CMP446: Computer Vision

CAIRO UNIVERSITY

Lecture 10: Alignment and Calibration

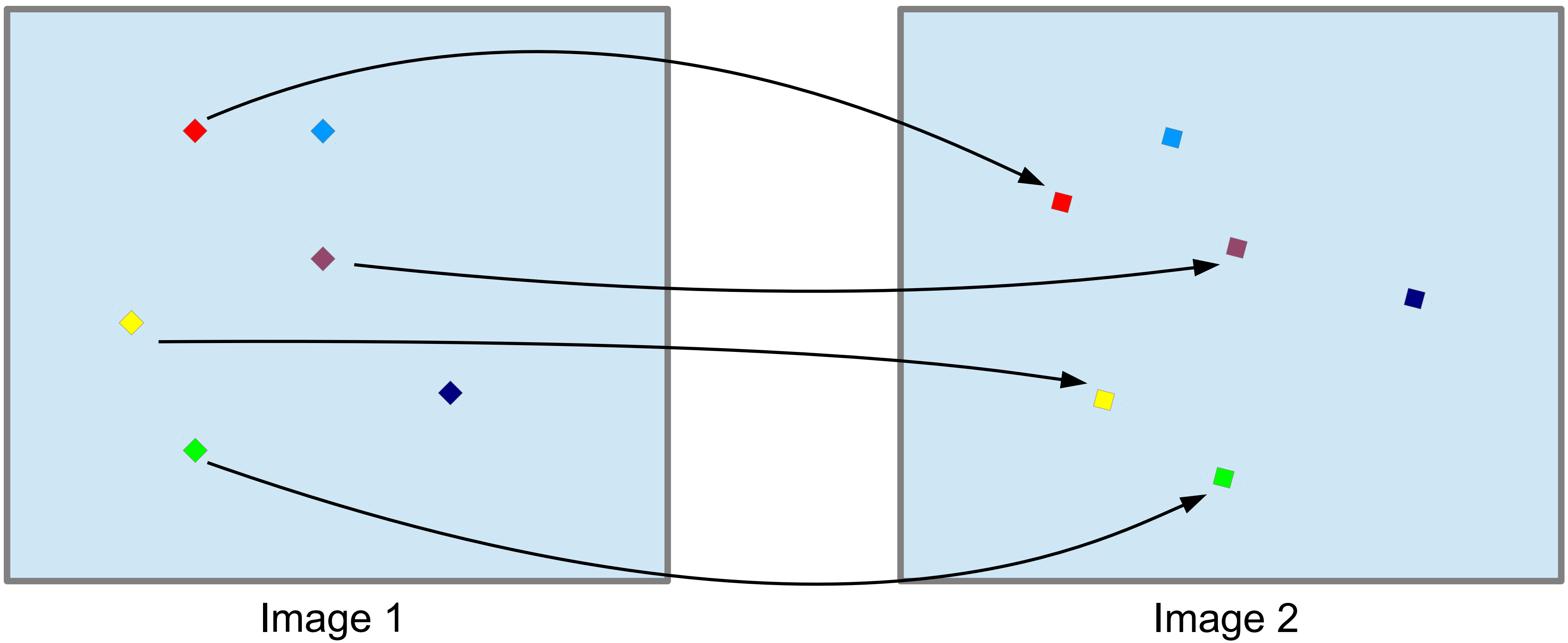
Mohamed Alaa El-Dien Aly
Computer Engineering Department
Cairo University
Fall 2012

Agenda

- Geometric Alignment
- Parameter Estimation
- Linear Least Squares
- Camera Calibration
  - Direct Linear Transformation
  - Nonlinear Least Squares

Some slides adapted from James Hays and Robert Collins

Geometric Alignment



Estimate the geometric transformation that transforms the matching points from 1 to 2

Geometric Alignment

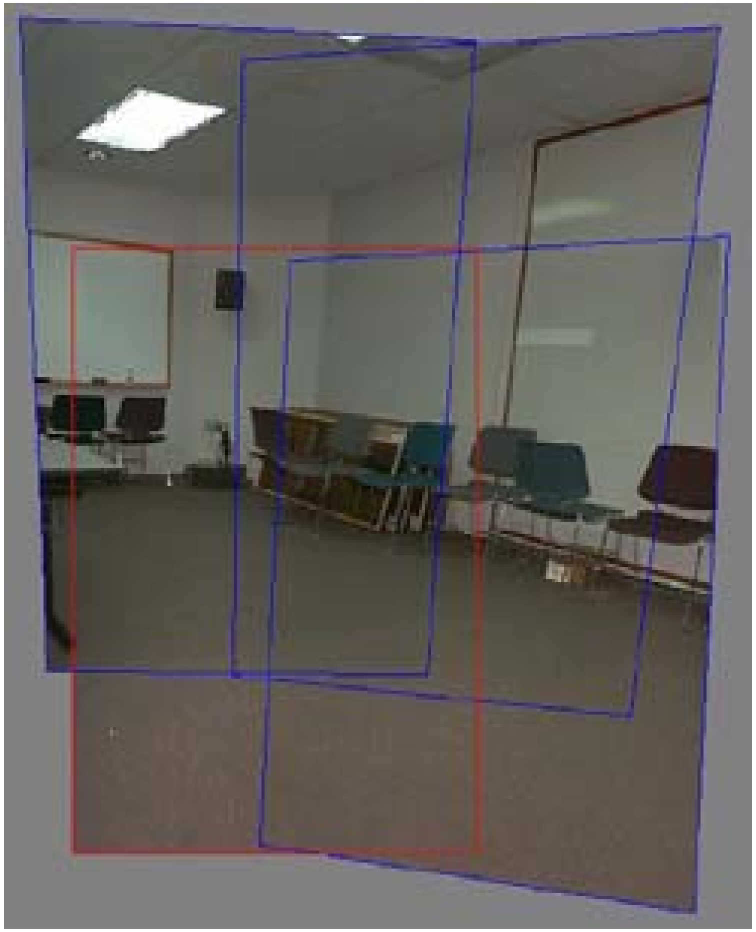


Image Stitching

Panography

Geometric Alignment

Transform	Matrix	Parameters p
translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$	(t_x, t_y)
Euclidean	$\begin{bmatrix} c_\theta & -s_\theta & t_x \\ s_\theta & c_\theta & t_y \end{bmatrix}$	(t_x, t_y, θ)
similarity	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	(t_x, t_y, a, b)
affine	$\begin{bmatrix} 1+a_{00} & a_{01} & t_x \\ a_{10} & 1+a_{11} & t_y \end{bmatrix}$	$(t_x, t_y, a_{00}, a_{01}, a_{10}, a_{11})$
projective	$\begin{bmatrix} 1+h_{00} & h_{01} & h_{02} \\ h_{10} & 1+h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix}$	$(h_{00}, h_{01}, \ldots, h_{21})$

