it's applications



Hartley Zisserman Ch 2. In particular: Ch 2.1-4, 2.7,

Estimation: HZ: Ch 4.1-4.2.5, 4.4.4-4.8 cursorly

(Szelisky: Ch 2.1.1, 2.1.2)

Richard Hartley and Andrew Zisserman, Multiple View Geometry, Cambridge University Publishers, 2nd ed. 2004

Homogeneous coordinates

Homogeneous representation of 2D points and lines

$$ax + by + c = 0 \qquad (a,b,c)^{\mathsf{T}}(x,y,1) = 0$$

The point x lies on the line l if and only if

$$\mathrm{l}^{\mathsf{T}}\mathrm{x} = 0$$

Note that scale is unimportant for incidence relation

$$(a,b,c)^{\mathsf{T}} \sim k(a,b,c)^{\mathsf{T}}, \forall k \neq 0 \qquad (x,y,1)^{\mathsf{T}} \sim k(x,y,1)^{\mathsf{T}}, \forall k \neq 0$$

equivalence class of vectors, any vector is representative

Set of all equivalence classes in $\mathbf{R}^3 - (0,0,0)^{\mathsf{T}}$ forms $\mathbf{P}^2$

Homogeneous coordinates $(x_1, x_2, x_3)^{\mathsf{T}}$ but only 2DOF

Inhomogeneous coordinates $(x, y)^{\mathsf{T}}$

Points from lines and vice-versa

Intersections of lines

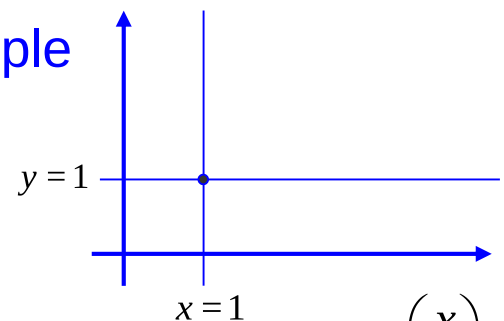
The intersection of two lines $\mathbf{l}$ and $\mathbf{l}'$ is $\mathbf{x} = \mathbf{l} \times \mathbf{l}'$

Line joining two points

The line through two points $\mathbf{x}$ and $\mathbf{x}'$ is $\mathbf{l} = \mathbf{x} \times \mathbf{x}'$

Example

$$(0,1,-1)\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$



$$(1,0,-1)\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Note:

$$\mathbf{x} \times \mathbf{x}' = [\mathbf{x}]_\times \mathbf{x}'$$

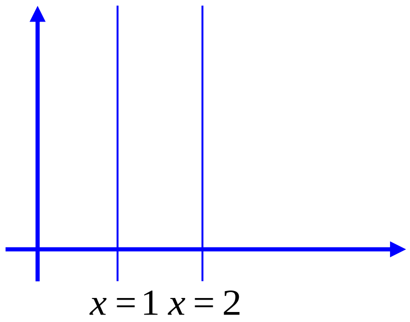
with $[\mathbf{x}]_\times = \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix}$

Ideal points and the line at infinity

Intersections of parallel lines

$\mathbf{l} = (a, b, c)^{\mathsf{T}}$ and $\mathbf{l}' = (a, b, c')^{\mathsf{T}}$ $\qquad \mathbf{l} \times \mathbf{l}' = (b, -a, 0)^{\mathsf{T}}$

Example



$(b, -a)$ tangent vector

(a, b) normal direction

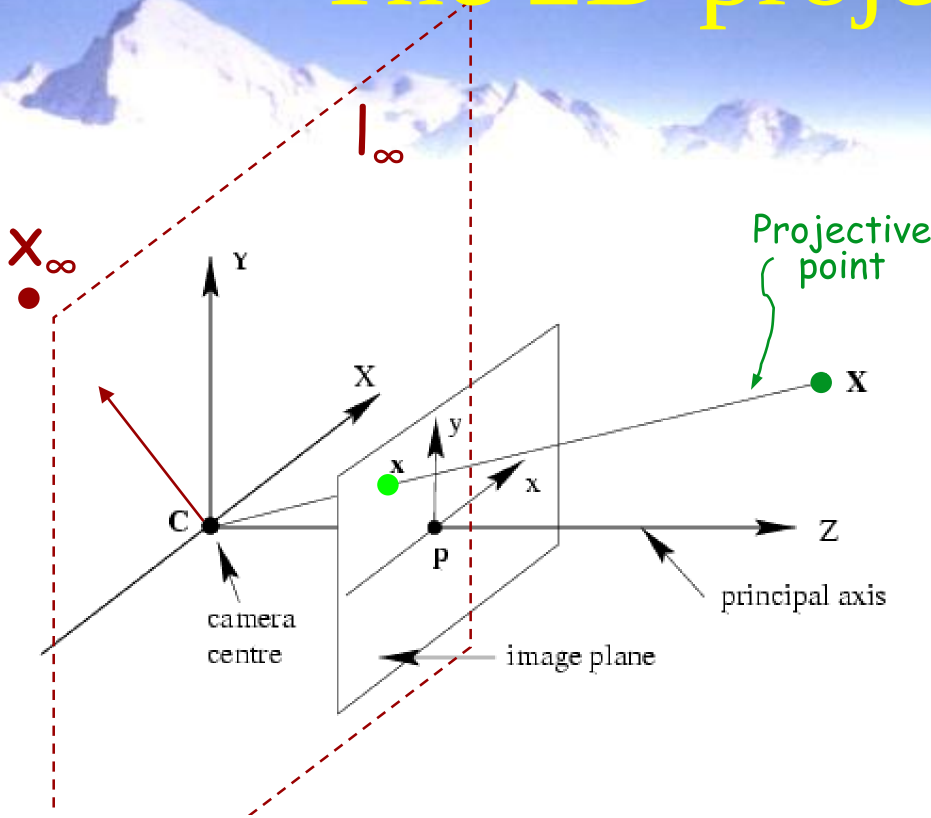
Ideal points $\quad (x_1, x_2, 0)^{\mathsf{T}}$

Line at infinity $\quad \mathbf{l}_\infty = (0,0,1)^{\mathsf{T}}$

$$\mathbf{P}^2 = \mathbf{R}^2 \cup \mathbf{l}_\infty$$

Note that in $\mathbf{P}^2$ there is no distinction between ideal points and others

The 2D projective plane



Homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \equiv s \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad s \neq 0$$

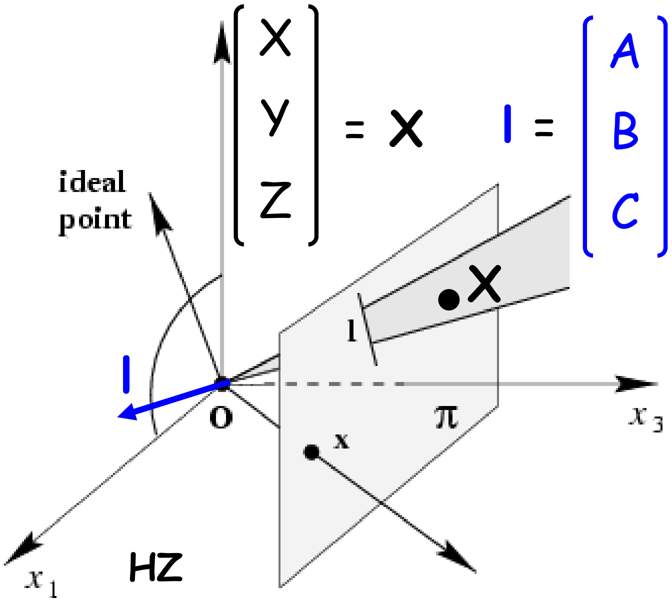
$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{z} \begin{bmatrix} x \\ y \end{bmatrix}$$

Inhomogeneous equivalent

- Perspective imaging models 2d projective space
- Each 3D ray is a point in P^2 : homogeneous coords.
- Ideal points
- P^2 is R^2 plus a “line at infinity” l_∞

$$x_\infty = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

Lines



- Projective line ~ a plane through the origin

$$l^T X = X^T l = AX + BY + CZ = 0$$

$$X_\infty = \begin{bmatrix} X \\ Y \\ 0 \end{bmatrix} \qquad l_\infty = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$ "line at infinity"

- Ideal line ~ the plane parallel to the image

Duality: For any 2d projective property, a dual property holds when the role of points and lines are interchanged.

$$l = X_1 \times X_2$$

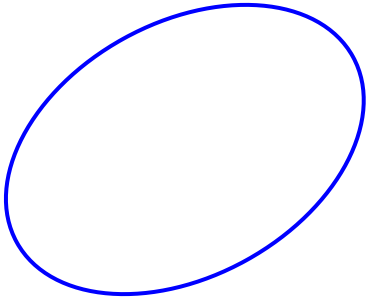
The line joining two points

$$X = l_1 \times l_2$$

The point joining two lines

Conics

Curve described by 2nd-degree equation in the plane



$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

or *homogenized* $x \mapsto x_1/x_3,\ y \mapsto x_2/x_3$

$$ax_1^{\ 2} + bx_1x_2 + cx_2^{\ 2} + dx_1x_3 + ex_2x_3 + fx_3^2 = 0$$

or in matrix form

$$\mathrm{x}^{\mathsf{T}}\mathbf{C}\,\mathrm{x} = 0 \quad \text{with} \quad \mathbf{C} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

5DOF: $\{a : b : c : d : e : f\}$

Five points define a conic

For each point the conic passes through

$$ax_i^2 + bx_i y_i + cy_i^2 + dx_i + ey_i + f = 0$$

or

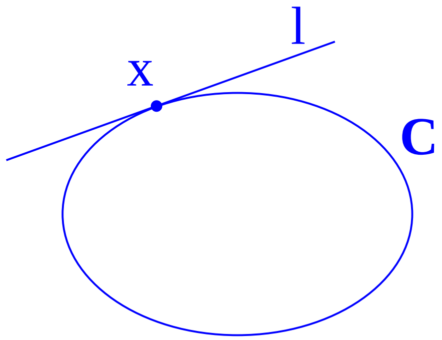
$$(x_i^2, x_i y_i, y_i^2, x_i, y_i, 1)\mathbf{c} = 0 \qquad \mathbf{c} = (a, b, c, d, e, f)^\mathsf{T}$$

stacking constraints yields

$$\begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2^2 & x_2 y_2 & y_2^2 & x_2 & y_2 & 1 \\ x_3^2 & x_3 y_3 & y_3^2 & x_3 & y_3 & 1 \\ x_4^2 & x_4 y_4 & y_4^2 & x_4 & y_4 & 1 \\ x_5^2 & x_5 y_5 & y_5^2 & x_5 & y_5 & 1 \end{bmatrix} \mathbf{c} = 0$$

