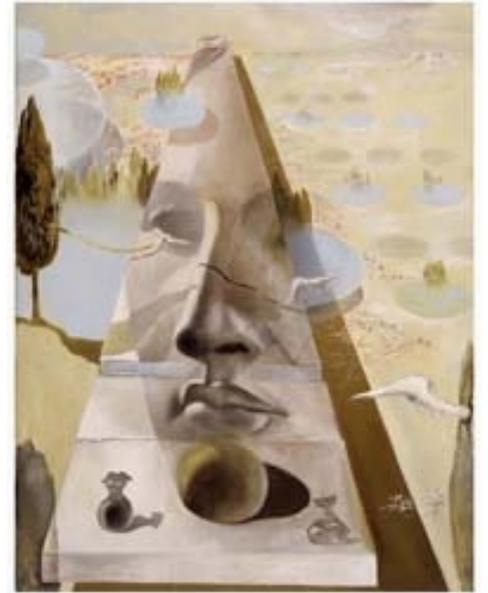


Lecture 3

Camera Models 2 & Camera Calibration



Professor Silvio Savarese
Stanford Vision and Learning Lab

Lecture 3

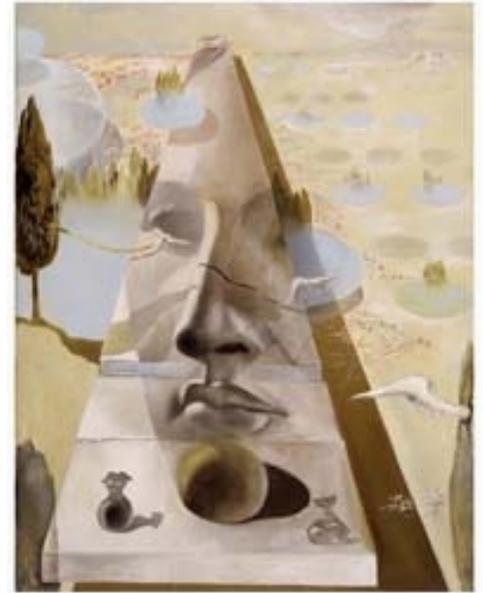
Camera Models 2 & Camera Calibration

- Recap of camera models
- Camera calibration problem
- Camera calibration with radial distortion
- Example

Reading:

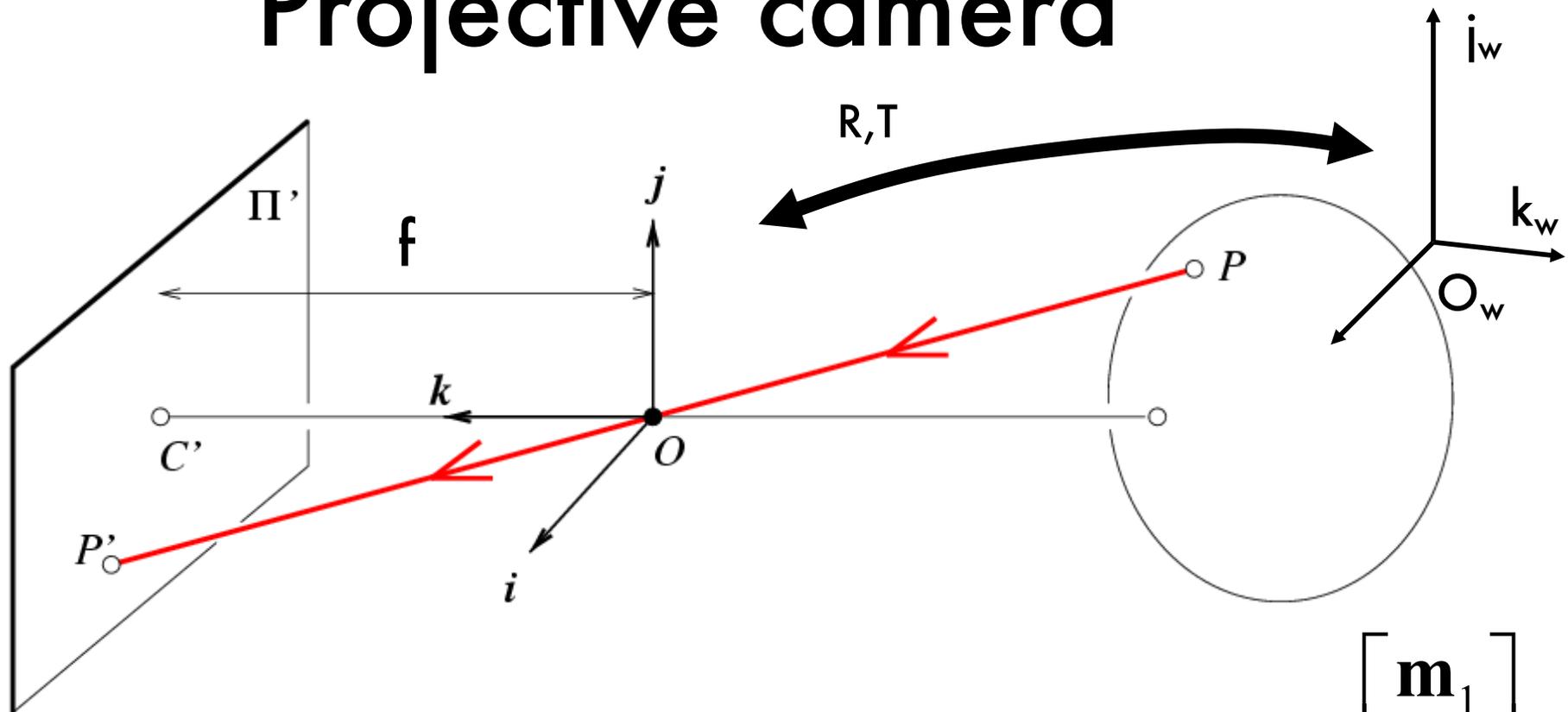
[FP] Chapter 1 "Geometric Camera Calibration"

[HZ] Chapter 7 "Computation of Camera Matrix P"



Some slides in this lecture are courtesy to Profs. J. Ponce, F-F Li

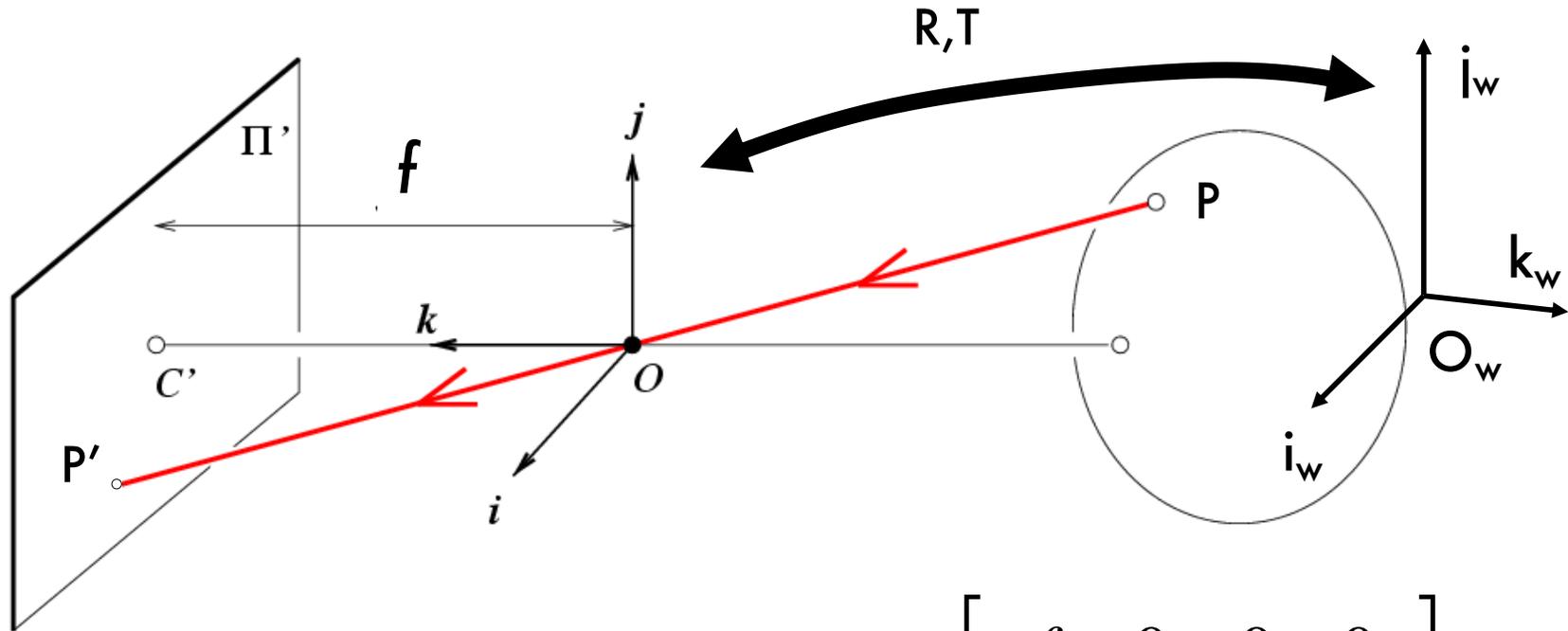
Projective camera



$$P'_{3 \times 1} = M P_w = K_{3 \times 3} \begin{bmatrix} R & T \end{bmatrix}_{3 \times 4} P_{w4 \times 1} \quad M = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_w = \begin{bmatrix} \mathbf{m}_1 P_w \\ \mathbf{m}_2 P_w \\ \mathbf{m}_3 P_w \end{bmatrix} \quad \xrightarrow{E} P'_E = \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w} \right) \quad \text{[Eq. 1]}$$

Exercise!



$$M = K \begin{bmatrix} R & T \end{bmatrix} = K \begin{bmatrix} I & 0 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\rightarrow P'_E = \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w} \right) = \left(f \frac{x_w}{z_w}, f \frac{y_w}{z_w} \right) \quad P_w = \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

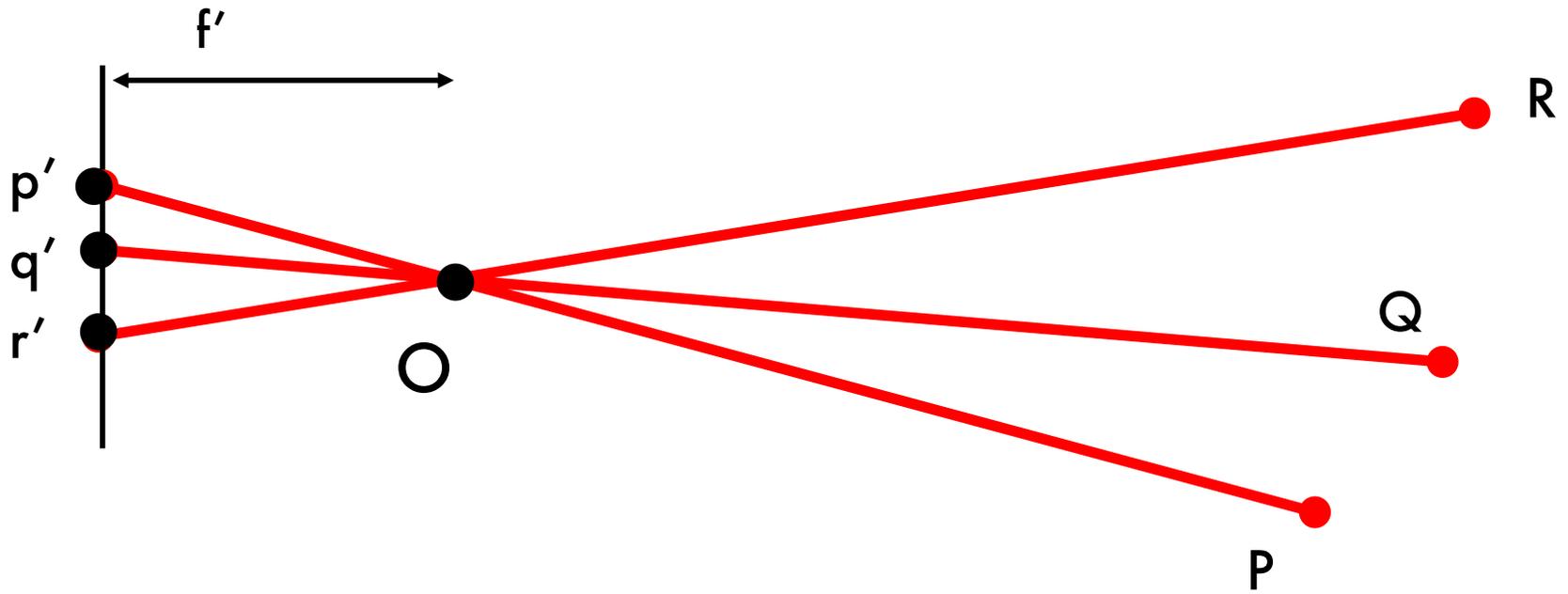
Canonical Projective Transformation

$$P' = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$P' = M P$$
$$\mathbb{R}^4 \xrightarrow{H} \mathbb{R}^3$$

