Camera: Geometry and Calibration

P J Narayanan

CVIT, IIIT, Hyderabad

CS5765. Computer Vision. Spring 2013

Course for Spring 2013

Course Topics:

- Geometry of imaging
- Visual recognition
- Segmentation and grouping

Pre-Requisites:

- Linear algebra, Image processing
- Pattern Analysis, Algorithms, Programming

Evaluating performance:

- Mid & final exams, handwritten assignments
- 3-4 programming assmnts, **one starting today!**
- A term-paper, **class participation**
- **I expect you to work hard in this *advanced elective***
- **Casual registrants will be disappointed!**

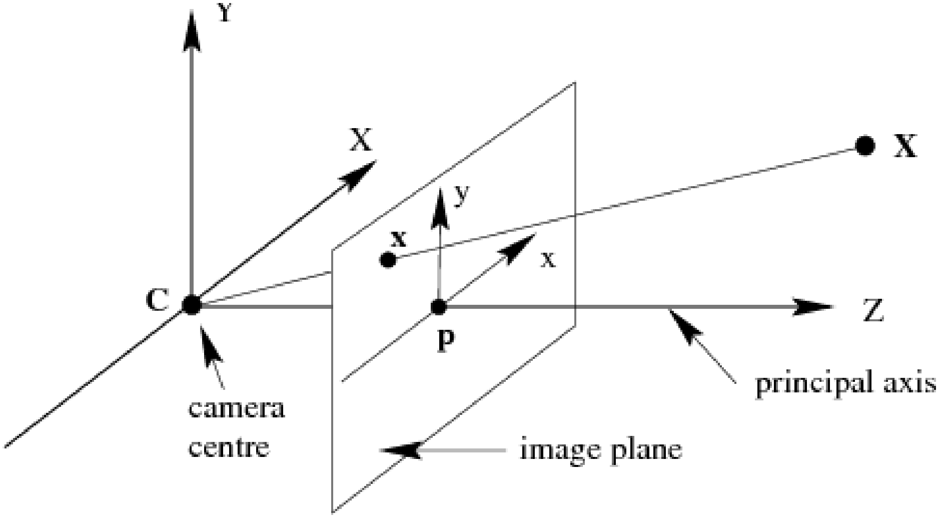
Camera Module: Objectives

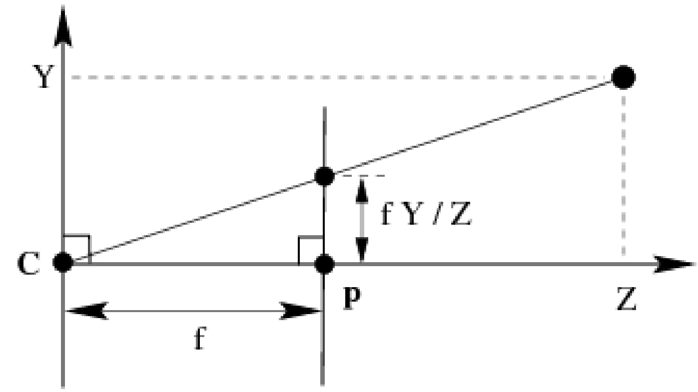
- Mathematically model what a camera does
  – Also understand what the model means
- Getting the model for a real-world camera
  – Estimation from real world measurements
- Special imaging configurations with simpler properties
  – Simpler relationships
- General theory on fitting linear models under noisy observations
  – Techniques that work across problems

What does a camera do?

- Form an image on the 2D image plane of the 3D world visible to it.
- Image is *behind* the lens; the scene is in front.
- 3D world is **projected** down to a 2D plane.
- Significant loss of information as one dimension is dropped.
- Mathematical depiction of this projection ...