Tangent lines to conics

The line l tangent to $\mathbf{C}$ at point x on $\mathbf{C}$ is given by $\mathrm{l}=\mathbf{C}\mathrm{x}$

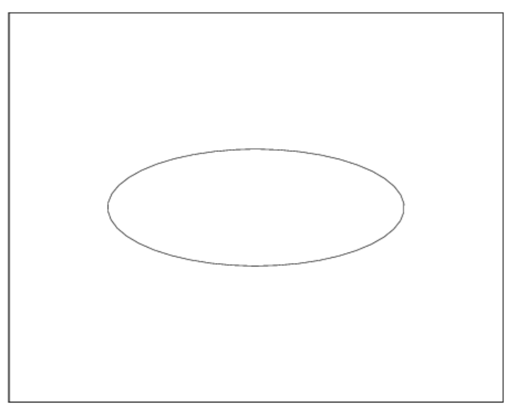


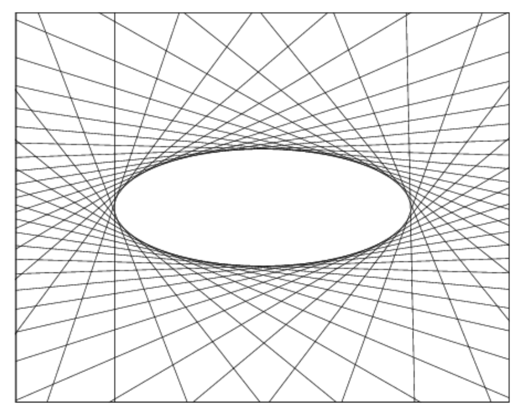
Dual conics

A line tangent to the conic **C** satisfies $\mathbf{l}^{\mathsf{T}}\,\mathbf{C}^{*}\,\mathbf{l} = 0$

In general (**C** full rank): $\mathbf{C}^{*} = \mathbf{C}^{-1}$

Dual conics = line conics = conic envelopes

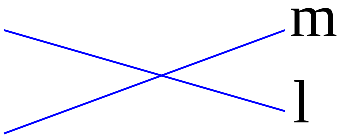




Degenerate conics

A conic is degenerate if matrix $\mathbf{C}$ is not of full rank

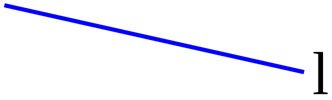
e.g. two lines (rank 2)



$$\mathbf{C} = \mathbf{l}\mathbf{m}^{\mathsf{T}} + \mathbf{m}\mathbf{l}^{\mathsf{T}}$$

e.g. repeated line (rank 1)

$$\mathbf{C} = \mathbf{l}\mathbf{l}^{\mathsf{T}}$$



Degenerate line conics: 2 points (rank 2), double point (rank1)

Note that for degenerate conics $\left(\mathbf{C}^{*}\right)^{*} \neq \mathbf{C}$

Conics: summary

- Conic:
  - Euclidean geometry: hyperbola, ellipse, parabola & degenerate
  - Projective geometry: equivalent under projective transform
  - Defined by 5 points

inhomogeneous

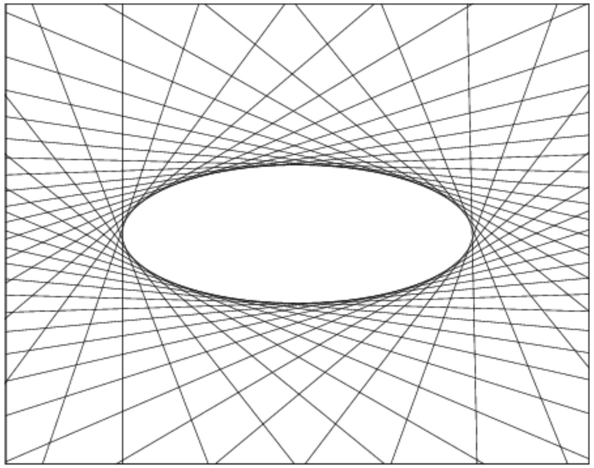
$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

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

homogeneous

$$C = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}$$

- Tangent line $\mathbf{l} = C\mathbf{x}$
- Dual conic C* $\mathbf{l}^T C^* \mathbf{l} = 0$



Projective transformations

Definition:

A *projectivity (=homography)* is an invertible mapping h from P^2 to itself such that three points $\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3$ lie on the same line if and only if $h(\mathrm{x}_1),h(\mathrm{x}_2),h(\mathrm{x}_3)$ do.

Theorem:

A mapping $h{:}\mathrm{P}^2 \rightarrow \mathrm{P}^2$ is a projectivity if and only if there exist a non-singular 3x3 matrix $\mathbf{H}$ such that for any point in P^2 represented by a vector x it is true that $h(\mathrm{x})=\mathbf{H}\mathrm{x}$

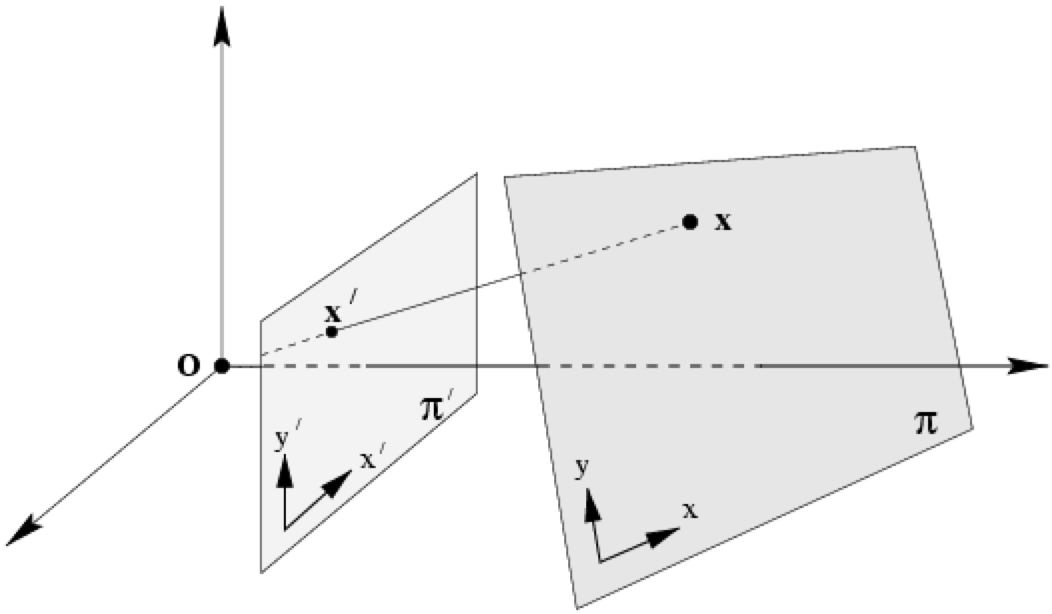
Definition: Projective transformation

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad \text{or} \quad \mathrm{x}' = \mathbf{H}\,\mathrm{x}$$

8DOF

projectivity=collineation=projective transformation=homography

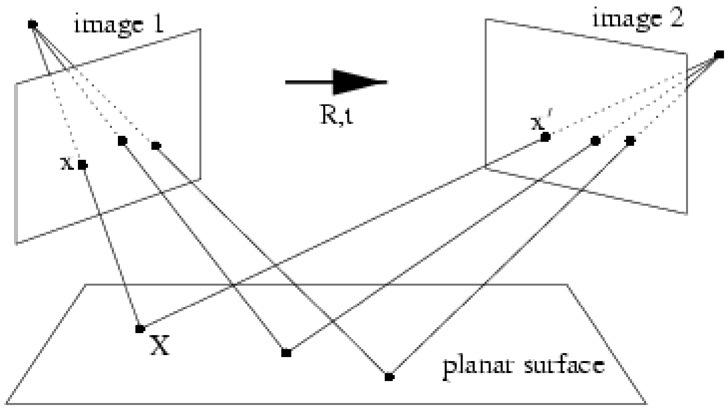
Mapping between planes

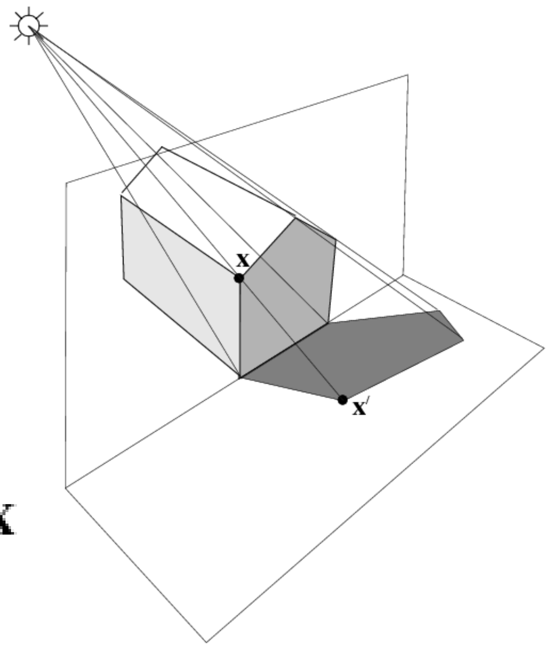


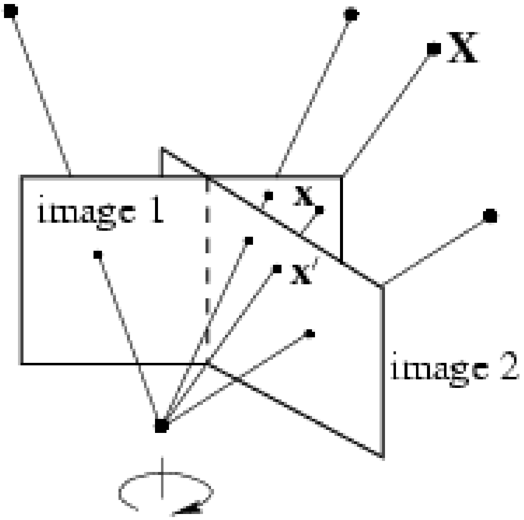
central projection may be expressed by x'=Hx

(application of theorem)

More examples







Projective transformations

- Homographies, collineations, projectivities
- 3x3 nonsingular H

maps $\boldsymbol{P}^2$ to $\boldsymbol{P}^2$
8 degrees of freedom
determined by 4 corresponding points

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

$$x' = Hx$$

- Transforming Lines?

subspaces preserved $\quad \mathbf{x}^T\mathbf{l} = 0 \qquad \mathbf{x}'^T\mathbf{l}' = 0$

substitution $\quad x^T H^T \mathbf{l}' = 0$

dual transformation $\quad \mathbf{l}' = H^{-T}\mathbf{l}$

Homographies a generalization of affine and Euclidean transforms

Group	Transformation	Invariants	Distortion
Projective 8 DOF	$H_P = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix}$	• Cross ratio • Intersection • Tangency	
Affine 6 DOF	$H_A = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Parallelism • Relative dist in 1d • Line at infinity $\mathbf{l}_\infty$	
Metric 4 DOF	$H_S = \begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Relative distances • Angles • Dual conic C^*_∞	
Euclidean 3 DOF	$H_E = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Lengths • Areas	