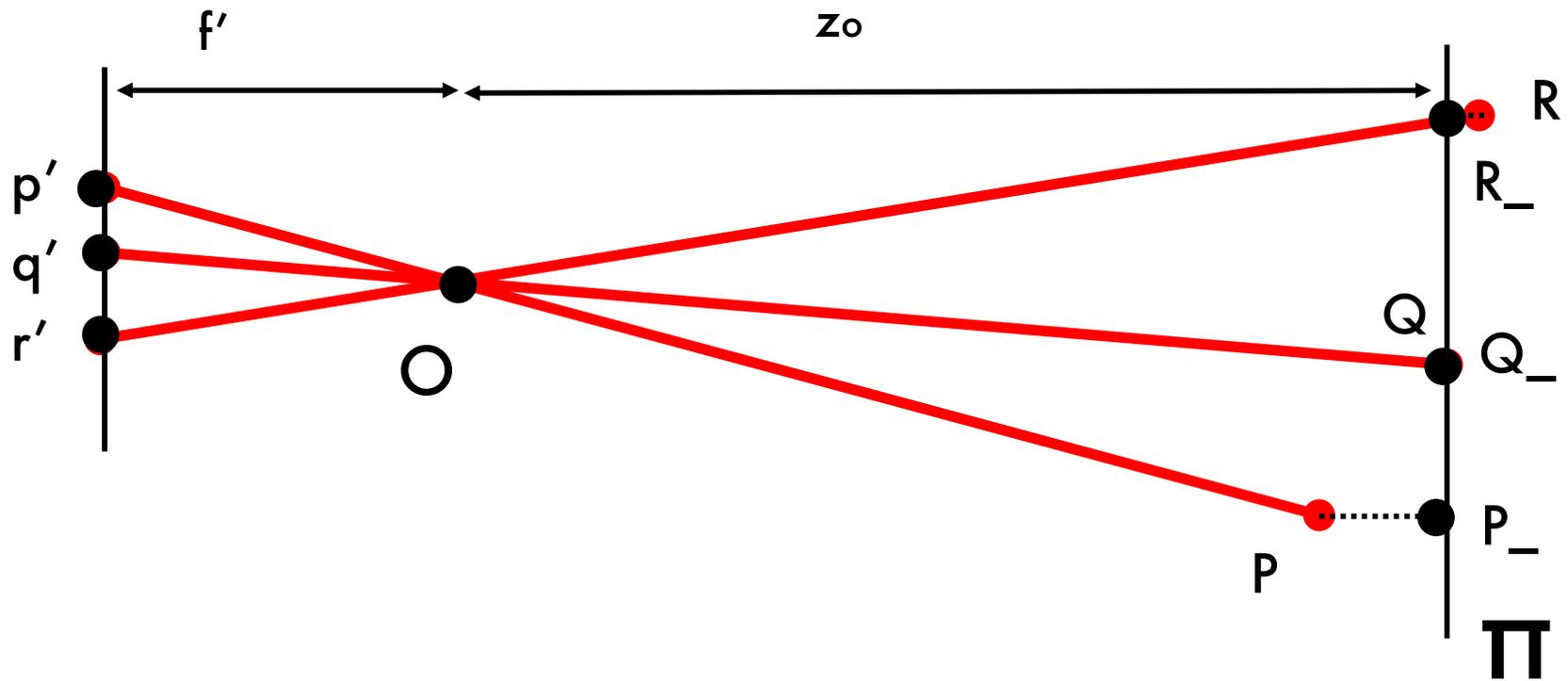
$$P'_i = \begin{bmatrix} \frac{x}{z} \\ \frac{y}{z} \end{bmatrix}$$

Projective camera

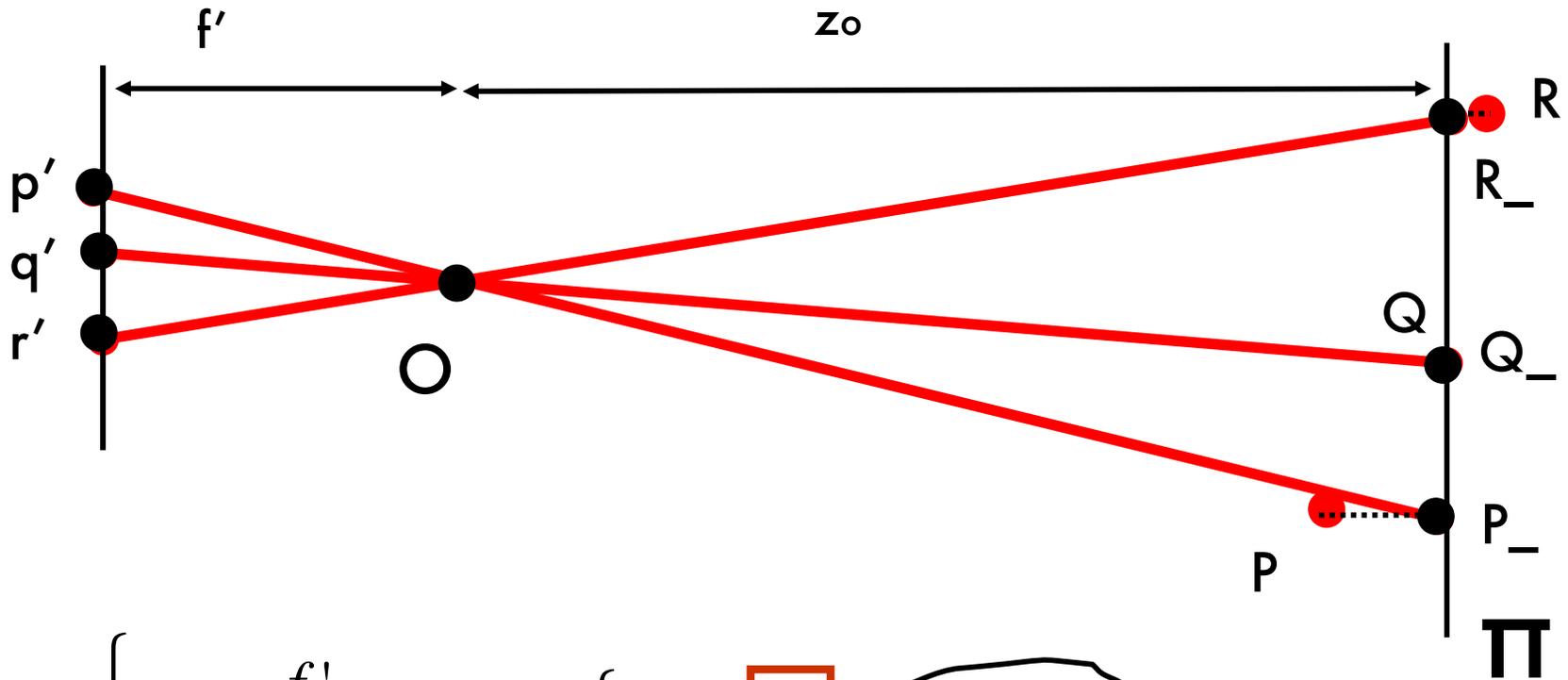


Weak perspective projection

When the relative scene depth is small compared to its distance from the camera



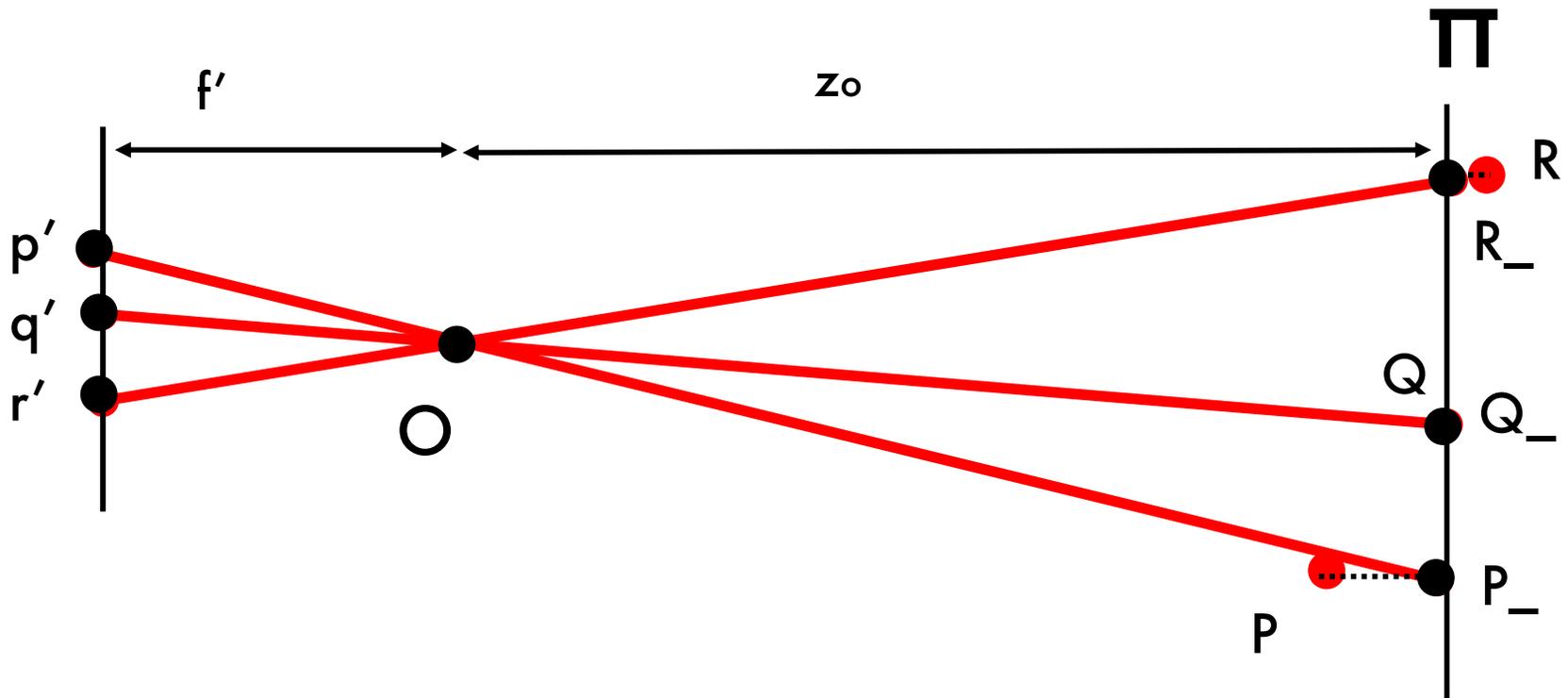
Weak perspective projection



$$\left\{ \begin{array}{l} x' = \frac{f'}{z} x \\ y' = \frac{f'}{z} y \end{array} \right. \rightarrow \left\{ \begin{array}{l} x' = \frac{f'}{z_0} x \\ y' = \frac{f'}{z_0} y \end{array} \right.$$

Magnification m

Weak perspective projection



Projective (perspective)

Weak perspective

$$M = K \begin{bmatrix} R & T \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{v} & \mathbf{1} \end{bmatrix} \Rightarrow M = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

$$P' = M P_w = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_w = \begin{bmatrix} \mathbf{m}_1 P_w \\ \mathbf{m}_2 P_w \\ \mathbf{m}_3 P_w \end{bmatrix}$$

$$M = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{v} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

$$\overset{\mathbf{E}}{\rightarrow} \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w} \right)$$

Perspective

$$P' = M P_w = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_w = \begin{bmatrix} \mathbf{m}_1 P_w \\ \mathbf{m}_2 P_w \\ 1 \end{bmatrix}$$

$$M = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 & & & \\ & \mathbf{m}_2 & & \\ & & 0 & 0 & 0 & 1 \end{bmatrix}$$

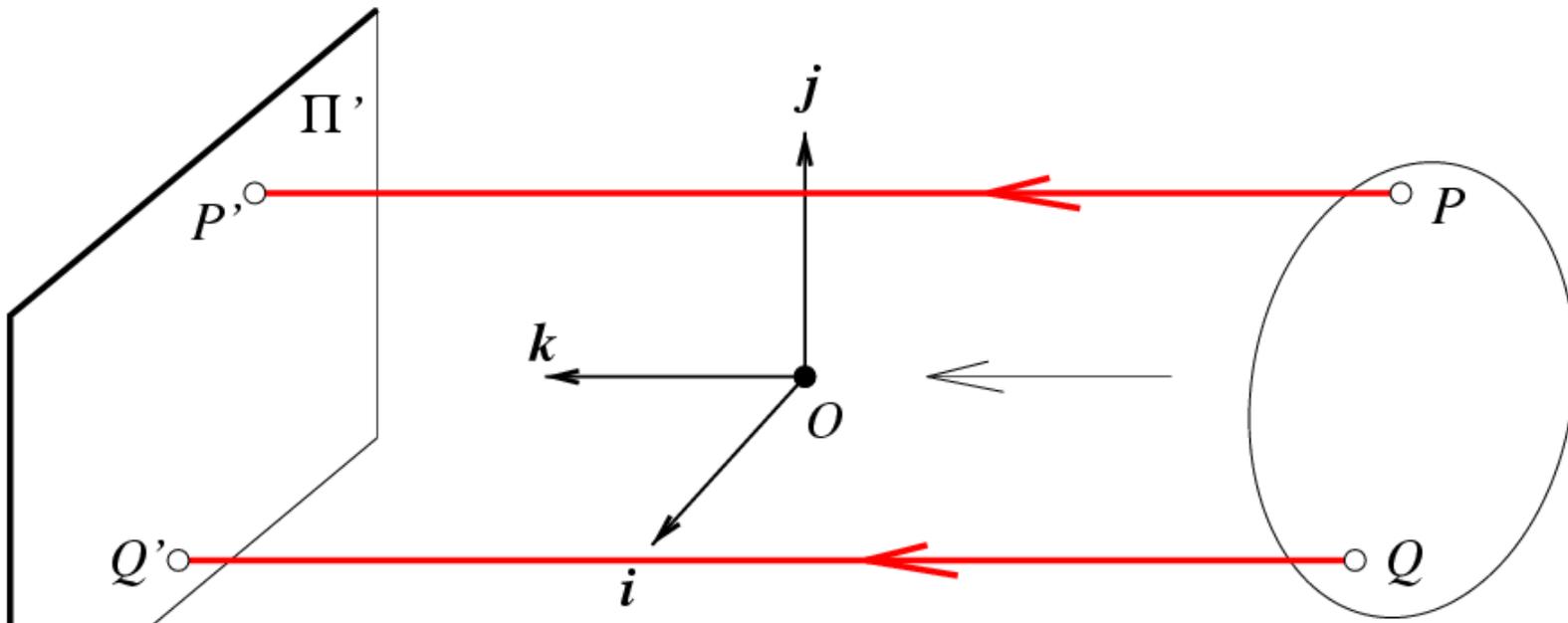
$$\overset{\mathbf{E}}{\rightarrow} (\mathbf{m}_1 P_w, \mathbf{m}_2 P_w)$$