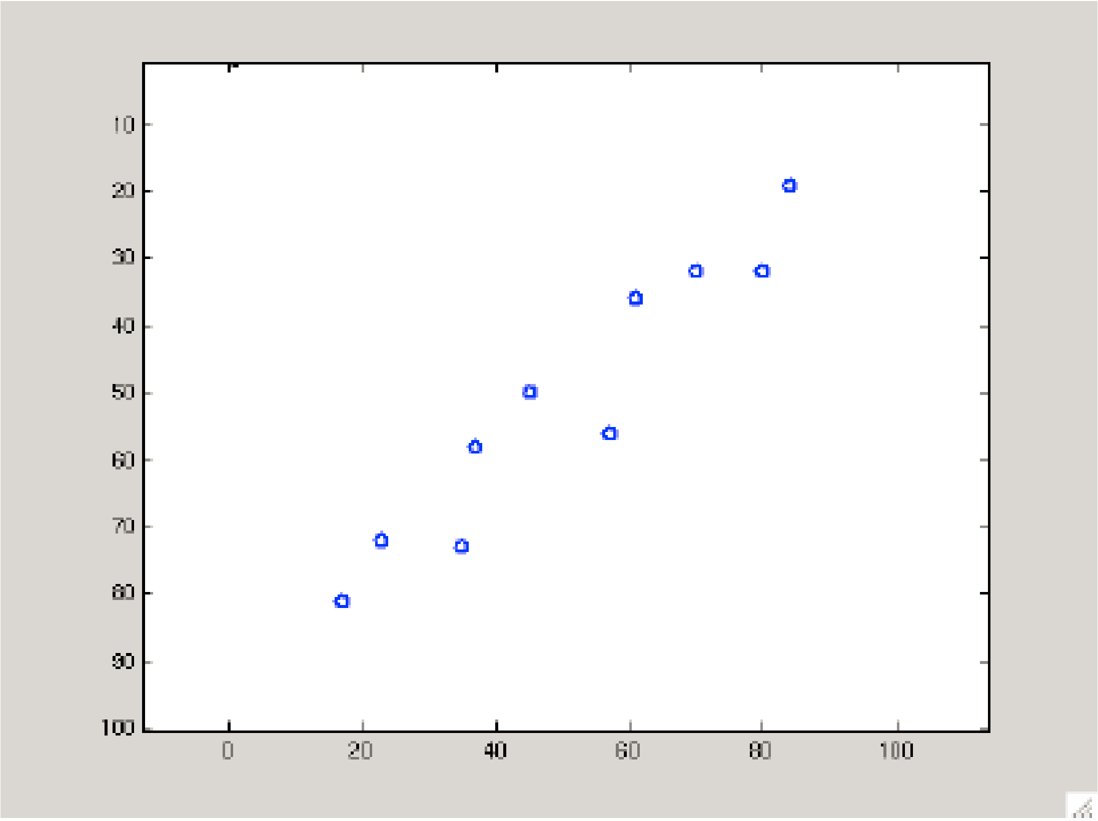
2D Parametric Geometric Transformations

Parameter Estimation

```
1. Given a number of data points

2. Define the model in terms of parameters

3. Define an error function that measures how
   well the model fits the data

4. Solve for the model the minimizes the error
```

Example: Line Fitting

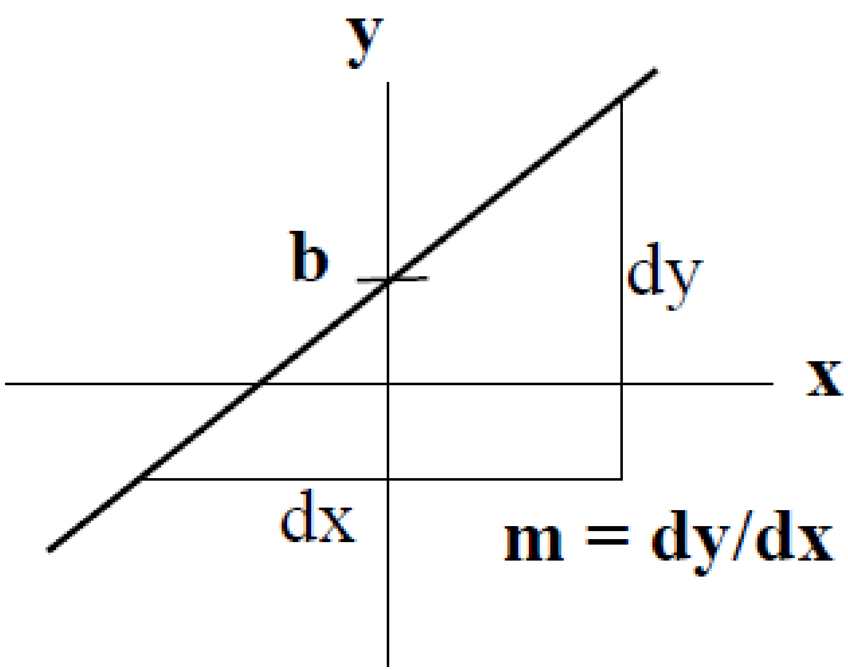
Data points: (x_i, y_i)



1. **Given a number of data points**
2. Define the model in terms of parameters
3. Define an error function that measures how well the model fits the data
4. Solve for the model the minimizes the error

Example: Line Fitting

Model: $y = mx + b$



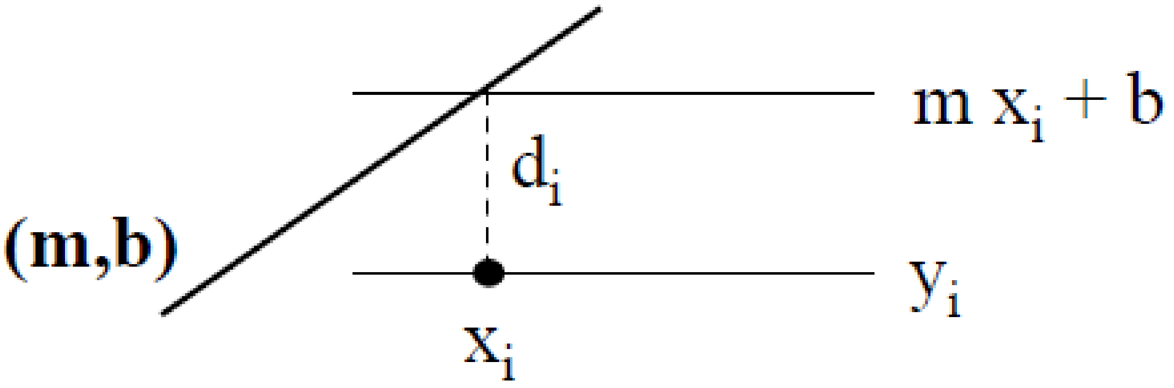
1. Given a number of data points
2. Define the model in terms of parameters
3. Define an error function that measures how well the model fits the data
4. Solve for the model the minimizes the error

