Lecture 3
Camera Models 2 & Camera Calibration

Professor Silvio Savarese

Computational Vision and Geometry Lab

Lecture 3

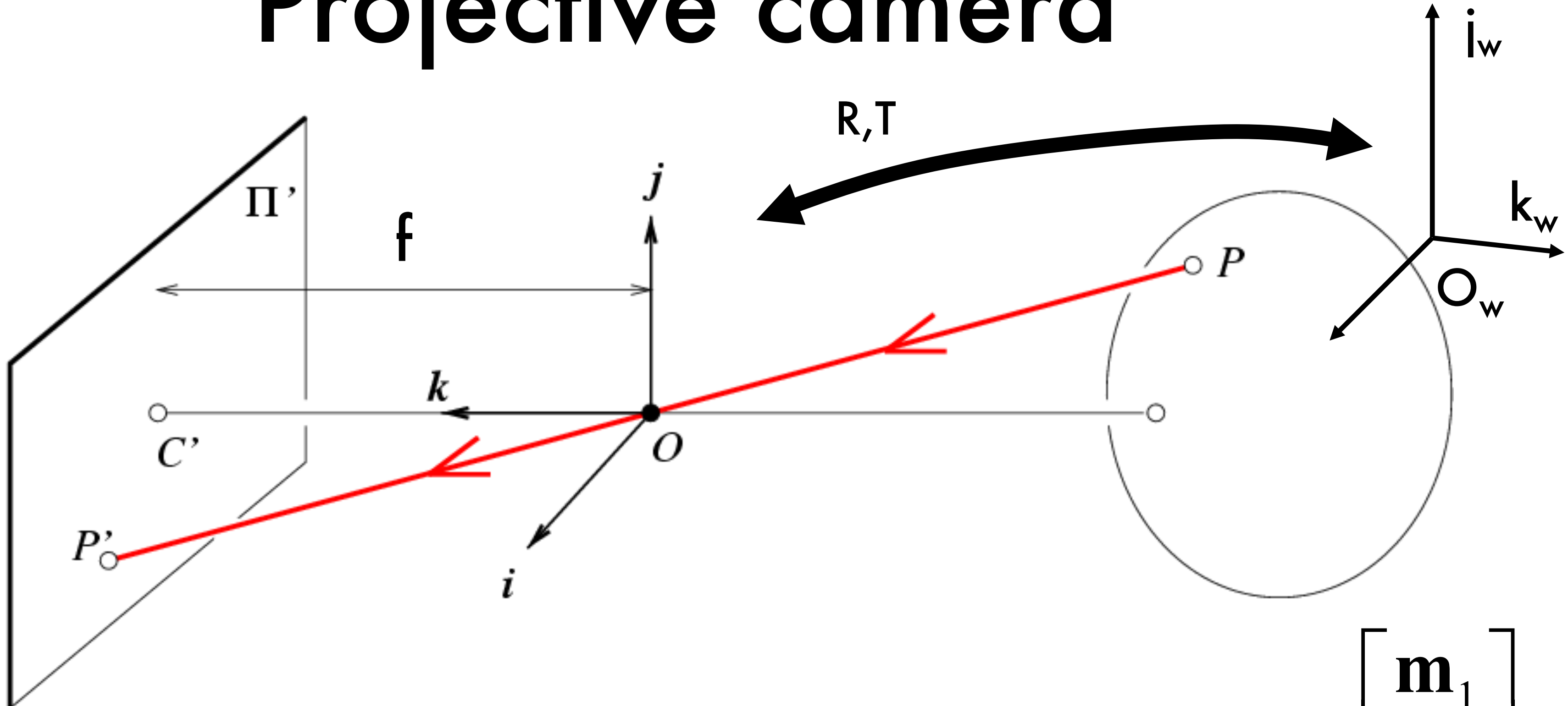
Camera Models 2 & Camera Calibration

- 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

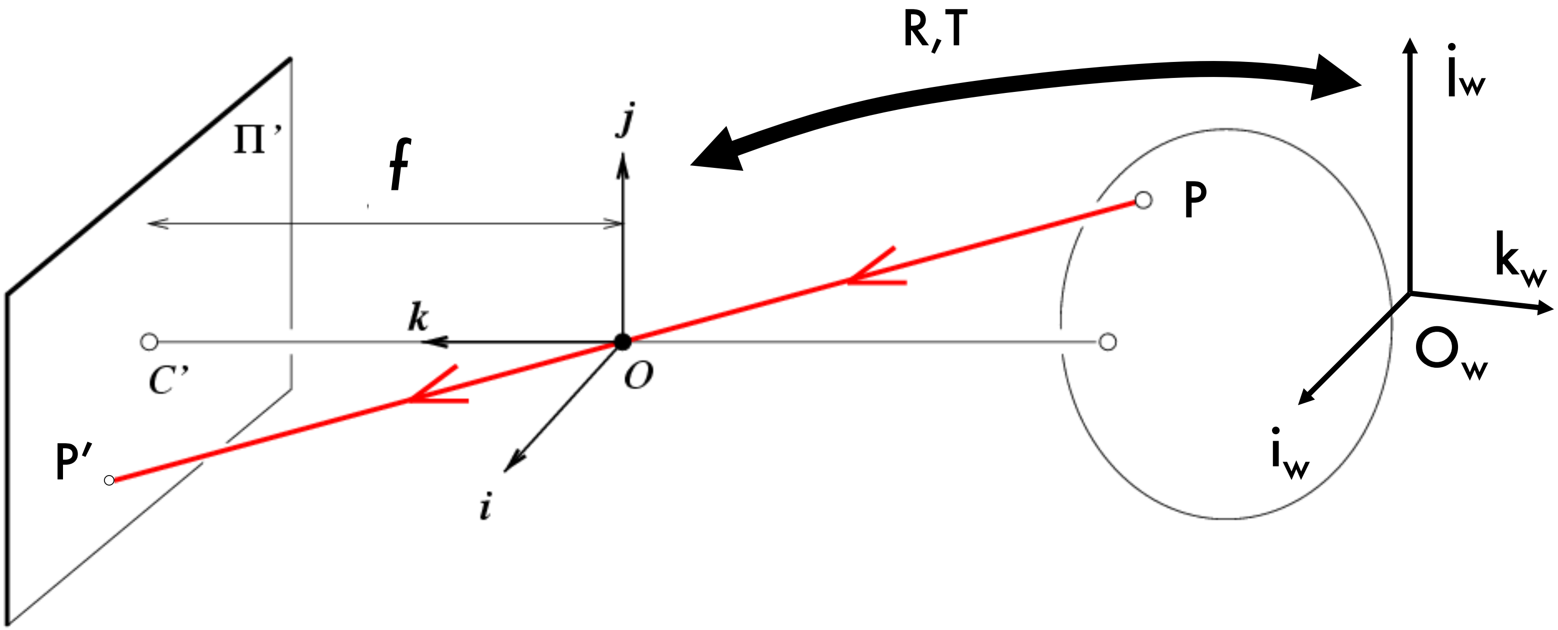
Projective camera



$$P'_{3\times1} = M\ P_w \quad = K_{3\times3}\begin{bmatrix} R & T \end{bmatrix}_{3\times4} P_{w\,4\times1} \qquad M = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_{\mathrm{W}} = \begin{bmatrix} \mathbf{m}_1 P_{\mathrm{W}} \\ \mathbf{m}_2 P_{\mathrm{W}} \\ \mathbf{m}_3 P_{\mathrm{W}} \end{bmatrix} \xrightarrow{\mathbf{E}} P'_E = \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w}\right)$$