↑ ↑
magnification

Weak perspective

Orthographic (affine) projection

Distance from center of projection to image plane is infinite

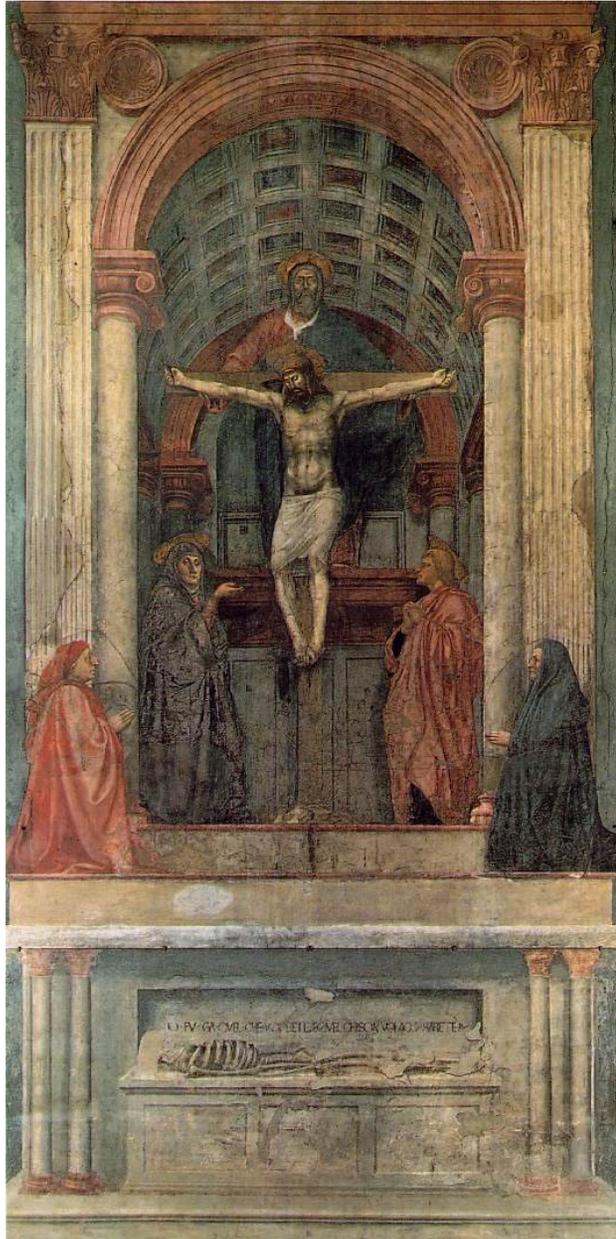


$$\begin{cases} x' = \frac{f'}{z} x \\ y' = \frac{f'}{z} y \end{cases} \rightarrow \begin{cases} x' = x \\ y' = y \end{cases}$$

Pros and Cons of These Models

- Weak perspective results in much simpler math.
 - Accurate when object is small and distant.
 - Most useful for recognition.
- Pinhole perspective is much more accurate for modeling the 3D-to-2D mapping.
 - Used in structure from motion or SLAM.

One-point perspective



Masaccio, *Trinity*,
Santa Maria
Novella, Florence,
1425-28

il Canaletto *The Piazzetta, Venice*,



Weak perspective projection



The Kangxi Emperor's Southern Inspection Tour (1691-1698) By Wang Hui

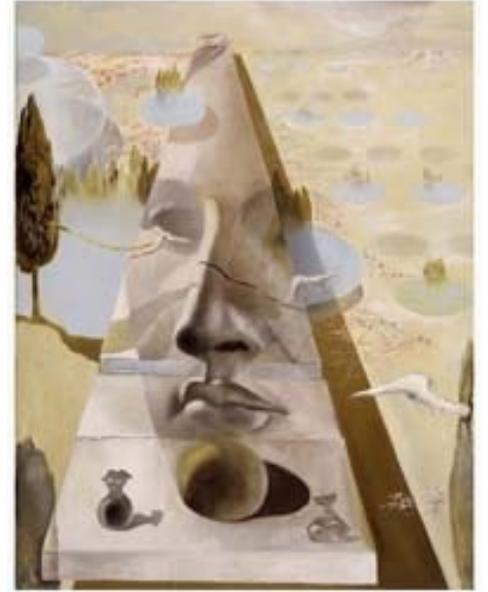
Weak perspective projection



The Kangxi Emperor's Southern Inspection Tour (1691-1698) By Wang Hui

Lecture 3

Camera Calibration



- Recap of camera models
- Camera calibration problem
- Camera calibration with radial distortion
- Example

Reading: **[FP]** Chapter 1 “Geometric Camera Calibration”
 [HZ] Chapter 7 “Computation of Camera Matrix P ”

Some slides in this lecture are courtesy to Profs. J. Ponce, F-F Li

Projective camera

$$P' = M P_w = \underbrace{K}_{\text{Internal parameters}} \underbrace{[R \quad T]}_{\text{External parameters}} P_w$$

Internal parameters

External parameters

$$M = \begin{pmatrix} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{pmatrix}_{3 \times 4}$$

$$K = \begin{bmatrix} \alpha & -\alpha \cot \theta & u_0 \\ 0 & \frac{\beta}{\sin \theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix}$$

$$T = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

Goal of calibration

$$P' = M P_w = K \begin{bmatrix} R & T \end{bmatrix} P_w$$

Internal parameters

External parameters

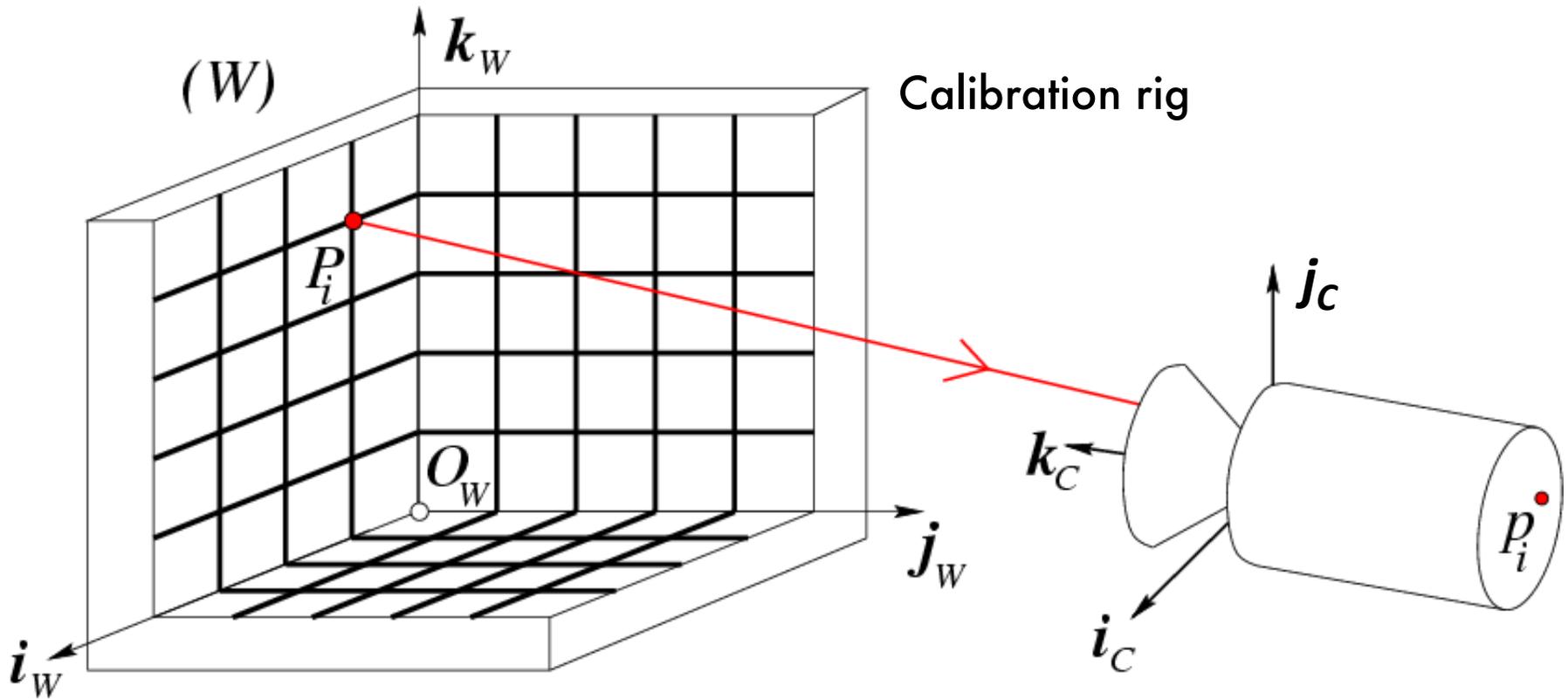
Estimate intrinsic and extrinsic parameters from 1 or multiple images

Change notation:

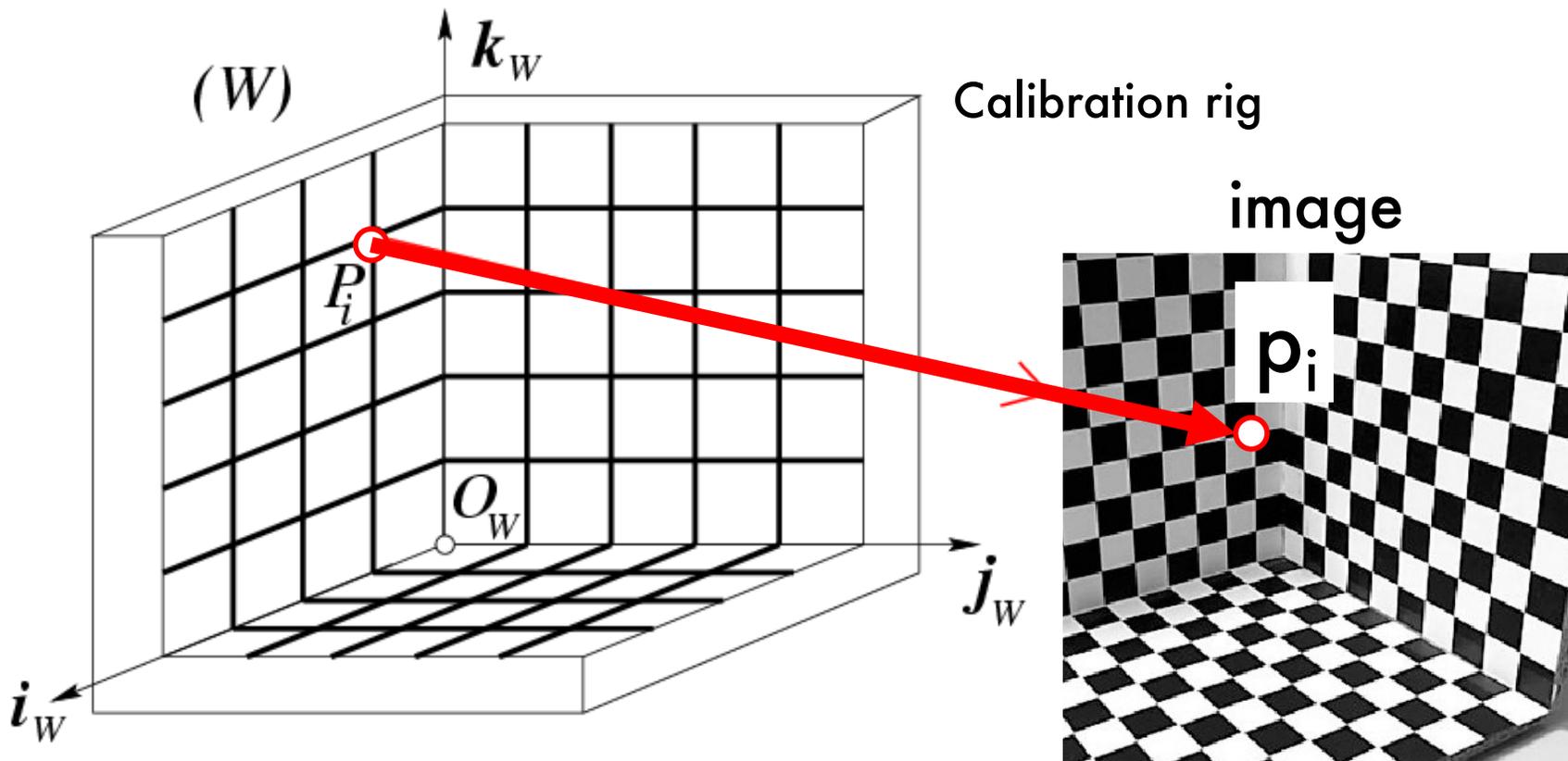
$$P = P_w$$

$$p = P'$$

Calibration Problem



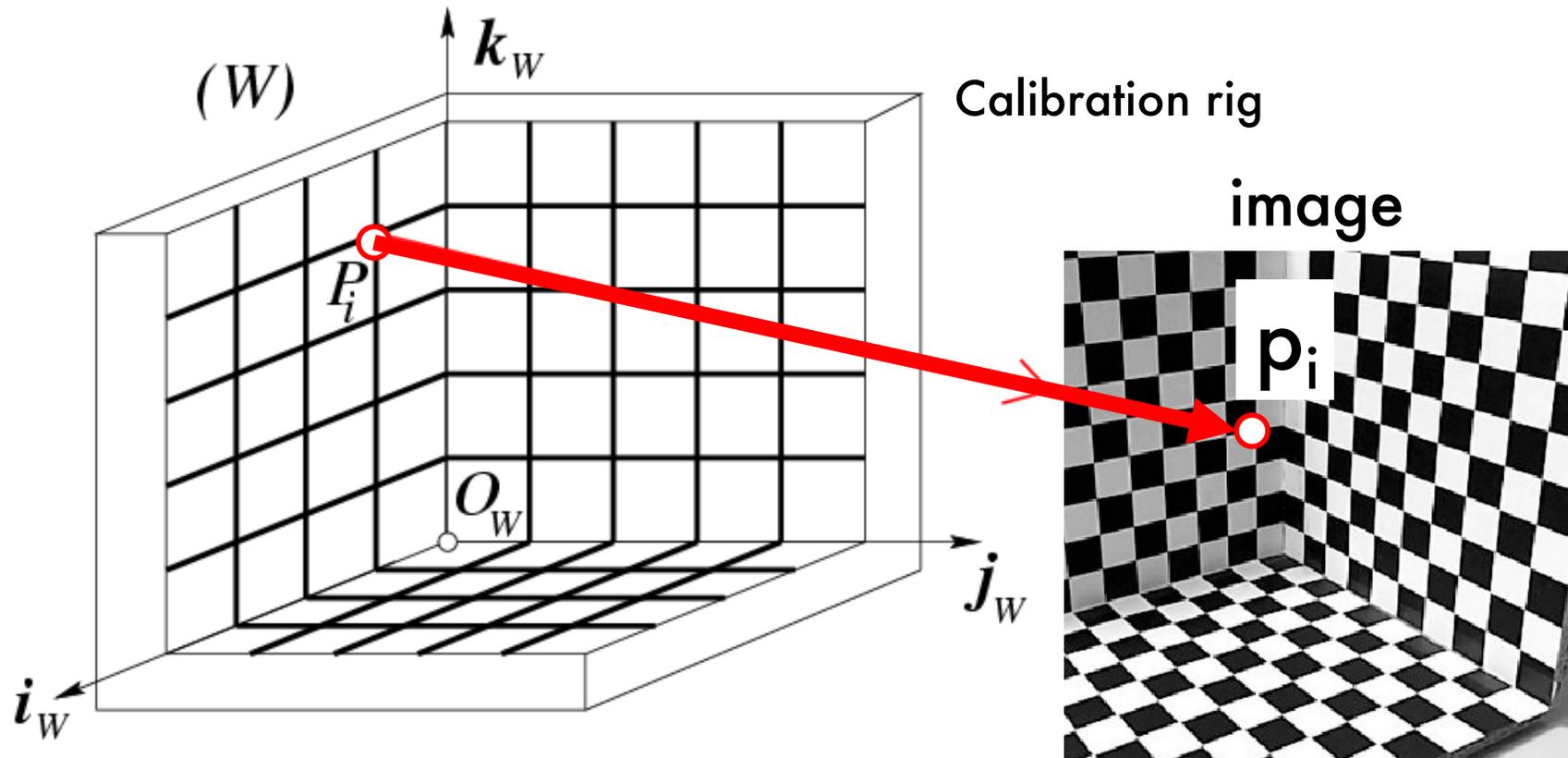
Calibration Problem



- $P_1 \dots P_n$ with **known** positions in $[O_w, i_w, j_w, k_w]$
- p_1, \dots, p_n **known** positions in the image

Goal: compute intrinsic and extrinsic parameters

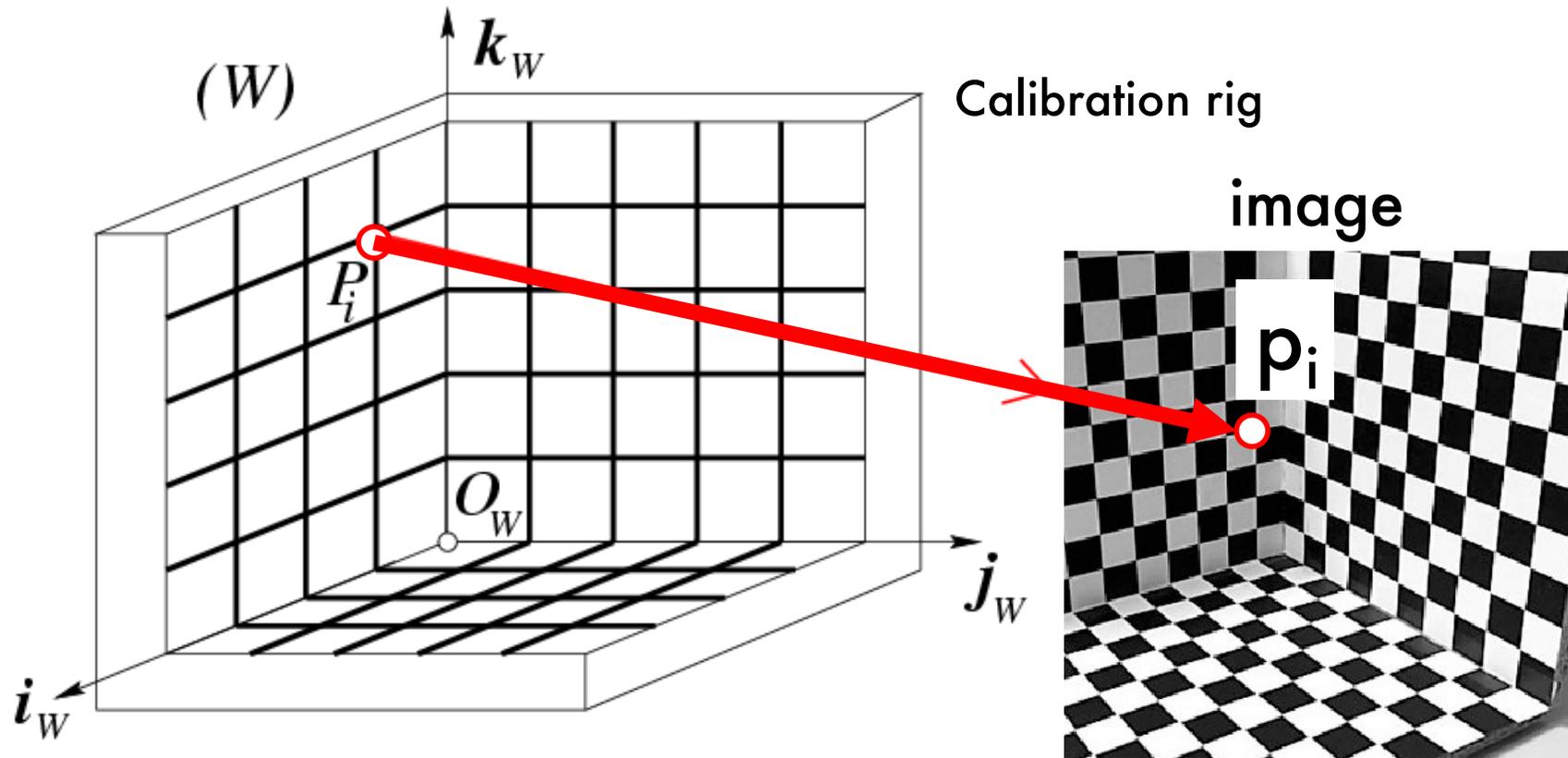
Calibration Problem



How many correspondences do we need?

- M has 11 unknowns • We need 11 equations

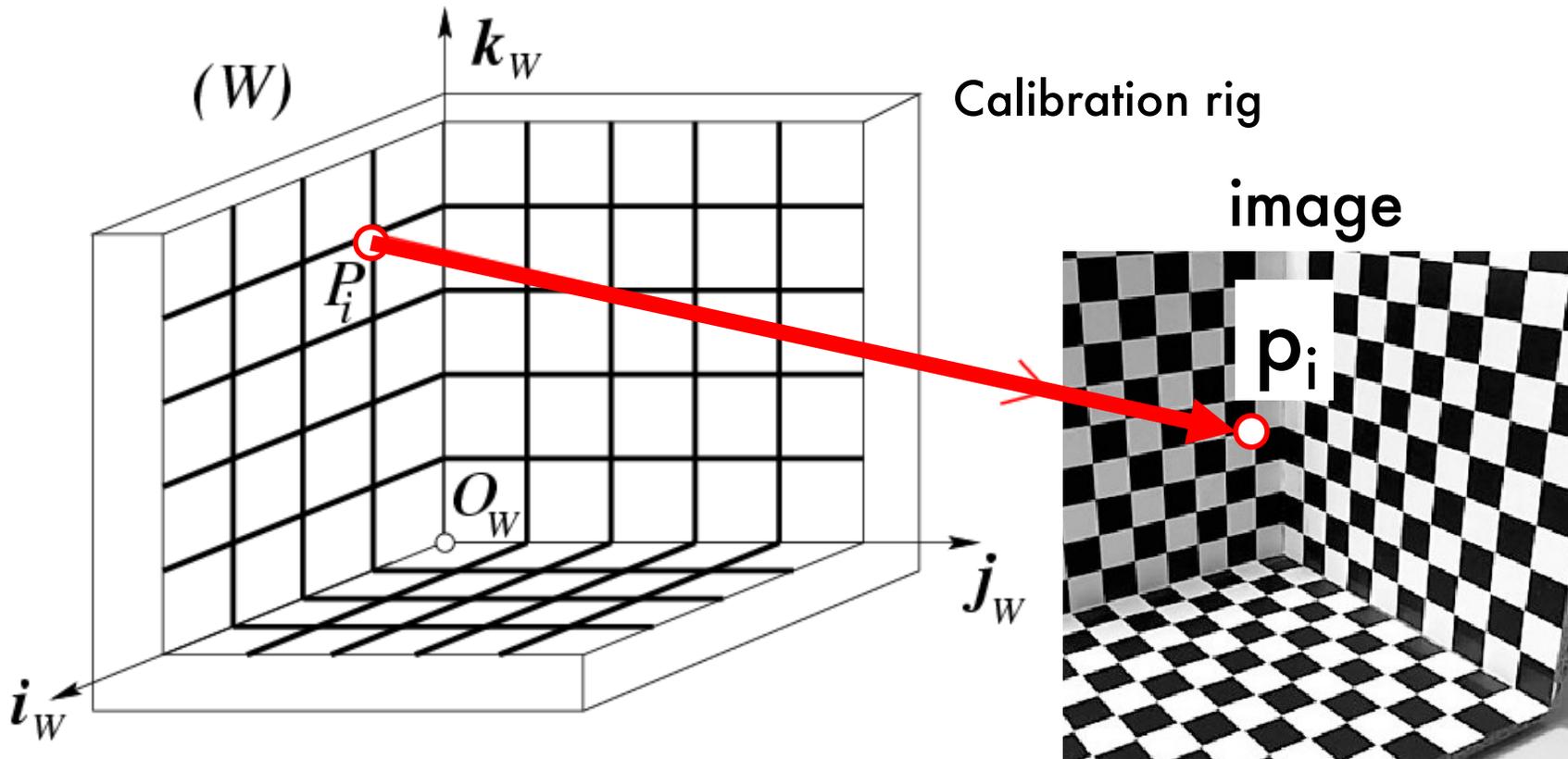
Calibration Problem



How many correspondences do we need?

