Multi View Geometry (Spring '08)

Computation of the Camera Matrix P

Prof. Kyoung Mu Lee
SoEECS, Seoul National University

Computation of the Camera Matrix - DLT

- **Problem:**
  - ✓ Given point correspondences $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$, how to find a camera matrix $\mathbf{P}$, such that $\mathbf{x}_i = \mathbf{P}\mathbf{X}_i$?
- **DLT:** $\boxed{\mathbf{x}_i \times \mathbf{P}\mathbf{X}_i = \mathbf{0}}$

$$\boxed{\begin{bmatrix} u_i \\ v_i \\ \omega_i \end{bmatrix} = \mathbf{P}\begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix},\ x_i = \frac{u_i}{\omega_i}, y_i = \frac{v_i}{\omega_i}}$$

$$\begin{bmatrix} \mathbf{0}^T & -\omega_i\mathbf{X}_i^T & v_i\mathbf{X}_i^T \\ \omega_i\mathbf{X}_i^T & \mathbf{0}^T & -u_i\mathbf{X}_i^T \\ -v_i\mathbf{X}_i^T & u_i\mathbf{X}_i^T & \mathbf{0}^T \end{bmatrix}\begin{bmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \mathbf{P}^3 \end{bmatrix} = \mathbf{0}$$

$$\begin{bmatrix} \mathbf{0}^T & -\omega_i\mathbf{X}_i^T & v_i\mathbf{X}_i^T \\ \omega_i\mathbf{X}_i^T & \mathbf{0}^T & -u_i\mathbf{X}_i^T \end{bmatrix}\begin{bmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \mathbf{P}^3 \end{bmatrix} = \mathbf{0}$$

$$\begin{bmatrix} \mathbf{0}^T & -\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ \mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \end{bmatrix}\begin{bmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \mathbf{P}^3 \end{bmatrix} = \mathbf{0} \quad \Rightarrow \quad \boxed{\mathbf{A}_i\mathbf{p} = \mathbf{0}}$$

Computation of the Camera Matrix - DLT

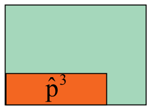
- **Using *n* correspondences, we have**

$$\boxed{\mathbf{A}_{2n\times 12}\mathbf{p} = \mathbf{0}}$$

✓ Minimal solution
- **P** has *11* dof, *2* independent eq./points
- Thus, 6 correspondences are needed

✓ Over-determined solution
- More than 6 correspondences
- This can be solved by minimizing the algebraic error (DLT)

$$\min \|\mathbf{Ap}\|$$

P = 

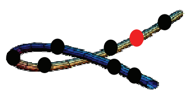
with constraints

$$\|\mathbf{p}\| = 1 \text{ or } \|\hat{\mathbf{p}}^3\| = 1, \text{ where } \hat{\mathbf{p}}^3 = (p_{31}, p_{32}, p_{33})^T$$

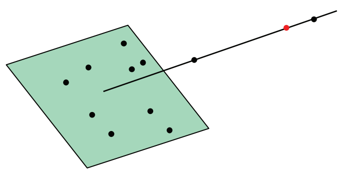
Degenerate Configurations

- **More complicate situations than 2D case**

✓ When the camera and points are on a twisted cubic

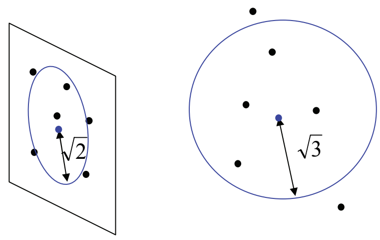


✓ When points lie on plane or single line passing through the projection center



DLT- Data Normalization

- **Normalization of 3D data**
  - ✓ Simple as before in 2D

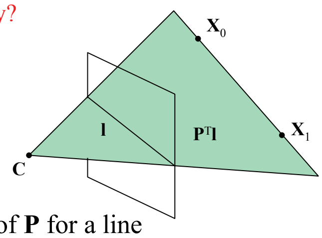


  - ✓ Anistropic scaling

DLT - Lines

- **Extend DLT to Line correspondences**
  - ✓ Assume an imaged line $\mathbf{l}$ and two points $\mathbf{X}_0$ and $\mathbf{X}_1$ on the 3D line
  - ✓ Since the plane back-projected from $\mathbf{l}$ is $\mathbf{P}^T\mathbf{l}$, the two points must satisfy

why?