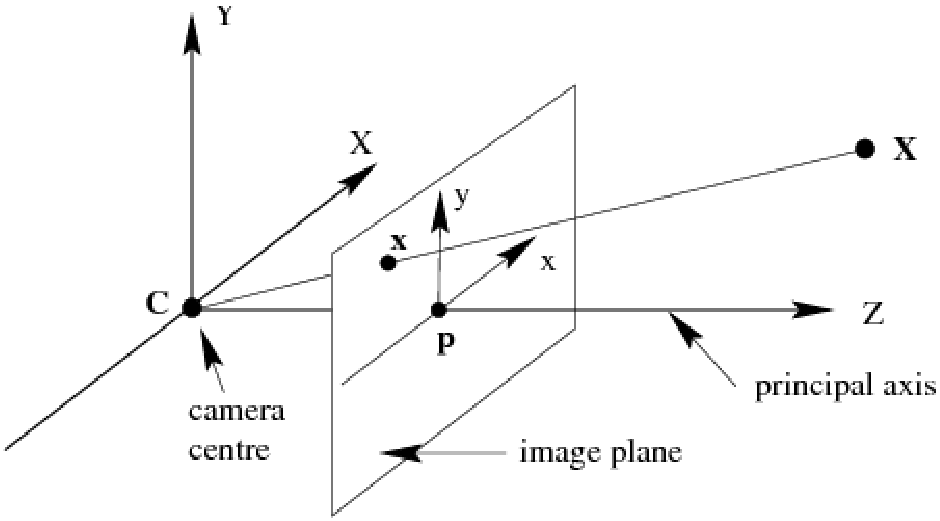
Perspective Projection

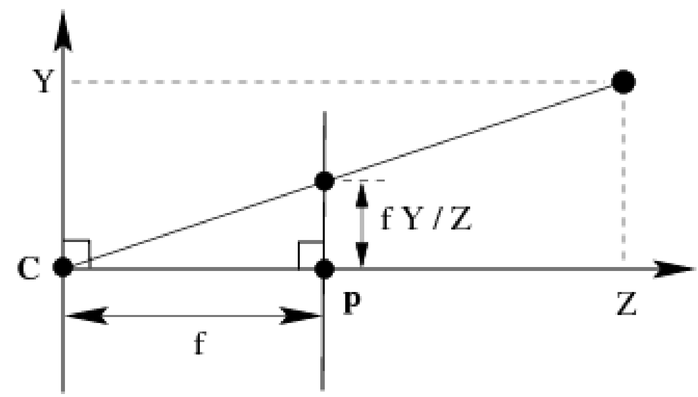




Cartesian image coordinates: $x = f\frac{X}{Z}, \quad y = f\frac{Y}{Z}$

Perspective Projection





Cartesian image coordinates: $x = f\frac{X}{Z}, \quad y = f\frac{Y}{Z}$

In matrix form using homogeneous coordinates:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{P}\,\mathbf{X}$$

Basic Camera Equation

A pin-hole camera projects a 3D point X_c in camera coords to an image point x via the 3×4 **camera matrix** P as:

$$x \;=\; P\,X_c = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \;[I \mid 0]\; X_c = K\;[I\,|\,0]\; X_c,$$

K: **(internal) camera calibration matrix**

Properties:

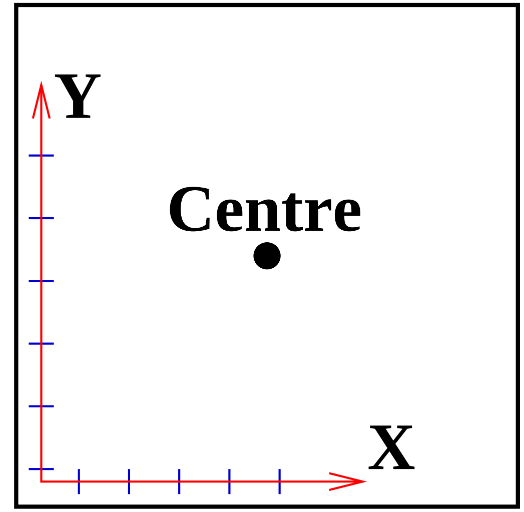
- Camera is located at origin
- Z-axis is the Camera or Optical axis
- *Principal point* or *image centre*: centre of the image
- Focal length expressed in pixel units
- Orthogonal image axes, uniform scales for each axis