- M has 11 unknowns
- We need 11 equations
- 6 correspondences would do it

Calibration Problem



In practice, using more than 6 correspondences enables more robust results

Calibration Problem

$$\text{[Eq. 1]} \quad \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \\ \frac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \end{bmatrix}$$

$$u_i = \frac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \rightarrow u_i(\mathbf{m}_3 P_i) = \mathbf{m}_1 P_i \rightarrow u_i(\mathbf{m}_3 P_i) - \mathbf{m}_1 P_i = 0$$

$$v_i = \frac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \rightarrow v_i(\mathbf{m}_3 P_i) = \mathbf{m}_2 P_i \rightarrow v_i(\mathbf{m}_3 P_i) - \mathbf{m}_2 P_i = 0$$

[Eqs. 2]

Calibration Problem

$$\left\{ \begin{array}{l} u_1(\mathbf{m}_3 P_1) - \mathbf{m}_1 P_1 = 0 \\ v_1(\mathbf{m}_3 P_1) - \mathbf{m}_2 P_1 = 0 \\ \vdots \\ u_i(\mathbf{m}_3 P_i) - \mathbf{m}_1 P_i = 0 \\ v_i(\mathbf{m}_3 P_i) - \mathbf{m}_2 P_i = 0 \\ \vdots \\ u_n(\mathbf{m}_3 P_n) - \mathbf{m}_1 P_n = 0 \\ v_n(\mathbf{m}_3 P_n) - \mathbf{m}_2 P_n = 0 \end{array} \right. \quad [\text{Eqs. 3}]$$

Block Matrix Multiplication

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

What is AB ?

$$AB = \begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{bmatrix}$$

Calibration Problem

$$\left\{ \begin{array}{l} -u_1(\mathbf{m}_3^T P_1) + \mathbf{m}_1^T P_1 = 0 \\ -v_1(\mathbf{m}_3^T P_1) + \mathbf{m}_2^T P_1 = 0 \\ \vdots \\ -u_n(\mathbf{m}_3^T P_n) + \mathbf{m}_1^T P_n = 0 \\ -v_n(\mathbf{m}_3^T P_n) + \mathbf{m}_2^T P_n = 0 \end{array} \right. \quad [\text{Eqs. 3}]$$

$$\mathbf{P} \mathbf{m} = \mathbf{0} \quad [\text{Eq. 4}]$$

known unknown

Homogenous
linear system

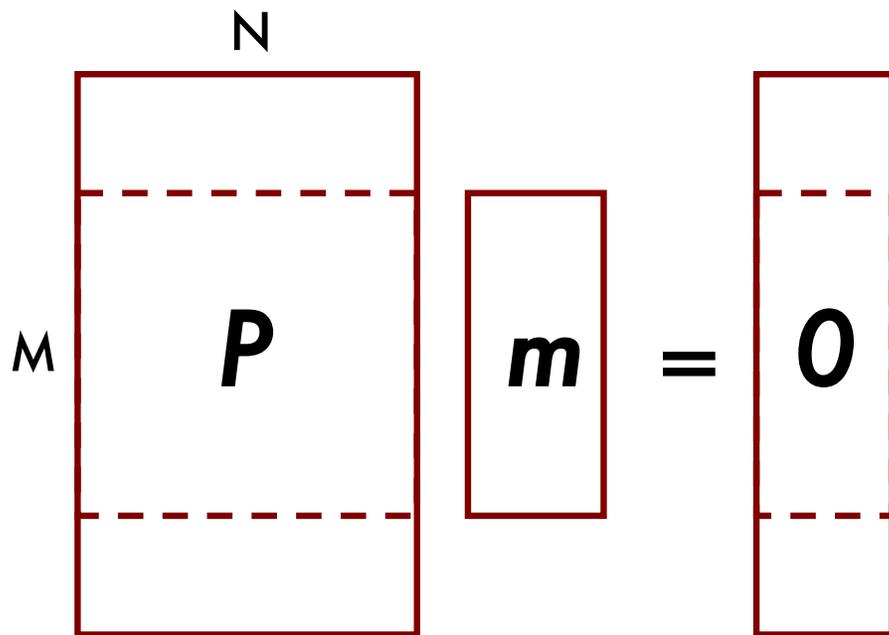
$$\mathbf{P} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{P}_1^T & \mathbf{0}^T & -u_1 \mathbf{P}_1^T \\ \mathbf{0}^T & \mathbf{P}_1^T & -v_1 \mathbf{P}_1^T \\ \vdots & \vdots & \vdots \\ \mathbf{P}_n^T & \mathbf{0}^T & -u_n \mathbf{P}_n^T \\ \mathbf{0}^T & \mathbf{P}_n^T & -v_n \mathbf{P}_n^T \end{pmatrix} \begin{matrix} 1 \times 4 \\ \\ \\ 2n \times 12 \end{matrix}$$

$$\mathbf{m} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{pmatrix} \begin{matrix} 4 \times 1 \\ \\ 12 \times 1 \end{matrix}$$

Homogeneous $M \times N$ Linear Systems

M =number of equations = $2n$

N =number of unknown = 11



Rectangular system ($M > N$)

- 0 is always a solution
- To find non-zero solution

Minimize $|Pm|^2$

under the constraint $|m|^2 = 1$

Calibration Problem

$$\mathbf{P} \mathbf{m} = 0$$

- How do we solve this homogenous linear system?
- Via SVD decomposition!

Calibration Problem

$$\mathbf{P} \mathbf{m} = 0$$

SVD decomposition of \mathbf{P}

$$\mathbf{U}_{2n \times 12} \mathbf{D}_{12 \times 12} \mathbf{V}^T_{12 \times 12}$$

Last column of \mathbf{V} gives \mathbf{m}

Why? See pag 592 of HZ

$$\mathbf{m} \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{pmatrix}$$

\mathbf{M}

Extracting camera parameters

$$M = \begin{pmatrix} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{pmatrix} \rho$$

Extracting camera parameters

See [FP],
Sec. 1.3.1

$$\frac{M}{\rho} = \left(\begin{array}{c|c} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \hline \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \hline \mathbf{r}_3^T & t_z \end{array} \right)$$

A \mathbf{b}

Box 1

$$A = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Estimated values

Intrinsic

$$\rho = \frac{\pm 1}{|\mathbf{a}_3|} \quad \begin{array}{l} u_o = \rho^2 (\mathbf{a}_1 \cdot \mathbf{a}_3) \\ v_o = \rho^2 (\mathbf{a}_2 \cdot \mathbf{a}_3) \end{array}$$

$$\cos \theta = \frac{(\mathbf{a}_1 \times \mathbf{a}_3) \cdot (\mathbf{a}_2 \times \mathbf{a}_3)}{|\mathbf{a}_1 \times \mathbf{a}_3| \cdot |\mathbf{a}_2 \times \mathbf{a}_3|}$$

Extracting camera parameters

See [FP],
Sec. 1.3.1

$$\frac{M}{\rho} = \left(\begin{array}{c|c} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{array} \right)$$

A \mathbf{b}

Box 1

$$A = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Estimated values

Intrinsic

$$\alpha = \rho^2 |\mathbf{a}_1 \times \mathbf{a}_3| \sin \theta$$

$$\beta = \rho^2 |\mathbf{a}_2 \times \mathbf{a}_3| \sin \theta$$

Extracting camera parameters

See [FP],
Sec. 1.3.1

$$\frac{M}{\rho} = \left(\begin{array}{c|c} \alpha \mathbf{r}_1^T - \alpha \cot \theta \mathbf{r}_2^T + u_0 \mathbf{r}_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \hline \frac{\beta}{\sin \theta} \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & \frac{\beta}{\sin \theta} t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{array} \right) = \mathbf{K}[\mathbf{R} \quad \mathbf{T}]$$

A \mathbf{b}

Box 1

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

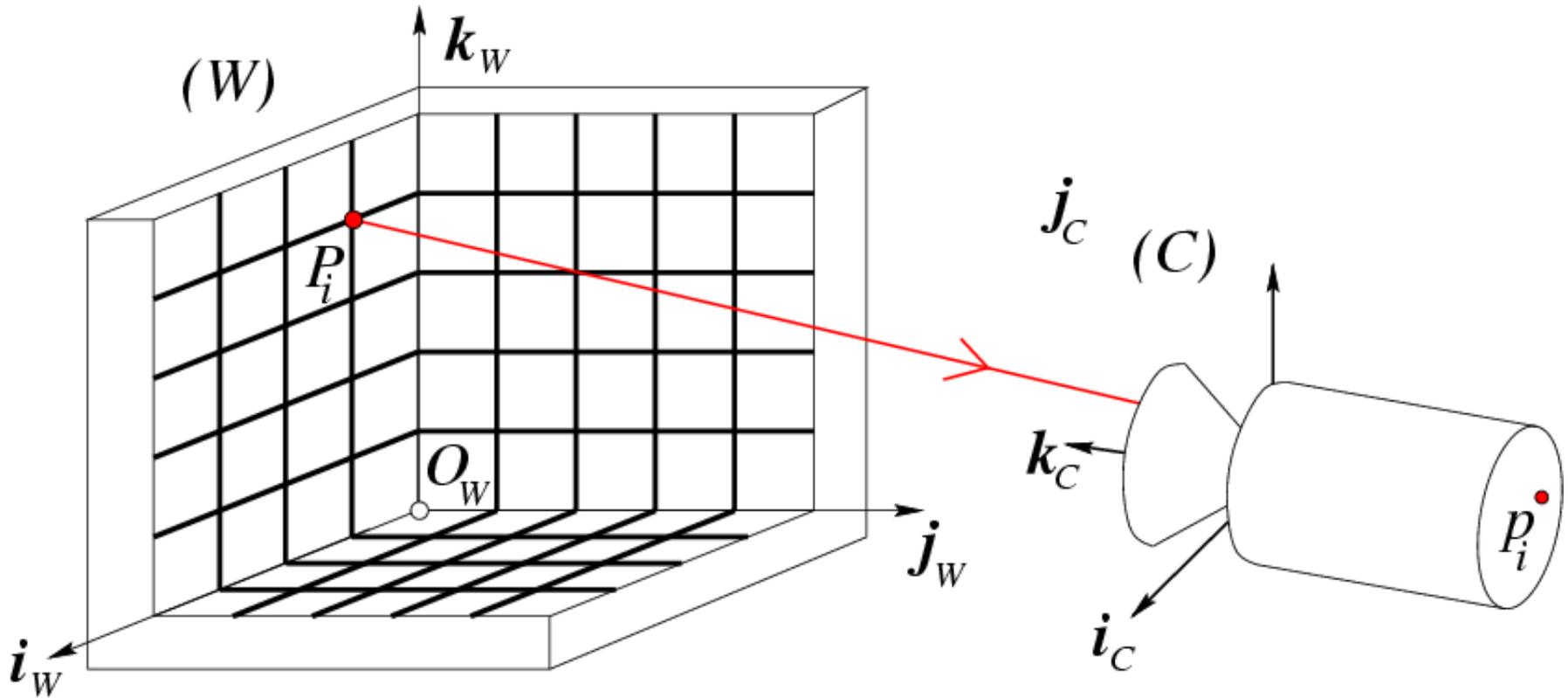
Estimated values

Extrinsic

$$\mathbf{r}_1 = \frac{(\mathbf{a}_2 \times \mathbf{a}_3)}{|\mathbf{a}_2 \times \mathbf{a}_3|} \quad \mathbf{r}_3 = \frac{\pm \mathbf{a}_3}{|\mathbf{a}_3|}$$

$$\mathbf{r}_2 = \mathbf{r}_3 \times \mathbf{r}_1 \quad \mathbf{T} = \rho \mathbf{K}^{-1} \mathbf{b}$$

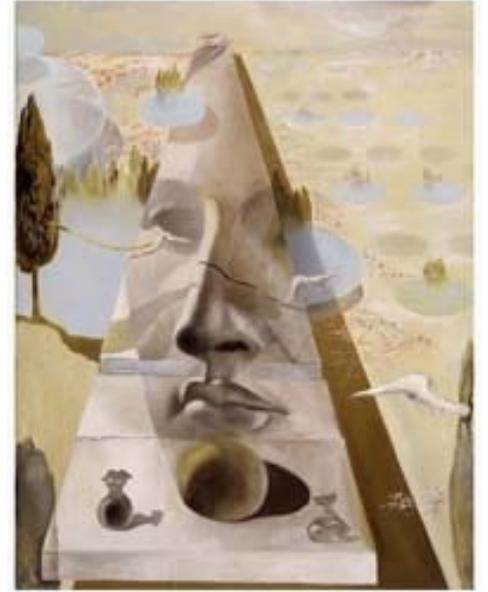
Degenerate cases



- P_i 's cannot lie on the same plane!
- Points cannot lie on the intersection curve of two quadric surfaces [FP] section 1.3

Lecture 3

Camera Calibration



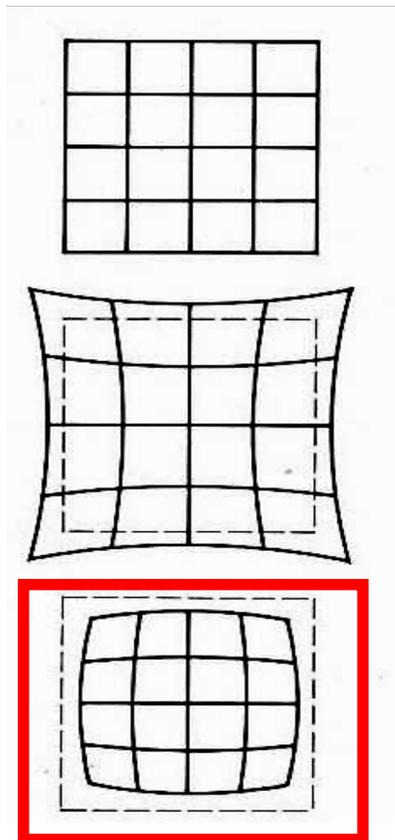
- Recap of projective cameras
- Camera calibration problem
- Camera calibration with radial distortion
- Example

Reading: **[FP]** Chapter 1 "Geometric Camera Calibration"
 [HZ] Chapter 7 "Computation of Camera Matrix P "

Some slides in this lecture are courtesy to Profs. J. Ponce, F-F Li

Radial Distortion

- Image magnification (in)decreases with distance from the optical axis
- Caused by imperfect lenses
- Deviations are most noticeable for rays that pass through the edge of the lens