$$\mathbf{l}^\top \mathtt{P} \mathbf{X}_j = 0 \quad \text{for } j = 0, 1.$$

⇨ Two linear equations on the entries of $\mathbf{P}$ for a line

  - ✓ Thus for n correspondences, we have

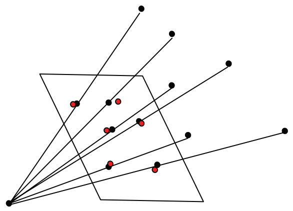
$$\mathbf{A}_{2n\times 12}\mathbf{p} = \mathbf{0}$$

Geometric error

- **Assume X_i are known**

$\mathbf{x}_i$: measured points

$\hat{\mathbf{x}}_i = \mathbf{P}\mathbf{X}_i$: estimated points



- **The geometric error in the image:**

$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2$$

- **MLE of P:** $$\min_{\mathrm{P}} \sum_i d(\mathbf{x}_i, \mathrm{P}\mathbf{X}_i)^2$$

Gold Standard Algorithm

Objective

Given $n \geq 6$ 3D to 2D point correspondences $\{\mathbf{X}_i \leftrightarrow \mathbf{x}_i'\}$, determine the Maximum Likelihood Estimation of $\mathbf{P}$

Algorithm

(i) **Linear solution:**

(a) Normalization: $\tilde{\mathbf{X}}_i = \mathbf{U}\mathbf{X}_i \quad \tilde{\mathbf{x}}_i = \mathbf{T}\mathbf{x}_i$

(b) DLT:

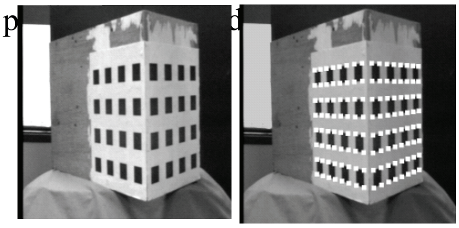
(ii) **Minimization of geometric error:** using the linear estimate as a starting point minimize the geometric error:

$$\sum_i d\left(\tilde{\mathbf{x}}_i, \tilde{\mathbf{P}}\tilde{\mathbf{X}}_i\right)^2$$

(iii) **Denormalization:** $\mathbf{P} = \mathbf{T}^{-1}\tilde{\mathbf{P}}\,\mathbf{U}$

Calibration Example

- Use Canny edge detector
- Detect straight lines from the edgels
- Find corners by intersecting the lines
- Subpixel accuracy < 1/10
- \# of constraints > 5 x # of unknowns (11) => more than 28 p



	f_y	f_x/f_y	skew	x_0	y_0	residual
linear	1673.3	1.0063	1.39	379.96	305.78	0.365
iterative	1675.5	1.0063	1.43	379.79	305.25	0.364

Errors in the world points

- **Errors in the world**

$$\sum_i d(\mathbf{X}_i, \hat{\mathbf{X}}_i)^2 \qquad \mathbf{x}_i = \mathbf{P}\hat{\mathbf{X}}_i$$

- **Errors in the image and in the world**

$$\sum_{i=1}^{n} d_{\mathrm{Mah}}(\mathbf{x}_i, \mathtt{P}\hat{\mathbf{X}}_i)^2 + d_{\mathrm{Mah}}(\mathbf{X}_i, \hat{\mathbf{X}}_i)^2$$

Geometric Interpretation of Algebraic Error

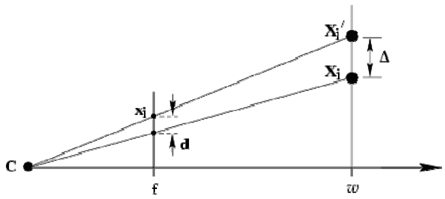
- **Normalized DLT minimizes** $\sum_i (\hat{\omega}_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i))^2$

$\mathbf{X}_i = (X_i, Y_i, Z_i, 1)^T$
$\mathbf{x}_i = \omega_i (x_i, y_i, 1)^T$

$$\hat{\omega}_i (\hat{x}_i, \hat{y}_i, 1)^T = \mathbf{P}\mathbf{X}_i \quad \hat{\omega}_i = \pm \|\hat{\mathbf{p}}^3\| \mathrm{depth}(\mathbf{X}; \mathbf{P})$$

therefore, if $\|\hat{\mathbf{p}}^3\| = 1$ then

$$\hat{\omega}_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i) \sim f d(\mathbf{X}_i, \hat{\mathbf{X}}_i)$$