2 dof $\mathbf{l}_\infty$

2 dof C^*_∞

Planar Projective Warping

HZ

$$x_i' = Hx_i$$

$$i = 1 \ldots 4$$

A novel view rendered via four points with known structure

Planar Projective Warping

Original

Top-down

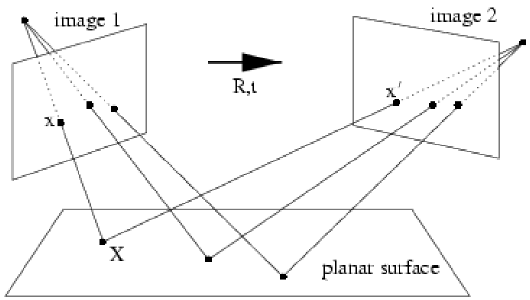
Facing right

HZ

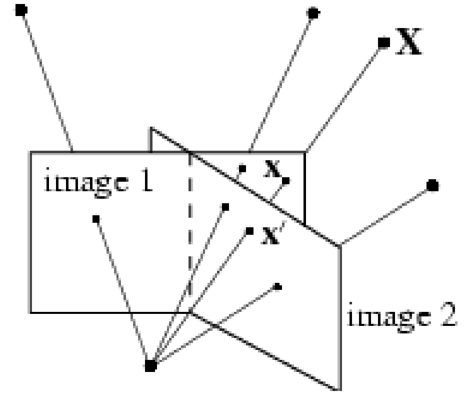
Artifacts are apparent where planarity is violated...

2d Homographies

2 images of a plane



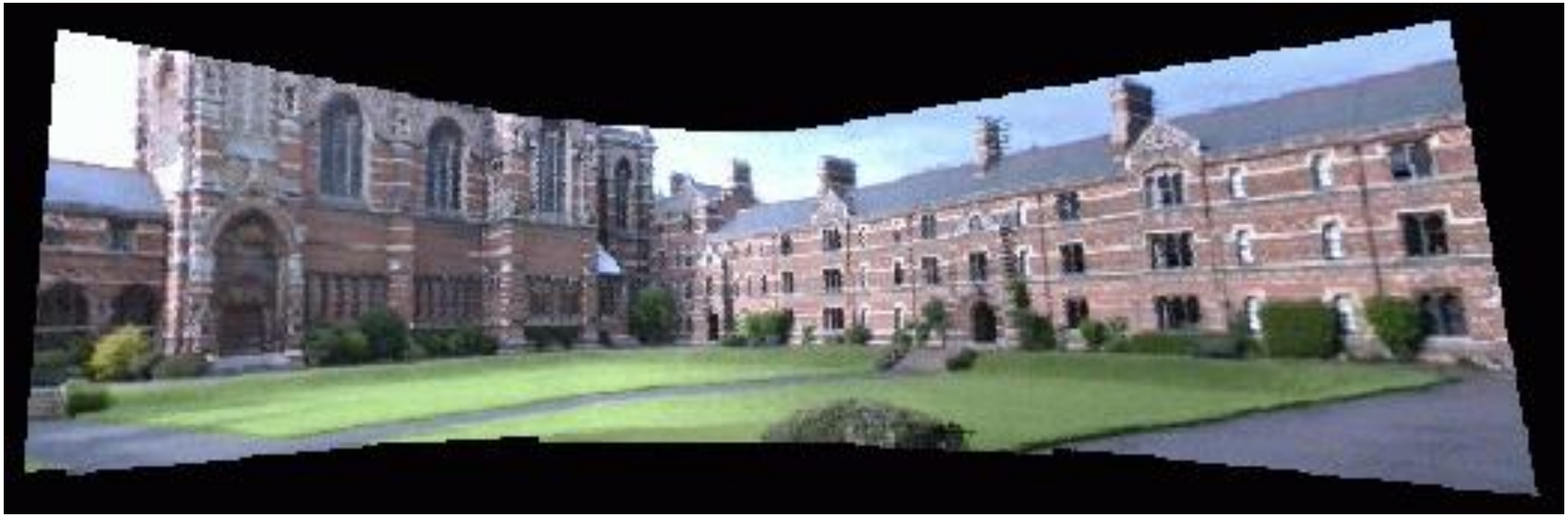
2 images from the same viewpoint (Perspectivity)



Panoramic imaging
Appl: Quicktime VR, robot navigation etc.

Homographies of the world, unite!

Image mosaics are stitched by homographies



HZ

Action of affinities and projectivities on line at infinity

$$\begin{bmatrix} \mathbf{A} & \mathrm{t} \\ 0^{\mathsf{T}} & v \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{A}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ 0 \end{pmatrix}$$

Line at infinity stays at infinity,
but points move along line

$$\begin{bmatrix} \mathbf{A} & \mathrm{t} \\ \mathrm{v}^{\mathsf{T}} & v \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{A}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ v_1 x_1 + v_2 x_2 \end{pmatrix}$$

Line at infinity becomes finite,
allows to observe vanishing points, horizon,

The line at infinity

The line at infinity l_∞ is a fixed line under a projective transformation H if and only if H is an affinity

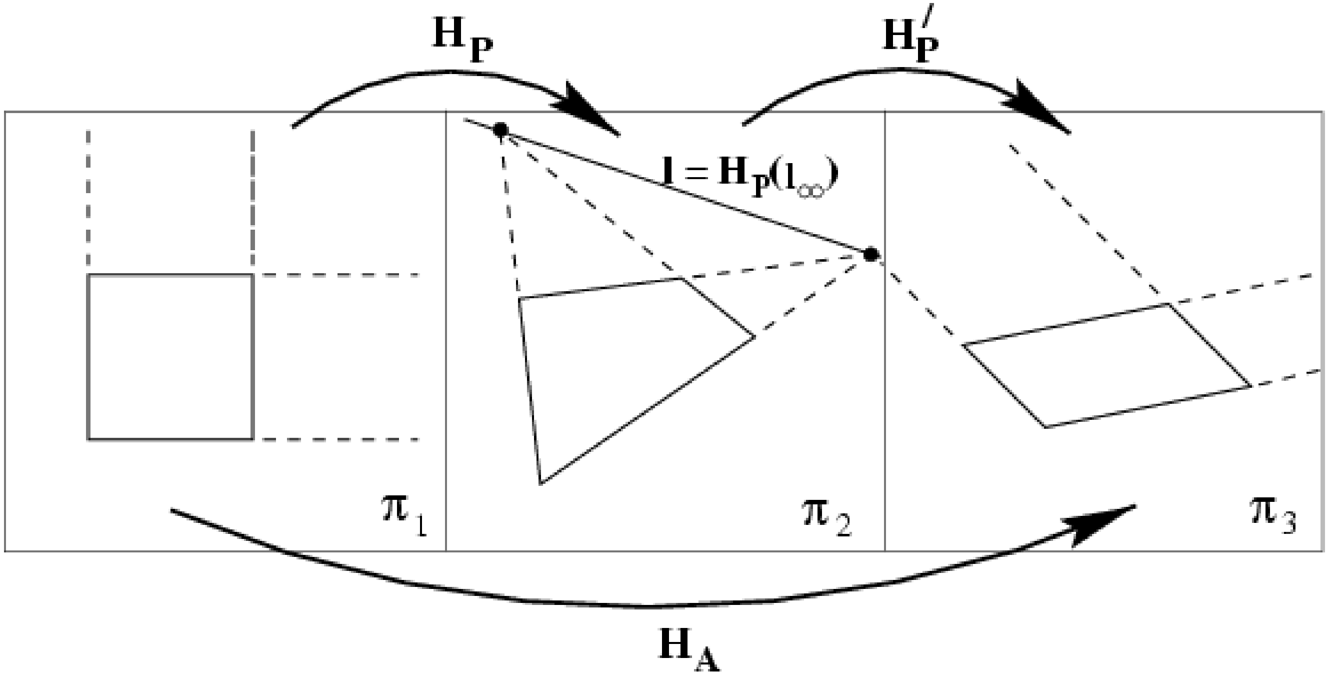
$$l'_\infty = \mathbf{H}_A^{-\mathsf{T}} l_\infty = \begin{bmatrix} \mathbf{A}^{-\mathsf{T}} & 0 \\ -t^{\mathsf{T}} \mathbf{A}^{-\mathsf{T}} & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = l_\infty$$

Note: But points on l_∞ can be rearranged to new points on l_∞

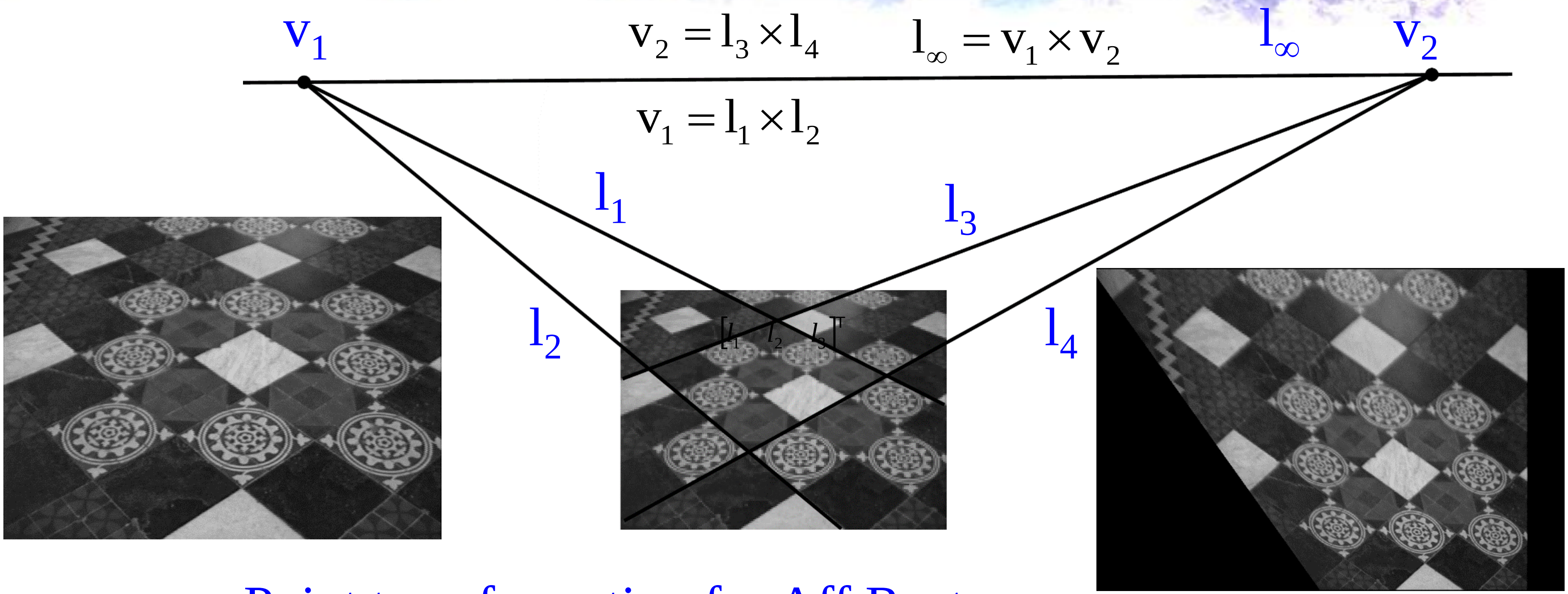
Affine properties from images

Projection
(Imaging)