Example: Line Fitting

Vertical distance: $d_i = (mx_i + b) - y_i$

Least Squares Error

Error: $E = \sum_i d_i^2 = \sum_i \left((mx_i + b) - y_i\right)^2$



1. Given a number of data points
2. Define the model in terms of parameters
3. Define an error function that measures how well the model fits the data
4. Solve for the model the minimizes the error

Example: Line Fitting

Error: $E = \sum_i d_i^2 = \sum_i \left((mx_i + b) - y_i \right)^2$

Minimum is at: $\nabla E = 0$

$$\frac{\partial E}{\partial m} = 0$$

$$\sum_i 2(mx_i + b - y_i)x_i = 0$$

$$\left(\sum_i x_i^2\right)m + \left(\sum_i x_i\right)b = \sum_i x_i y_i$$

$$\frac{\partial E}{\partial b} = 0$$

$$\sum_i 2(mx_i + b - y_i) = 0$$

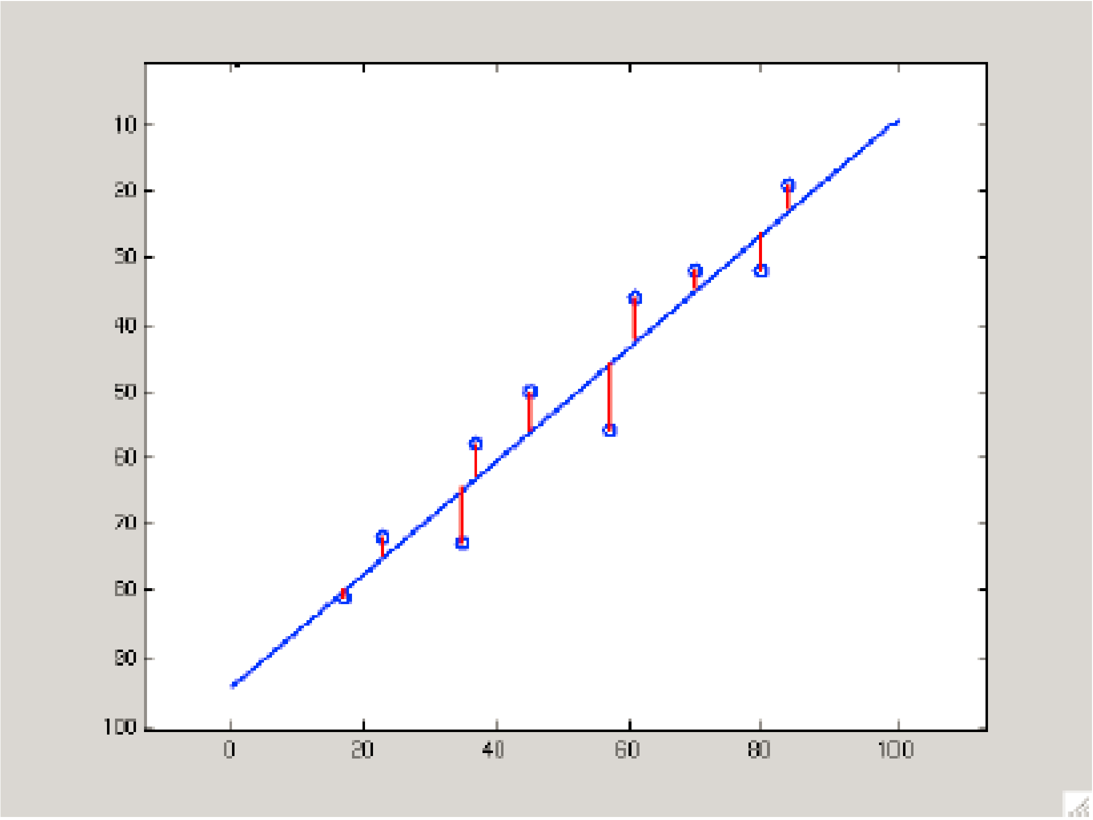
$$\left(\sum_i x_i\right)m + \left(\sum_i 1\right)b = \sum_i y_i$$

Normal Equations: $\begin{bmatrix} \sum_i x^2 & \sum_i x \\ \sum_i x & \sum_i 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} \sum_i x_i y_i \\ \sum_i y_i \end{bmatrix}$

1. Given a number of data points
2. Define the model in terms of parameters
3. Define an error function that measures how well the model fits the data
4. Solve for the model the minimizes the error

Example: Line Fitting

$$\begin{bmatrix} \sum_i x^2 & \sum_i x \\ \sum_i x & \sum_i 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} \sum_i x_i y_i \\ \sum_i y_i \end{bmatrix}$$



1. Given a number of data points
2. Define the model in terms of parameters
3. Define an error function that measures how well the model fits the data
4. Solve for the model the minimizes the error

Linear Least Squares

Data points: $\{x_i\}_{i=1}^N \in \mathbb{R}^f, \{y_i\}_{i=1}^N \in \mathbb{R}$

Parameters: $p \in \mathbb{R}^f$

Linear Model: $$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1f} \\ x_{21} & x_{22} & \cdots & x_{2f} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nf} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_f \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \rightarrow \quad Ap = y$$