A General Camera

Image centre at (x_0, y_0) in image co-ordinates

Non-orthogonal axes, with skew s

Different scales for each axis,
with focal lengths α_x and α_y

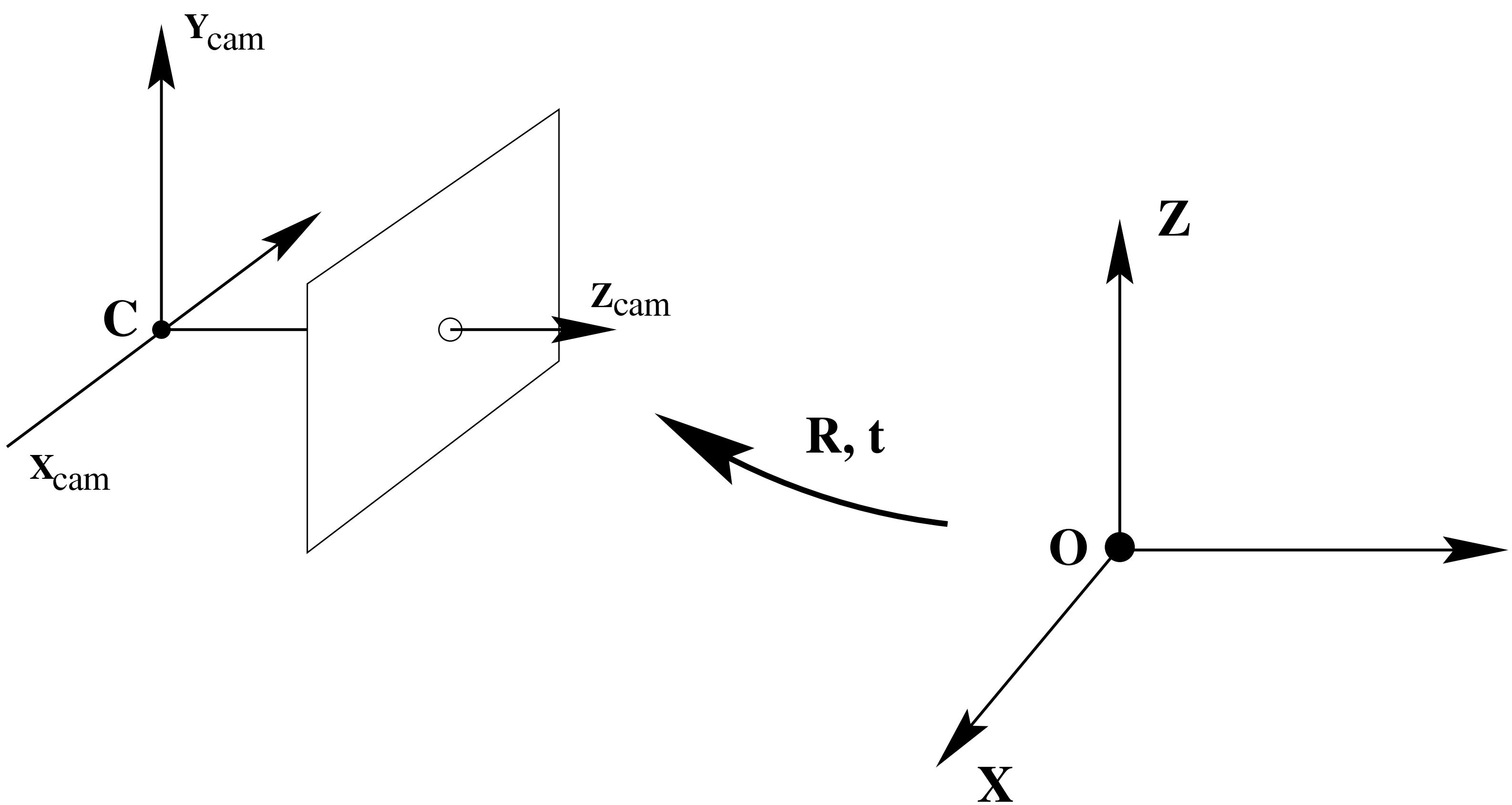


$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

K: An upper diagonal matrix with 5 degrees of freedom

Camera Orientation

- General setting: A *world coordinate system*. Camera not at origin, Z not the optical axis.
- Camera is a point C in world coords. A rotation given by a matrix R relates the coordinate sytems.



General Camera Equation

Camera and world coordintes related by

$$X_c = \begin{bmatrix} R & -RC \\ 0 & 1 \end{bmatrix} X_w$$

2D projection x of a 3D point X_w given by

$$x = K\,[I|0]\,X_c = K\,[R \mid -RC]\,X_w$$

$x = PX_w$ with camera matrix $P = [KR \mid -KRC] = [M \mid \mathbf{p_4}]$

Common $K = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix}$. General $K = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$

General Camera Equation

General projection equation in world coordinates:

$$x = K\,[R|\,{-RC}]\,X_w = [KR\mid -KRC]\,X_w = [M\mid \mathbf{p_4}]\,X_w$$

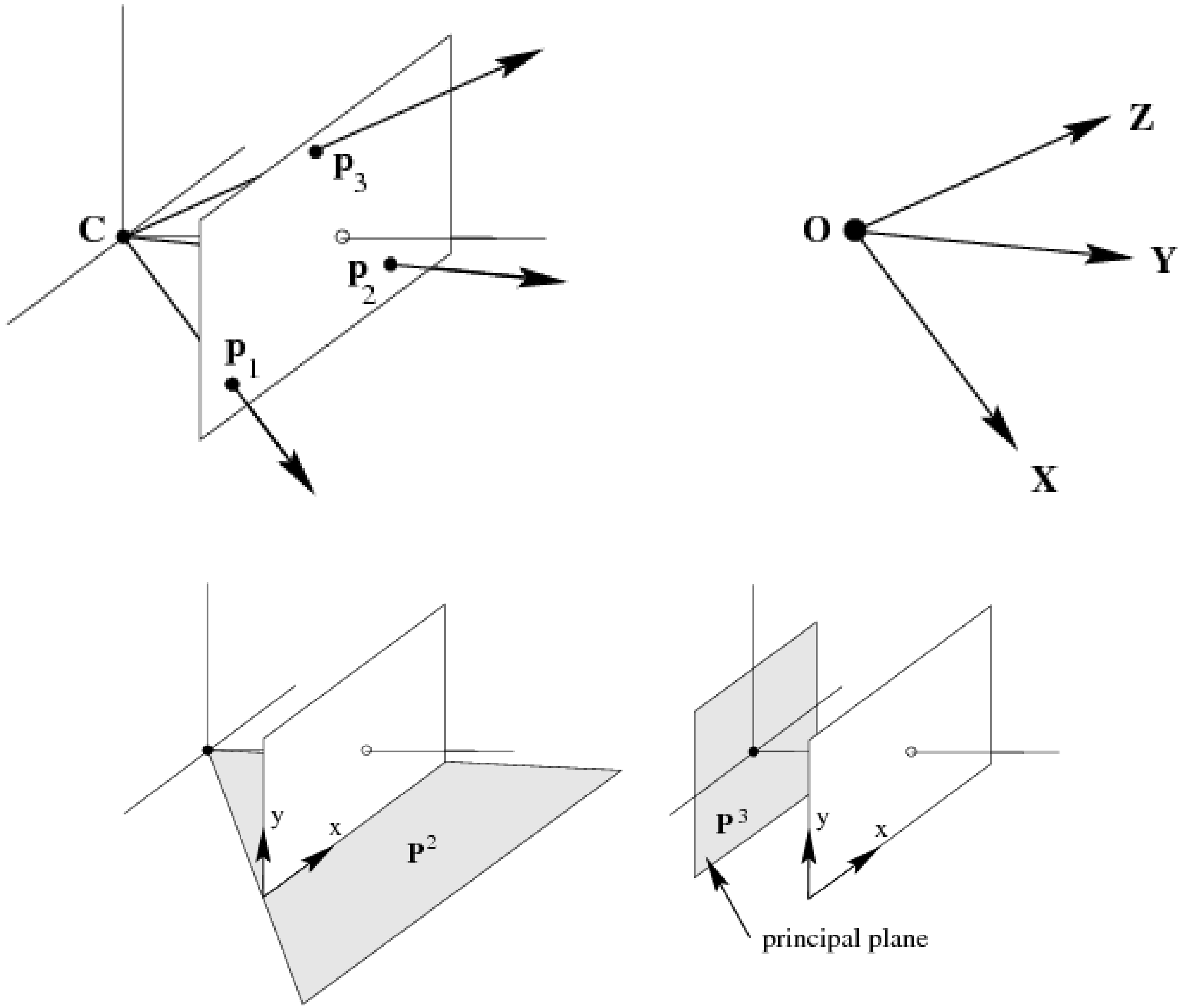
3×4 matrix P maps/projects world coords to image coords.
– Left 3×3 submatrix is non-singluar for finite cameras
– Orthographic projection: left submatrix is singular