- **Note invariance to 2D and 3D similarity transform given proper normalization**

Estimation of an Affine Camera Matrix

- **Affine camera matrix:** $\mathbf{P}_A = (\mathbf{P}^1, \mathbf{P}^2, \mathbf{P}^3)^T, \quad \mathbf{P}^{3T} = (0,0,0,1)$
- **For** $\mathbf{X}_i = (X_i, Y_i, Z_i, 1)^T$ and $\mathbf{x}_i = \omega_i (x_i, y_i, 1)^T$

$$\begin{bmatrix} \mathbf{0}^T & -\mathbf{X}_i^T \\ \mathbf{X}_i^T & \mathbf{0}^T \end{bmatrix} \begin{pmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \end{pmatrix} + \begin{pmatrix} y_i \\ -x_i \end{pmatrix} = \mathbf{0}$$

$$\Rightarrow \begin{bmatrix} \mathbf{X}_i^T & \mathbf{0}^T \\ \mathbf{0}^T & \mathbf{X}_i^T \end{bmatrix} \begin{pmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$$

$$\Rightarrow \mathbf{A}_i \mathbf{p}_8 = \mathbf{b}_i$$

- **By stacking *n* correspondences, and using pseudo inverse** $\mathbf{A}_8 \mathbf{p}_8 = \mathbf{b} \Rightarrow \mathbf{p}_8 = \mathbf{A}_8^+ \mathbf{b}$
- **Algebraic error: (=geometric error, in this case)**

$$\|\mathbf{b} - \mathbf{A}\mathbf{p}\|^2 = \sum_i (x_i - \mathbf{P}^{1T}\mathbf{X}_i)^2 + (y_i - \mathbf{P}^{2T}\mathbf{X}_i)^2 = \sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2$$

Gold Standard Algorithm - Affine Camera

Objective

Given $n \geq 4$ 3D to 2D point correspondences $\{\mathbf{X}_i \leftrightarrow \mathbf{x}_i'\}$, determine the Maximum Likelihood Estimation of $\mathbf{P}$

(remember $\mathbf{P}^{3T}=(0,0,0,1)$)

Algorithm

(i) **Normalization:** $\tilde{\mathbf{X}}_i = \mathbf{U}\mathbf{X}_i \quad \tilde{\mathbf{x}}_i = \mathbf{T}\mathbf{x}_i$

(ii) For each correspondence

$$\begin{bmatrix} \tilde{\mathbf{X}}_i^T & \mathbf{0}^T \\ \mathbf{0}^T & \tilde{\mathbf{X}}_i^T \end{bmatrix} \begin{pmatrix} \tilde{\mathbf{P}}^1 \\ \tilde{\mathbf{P}}^2 \end{pmatrix} = \begin{pmatrix} \tilde{x}_i \\ \tilde{y}_i \end{pmatrix}$$

$$\mathbf{A}_8 \mathbf{p}_8 = \mathbf{b}$$

(iii) solution is $\mathbf{p}_8 = \mathbf{A}_8^{+}\mathbf{b}$ and $\tilde{\mathbf{P}}^{3T} = (0,0,0,1)$

(iv) **Denormalization:** $\mathbf{P} = \mathbf{T}^{-1}\tilde{\mathbf{P}}\,\mathbf{U}$

Decompose P into K, R, and t

- **We note that M=KR is the product of an upper triangular and rotation matrix**
  - ✓ Factor **M** into **KR** using the QR matrix decomposition. This determines **K** and **R**.
  - ✓ Then

$$\mathbf{t} = \mathbf{K}^{-1}(p_{14}, p_{24}, p_{34})^T$$

- **Note that this produces a matrix with nonzero skew** s

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

with , $s = \tan\theta$ and θ is the angle between the image axes

Restricted Camera Estimation

$$\mathtt{P} = [\mathtt{M} \mid -\mathtt{M}\tilde{\mathbf{C}}] = \mathtt{K}[\mathtt{R} \mid -\mathtt{R}\tilde{\mathbf{C}}]. \qquad \mathtt{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- **Camera estimation with some known parameter information:**
  - ✓ The skew is zero $s = 0$
  - ✓ The pixels are square $\alpha_x = \alpha_y$
  - ✓ The principal point (x_0, y_0) is known
  - ✓ The internal parameter matrix **K** is known
- **Minimizing geometric error:**
  - ✓ impose constraint through parameterization
  - ✓ Suppose $s = 0, \alpha_x = \alpha_y$, then
  - ✓ 2D Image error case: LM minimization $f : \mathrm{R}^9 \to \mathrm{R}^{2n}$
  - ✓ 3D and 2D error case: LM minimization $f : \mathrm{R}^{3n+9} \to \mathrm{R}^{5n}$