Error function: $E = \|Ap - y\|^2 = (Ap - y)^T (Ap - y)$

$$E = p^T A^T A p - 2 p^T A^T y + y^T y$$

Normal Equations: $\nabla E = 0 \quad \rightarrow \quad (A^T A) p = A^T y$

Optimal Parameters: $p^* = (A^T A)^{-1} A^T y$

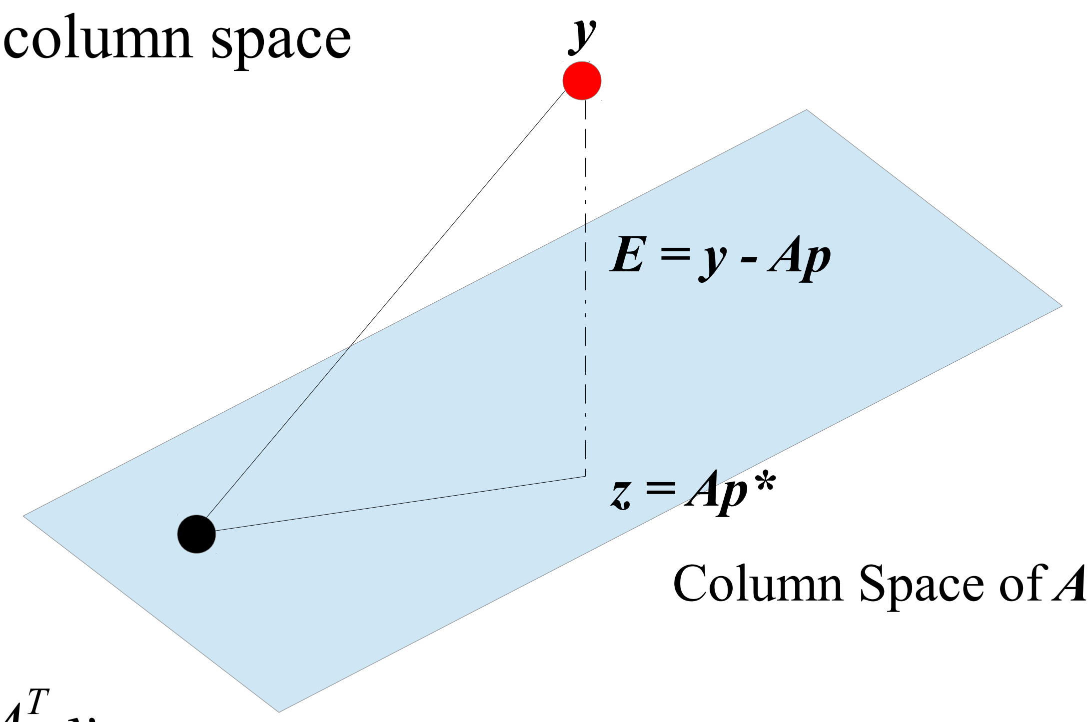
Linear Least Squares

Want to solve: $Ap = y$

But y is not in the column space of A i.e. no solution

The neasrest point to y in the column space is its normal projection z

The error vector E is normal to the column space



$$A^T E = 0$$

$$A^T (y - Ap) = 0$$

Normal Equations: $A^T A p = A^T y$

$$p^* = \left(A^T A\right)^{-1} A^T y$$

Projection: $z = Ap^* = A\left(A^T A\right)^{-1} A^T y$

Linear Least Squares

Verify that $E \perp z$

$$E^T z = (y - A p^*)^T A p^*$$

$$= (y^T - p^{*T} A^T) A p^*$$

$$= y^T A (A^T A)^{-1} A^T y - y^T A (A^T A)^{-1} A^T A (A^T A)^{-1} A^T y$$

$$= y^T A (A^T A)^{-1} A^T y - y^T A (A^T A)^{-1} A^T y$$

$$= 0$$

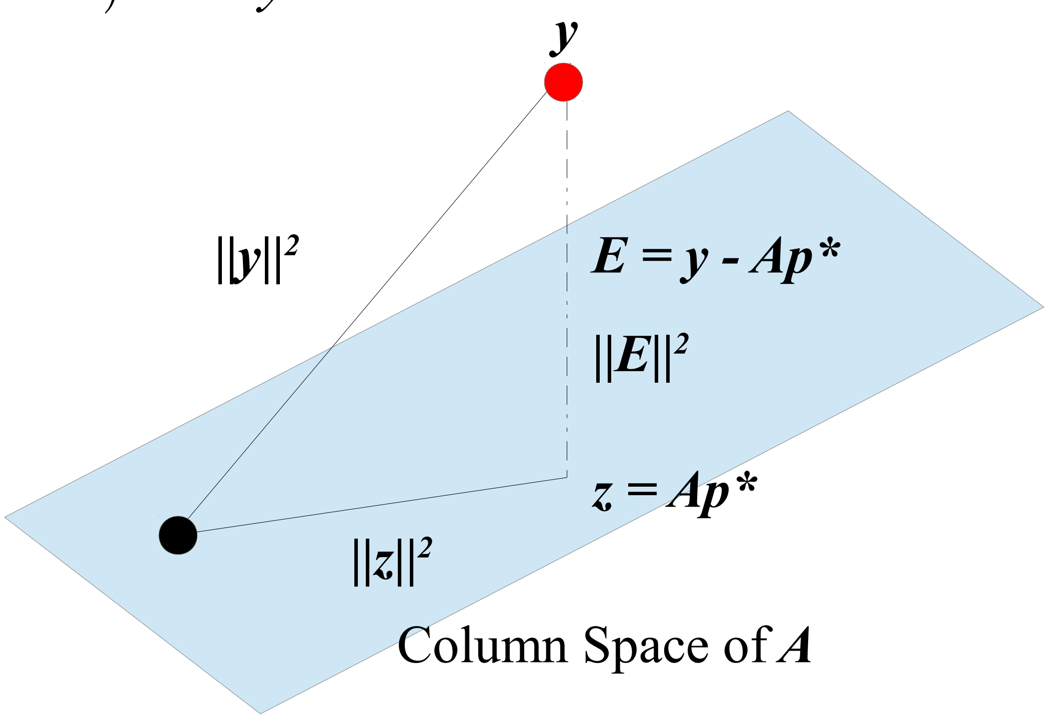
Right triangle with sides: y, E, z

$$\|y\|^2 = \|E + z\|^2$$

$$= \|E\|^2 + 2 E^T z + \|z\|^2$$

$$= \|E\|^2 + \|z\|^2$$

$$\rightarrow \|E\|^2 = \|y\|^2 - \|z\|^2$$



Linear Least Squares: Line Fitting

$$A = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_N & 1 \end{bmatrix} \qquad p = \begin{bmatrix} m \\ b \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

Error: $E = \| Ap - y \|^2 = \sum_i (m x_i + b - y_i)^2$

Normal Equations: $(A^T A) p = A^T y$ → $\begin{bmatrix} \sum_i x^2 & \sum_i x \\ \sum_i x & \sum_i 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix} = \begin{bmatrix} \sum_i x_i y_i \\ \sum_i y_i \end{bmatrix}$

Solve to find (m^*, b^*) that minimize E

Linear Least Squares: Recap

Equation: $Ap = y$

Want to find: p s.t. $\|Ap - y\|^2$ is minimum

Set up Normal Equations: $A^T A\, p = A^T y$

Solve for: p^*

Geometric Alignment

We have point correspondences x_i & x_i' s.t. $x_i' = f(x_i)$

We want to estimate the parameters of the transformation function $f(x)$

Estimate parameters that minimize the sum of squared residuals

$$r_i^2 = \| x_i' - f(x_i) \|^2$$

Two questions:

1) How many Degrees of Freedom?
2) How many point correspondences are needed?