Any 3×3 matrix P with non-singular left submatrix represents a camera! It can also be decomposed into:

– A non-singular upper diagonal matrix K
– An orthonormal matrix R and a vector C

with the usual meanings!!

Axis Points and Planes



Camera Matrix Anatomy

$$P = [\mathbf{p_1} \;\; \mathbf{p_2} \;\; \mathbf{p_3} \;\; \mathbf{p_4}] = [\mathbf{P^1} \;\; \mathbf{P^2} \;\; \mathbf{P^3}]^T$$

- 4-vector C with $PC = 0$ is the camera centre.

  Camera centre is the only point with no projection or projects to vector $\mathbf{0}$ which is undefined in P^2

- Columns $\mathbf{p_1}, \mathbf{p_2}, \mathbf{p_3}$ are the images of vanishing points of world X, Y, Z directions.

  $\mathbf{p_1} = \mathbf{P}[\mathbf{1} \;\; \mathbf{0} \;\; \mathbf{0} \;\; \mathbf{0}]^{\mathbf{T}}$, the image of ideal point in X direction. Similary rest

- $\mathbf{p_4}$ is image of world origin. $\mathbf{p_4} = \mathbf{P}[\mathbf{0} \;\; \mathbf{0} \;\; \mathbf{0} \;\; \mathbf{1}]^{\mathbf{T}}$

Camera Anatomy ... contd

- Row vector $\mathbf{P^3}$ is the principal plane.

  $\mathbf{P^{3T}X} = \mathbf{0}$ is the plane projecting to line at infinity, or the principal plane

- Row vectors $\mathbf{P^1}$ and $\mathbf{P^2}$ are axis planes for image Y and X axes.

  $\mathbf{P^{1T}X} = \mathbf{0}$ is the plane projecting to points $(0, y)$ or to the camera Y-axis

- The principal point (or image center) is given by $\mathbf{x_0} = M\,\mathbf{m^3}$, with $\mathbf{m^3}$ the third row vector of matrix M.

  P^3: principal axis direction, $[m^3\ 0]^T$: its ideal point, Mm^3 of that

- $det(M)\mathbf{m^3}$ gives the principal axis as a vector from the camera centre through the prinicipal point to the front of the camera.

  For $K[I|0]$, m^3 is Z-axis, $det(M)$ is positive. Hence definition

Backprojecting Points

- An image point x corresponds to a ray starting from C through point x. Camera matrix is $P = [M \mid t]$
- If $\mathbf{P^+ = P^T(PP^T)^{-1}}$, P^+x projects to x
- The ray can be given by: $C + \lambda P^+ x$ for $\lambda > 0$.
- $C = \begin{bmatrix} -M^{-1}t \\ 1 \end{bmatrix}$ and $\begin{bmatrix} M^{-1}x \\ 0 \end{bmatrix}$ and hence $\begin{bmatrix} M^{-1}(\lambda x - t) \\ 1 \end{bmatrix}$ are on the ray for $\lambda > 0$