Rectification
Post-processing



Affine rectification



Point transformation for Aff Rect:

$$\mathbf{l}_\infty = [l_1 \quad l_2 \quad l_3]^\top, l_3 \neq 0$$

$$\mathbf{H}_{PA} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix} \mathbf{H}_A$$

Exercise: Verify $\mathbf{H}_{PA}[l_1 \quad l_2 \quad l_3]^\top = [0,0,1]^\top$

Geometric strata: 2d overview

Group	Transformation	Invariants	Distortion
Projective 8 DOF	$H_P = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix}$	• Cross ratio • Intersection • Tangency	
Affine 6 DOF	$H_A = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Parallelism • Relative dist in 1d • Line at infinity $\mathbf{l}_\infty$	
Metric 4 DOF	$H_S = \begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Relative distances • Angles • Dual conic C^*_∞	
Euclidean 3 DOF	$H_E = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Lengths • Areas	

2 dof $\mathbf{l}_\infty$

2 dof C^*_∞

Parameter estimation in geometric transforms

cs428

- 2D homography

  Given a set of (x_i, x_i'), compute H $(x_i' = Hx_i)$

- 3D to 2D camera projection

  Given a set of (X_i, x_i), compute P $(x_i = PX_i)$

- Fundamental matrix

  Given a set of (x_i, x_i'), compute F $(x_i'^T F x_i = 0)$

Useful in Grad research

- Trifocal tensor

  Given a set of (x_i, x_i', x_i''), compute T

Math tools 1: Solving Linear Systems

- If $m = n$ (A is a square matrix), then we can obtain the solution by simple inversion:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$

- If $m > n$, then the system is **over-constrained** and A is not invertible
  – Use Matlab "\\" to obtain **least-squares solution** $x = A\backslash b$ to $Ax = b$ internally Matlab uses QR-factorization (cmput418/340) to solve this.
  – Can also write this using pseudoinverse $A^+ = (A^TA)^{-1}A^T$ to obtain **least-squares solution** $x = A^+b$

Fitting Lines

- A 2-D point $\mathbf{x} = (x, y)$ is on a line with slope m and intercept b if and only if $y = mx + b$
- Equivalently,

$$\begin{pmatrix} x & 1 \end{pmatrix} \begin{pmatrix} m \\ b \end{pmatrix} = y$$

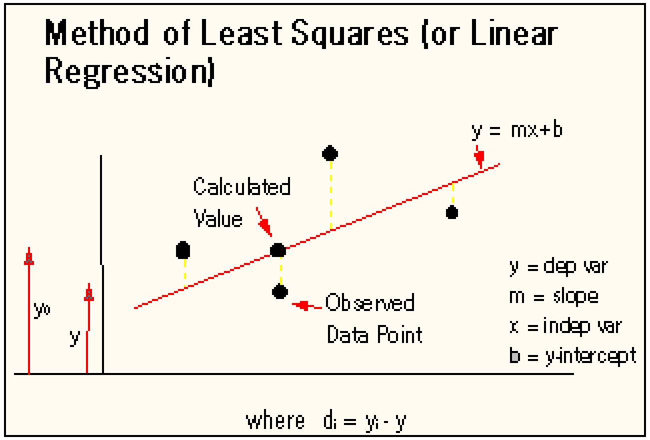
- So the line defined by two points $\mathbf{x}_1$, $\mathbf{x}_2$ is the solution to the following system of equations:

$$\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \end{pmatrix} \begin{pmatrix} m \\ b \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

Heath ch 3,7

Fitting Lines

- With more than two points, there is no guarantee that they will all be on the same line
- Least-squares solution obtained from pseudoinverse is line that is “closest” to all of the points



courtesy of Vanderbilt U.

Example: Fitting a Line

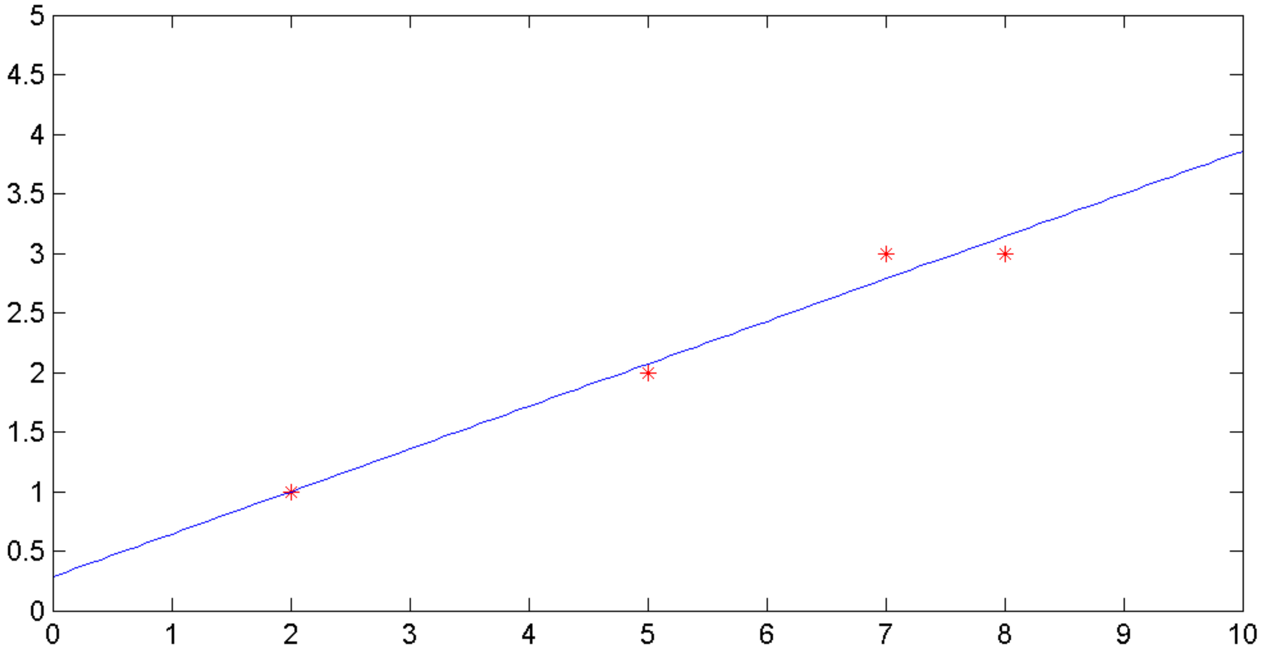
- Suppose we have points (2, 1), (5, 2), (7, 3), and (8, 3)
- Then

$$\begin{pmatrix} 2 & 1 \\ 5 & 1 \\ 7 & 1 \\ 8 & 1 \end{pmatrix} \begin{pmatrix} m \\ b \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 3 \end{pmatrix} \to \mathbf{A}^+ = \begin{pmatrix} -0.1667 & -0.0238 & 0.0714 & 0.1190 \\ 1.1667 & 0.3810 & -0.1429 & -0.4048 \end{pmatrix}$$

and $\mathbf{x} = \mathbf{A}^+\mathbf{b} = (0.3571, 0.2857)^T$

Matlab: **x = A\b**

Example: Fitting a Line



Homogeneous Systems of Equations

- Suppose we want to solve $\mathbf{Ax} = \mathbf{0}$
- There is a trivial solution $\mathbf{x} = \mathbf{0}$, but we don't want this. For what other values of $\mathbf{x}$ is $\mathbf{Ax}$ close to $\mathbf{0}$?
- This is satisfied by computing the **singular value decomposition** (SVD) $\mathbf{A} = \mathbf{UDV}^T$ (a non-negative diagonal matrix between two orthogonal matrices) and taking $\mathbf{x}$ as the last column of $\mathbf{V}$
  - Note that Matlab returns `[U, D, V] = svd(A)`

Line-Fitting as a Homogeneous System

- A 2-D homogeneous point $\mathbf{x} = (x, y, 1)^T$ is on the line $\mathbf{l} = (a, b, c)^T$ only when

$$ax + by + c = 0$$

- We can write this equation with a dot product:

$\mathbf{x} \cdot \mathbf{l} = 0$, and hence the following system is implied for multiple points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$:

$$\begin{pmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n^T \end{pmatrix} \mathbf{l} = \mathbf{0}$$

Heath example 3.21

Example: Homogeneous Line-Fitting

- Again we have 4 points, but now in homogeneous form:

  (2, 1, 1), (5, 2, 1), (7, 3, 1), and (8, 3, 1)

- Our system is:

$$\mathbf{A}\mathbf{x} = \begin{pmatrix} 2 & 1 & 1 \\ 5 & 2 & 1 \\ 7 & 3 & 1 \\ 8 & 3 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \mathbf{0}$$

- Taking the SVD of **A**, we get:

compare to **x** = $(0.3571, 0.2857)^T$

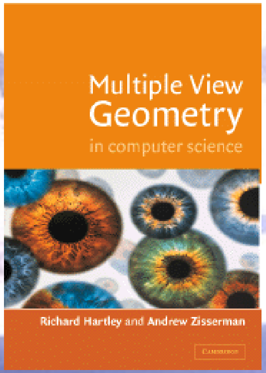
$$\mathbf{V} = \begin{pmatrix} -0.9183 & 0.2334 & 0.3197 \\ -0.3690 & -0.2128 & -0.9047 \\ -0.1431 & -0.9488 & 0.2816 \end{pmatrix}$$

a/-b=0.3534

c/-b=0.3113

Computer Vision

The 2D projective plane and it's applications



Hartley Zisserman Ch 2. In particular: Ch 2.1-4, 2.7,

Estimation: HZ: Ch 4.1-4.2.5, 4.4.4-4.8 cursorly

(Szelisky: Ch 2.1.1, 2.1.2)

Richard Hartley and Andrew Zisserman, Multiple View Geometry, Cambridge University Publishers, 2nd ed. 2004