Restricted Camera Estimation

- **Minimizing algebraic error:**
  - ✓ Assume map from param $\mathbf{q} \to \mathbf{P} = \mathbf{K}[\mathbf{R} \mid -\mathbf{R}\tilde{\mathbf{C}}]$, i.e. $\mathbf{p}=g(\mathbf{q})$
  - ✓ Minimize $\|\mathbf{Ap}\| = \|\mathbf{A}g(\mathbf{q})\|$, where $\mathbf{q}$ is the parameters to be found
- **Reduced measurement matrix**
  - ✓ Replace $\mathbf{A}_{2n \times 12}$ with $\hat{\mathbf{A}}_{12 \times 12}$ such that

    $$\|\mathbf{Ap}\| = \|\hat{\mathbf{A}}\mathbf{p}\| \Rightarrow \mathbf{P}^T\mathbf{A}^T\mathbf{AP} = \mathbf{P}^T\hat{\mathbf{A}}^T\hat{\mathbf{A}}\mathbf{P} \Rightarrow \mathbf{A}^T\mathbf{A} = \hat{\mathbf{A}}^T\hat{\mathbf{A}}$$
  - ✓ By using SVD

    Let $\mathbf{A} = \mathbf{UDV}^T$ and $\hat{\mathbf{A}} = \mathbf{DV}^T$, then

    $$\mathbf{A}^T\mathbf{A} = (\mathbf{VDU}^T)(\mathbf{UDV}^T) = (\mathbf{VD})(\mathbf{DV}^T) = \hat{\mathbf{A}}^T\hat{\mathbf{A}}$$
  - ✓ By using QR factorization

    Let $\mathbf{A} = \mathbf{Q}\hat{\mathbf{A}}$, then

    $$\mathbf{A}^T\mathbf{A} = (\hat{\mathbf{A}}^T\mathbf{Q}^T)(\mathbf{Q}\hat{\mathbf{A}}) = \hat{\mathbf{A}}^T\hat{\mathbf{A}}$$
  - ✓ Thus, $f : \mathbf{q} \to \hat{\mathbf{A}}g(\mathbf{q}) \Rightarrow f : \mathrm{R}^9 \to \mathrm{R}^{12}$

Restricted Camera Estimation

- **Initialization**
  - ✓ Find initial **P** using DLT
  - ✓ Decompose **P** and clamp fixed parameters to their desired values e.g. $s=0$, $\alpha_x = \alpha_y$ (*hard constraint*).
  - ✓ Set other variables to their values obtained by the new decomposition
  - ✓ It can sometimes cause big jump in error

- **Alternative initialization**
  - ✓ Use general DLT
  - ✓ Impose *soft constraints*

$$\sum_i d(\mathbf{x}_i, \mathrm{P}\mathbf{X}_i)^2 + ws^2 + w(\alpha_x - \alpha_y)^2$$

  - ✓ gradually increase weights

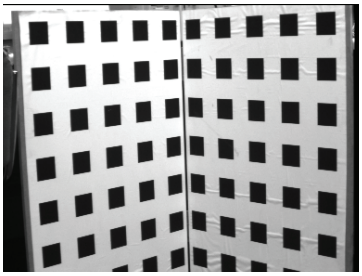
Exterior Orientation

- **If the camera is calibrated (internal parameters are known), then the position and orientation of the camera should be determined**

  → Pose estimation

- **6 DOF ⇒ 3 points minimal (4 solutions in general)**

Experimental Evaluation



	f_y	f_x/f_y	skew	x_0	y_0	residual
algebraic	1633.4	1.0	0.0	371.21	293.63	0.601
geometric	1637.2	1.0	0.0	371.32	293.69	0.601

	f_y	f_x/f_y	skew	x_0	y_0	residual
linear	1673.3	1.0063	1.39	379.96	305.78	0.365
iterative	1675.5	1.0063	1.43	379.79	305.25	0.364