  $P = [M|t] = [KR| -KRC]$. So, $C = -M^{-1}t$. $P[M^{-1}x \;\; 0]^T = MM^{-1}x$
- Depth of a 3D point X: $sgn(det(M))\mathbf{m^{3T}}(X - C)$, if $\mathbf{m^3}$ is normalized. This also equals $sgn(det(M))w$ where $\mathbf{PX} = w[x\ y\ 1]^{\mathbf{T}}$

$w =$ third component of $[M| -MC][X \;\; 1]^T$ equals $m^{3T}X - m^{3T}C = m^{3T}(X - C)$

Calibration: Estimating the Camera

Camera Calibration

- Given a point match $\mathbf{x} \leftrightarrow \mathbf{X}$, $\mathbf{x} = \mathbf{PX}$, upto scale.
- In cartesian coordinates (u, v): $u = \frac{P^{1T}X}{P^{3T}X}$, $v = \frac{P^{1T}X}{P^{3T}X}$ or

$$P^{1T}X - uP^{3T}X = 0, \quad P^{2T}X - vP^{3T}X = 0$$

  These equations are linear in entries P^i_j

- Two equations per match. P has 11 real unknowns. Need $\mathbf{6}$ pairs $\mathbf{x}_i \leftrightarrow \mathbf{X}_i$ to estimate P.
- In practice, use many point correspondences! From n matches, form a $2n \times 12$ matrix $\mathbf{A}$ such that $\mathbf{Ap} = \mathbf{0}$ where p is the vector of P^i_j values.
- Solve the above *homogeneous* linear equation for p to get the camera matrix P.

Direct Linear Transformation

- A minimum of 6 points for an exact solution. A Least Squared Error (**LSE**) solution is better in the presence of noise, with as many matches as possible.
- Measurement matrix A is $2n \times 12$ for n observations.
- Solve for $\mathbf{Ap} = \mathbf{0}$, subject to $||p|| = 1$.
  With noise, find $\min_h ||Ap||$, such that $||p|| = 1$
- Using SVD: $\mathbf{A} = \mathbf{U}\,\mathbf{D}\,\mathbf{V}^{\mathbf{T}}$, with singular values of the diagonal matrix $\mathbf{D}$ in decreasing order. $\mathbf{U}$ and $\mathbf{V}$ are orthogonal. Rows of $\mathbf{D}$ are 0 beyond the 12th. Thus,
  $\mathbf{A}_{2n\times 12} = \mathbf{U}_{2n\times 2n}\mathbf{D}_{2n\times 12}\mathbf{V}^T_{12\times 12} = \mathbf{U}_{2n\times 12}\mathbf{D}_{12\times 12}\mathbf{V}^T_{12\times 12}$
- LSE Solution: $\mathbf{V}_i$ corresponding to smallest $D_{ii} > 0$.

SVD-Based Algorithm: Proof

- To solve the homogeneous equation: $Ap = UDV^T p = 0$, such that $||p|| = 1$
- Formulate as the following **LSE** problem: minimize $||UDV^T p|| = ||DV^T p||$ (as U is orthogonal).
- Define $y = V^T p$. $||y|| = ||V^T p|| = ||p||$ (orthogonal V)
- Reduces to: minimize $||Dy||$ subject to $||y|| = 1$.
- Since D has singular values in decreasing order, the solution is: $y = [0, 0, \cdots, 1]^T$, with a 1 for the least singular value.
- $p = Vy$, the last column of V.

Robust Estimation using RANSAC

- DLT estimates the least squared error solution.
  LSE is sensitive to outliers
- Data usually contains *outliers* which don't fit the model.
  How can we recognize them explicitly?
- **Inlier:** points within a small distance of solution.
  Outlier: data points far from the solution
- RANdom SAmpling Consensus (RANSAC) approach fits the model with *maximum* number of inliers

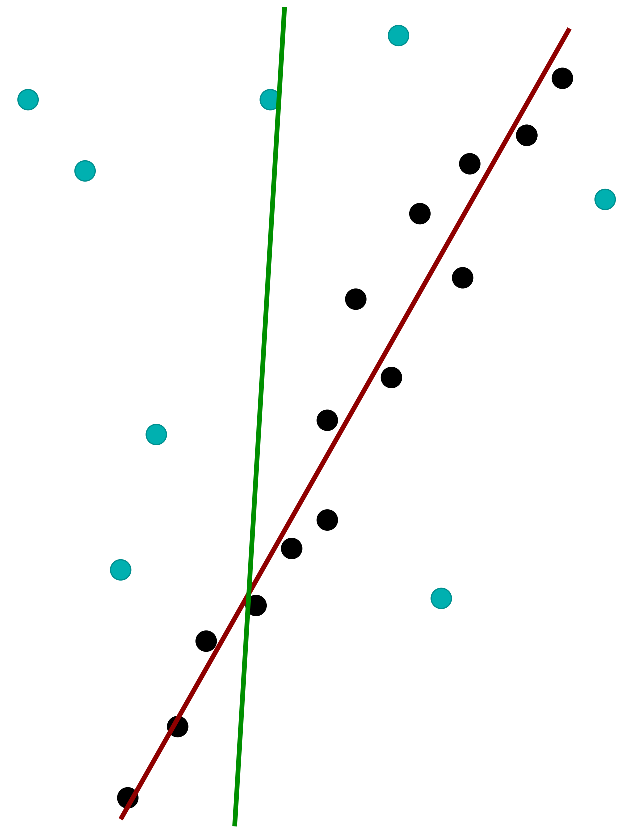
RANSAC [Fischler&Bolles 1981] is a very influential work.