Geometric Alignment

Transform	Matrix	Parameters p
translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$	(t_x, t_y)
Euclidean	$\begin{bmatrix} c_\theta & -s_\theta & t_x \\ s_\theta & c_\theta & t_y \end{bmatrix}$	(t_x, t_y, θ)
similarity	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	(t_x, t_y, a, b)
affine	$\begin{bmatrix} 1+a_{00} & a_{01} & t_x \\ a_{10} & 1+a_{11} & t_y \end{bmatrix}$	$(t_x, t_y, a_{00}, a_{01}, a_{10}, a_{11})$
projective	$\begin{bmatrix} 1+h_{00} & h_{01} & h_{02} \\ h_{10} & 1+h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix}$	$(h_{00}, h_{01}, \ldots, h_{21})$

Geometric Alignment: Translation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix}$$

$$f(x) = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\text{Parameters: } p = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

- DOF?
  - 2
- Point Correspondences?
  - 1

Geometric Alignment: Translation

Least Squares: $E = \sum_i r_i^2 = \sum_i (x_i + t_x - x'_i)^2 + (y_i + t_y - y'_i)^2$

$$\frac{\partial E}{\partial t_x} = 0$$

$$\sum_i 2(x_i + t_x - x'_i) = 0$$

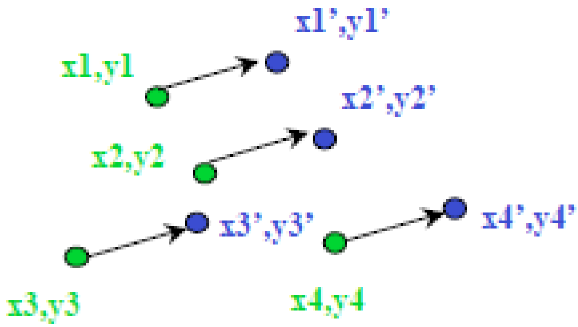
$$\rightarrow t_x = \sum_i (x'_i - x_i) / N$$

$$\frac{\partial E}{\partial t_y} = 0$$

$$\sum_i 2(y_i + t_y - y'_i) = 0$$

$$\rightarrow t_y = \sum_i (y'_i - y_i) / N$$

$$\begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} \bar{x}' - \bar{x} \\ \bar{y}' - \bar{y} \end{bmatrix}$$



Geometric Alignment

Least Squares: $E = \sum_i r_i^2 = \sum_i \| f(x_i) - x'_i \|^2$

Most transformations are such that: $f(x) = x + J(x)\, p$

where: $J(x)$ is the Jacobian matrix i.e. $J_{ij} = \frac{\partial f_i}{\partial p_j}$

For example, for translation: $J(x) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

and $f(x) = \begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} t_x \\ t_y \end{bmatrix}$

Least Squares: $E = \sum_i r_i^2 = \sum_i \| J(x)\, p + x_i - x'_i \|^2$

Geometric Alignment

Least Squares: $E = \sum_i r_i^2 = \sum_i \| J(x) p - \Delta x_i \|^2$ $\qquad \Delta x_i = x'_i - x_i$

$$E = p^T \left(\sum_i J^T(x_i) J(x_i) \right) p - 2 p^T \left(\sum_i J^T(x_i) \Delta x_i \right) + \sum_i \| \Delta x_i \|^2$$

$$E = p^T A p - 2 p^T b + c$$

$$A = \sum_i J^T(x_i) J(x_i) \qquad b = \sum_i J^T(x_i) \Delta x_i$$

$\nabla E = 0 \quad \rightarrow \quad Ap = b$

find $p^* = A^{-1} b$

Geometric Alignment: Translation

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

$$A=\sum_i J^T(x_i)\,J(x_i) \quad \rightarrow \quad A=\begin{bmatrix}N & 0\\ 0 & N\end{bmatrix}$$

$$b=\sum_i J^T(x_i)\,\Delta x_i \quad \rightarrow \quad b=\begin{bmatrix}\sum_i x'_i-x_i\\ \sum_i y'_i-y_i\end{bmatrix}$$