Parameter estimation in geometric transforms

cs428

- 2D homography

  Given a set of (x_i, x_i'), compute H $(x_i' = Hx_i)$

- 3D to 2D camera projection

  Given a set of (X_i, x_i), compute P $(x_i = PX_i)$

- Fundamental matrix

  Given a set of (x_i, x_i'), compute F $(x_i'^T F x_i = 0)$

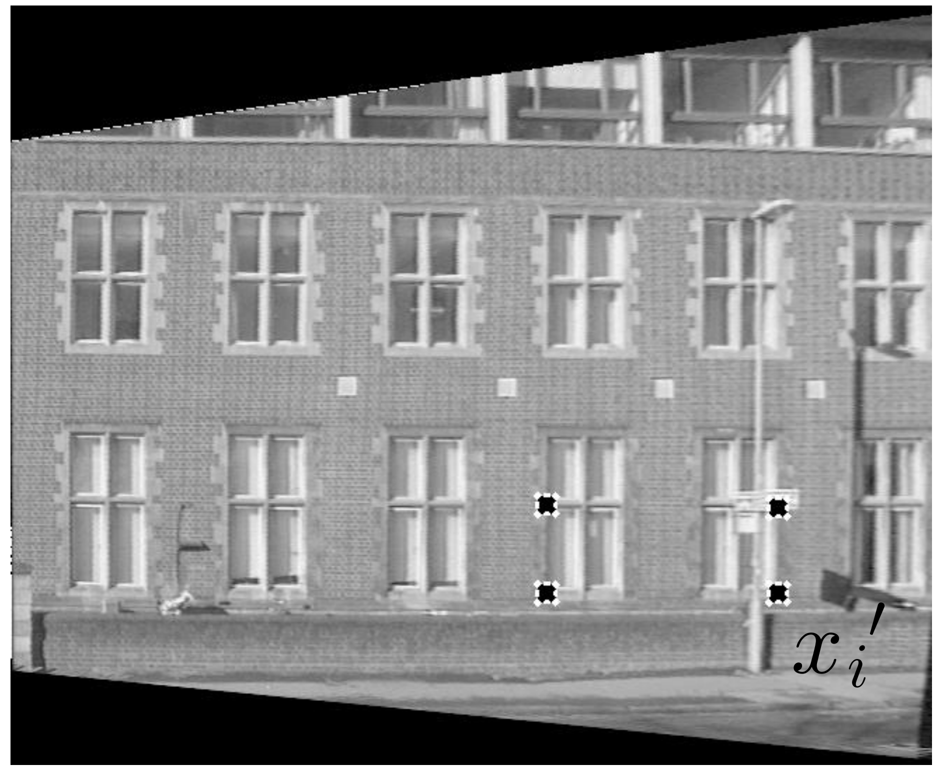
Useful in Grad research

- Trifocal tensor

  Given a set of (x_i, x_i', x_i''), compute T

Estimating Homography H given image points x

HZ Ch 4



HZ

A novel view rendered via four points with known structure

$$x_i' = Hx_i$$

$$i = 1 \ldots 4$$

Number of measurements required

- At least as many independent equations as degrees of freedom required
- Example:

$$\lambda \begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\mathbf{x}' = \mathbf{H}\mathbf{x}$$

2 independent equations / point
8 degrees of freedom

4x2≥8

Approximate solutions

- Minimal solution

    4 points yield an exact solution for H

- More points
    - No exact solution, because measurements are inexact ("noise")
    - Search for "best" according to some cost function
    - Algebraic or geometric/statistical cost

Estimating **H**: The Direct Linear Transformation (DLT) Algorithm

HZ Ch 4.1

- $\mathbf{x}_i = \mathbf{H}\mathbf{X}_i$ is an equation involving homogeneous vectors, so $\mathbf{H}\mathbf{X}_i$ and $\mathbf{x}_i$ need only be in the same direction, not strictly equal

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} \lambda\mathbf{x} \\ \lambda \end{pmatrix}$$

- We can specify "same directionality" by using a cross product formulation:

$$\mathrm{x}'_i \times \mathrm{H}\mathrm{x}_i = 0$$

Direct Linear Transformation (DLT)

$$\lambda \mathrm{x}'_i = \mathrm{H}\mathrm{x}_i$$

$$\mathrm{x}'_i \times \mathrm{H}\mathrm{x}_i = 0 \qquad \mathrm{x}'_i = (x'_i, y'_i, w'_i)^\mathsf{T} \qquad \mathrm{H}\mathrm{x}_i = \begin{pmatrix} \mathrm{h}^{1\mathsf{T}}\mathrm{x}_i \\ \mathrm{h}^{2\mathsf{T}}\mathrm{x}_i \\ \mathrm{h}^{3\mathsf{T}}\mathrm{x}_i \end{pmatrix}$$

$$\mathrm{x}'_i \times \mathrm{H}\mathrm{x}_i = \begin{pmatrix} y'_i\mathrm{h}^{3\mathsf{T}}\mathrm{x}_i - w'_i\mathrm{h}^{2\mathsf{T}}\mathrm{x}_i \\ w'_i\mathrm{h}^{1\mathsf{T}}\mathrm{x}_i - x'_i\mathrm{h}^{3\mathsf{T}}\mathrm{x}_i \\ x'_i\mathrm{h}^{2\mathsf{T}}\mathrm{x}_i - y'_i\mathrm{h}^{1\mathsf{T}}\mathrm{x}_i \end{pmatrix}$$

$$\begin{bmatrix} 0^\mathsf{T} & -w'_i\mathrm{x}_i^\mathsf{T} & y'_i\mathrm{x}_i^\mathsf{T} \\ w'_i\mathrm{x}_i^\mathsf{T} & 0^\mathsf{T} & -x'_i\mathrm{x}_i^\mathsf{T} \\ -y'_i\mathrm{x}_i^\mathsf{T} & x'_i\mathrm{x}_i^\mathsf{T} & 0^\mathsf{T} \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

$$\mathrm{A}_i\mathrm{h} = 0$$

Direct Linear Transformation (DLT)

- Equations are linear in $\mathbf{h}$

$$A_i \mathbf{h} = 0$$

- Only 2 out of 3 are linearly independent (indeed, 2 eq/pt)

$$\begin{bmatrix} 0^\top & -w_i' \mathbf{x}_i^\top & y_i' \mathbf{x}_i^\top \\ w_i' \mathbf{x}_i^\top & 0^\top & -x_i' \mathbf{x}_i^\top \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = 0$$

$$x_i' A_i^1 + y_i' A_i^2 + w_i' A_i^3 = 0$$

(only drop third row if $w_i' \neq 0$)

- Holds for any homogeneous representation, e.g. $(x_i', y_i', 1)$

Direct Linear Transformation (DLT)

- Solving for H

$$\begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{bmatrix} h = 0$$

$$Ah = 0$$

size A is 8x9 or 12x9, but rank 8

Trivial solution is $h=0_9^T$ is not interesting

1-D null-space yields solution of interest

pick for example the one with $\|h\| = 1$

Obtain SVD of A. Solution for h is last column of V

Direct Linear Transformation (DLT)

- **Over-determined solution:** more than 4 p-p corresp

$$\begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_n \end{bmatrix} h = 0$$

$$Ah = 0$$

No exact solution because of inexact measurement i.e. "noise"

Find approximate solution

- Additional constraint needed to avoid 0, e.g. $\|h\| = 1$
- $Ah = 0$ not possible, so minimize $\|Ah\|$

DLT algorithm

<u>Objective</u>

Given $n \geq 4$ 2D to 2D point correspondences $\{x_i \leftrightarrow x_i'\}$, determine the 2D homography matrix H such that $x_i' = Hx_i$

<u>Algorithm</u>

(i) For each correspondence $x_i \leftrightarrow x_i'$ compute A_i. Usually only two first rows needed.

(ii) Assemble *n* 2x9 matrices A_i into a single 2*n*x9 matrix A

(iii) Obtain SVD of A. Solution for h is last column of V

(iv) Determine H from h (reshape)

Inhomogeneous solution

Since h can only be computed up to scale, pick $h_j=1$, e.g. $h_9=1$, and solve for 8-vector $\tilde{h}$

$$\begin{bmatrix} 0 & 0 & 0 & -x_i w_i' & -y_i w_i' & -w_i w_i' & x_i y_i' & y_i y_i' \\ x_i w_i' & y_i w_i' & w_i w_i' & 0 & 0 & 0 & x_i x_i' & y_i x_i' \end{bmatrix} \tilde{h} = \begin{pmatrix} -w_i y_i' \\ w_i x_i' \end{pmatrix}$$

Solve using Gaussian elimination (4 points) or using linear least-squares (more than 4 points)

However, if $h_9=0$ this approach fails
also poor results if h_9 close to zero
Therefore, not recommended for general homographies

Note $h_9=H_{33}=0$ if origin is mapped to infinity

$$l_\infty^T H x_0 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} H \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

Algebraic distance

DLT minimizes $\|\mathrm{Ah}\|$

$e = \mathrm{Ah}$ residual vector

e_i partial vector for each ($\mathrm{x}_i \leftrightarrow \mathrm{x}_i'$)

algebraic error vector

$$d_{\mathrm{alg}}(\mathrm{x}_i', \mathrm{Hx}_i)^2 = \|e_i\|^2 = \left\| \begin{bmatrix} 0^\mathsf{T} & -w_i'\mathrm{x}_i^\mathsf{T} & -y_i'\mathrm{x}_i^\mathsf{T} \\ -w_i'\mathrm{x}_i^\mathsf{T} & 0^\mathsf{T} & -x_i'\mathrm{x}_i^\mathsf{T} \end{bmatrix} \mathrm{h} \right\|^2$$

algebraic distance

$$d_{\mathrm{alg}}(\mathrm{x}_1, \mathrm{x}_2)^2 = a_1^2 + a_2^2 \quad \text{where} \quad \mathrm{a} = (a_1, a_2, a_3)^\mathsf{T} = \mathrm{x}_1 \times \mathrm{x}_2$$

$$\sum_i d_{\mathrm{alg}}(\mathrm{x}_i', \mathrm{Hx}_i)^2 = \sum_i \|e_i\|^2 = \|\mathrm{Ah}\|^2 = \|e\|^2$$

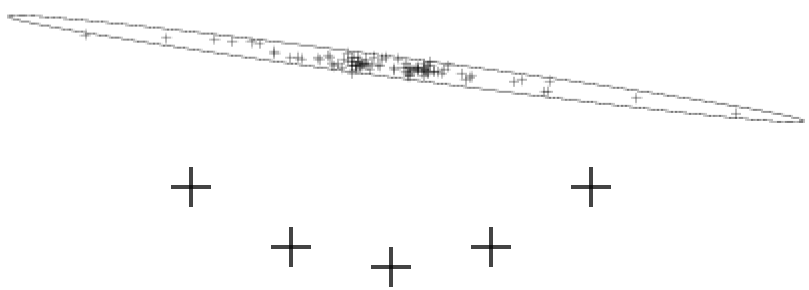
Not geometrically/statistically meaningfull, but given good normalization it works fine and is very fast (use for

DLT:Importance of normalization

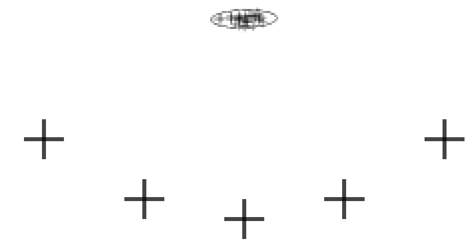
$$\begin{bmatrix} 0 & 0 & 0 & -x_i' & -y_i' & -1 & y_i'x_i & y_i'y_i & y_i' \\ x_i & y_i & 1 & 0 & 0 & 0 & -x_i'x_i & -x_i'y_i & -x_i' \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

$\sim 10^2$ $\sim 10^2$ 1 $\sim 10^2$ $\sim 10^2$ 1 $\sim 10^4$ $\sim 10^4$ $\sim 10^2$

orders of magnitude difference!