No distortion

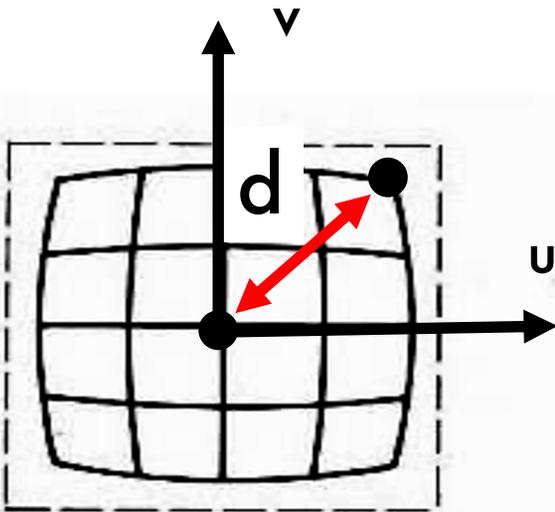
Pin cushion

Barrel



Radial Distortion

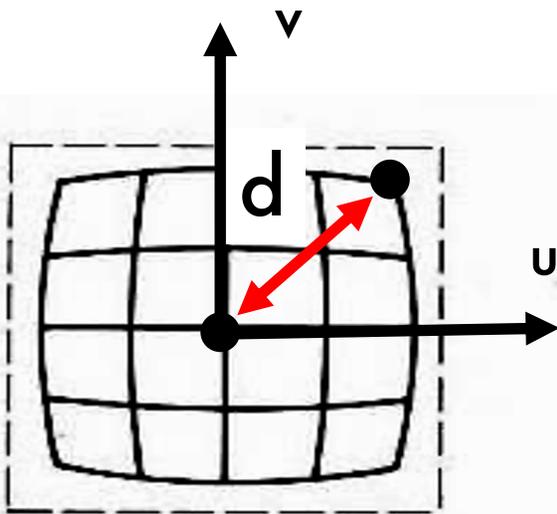
Image magnification decreases with distance from the optical center



$$\begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{M} \mathbf{P}_i \rightarrow \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \mathbf{p}_i \quad \mathbf{S}_\lambda$$

Radial Distortion

Image magnification decreases with distance from the optical center



$$\begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{M} \mathbf{P}_i \rightarrow \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \mathbf{p}_i \quad \mathbf{S}_\lambda$$

Distortion coefficient

$$\lambda = 1 \pm \sum_{p=1}^3 \kappa_p d^{2p}$$

[Eq. 5] Polynomial function

$$d^2 = a u^2 + b v^2 + c u v$$

To model radial behavior **[Eq. 6]**

Radial Distortion

$$\begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{M} \mathbf{P}_i \rightarrow \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} = \mathbf{p}_i \quad \mathbf{Q} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

\mathbf{Q}

Is this a linear system of equations?

$$\mathbf{p}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \frac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} \rightarrow \begin{cases} u_i \mathbf{q}_3 P_i = \mathbf{q}_1 P_i \\ v_i \mathbf{q}_3 P_i = \mathbf{q}_2 P_i \end{cases}$$

No!

[Eqs.7]

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \frac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} \xrightarrow{\text{red arrow}} X = f(Q) \quad [\text{Eq .8}]$$

$i=1 \dots n$

$f()$ is the nonlinear mapping

measurements parameters

-Newton Method

-Levenberg-Marquardt Algorithm

- Iterative, starts from initial solution
- May be slow if initial solution far from real solution
- Estimated solution may be function of the initial solution (because of local minima)
- Newton requires the computation of J, H
- Levenberg-Marquardt doesn't require the computation of H

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \frac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} \xrightarrow{\text{red arrow}} X = f(Q) \quad [\text{Eq .8}]$$

$i=1 \dots n$

$f(\)$ is the nonlinear mapping

measurements parameters

A possible algorithm

1. Solve linear part of the system to find approximated solution
2. Use this solution as initial condition for the full system
3. Solve full system using Newton or L.M.

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \frac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} \xrightarrow{\text{red arrow}} X = f(Q) \quad [\text{Eq .8}]$$

$i=1 \dots n$

measurements parameters

$f()$ is the nonlinear mapping

Typical assumptions:

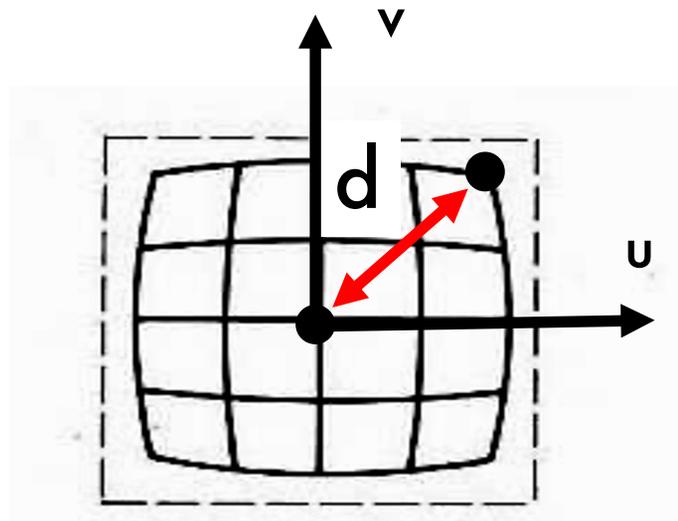
- zero-skew, square pixel
- $u_o, v_o =$ known center of the image

Radial Distortion

$$\mathbf{p}_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \frac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \frac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \\ \frac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \end{bmatrix}$$

Can we estimate m_1 and m_2 and ignore the radial distortion?

Hint:



$$\frac{u_i}{v_i} = \text{slope}$$

Radial Distortion Tsai [87]

Estimating m_1 and $m_2 \dots$

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_3 P_i}{m_2 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix} \rightarrow \frac{u_i}{v_i} = \frac{\frac{(m_1 P_i)}{(m_3 P_i)}}{\frac{(m_3 P_i)}{(m_2 P_i)}} = \frac{m_1 P_i}{m_2 P_i} \quad \text{[Eq .9]}$$

[Eq .10]

$$\begin{cases} v_1(m_1 P_1) - u_1(m_2 P_1) = 0 \\ v_i(m_1 P_i) - u_i(m_2 P_i) = 0 \\ \vdots \\ v_n(m_1 P_n) - u_n(m_2 P_n) = 0 \end{cases}$$

[Eq .11]

$$L \mathbf{n} = 0$$



Get m_1 and m_2 by SVD

$$L \stackrel{\text{def}}{=} \begin{pmatrix} v_1 P_1^T & -u_1 P_1^T \\ v_2 P_2^T & -u_2 P_2^T \\ \vdots & \vdots \\ v_n P_n^T & -u_n P_n^T \end{pmatrix}$$

$$\mathbf{n} = \begin{bmatrix} m_1^T \\ m_2^T \end{bmatrix}$$

Radial Distortion

Once that \mathbf{m}_1 and \mathbf{m}_2 are estimated...

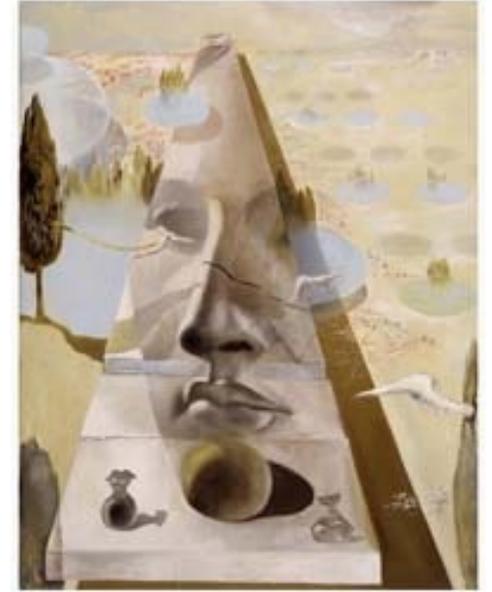
$$\mathbf{p}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \frac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \\ \frac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \\ \mathbf{m}_3 P_i \end{bmatrix}$$

\mathbf{m}_3 is non linear function of \mathbf{m}_1 , \mathbf{m}_2 , λ

There are some degenerate configurations for which \mathbf{m}_1 and \mathbf{m}_2 cannot be computed

Lecture 3

Camera Calibration



- Recap of projective cameras
- Camera calibration problem
- Camera calibration with radial distortion
- Example

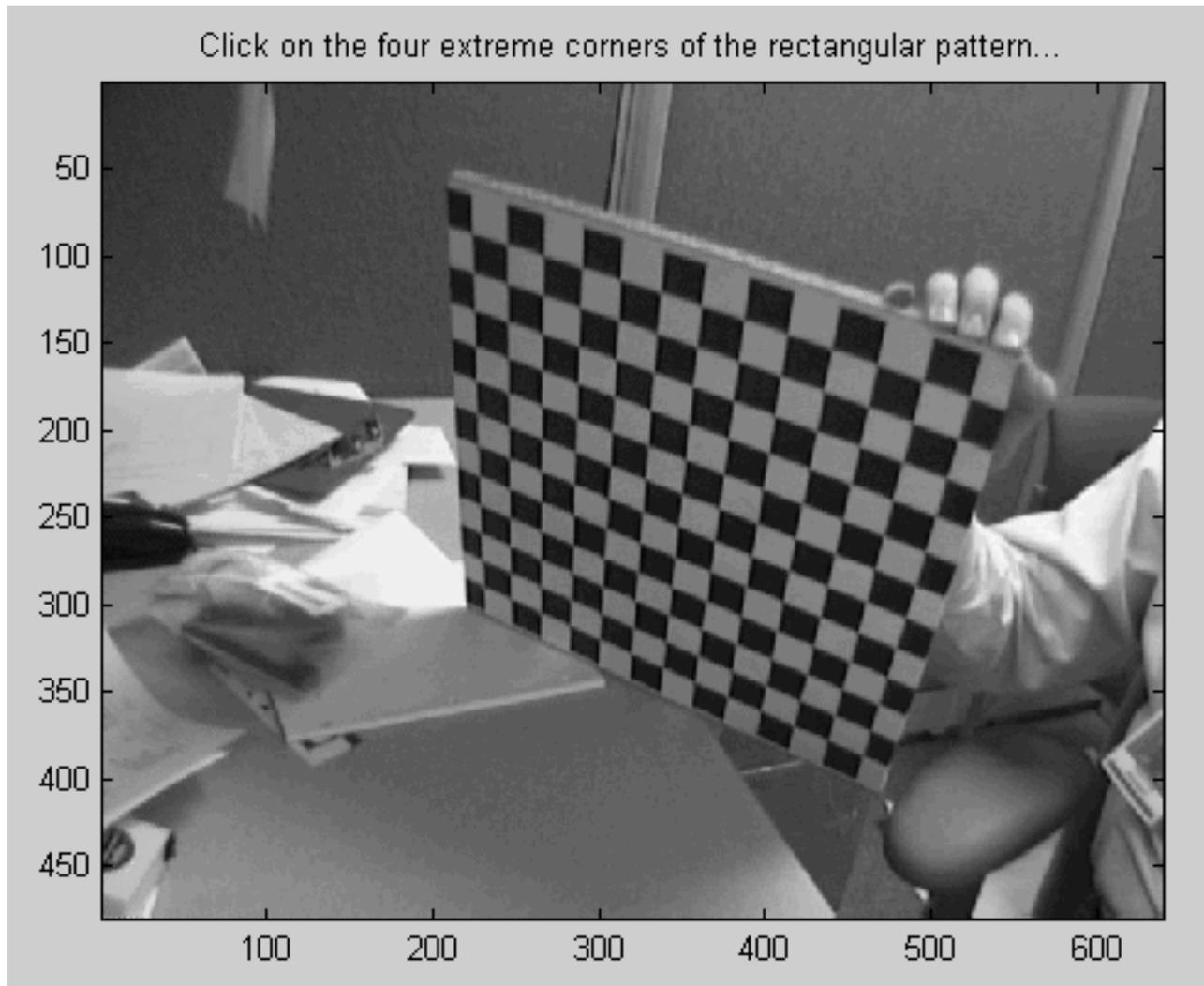
Reading: **[FP]** Chapter 1 "Geometric Camera Calibration"
 [HZ] Chapter 7 "Computation of Camera Matrix P "