Covariance Estimation

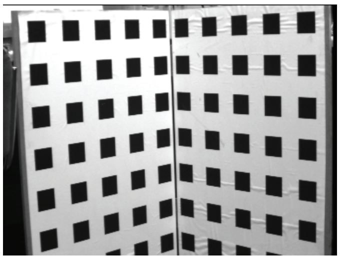
- **ML residual error**

$$\varepsilon_{res} = \sigma(1 - d/2n)^{1/2}$$

$$\varepsilon_{res} \leftrightarrow \sigma$$

- **Example:**

$$n = 197, \quad \varepsilon_{res} = 0.365, \quad \sigma = 0.37$$



Covariance for Estimated Camera

- **Compute Jacobian at ML solution, then**

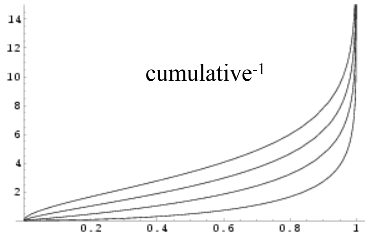
$$\Sigma_{\mathbf{P}} = \left(\mathbf{J}^{\mathsf{T}} \Sigma_{\mathbf{X}}^{-1} \mathbf{J}\right)^{+}$$

✓ variance per parameter can be found on diagonal

- **Confidence ellipsoid for the camera center**

$$(\mathbf{C} - \bar{\mathbf{C}})^{\top} \Sigma_{\mathbf{C}}^{-1} (\mathbf{C} - \bar{\mathbf{C}}) = k^2 \qquad k^2 = F_n^{-1}(\alpha)$$

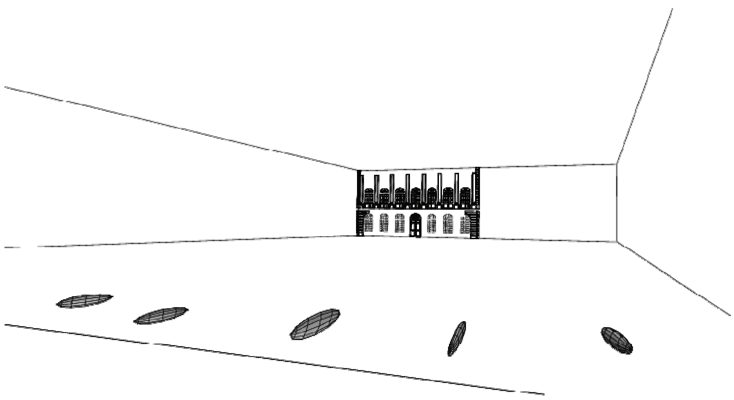
k^2



χ^2

(chi-square distribution
=distribution of sum of squares)

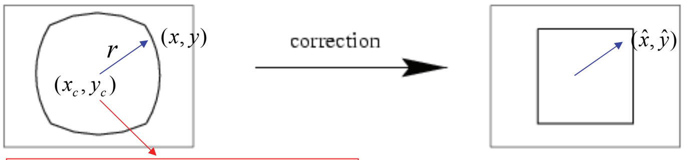
Covariance for Estimated Camera



Camera center covariance ellipsoids

Radial Distortion

short focal length   long focal length



Radial Distortion

- **Correction of distortion**

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

$$r^2 = (x - x_c)^2 + (y - y_c)^2$$

- **Distortion function: Taylor series**

$$L(r) = 1 + \kappa_1 r + \kappa_2 r^2 + \kappa_3 r^3 + \cdots$$

$$\boxed{L(r) = 1 + \kappa r^2 \text{ is sufficient}}$$

- **Distortion parameters can be determined by some constraints or cost function:**
  - ✓ Straight line should be straight
  - ✓ Deviation from the linear mapping

Radial Distortion

Radial Distortion

	f_y	f_x/f_y	skew	x_0	y_0	residual
linear	1580.5	1.0044	0.75	377.53	299.12	0.179
iterative	1580.7	1.0044	0.70	377.42	299.02	0.179
algebraic	1556.0	1.0000	0.00	372.42	291.86	0.381
iterative	1556.6	1.0000	0.00	372.41	291.86	0.380
linear	1673.3	1.0063	1.39	379.96	305.78	0.365
iterative	1675.5	1.0063	1.43	379.79	305.25	0.364
algebraic	1633.4	1.0000	0.00	371.21	293.63	0.601
iterative	1637.2	1.0000	0.00	371.32	293.69	0.601