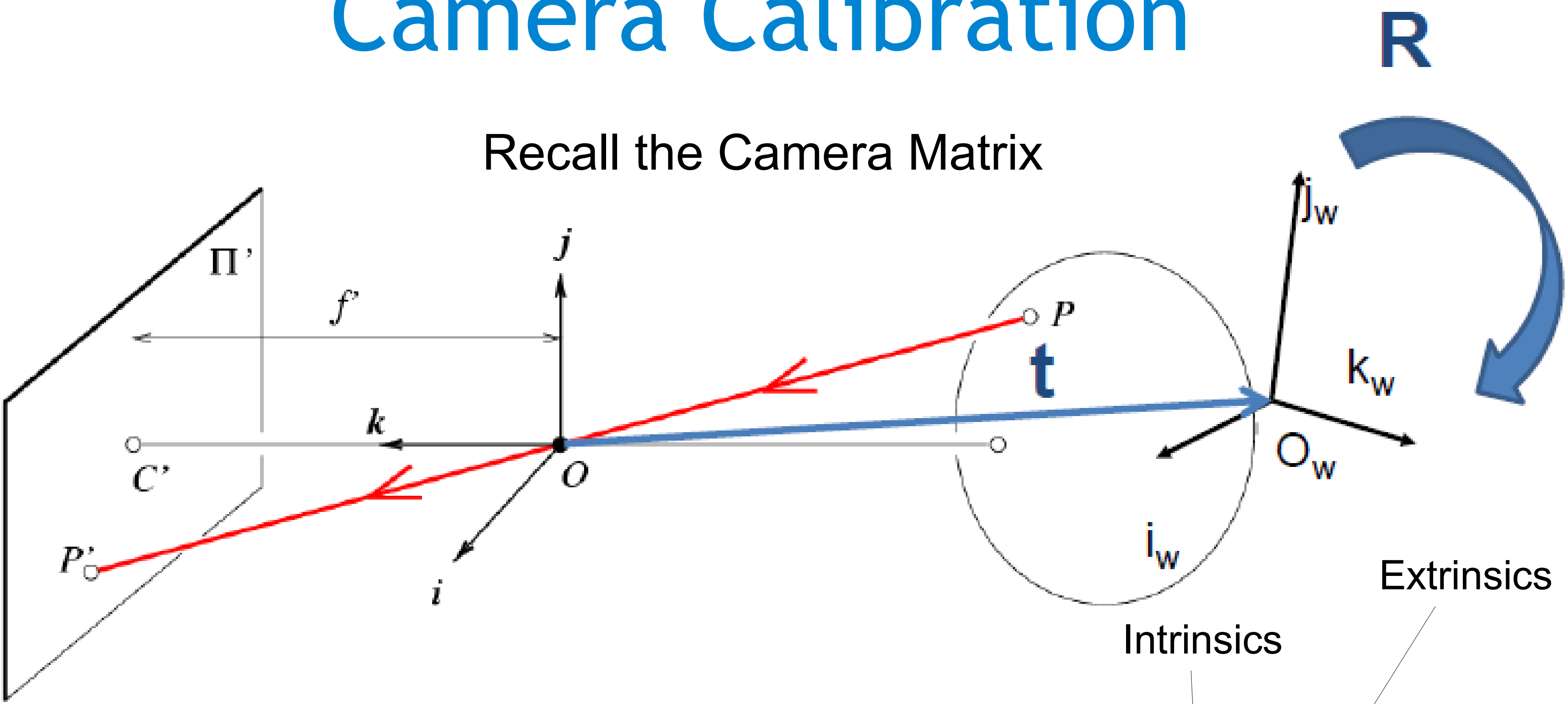
$$\nabla E=0 \quad \rightarrow \quad Ap=b \quad \rightarrow \quad p=A^{-1}b=\begin{bmatrix}\sum_i (x'_i-x_i)/N\\ \sum_i (y'_i-y_i)/N\end{bmatrix}$$

Geometric Alignment

Transform	Matrix	Parameters p	Jacobian J
translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$	(t_x, t_y)	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
Euclidean	$\begin{bmatrix} c_\theta & -s_\theta & t_x \\ s_\theta & c_\theta & t_y \end{bmatrix}$	(t_x, t_y, θ)	$\begin{bmatrix} 1 & 0 & -s_\theta x - c_\theta y \\ 0 & 1 & c_\theta x - s_\theta y \end{bmatrix}$
similarity	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	(t_x, t_y, a, b)	$\begin{bmatrix} 1 & 0 & x & -y \\ 0 & 1 & y & x \end{bmatrix}$
affine	$\begin{bmatrix} 1+a_{00} & a_{01} & t_x \\ a_{10} & 1+a_{11} & t_y \end{bmatrix}$	$(t_x, t_y, a_{00}, a_{01}, a_{10}, a_{11})$	$\begin{bmatrix} 1 & 0 & x & y & 0 & 0 \\ 0 & 1 & 0 & 0 & x & y \end{bmatrix}$

Camera Calibration

Recall the Camera Matrix



$$\tilde{\boldsymbol{x}} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \boldsymbol{K} [\boldsymbol{R} | \boldsymbol{t}] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

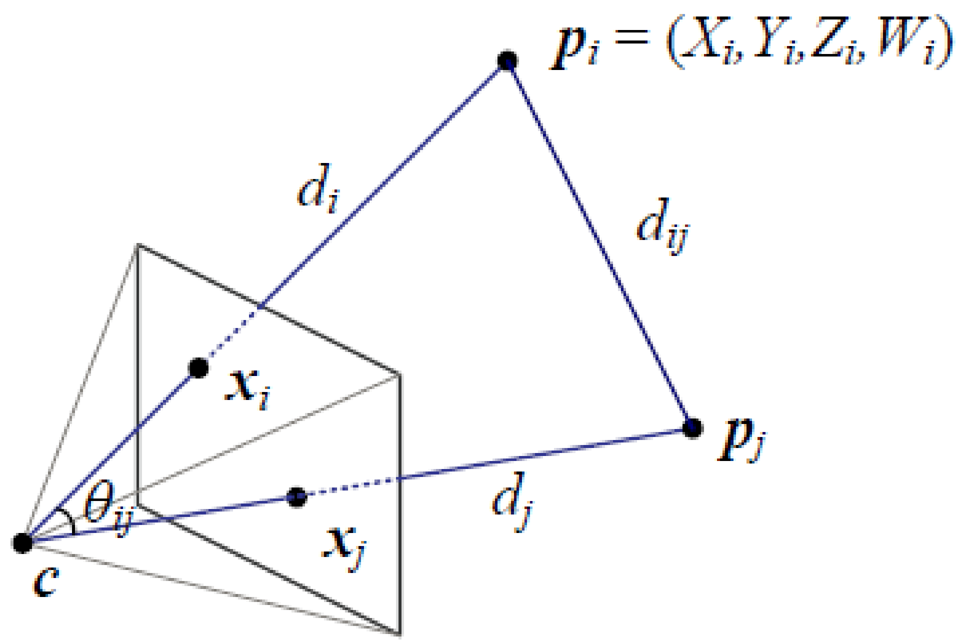
$$\tilde{\boldsymbol{x}} = \boldsymbol{P} \, \boldsymbol{X}$$