Un-normnalized



normnalized

Normalizing transformations

- Since DLT is not invariant to coordinate transforms, what is a good choice of coordinates?

  e.g.

  – Translate centroid to origin

  – Scale to a $\sqrt{2}$ average distance to the origin

  – Independently on both images

Or

$$\mathrm{T}_{\mathrm{norm}} = \begin{bmatrix} w+h & 0 & w/2 \\ 0 & w+h & h/2 \\ 0 & 0 & 1 \end{bmatrix}^{-1}$$

Normalized DLT algorithm

Objective

Given n≥4 2D to 2D point correspondences $\{x_i \leftrightarrow x_i'\}$, determine the 2D homography matrix H such that $x_i' = Hx_i$

Algorithm

(i) Normalize points $\tilde{x}_i = T_{norm} x_i, \tilde{x}_i' = T'_{norm} x_i'$

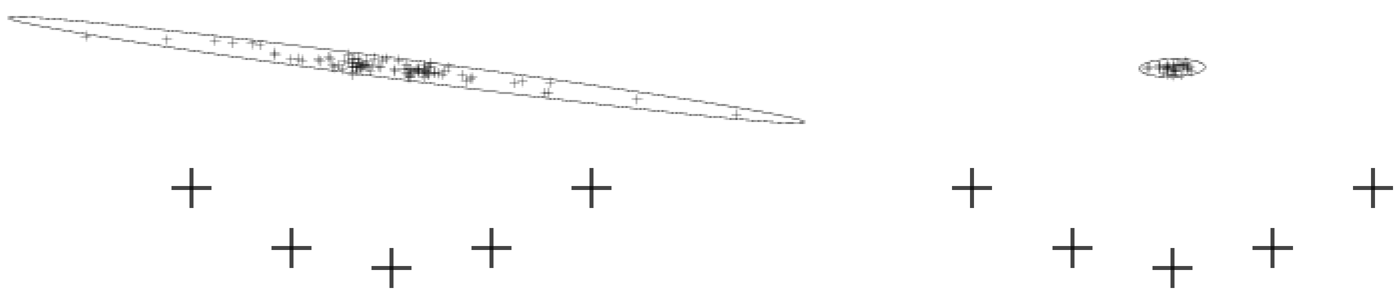
(ii) Apply DLT algorithm to $\tilde{x}_i \leftrightarrow \tilde{x}_i',$

(iii) Denormalize solution $H = T'^{-1}_{norm} \tilde{H} T_{norm}$

DLT:Importance of normalization

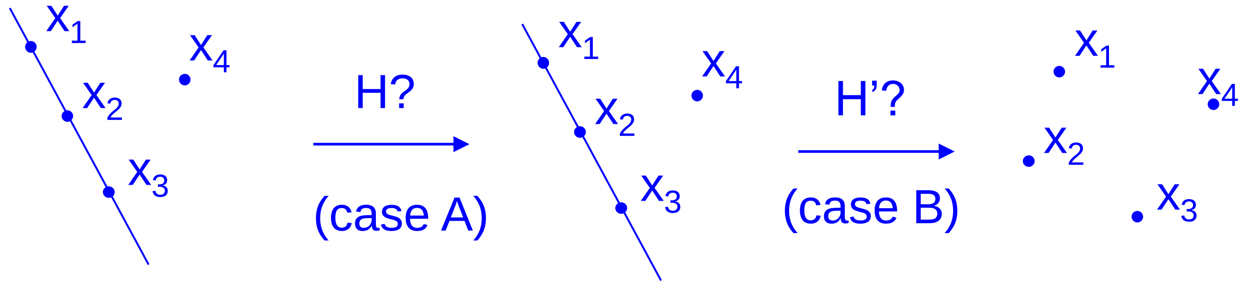
$$\begin{bmatrix} 0 & 0 & 0 & -x'_i & -y'_i & -1 & y'_i x_i & y'_i y_i & y'_i \\ x_i & y_i & 1 & 0 & 0 & 0 & -x'_i x_i & -x'_i y_i & -x'_i \end{bmatrix} \begin{pmatrix} \mathrm{h}^1 \\ \mathrm{h}^2 \\ \mathrm{h}^3 \end{pmatrix} = 0$$

Un-normalized	$\sim 10^2$	$\sim 10^2$	1	$\sim 10^2$	$\sim 10^2$	1	$\sim 10^4$	$\sim 10^4$	$\sim 10^2$
normalized	~1	~1	1	~1	~1	1	~1	~1	~1



Un-normalized normalized

Degenerate configurations



Constraints: $\mathrm{x}_i' \times \mathrm{H}\mathrm{x}_i = 0$ i=1,2,3,4

Define: $\mathrm{H}^* = \mathrm{x}_4' \mathrm{l}^{\mathsf{T}}$

Then, $\mathrm{H}^*\mathrm{x}_i = \mathrm{x}_4'\left(\mathrm{l}^{\mathsf{T}}\mathrm{x}_i\right) = 0,\ i = 1,2,3$

$\mathrm{H}^*\mathrm{x}_4 = \mathrm{x}_4'\left(\mathrm{l}^{\mathsf{T}}\mathrm{x}_4\right) = k\mathrm{x}_4'$

H^* is rank-1 matrix and thus not a homography

If H^* is unique solution, then no homography mapping $x_i \rightarrow x_i'$(case B)
If further solution H exist, then also $\alpha\mathrm{H}^* + \beta\mathrm{H}$ (case A)
(2-D null-space in stead of 1-D null-space)

Solutions from lines, etc.

2D homographies from 2D lines

$$l'_i = H^T l_i \qquad Ah = 0$$

Minimum of 4 lines

3D Homographies (15 dof)

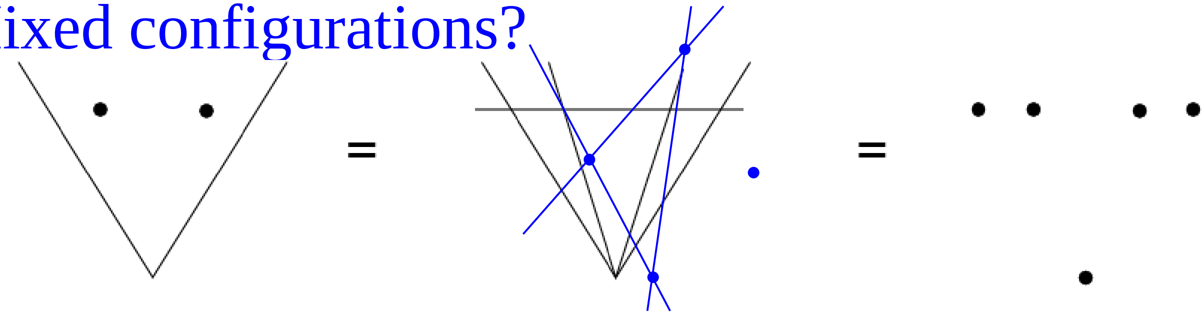
Minimum of 5 points or 5 planes

2D affinities (6 dof)

Minimum of 3 points or lines

Conic provides 5 constraints

Mixed configurations?



Homography: Summary

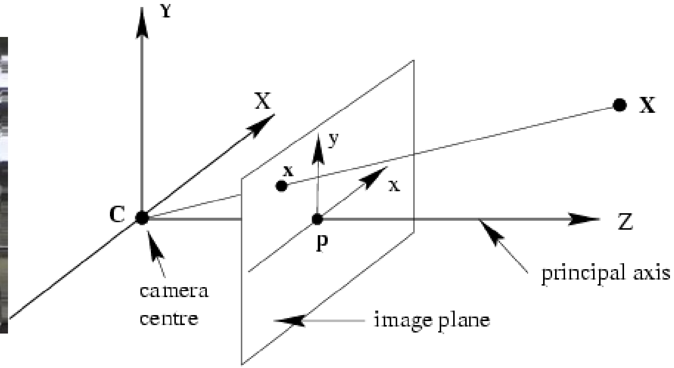
- Direct Linear Transform
- Inhomogenous solution
- Projective – Affine (- Metric) upgrade

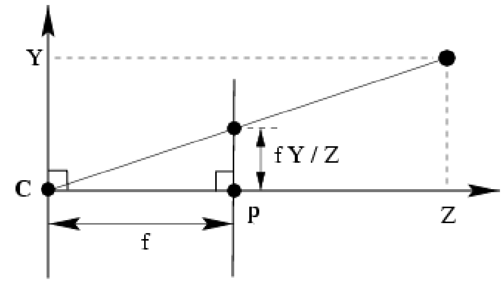
- Non-linear computation (Tracking)

- First 3D proj geom
- Then review and more on camera models
- Then following P estimation