Some slides in this lecture are courtesy to Profs. J. Ponce, F-F Li

Calibration Procedure

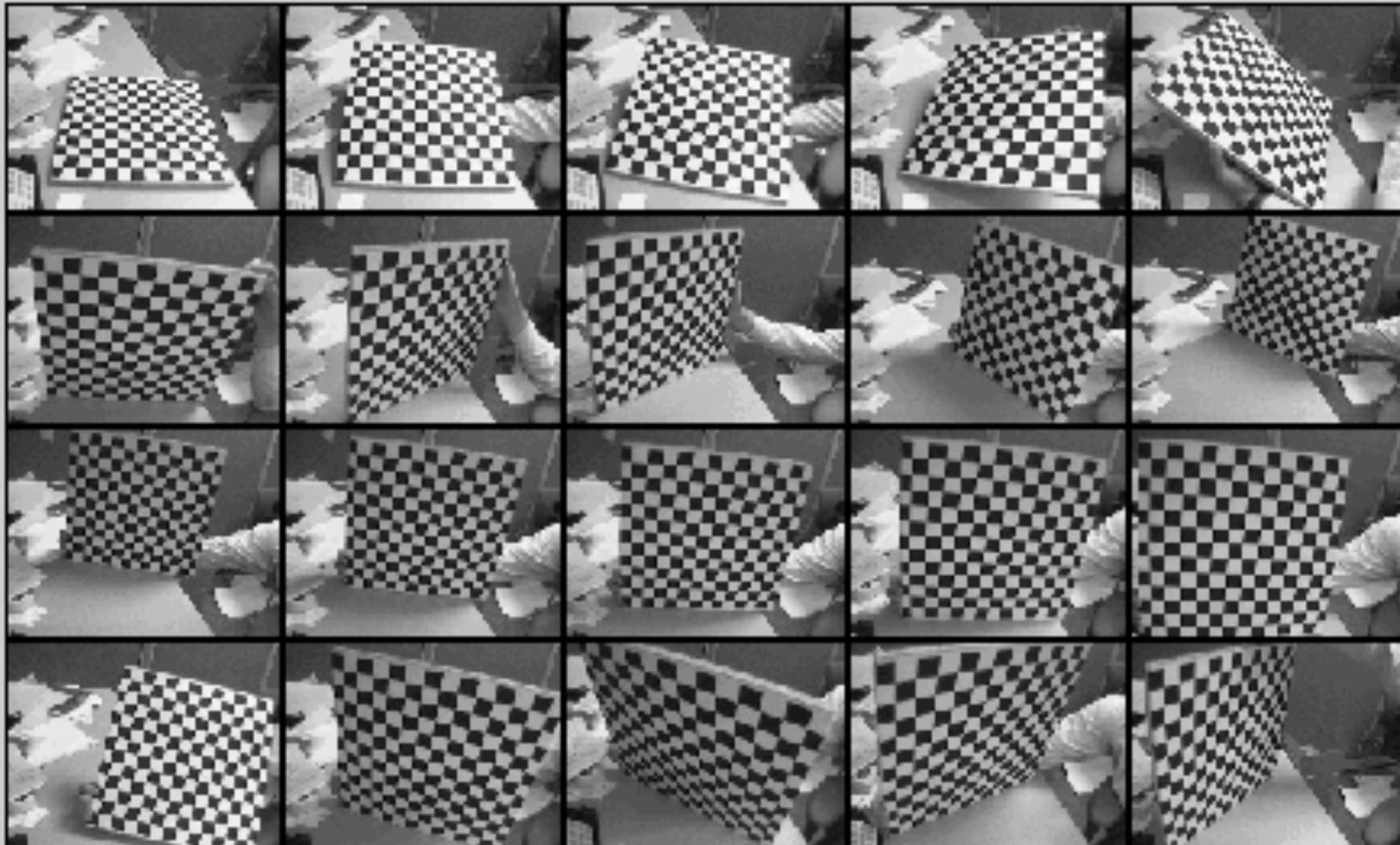
Camera Calibration Toolbox for OpenCV
J. Bouquet – [1998-2000]

http://www.vision.caltech.edu/bouquetj/calib_doc/index.html#examples



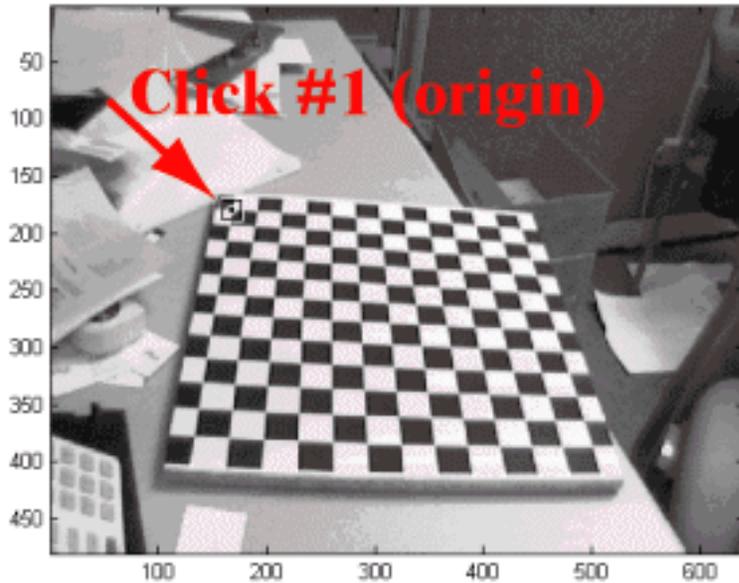
Calibration Procedure

Calibration images

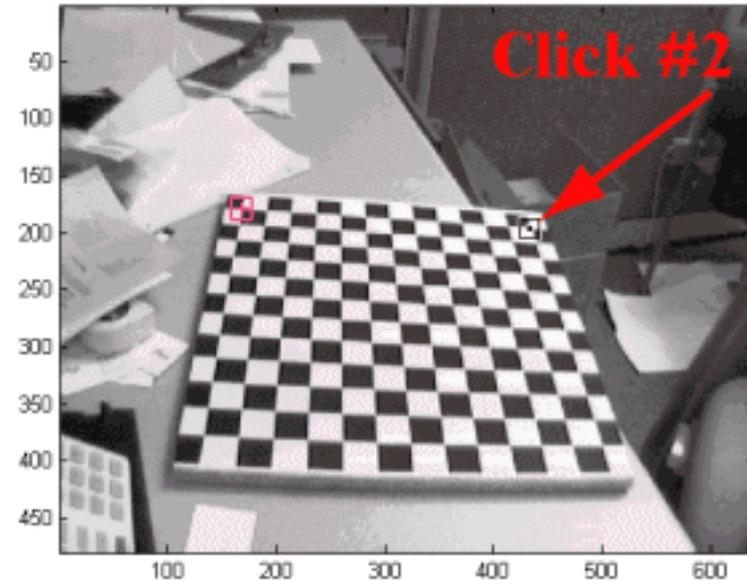


Calibration Procedure

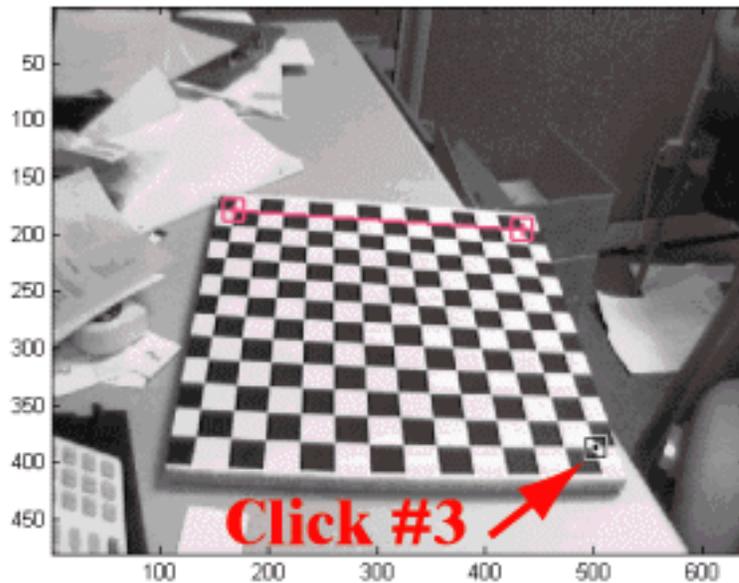
Click on the four extreme corners of the rectangular pattern (first corner = origin)... Image 1



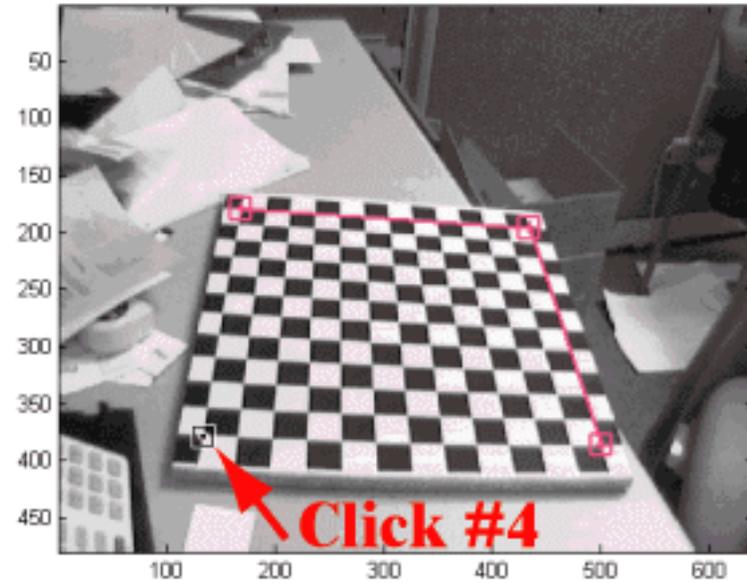
Click on the four extreme corners of the rectangular pattern (first corner = origin)... Image 1



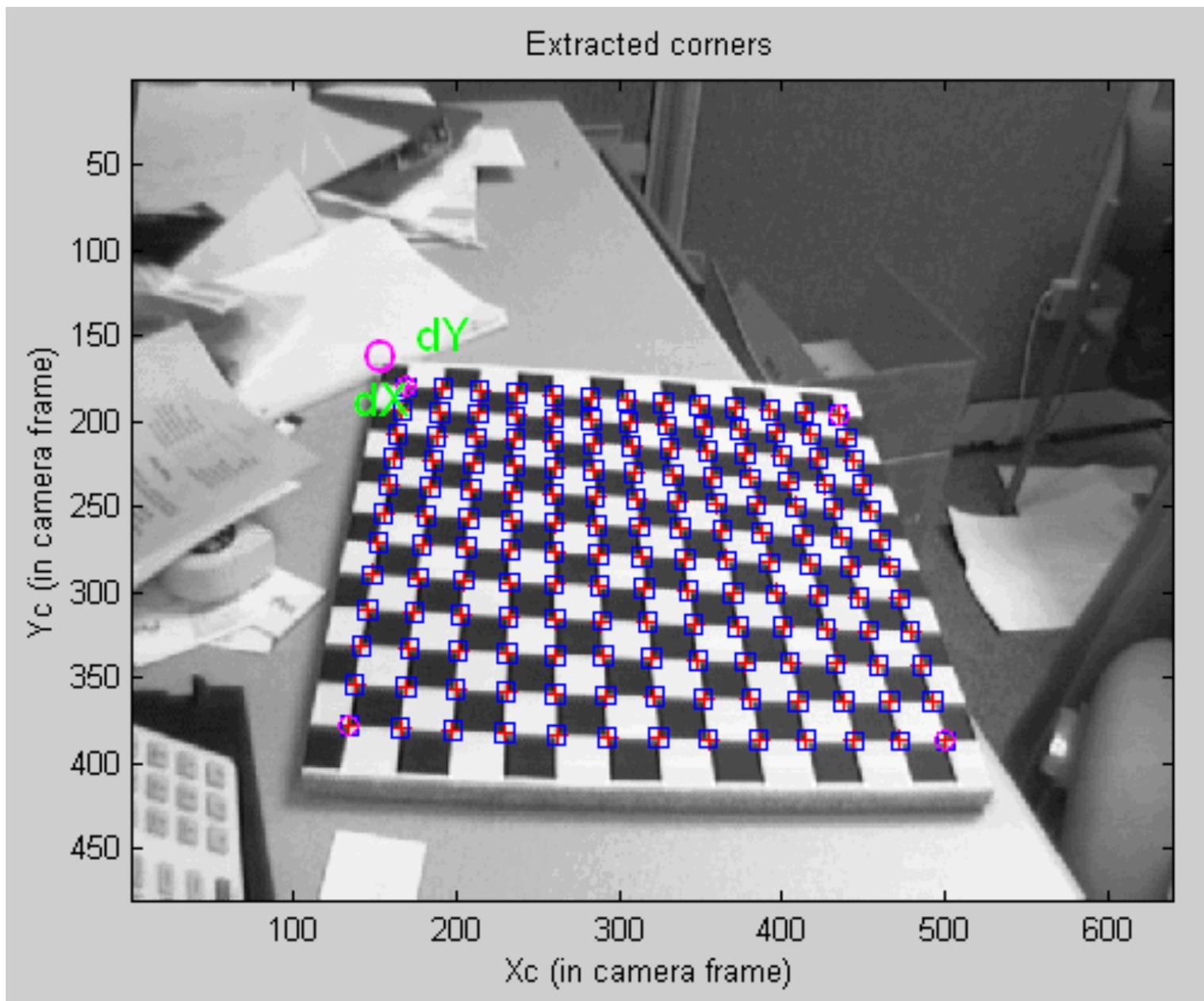
Click on the four extreme corners of the rectangular pattern (first corner = origin)... Image 1