Camera Calibration

Given N point correspondences: $x_i \in \mathbb{R}^2$ & $X_i \in \mathbb{R}^3$

such that: $\tilde{x}_i = P X_i$

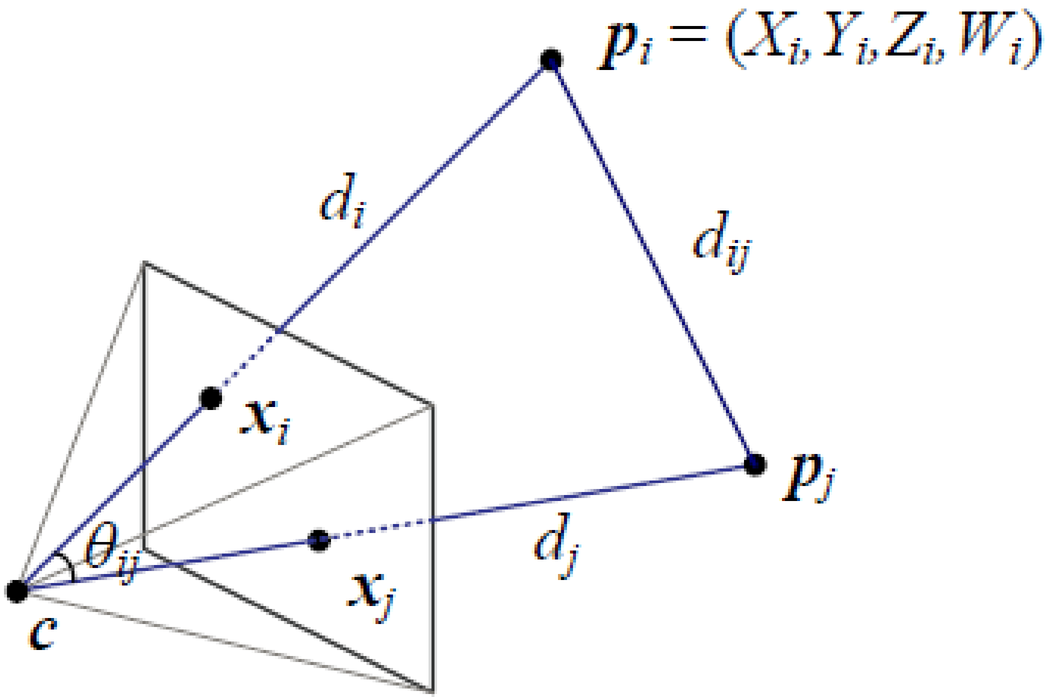
Find the 3x4 Camera Matrix: $\boldsymbol{P} = \boldsymbol{K}[\boldsymbol{R} | \boldsymbol{t}]$



$$\tilde{\boldsymbol{x}} = \boldsymbol{P} \boldsymbol{X}$$

Camera Calibration

- Two ways:
  - Linear Algorithm: Direct Linear Transformation (DLT)
  - Iterative Algorithm: Nonlinear Least Squares (NNLS)



$$\tilde{x} = P\,X$$

Camera Calibration: DLT

Direct Linear Transformation

Let: $P = \begin{bmatrix} p_1^T \\ p_2^T \\ p_3^T \end{bmatrix}$ where $p_i \in \mathbb{R}^4$ $\quad \tilde{\boldsymbol{x}}_{\boldsymbol{i}} = \boldsymbol{P}\boldsymbol{X}_{\boldsymbol{i}} \quad \rightarrow$

$$u_i = \frac{p_1^T \boldsymbol{X}_{\boldsymbol{i}}}{p_3^T \boldsymbol{X}_{\boldsymbol{i}}}$$

$$v_i = \frac{p_2^T \boldsymbol{X}_{\boldsymbol{i}}}{p_3^T \boldsymbol{X}_{\boldsymbol{i}}}$$

$$(p_3^T \boldsymbol{X}_{\boldsymbol{i}}) u_i = p_1^T \boldsymbol{X}_{\boldsymbol{i}} \qquad (p_3^T \boldsymbol{X}_{\boldsymbol{i}}) v_i = p_2^T \boldsymbol{X}_{\boldsymbol{i}}$$

$$(p_1^T - u_i p_3^T) \boldsymbol{X}_{\boldsymbol{i}} = 0 \qquad (p_2^T - v_i p_3^T) \boldsymbol{X}_{\boldsymbol{i}} = 0$$

We have 11 DoF in $\boldsymbol{P}$. Why?

Every point gives 2 equations in entries of $\boldsymbol{P} \rightarrow$ need 6 points

Camera Calibration: DLT

$$(p_3^T \boldsymbol{X_i}) u_i = p_1^T \boldsymbol{X_i} \qquad (p_3^T \boldsymbol{X_i}) v_i = p_2^T \boldsymbol{X_i}$$

$$(p_1^T - u_i p_3^T) \boldsymbol{X_i} = 0 \qquad (p_2^T - v_i p_3^T) \boldsymbol{X_i} = 0$$

Can be written as: $A p = 0$

$$A = \begin{bmatrix} \boldsymbol{X_i^T} & \boldsymbol{0^T} & -u_i \boldsymbol{X_i^T} \\ \boldsymbol{0^T} & \boldsymbol{X_i^T} & -v_i \boldsymbol{X_i^T} \end{bmatrix}_{2\times 12} \qquad p = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}_{12\times 1}$$

Stacking at least 6 points:

$$A = \begin{bmatrix} \boldsymbol{X_1^T} & \boldsymbol{0^T} & -u_1 \boldsymbol{X_1^T} \\ \boldsymbol{0^T} & \boldsymbol{X_1^T} & -v_1 \boldsymbol{X_1^T} \\ \vdots & \vdots & \vdots \\ \boldsymbol{X_N^T} & \boldsymbol{0^T} & -u_N \boldsymbol{X_N^T} \\ \boldsymbol{0^T} & \boldsymbol{X_N^T} & -v_N \boldsymbol{X_N^T} \end{bmatrix}_{2\mathrm{N}\times 12}$$

Camera Calibration: DLT

We have: $A\,p=0$ and want to find p s.t. $\|A\,p\|^2$ is minimum

Linear Least Squares: get normal equations and solve

Normal equations: $\left(A^T A\right)p=0$

p is in the *null space* of $A^T A$

But $p=\mathbf{0}$ minimizes $\|Ap\|^2$

Instead find p that minimizes $\|Ap\|^2$ such that $\|p\|=1$

Camera Calibration: DLT

Find p that minimizes $\|Ap\|^2$ such that $\|p\|=1$

Lagrange Multiplier: min $L(p)=\|Ap\|^2-\lambda(\|p\|^2-1)$

Differentiate w.r.t. p and λ and set to zero:

$$A^T A\, p=\lambda\, p \qquad\qquad p^T p=1$$

p is a unit eigenvector of $A^T A$ with eigenvalue λ

Let x be a unit eigenvector with eigenvalue λ:

$$L(x)=x^T A^T A\, x-\lambda(x^T x-1)=x^T(\lambda\, x)=\lambda$$

So p is the unit eigenvector of $A^T A$ with smallest eigenvalue λ_0

Camera Calibration: DLT

Singular Value Decomposition (SVD)

$$A = U\, S\, V^T$$

$A:\ m \times n$ matrix
$U:\ m \times m$ matrix with orthonormal columns
$V:\ n \times n$ matrix with orthonormal columns
$D:\ m \times n$ diagonal matrix with *singular values* σ_i