Camera Models

Mostly pinhole camera model



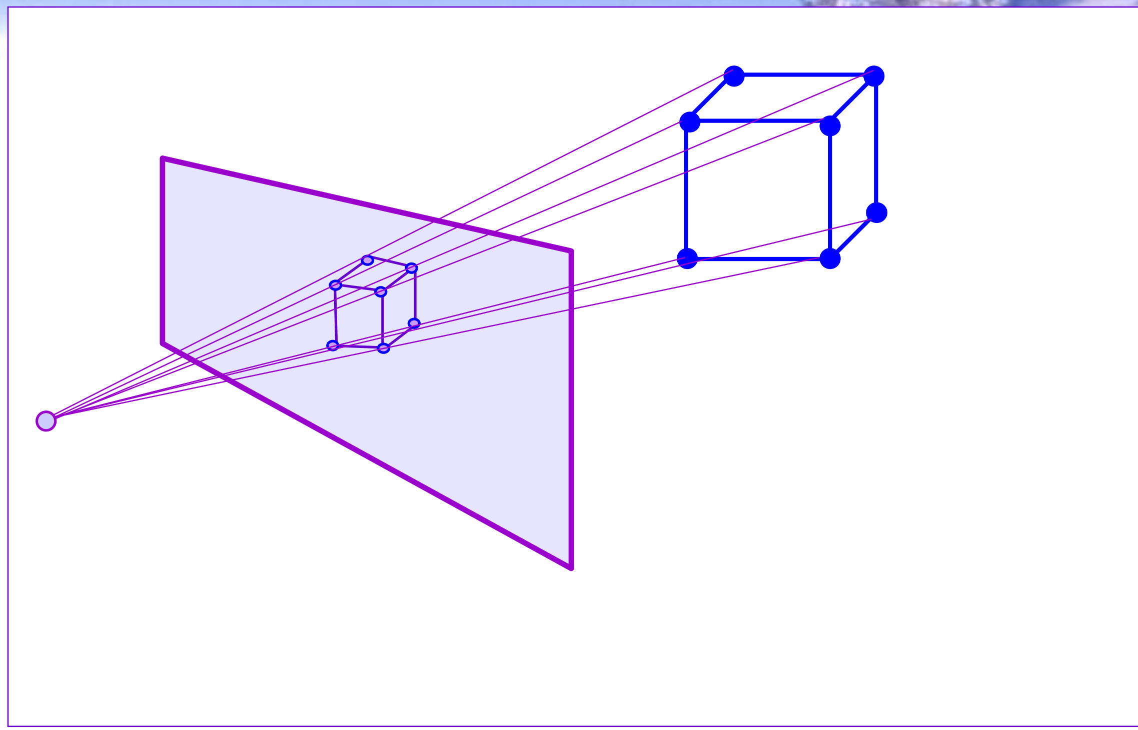


$$\lambda\begin{bmatrix}x\\y\\1\end{bmatrix}=\begin{bmatrix}f & & p_x\\ & f & p_y\\ & & 1\end{bmatrix}\begin{bmatrix}1 & & & \\ & 1 & & \\ & & 1 & \end{bmatrix}\begin{bmatrix}\mathbf{R} & \mathbf{t}\\ 0_3^{\mathsf{T}} & 1\end{bmatrix}\begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}$$

$$\text{or} \quad \lambda\,\mathrm{x}=\mathbf{P}.\mathrm{X}$$

$$\begin{pmatrix}U\\V\\W\end{pmatrix}=\begin{pmatrix}\text{Transforma tion}\\\text{representi ng}\\\text{intrinsic parameters}\end{pmatrix}\begin{pmatrix}\text{Transforma tion}\\\text{representi ng}\\\text{projection model}\end{pmatrix}\begin{pmatrix}\text{Transforma tion}\\\text{representi ng}\\\text{extrinsic parameters}\end{pmatrix}\begin{pmatrix}X\\Y\\Z\\T\end{pmatrix}$$

A 3D Vision Problem: Multi-view geometry - resection

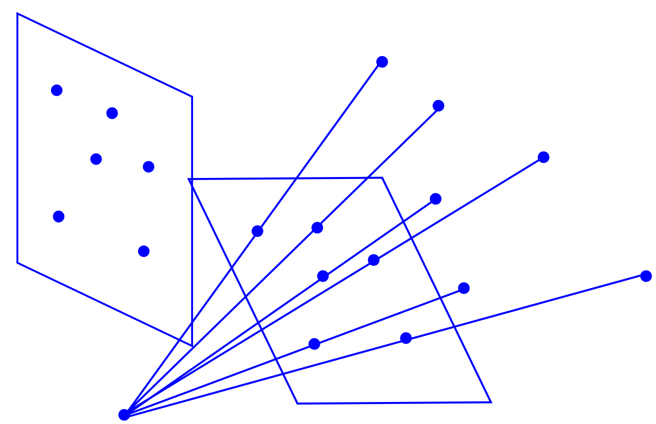


- Projection equation

  $x_i = P_i X$

- Resection:
  - $x_i, X \longrightarrow P_i$

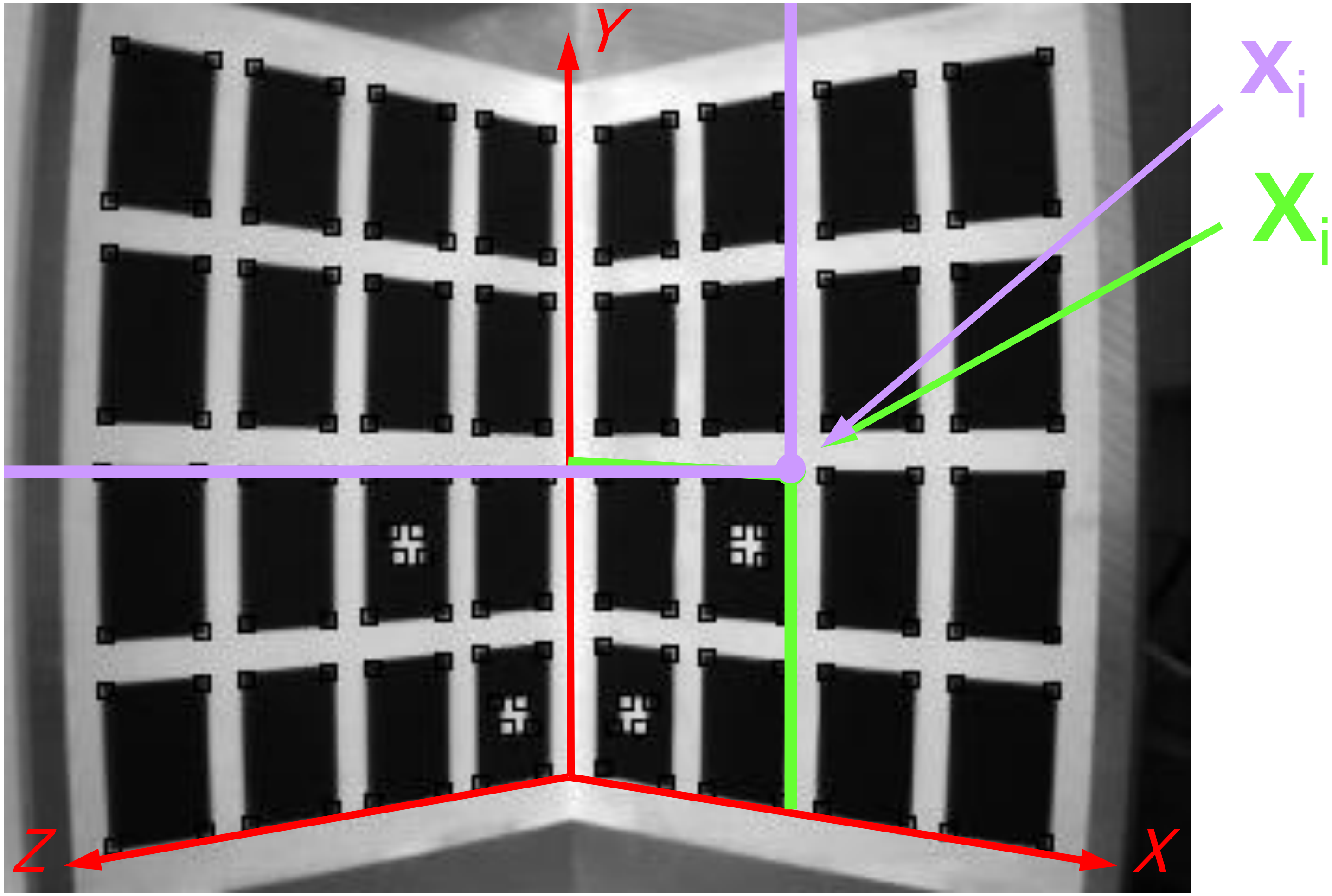
Given image points and 3D points calculate camera projection matrix.



Estimating camera matrix $\mathbf{P}$

- Given a number of correspondences between 3-D points and their 2-D image projections $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$, we would like to determine the **camera projection matrix** $\mathbf{P}$ such that $\mathbf{x}_i = \mathbf{P}\mathbf{X}_i$ for all i

A Calibration Target



courtesy of B. Wilburn

Estimating **P**: The Direct Linear Transformation (DLT) Algorithm

- $\mathbf{x}_i = \mathbf{P}\mathbf{X}_i$ is an equation involving homogeneous vectors, so $\mathbf{P}\mathbf{X}_i$ and $\mathbf{x}_i$ need only be in the same direction, not strictly equal

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} \lambda\mathbf{x} \\ \lambda \end{pmatrix}$$

- We can specify "same directionality" by using a cross product formulation:

$$\mathbf{x}_i \times \mathbf{P}\mathbf{X}_i = \mathbf{0}$$

DLT Camera Matrix Estimation: Preliminaries

- Let the image point $\mathbf{x}_i = (x_i, y_i, w_i)^T$ (remember that $\mathbf{X}_i$ has 4 elements)
- Denoting the jth row of $\mathbf{P}$ by $\mathbf{p}^{jT}$ (a 4-element row vector), we have:

$$\mathbf{P}\mathbf{X}_i = \begin{pmatrix} \mathbf{p}^{1T} \\ \mathbf{p}^{2T} \\ \mathbf{p}^{3T} \end{pmatrix} \mathbf{X}_i = \begin{pmatrix} \mathbf{p}^{1T}\mathbf{X}_i \\ \mathbf{p}^{2T}\mathbf{X}_i \\ \mathbf{p}^{3T}\mathbf{X}_i \end{pmatrix}$$

DLT Camera Matrix Estimation: Step 1

- Then by the definition of the cross product,

$\mathbf{x}_i \times \mathbf{P}\mathbf{X}_i$ is:

$$\begin{pmatrix} y_i \mathbf{p}^{3T}\mathbf{X}_i - w_i \mathbf{p}^{2T}\mathbf{X}_i \\ w_i \mathbf{p}^{1T}\mathbf{X}_i - x_i \mathbf{p}^{3T}\mathbf{X}_i \\ x_i \mathbf{p}^{2T}\mathbf{X}_i - y_i \mathbf{p}^{1T}\mathbf{X}_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \mathbf{0}$$

DLT Camera Matrix Estimation: Step 2

- The dot product commutes, so $\mathbf{p}^{jT}\mathbf{X}_i = \mathbf{X}^T_i \mathbf{p}^j$, and we can rewrite the preceding as:

$$\begin{pmatrix} y_i \mathbf{X}_i^T \mathbf{p}^3 - w_i \mathbf{X}_i^T \mathbf{p}^2 \\ w_i \mathbf{X}_i^T \mathbf{p}^1 - x_i \mathbf{X}_i^T \mathbf{p}^3 \\ x_i \mathbf{X}_i^T \mathbf{p}^2 - y_i \mathbf{X}_i^T \mathbf{p}^1 \end{pmatrix} = \mathbf{0}$$