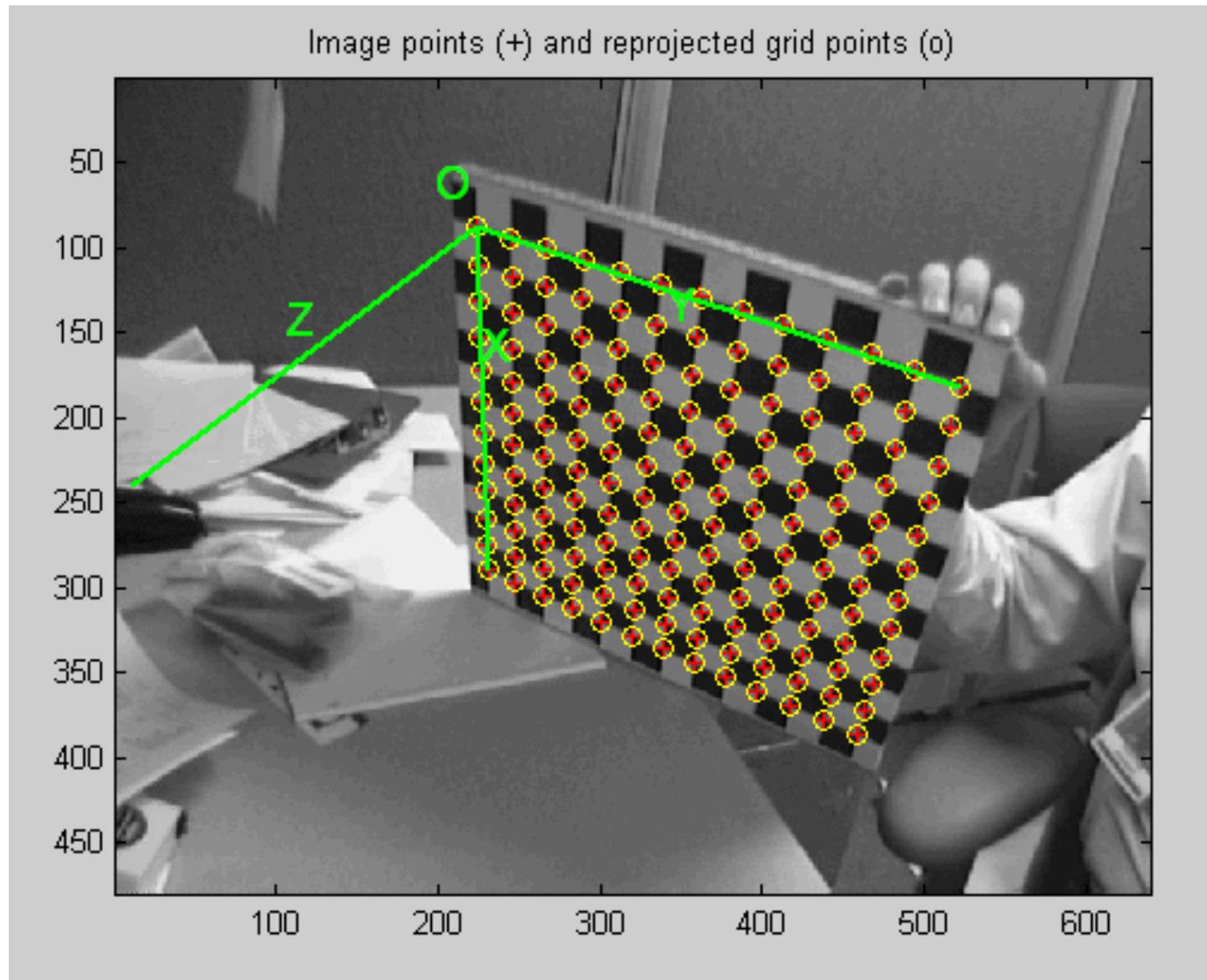
Click on the four extreme corners of the rectangular pattern (first corner = origin)... Image 1



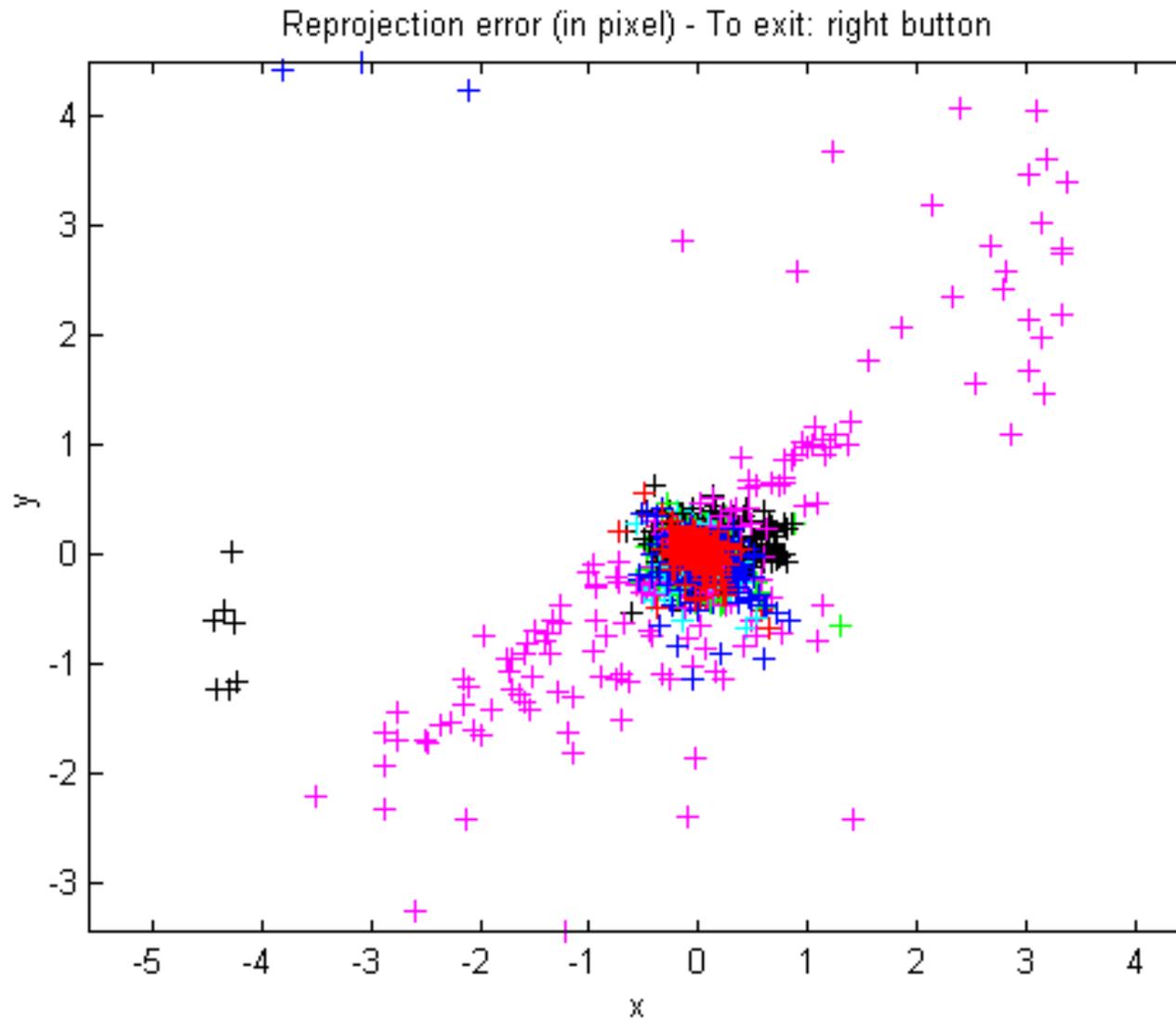
Calibration Procedure



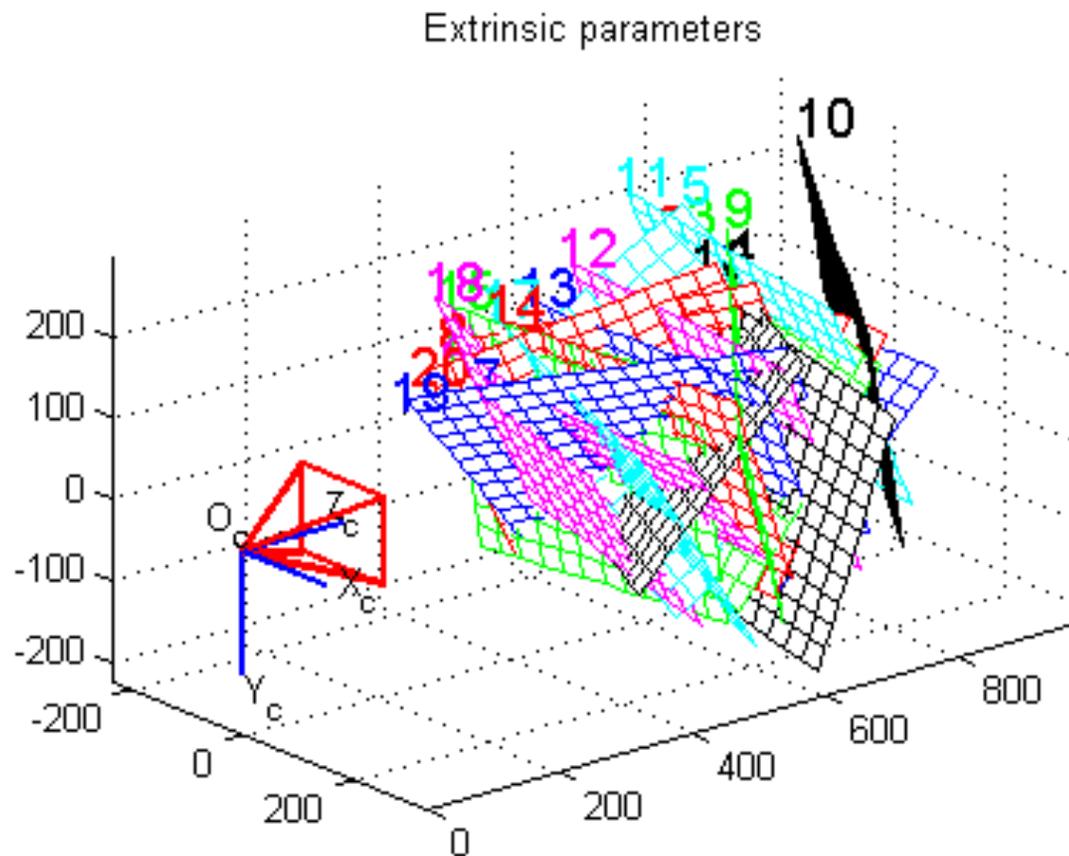
Calibration Procedure



Calibration Procedure

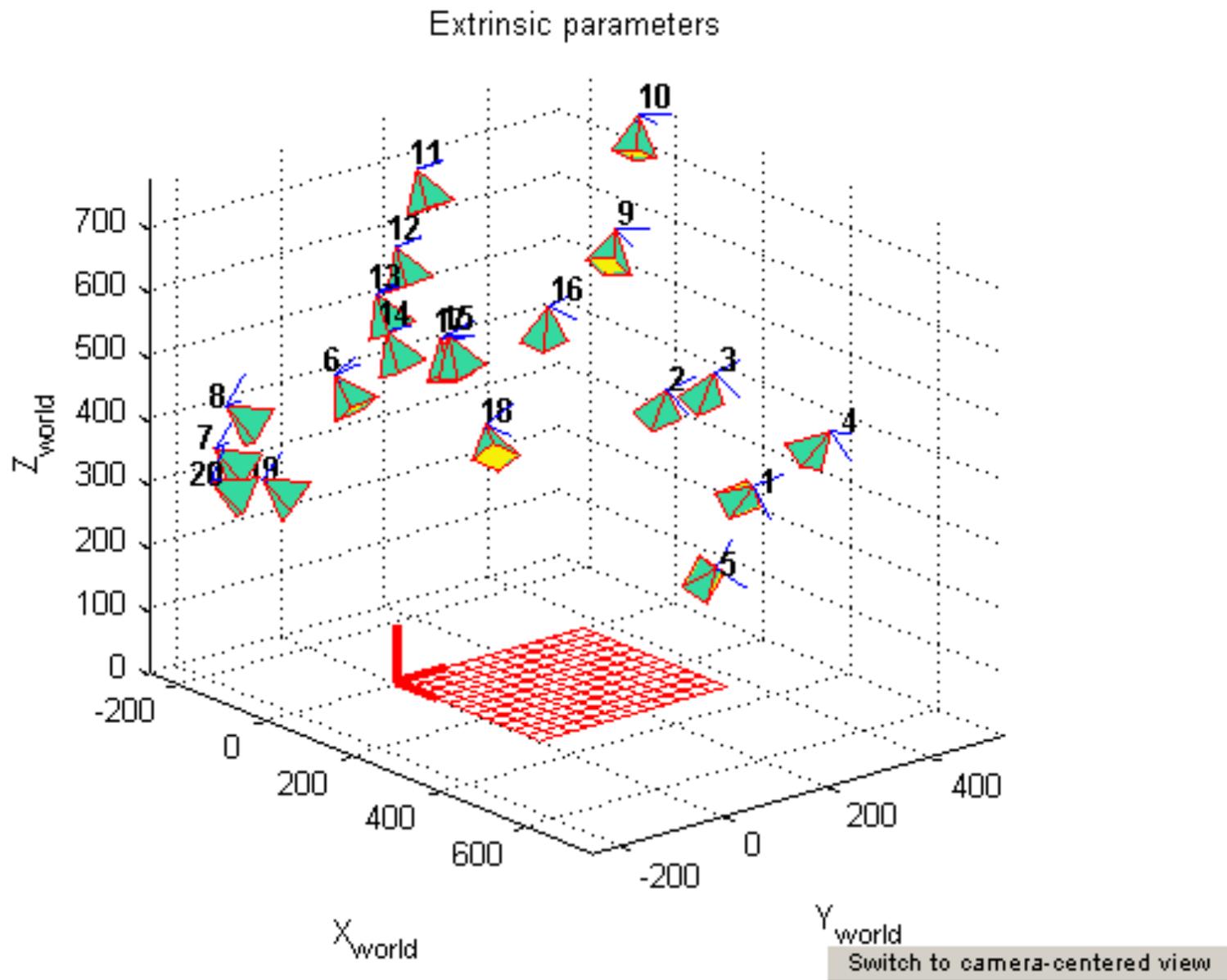


Calibration Procedure



Switch to world-centered view

Calibration Procedure



Next lecture

- **Single view reconstruction**