Camera Calibration: DLT

Singular Value Decomposition (SVD)

$$A = U\,S\,V^T$$

Properties:

Columns of U are eigenvectors of AA^T

$$A\,A^T = \left(U\,S\,V^T\right)\left(V\,S\,U^T\right) = U\,S^2\,U^T \text{ i.e. } A\,A^T\,U = U\,S^2$$

Columns of V are eigenvectors of $A^T A$

$$A^T A = \left(V\,S\,U^T\right)\left(U\,S\,V^T\right) = V\,S^2\,V^T \text{ i.e. } A^T A\,V = V\,S^2$$

Square of singular values are eigenvalues of AA^T and $A^T A$

Camera Calibration: DLT

Find p that minimizes $\|Ap\|^2$ such that $\|p\| = 1$

→ p is the unit eigenvector of $A^T A$ with smallest eigenvalue λ_0

SVD: $A = U\,S\,V^T$

p is the column of V with smallest magnitude singular value σ_0

Camera Calibration: DLT

Now we have an estimate for $P = K[R|t]$

and want to estimate K, R, and t

RQ factorization of $P_{3\times3}$ where:

$P = [P_{3\times3} \quad P_{3\times1}]$

R is upper triangular (i.e. K)

Q is orthonormal (i.e. R)

Then compute t such that $K t = P_{3\times1}$

Camera Calibration: NNLS

Nonlinear Least Squares

Projection Equation: $x_i = f(X_i; \boldsymbol{K}, \boldsymbol{R}, \boldsymbol{t}) = \begin{bmatrix} u(X_i; \boldsymbol{K}, \boldsymbol{R}, \boldsymbol{t}) \\ v(X_i; \boldsymbol{K}, \boldsymbol{R}, \boldsymbol{t}) \end{bmatrix}$

Parametrization: $\boldsymbol{\xi}$ e.g. translation parameters, focal lengths, ...

Projection Equation: $x_i = f(X_i; \boldsymbol{\xi})$

Taylor's Expansion: $f(X_i; \xi + \Delta\xi) \approx f(X_i; \xi) + J(X_i)\Delta\xi$

$$\text{Jacobian: } J(X_i) = \begin{bmatrix} \nabla u^T \\ \nabla v^T \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial \xi_1} & \cdots & \frac{\partial u}{\partial \xi_p} \\ \frac{\partial v}{\partial \xi_1} & \cdots & \frac{\partial v}{\partial \xi_p} \end{bmatrix}$$

Camera Calibration: NNLS

Taylor's Expansion: $f(X_i;\xi+\Delta\xi)\approx f(X_i;\xi)+J(X_i)\Delta\xi$

Reprojection Error:

$$E(\xi)=\sum_i\left[u(X_i;\xi)-u_i\right]^2+\left[v(X_i;\xi)-v_i\right]^2=\|f(X_i;\xi)-x_i\|^2$$

$$E(\xi+\Delta\xi)\approx\sum_i\|f(X_i;\xi)+J(X_i)\Delta\xi-x_i\|^2$$

$$E(\xi+\Delta\xi)\approx\sum_i\|J(X_i)\Delta\xi-r_i\|^2 \text{ where } r_i=f(X_i;\xi)-x_i$$

$$E(\xi+\Delta\xi)\approx\Delta\xi^T\left(\sum_i J^T J\right)\Delta\xi-2\Delta\xi^T\left(\sum_i J^T r_i\right)+\sum_i\|r_i\|^2$$

$$E(\xi+\Delta\xi)\approx\Delta\xi^T A\Delta\xi-2\Delta\xi^T b+c$$

Linear Least Squares ! → find $\Delta\xi$ s.t. $A^T A\Delta\xi=A^T b$

Camera Calibration: NNLS

Nonlinear Least Squares

1. Start at time t = 0

$$\xi^0$$

2. Setup the normal equations:

$$A^T A \Delta \xi^t = A^T b$$

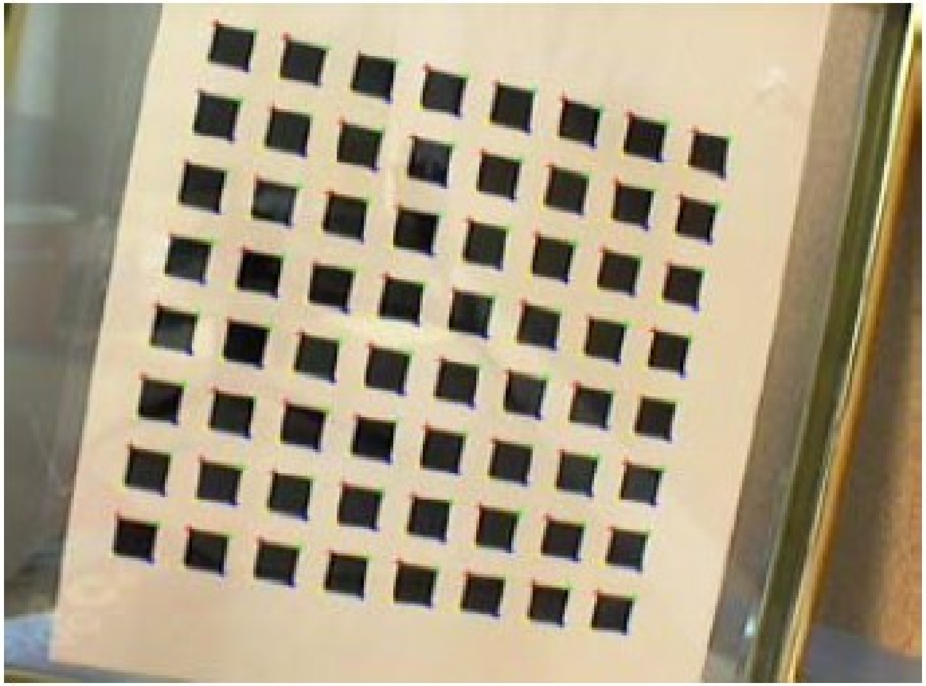
3. Solve for: ξ^t

4. Update: $\xi^{t+1} = \xi^t + \Delta \xi^t$

5. Repeat 2-5 until convergence

Camera Calibration: NNLS

- How to get initial guess?
  - Direct Linear Transformation !
- Parametrization
  - Rotation: Euler's angles, quaternion, …
  - Intrinsics
- Correspondences
  - Calibration Targets



Recap

- Geometric Alignment
- Parameter Estimation
- Linear Least Squares
- Camera Calibration
  - Direct Linear Transformation
  - Nonlinear Least Squares