It is used extensively in Computer Vision to estimate camera matrix, homography, fundamental matrix, etc.

RANSAC Line Fitting

Fit a line to points with noise and outliers.

- Pick a random pair of points. Fit a line through them
- Check other points against the line. Count #inliers using a threshold
- Stop if #inliers is high enough
- Else repeat previous steps for a number of iterations
- Pick line with maximum #inliers
- Recompute the line using all inliers suitably



Number of Samples/Iterations

- Model needs n points. We need to find all-inliers model with probability p (say, 0.99)
- Inlier probability: w. Probability of picking n inliers: w^n
- Probability of at least one outlier: $1 - w^n$
- Given N samples, we have: $1 - p = (1 - w^n)^N$. Thus, $N = \log(1-p)/\log(1-w^n)$
- w may be known roughly or start with $w = 0, N = \infty$
  - Draw a sample and find #inliers
  - Update w based on this and recompute N using above formula
  - Repeat this until the reduced N equals the number of samples evaluated

RANSAC to Estimate $\mathbf{P}$

- Find sufficient number 2D to 3D matches.
  (How? Identify images of known 3D points..)
- Estimate $\mathbf{P}$ using RANSAC, using #samples N
  - Select a random sample of 6 points, estimate $\mathbf{P}$
  - Compute number of inliers among matches to $\mathbf{P}$
  - Keep the solution with maximum number of inliers
- Estimate $\mathbf{P}$ from all matches classified as inliers.
  Can use a more accurate, non-linear optimization of geometric distance for this step

Non-Linear Calibration

- Geometric or Reprojection error of a point: $d(x, PX)$. Over all matches, minimize $\sum_i d(x_i, PX_i)$.
- Adjust values of P to minimize error. This is a non-linear minimization problem.
- The DLT solution can be the starting point
- Iterative methods, like Levenberg-Marquardt is popular
- Parametrize P suitably and move in the parameter space towards (hopefully the) global minimum.
- Constrained calibration when α_x, α_y, s or entire K matrix is known. Parametrization of P could be different.

Components of the Camera

- Estimated camera matrix has the form:

$$\mathbf{P} = [\mathbf{M} \,|\, \mathbf{t}] = [\mathbf{KR} \,|\, -\mathbf{KRC}]$$

- Decompose matris $\mathbf{M} = \mathbf{R}\,\mathbf{Q}$, where $\mathbf{R}$ is an upper triangular matrix and $\mathbf{Q}$ is an orthogonal matrix. Several matrix decomposition methods can be used.
- Upper triangular matrix: internal calibration matrix K.
  Orthogonal matrix: rotation R
  $-M^{\mathbf{-1}}t$: Camera centre C

Radial Distortion

- Extreme lenses can distort images.
- Distorted coordinates: $[x_d \quad y_d]^T = L(r)\,[x \quad y]^T$
- Radial distortion: $L(r) = 1 + \kappa_1 r + \kappa_2 r^2 + \cdots$

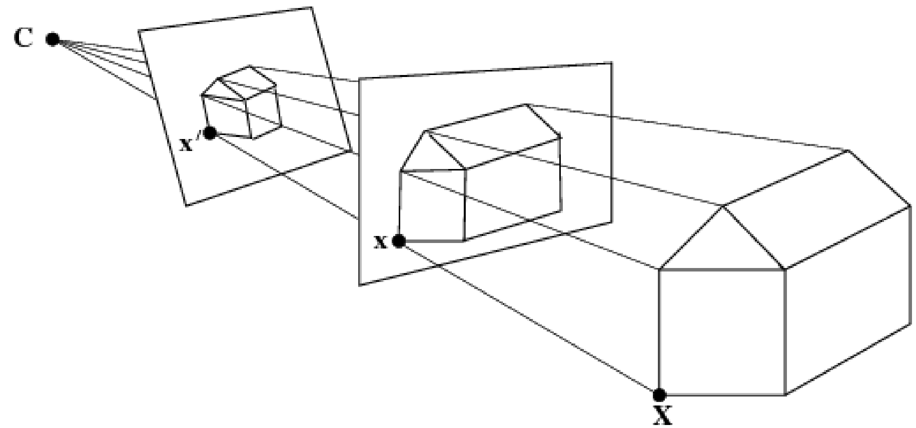
- Estimate κ values by identifying (curved) lines in image as straight lines.
- Correct radial distortion first

Some Camera Imaging Configurations

Common Camera Centre

Consider two cameras:
$P = KR[I|-C]$ and
$P' = K'R'[I|-C]$

$x = PX$ and $x' = P'X$



$$x' = K'R'[I|-C]X = K'R'(KR)^{-1}KR[I|-C]X = (K'R')(KR)^{-1}x$$

Or, $x' = \mathbf{H}x$, where $\mathbf{H} = K'R'R^{-1}K^{-1}$ is a **homography**

A 3×3 homography represents a general linear projective transformation, defined upto an unknown scale factor.