DLT Camera Matrix Estimation: Step 3

- Collecting terms, this can be rewritten as a matrix product:

$$\begin{pmatrix} \mathbf{0}^T & -w_i\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ w_i\mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \\ -y_i\mathbf{X}_i^T & x_i\mathbf{X}_i^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix}$$

where $\mathbf{0}^T = (0, 0, 0, 0)$. This is a 3 x 12 matrix times a 12-element column vector $\mathbf{p} = (\mathbf{p}^{1T}, \mathbf{p}^{2T}, \mathbf{p}^{3T})^T$

What We Just Did

$$\begin{pmatrix} y_i \mathbf{X}_i^T \mathbf{p}^3 - w_i \mathbf{X}_i^T \mathbf{p}^2 \\ w_i \mathbf{X}_i^T \mathbf{p}^1 - x_i \mathbf{X}_i^T \mathbf{p}^3 \\ x_i \mathbf{X}_i^T \mathbf{p}^2 - y_i \mathbf{X}_i^T \mathbf{p}^1 \end{pmatrix}$$

$$\Downarrow$$

$$\begin{pmatrix} \mathbf{0}^T & -w_i \mathbf{X}_i^T & y_i \mathbf{X}_i^T \\ w_i \mathbf{X}_i^T & \mathbf{0}^T & -x_i \mathbf{X}_i^T \\ -y_i \mathbf{X}_i^T & x_i \mathbf{X}_i^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix}$$

DLT Camera Matrix Estimation: Step 4

$$\begin{pmatrix} \mathbf{0}^T & -w_i\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ w_i\mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \\ -y_i\mathbf{X}_i^T & x_i\mathbf{X}_i^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix}$$

- There are only two linearly independent rows here
  - The third row is obtained by adding x_i times the first row to y_i times the second and scaling the sum by $-1/w_i$

DLT Camera Matrix Estimation: Step 4

- So we can eliminate one row to obtain the following linear matrix equation for the ith pair of corresponding points:

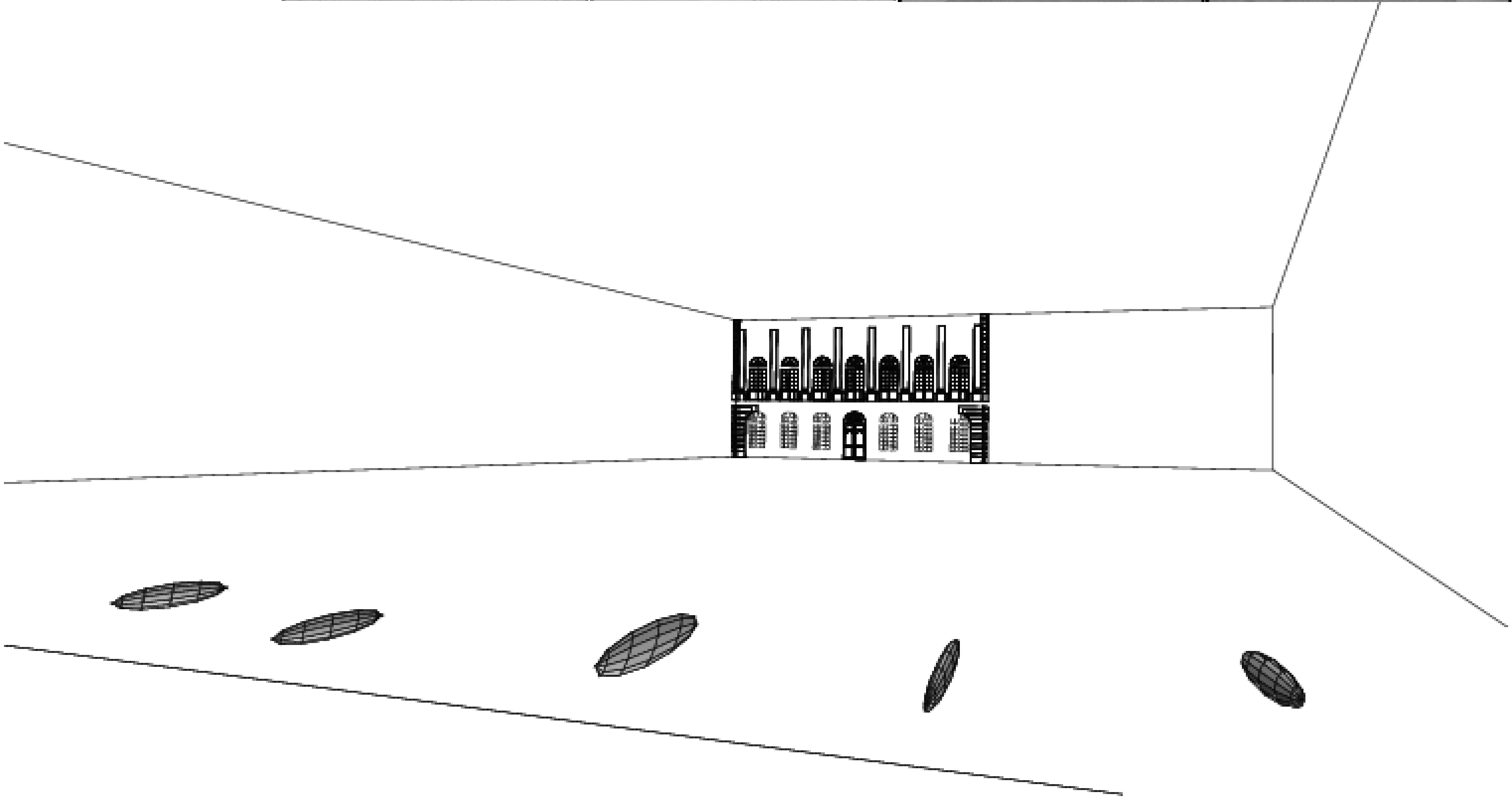
$$\begin{pmatrix} \mathbf{0}^T & -w_i\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ w_i\mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = 0$$

- Write this as $\mathbf{A}_i\,\mathbf{p} = \mathbf{0}$

DLT Camera Matrix Estimation: Step 5

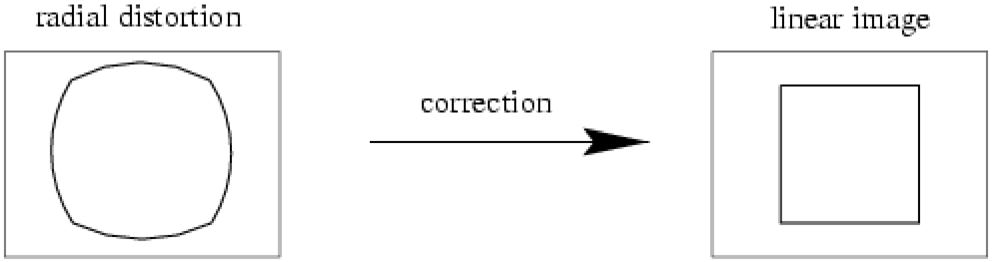
- Remember that there are 11 unknowns which generate the 3 x 4 homogeneous matrix **P** (represented in vector form by **p**)
- Each point correspondence yields 2 equations (the two row of $\mathbf{A}_i$)
- We need at least 5 ½ point correspondences to solve for **p**
- Stack $\mathbf{A}_i$ to get homogeneous linear system $\mathbf{A}\,\mathbf{p} = \mathbf{0}$

Experiment:



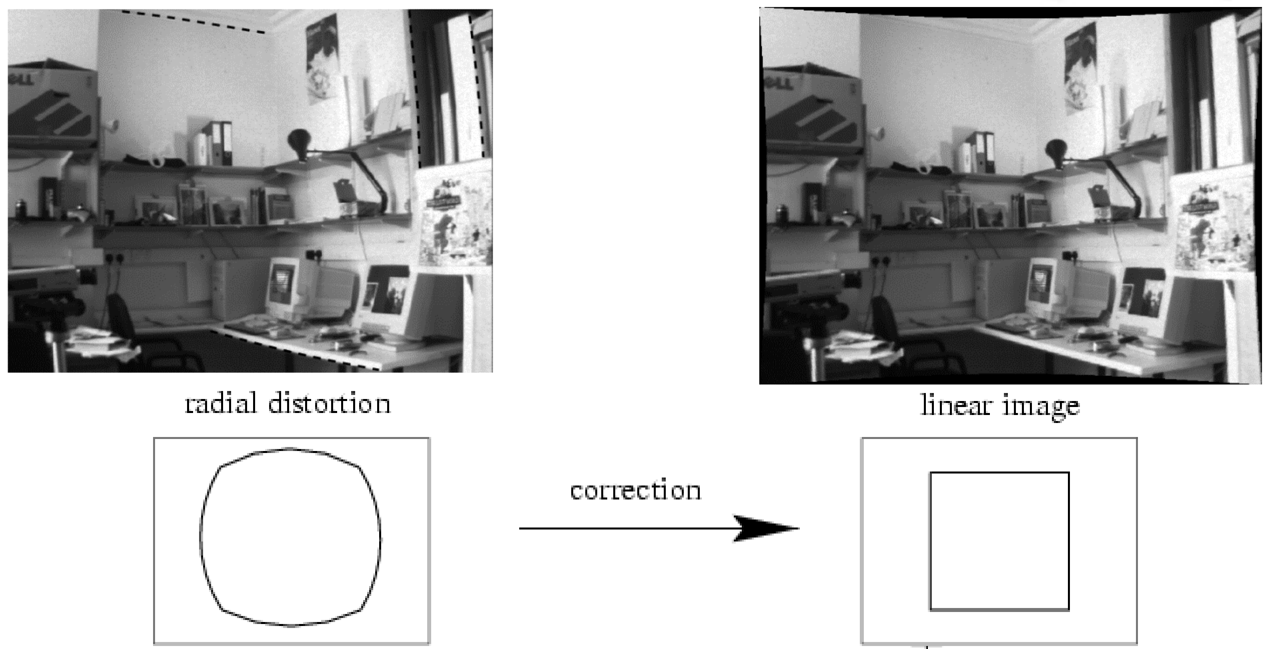
Radial Distortion

short and long focal length



Radial Distortion

Radial Distortion



$$(\tilde{x}, \tilde{y}, 1)^{\top} = [\mathtt{I} \mid \mathbf{0}]\mathbf{X}_{\text{cam}}$$

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = L(\tilde{r}) \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix}$$

Radial Distortion

Correction of distortion

$$\hat{x} = x_c + L(r)(x - x_c) \qquad \hat{y} = y_c + L(r)(y - y_c)$$

Choice of the distortion function and center

$$x = x_o + (x_o - c_x)(K_1 r^2 + K_2 r^4 + \ldots)$$
$$y = y_o + (y_o - c_y)(K_1 r^2 + K_2 r^4 + \ldots)$$

$$r = (x_o - c_x)^2 + (y_o - c_y)^2 \ .$$

Computing the parameters of the distortion function

(i) Minimize with additional unknowns

(ii) Straighten lines

(iii) …