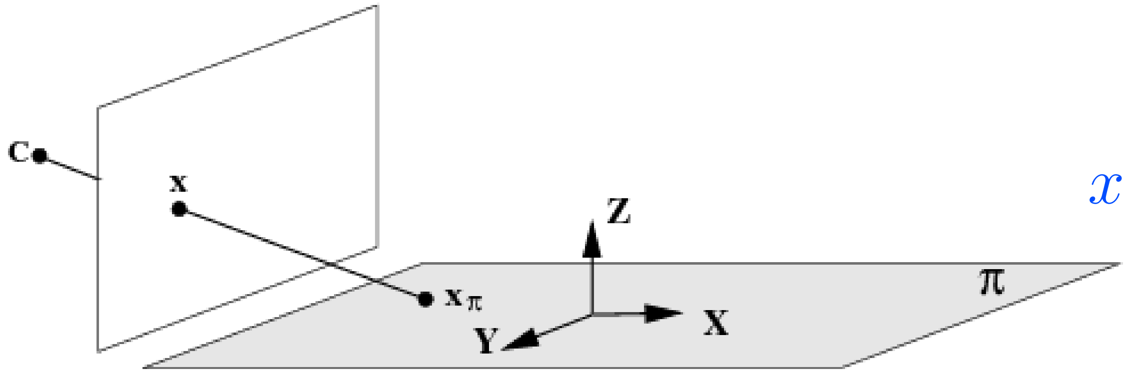
Special Cases

- Changing calibration K (focal length, image centre): $H = K'K^{-1}$. Reduces to a simple scaling by the ratio of focal lengths if only focal length changes.
- Pure rotation: $H = KR'R^{-1}K^{-1} = KR_{12}K^{-1}$, where R_{12} is rotation of the second camera w.r.t the first.

  In all cases, the rays through the same point are being sampled by planes of different orientations and distances. No new information is brought in!

Homography H is a 3×3 (non-singular) matrix defined only upto scale. It has 8 degrees of freedom and can be estimated from 4 world points for which image points in both views are known. Each such point gives 2 equations.

Image of a World Plane



$$x = [\mathbf{p_1}\,\mathbf{p_2}\,\mathbf{p_3}\,\mathbf{p_4}] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}$$

$$x = [\mathbf{p_1} \;\; \mathbf{p_2} \;\; \mathbf{p_4}]\, X_\pi = H X_\pi$$

A homography relates world Z-plane and the image plane. Similarly for X and Y planes.

For an arbitrary plane, there is a transformation M to bring it to world Z plane. Use $P' = PM$ to get its homography.

Projection: a 3D to 2D txformation with loss of information

Planar points: a general linear transformation, no loss!

Homography Due to a Plane

- Consider 2 cameras watching the same plane.

$$\mathbf{x_1} = \mathbf{H_1 X_\pi}, \qquad \mathbf{x_2} = \mathbf{H_2 X_\pi}$$

- $\mathbf{X_\pi} = \mathbf{H_1^{-1}}$. Therefore, $\mathbf{x_2} = \mathbf{H_2 H_1^{-1} x_1} = \mathbf{H x_1}$
- Images of a plane in two cameras are related by a homography.
- The plane brings more constraints.

Comparison of Homographies

Common Camera Centre

$$\mathbf{x_2} = \mathbf{H}\mathbf{x_1}$$

Arbitrary, unconstrained points

Cameras constrained to share a common centre.

Common Plane

$$\mathbf{x_2} = \mathbf{H}\mathbf{x_1}$$

Arbitrary, unconstrained cameras

Points constrained to lie on a plane

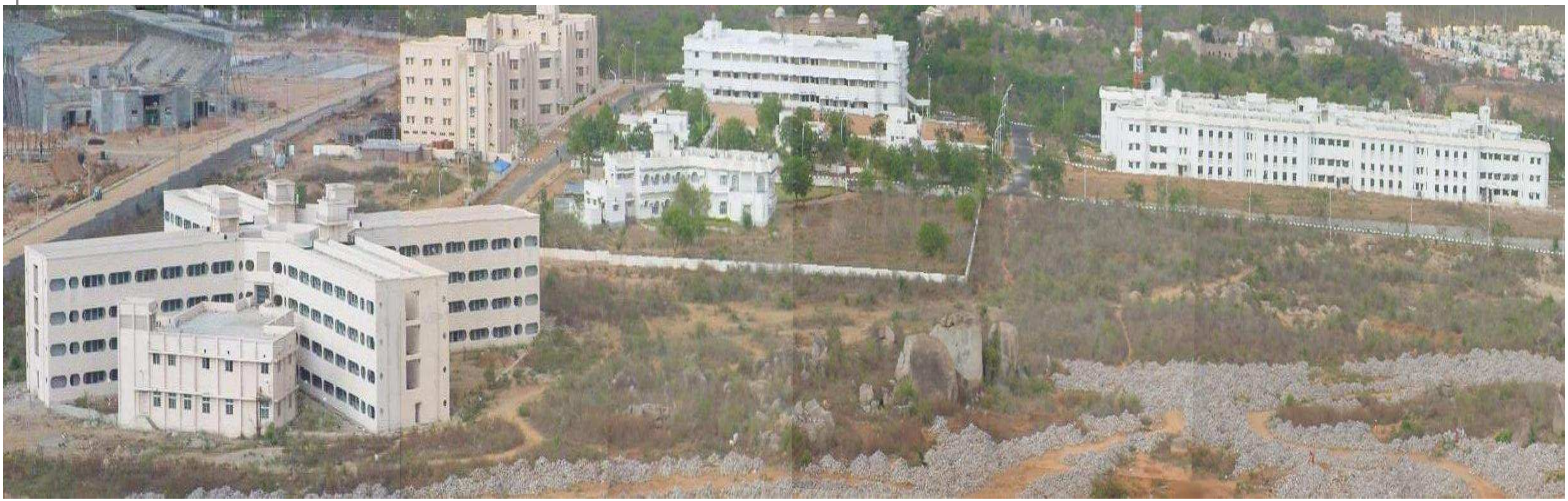
Applications of Homography

- **Mosaicing** to increase the field of view of a camera. Pan the camera by rotating to take many views. Relate pairs by their homography. Bring all to a reference plane!
- **Rectification** to get a fronto parallel view of a plane. A homography relates perspective and fronto-parallel views. Use 4 corners of a known world rectangle to obtain it
- **View generation** of planar scenes. Any plane can be viewed from any viewpoint if one view is given. The camera matrices give the homography using view can be synthesized

Image Mosaicing: Examples




Historical IIIT-H



Metric Rectification

Estimating Homography

- H has a maximum of 8 degrees of freedom. Each point match gives: $\mathbf{x}' = \mathbf{Hx}$, equality upto scale. Need 4 point matches to get 8 H values.
- $\mathbf{x}'$ and $\mathbf{Hx}$ are parallel. Write as $\mathbf{x}' \times \mathbf{Hx} = \mathbf{0}$.

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

Multiply row 1 by x' and row 2 by y' and add

- Use first 2 equations, as only 2 are linearly independent.
- From all point matches, form an $\mathbf{A}$ matrix of these rows, such that $Ah = 0$. h is a 9-vector of entries of $\mathbf{H}$.

Homography using DLT Algorithm

- A minimum of 4 points are sufficient for an exact solution. An **LSE** solution is better in the presence of noise, with as many matches as possible.
- Measurement matrix A is $2N \times 9$ for N observations.
- Solve for $\mathbf{Ah} = \mathbf{0}$, subject to $||h|| = 1$. With more entries, minimize $\min_h ||Ah||$, subject to $||h|| = 1$
- Using SVD: $\mathbf{A} = \mathbf{U}\,\mathbf{D}\,\mathbf{V}^{\mathbf{T}}$, with singular values of the diagonal matrix D in decreasing order. $\mathbf{U}$ and $\mathbf{V}$ are orthogonal. Rows of $\mathbf{D}$ are 0 beyond the 9th. Thus, $\mathbf{A}_{2N\times 9} = \mathbf{U}_{2N\times 2N}\mathbf{D}_{2N\times 9}\mathbf{V}^T_{9\times 9} = \mathbf{U}_{2N\times 9}\mathbf{D}_{9\times 9}\mathbf{V}^T_{9\times 9}$.
- Solution: column of $\mathbf{V}$ corresponding to smallest $D_{ii} > 0$.

Condition Number & Normalization

- x, y coordinates are in 100s and 1000s, products in the range 10^4 to 10^5. and w is in the range 1.
- Condition number of a matrix: ratio of the largest to the smallest singular value. High when ratios of matrix elements is high.
- Normalization: Reduce impact of high condition number using a translation and scaling of each image.
  - Shift origin to centroid of points
  - Scale points to have average distance to them from origin as $\sqrt{2}$.
  - Estimage homography $\hat{H}$ using DLT procedure.
  - Real homography $H = T'^{-1}\hat{H}T$ where T' and T are the conditioning transformations.

Point Matches

- Homography needs point matches in the given two images.
- This can be done by automatically identifying unique points. Or can be done manually!

RANSAC to Estimate $\mathbf{H}$

- Find matches using similarity, proximity, etc., of points in both images.
- Estimate $\mathbf{H}$ using RANSAC, using #samples N
  - Select a random sample of 4 points, estimate $\mathbf{H}$
  - Compute number of inliers among matches to $\mathbf{H}$
  - Keep the solution with maximum number of inliers
- Estimate $\mathbf{H}$ from all matches classified as inliers. Can use a more accurate, non-linear optimization of geometric distance for this step
- Use $\mathbf{H}$ matrix to find better interest point matches
- Repeat these steps until matches converge

Thank You!

Many figures are from the book

Multiview Geometry in Computer Vision

by **Richard Hartley and Andrew Zisserman**