Exercise!



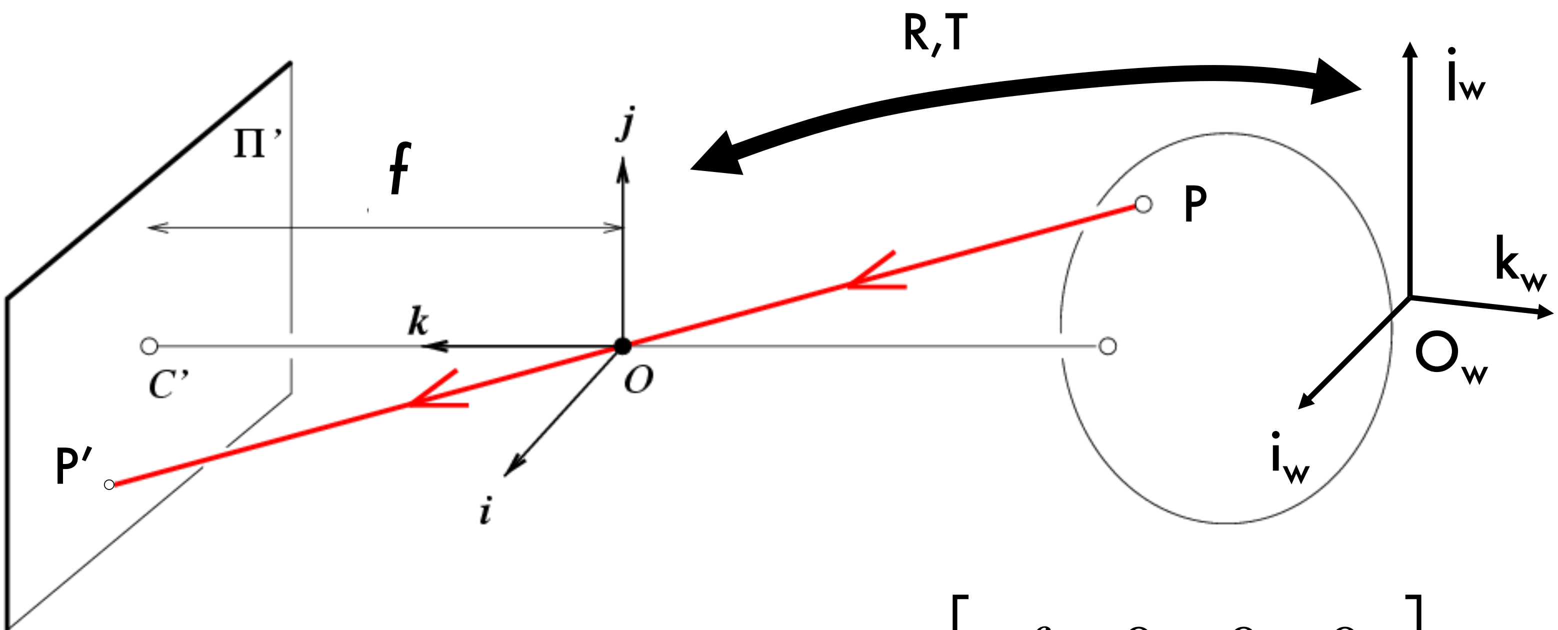
$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

Suppose we have no rotation or translation
Zero skew, square pixels, no distortion, no off-set

$$\rightarrow \; P'_E = \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w}\right)$$

$$P_w = \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

Exercise!



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

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

$$P_w = \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

Canonical Projective Transformation

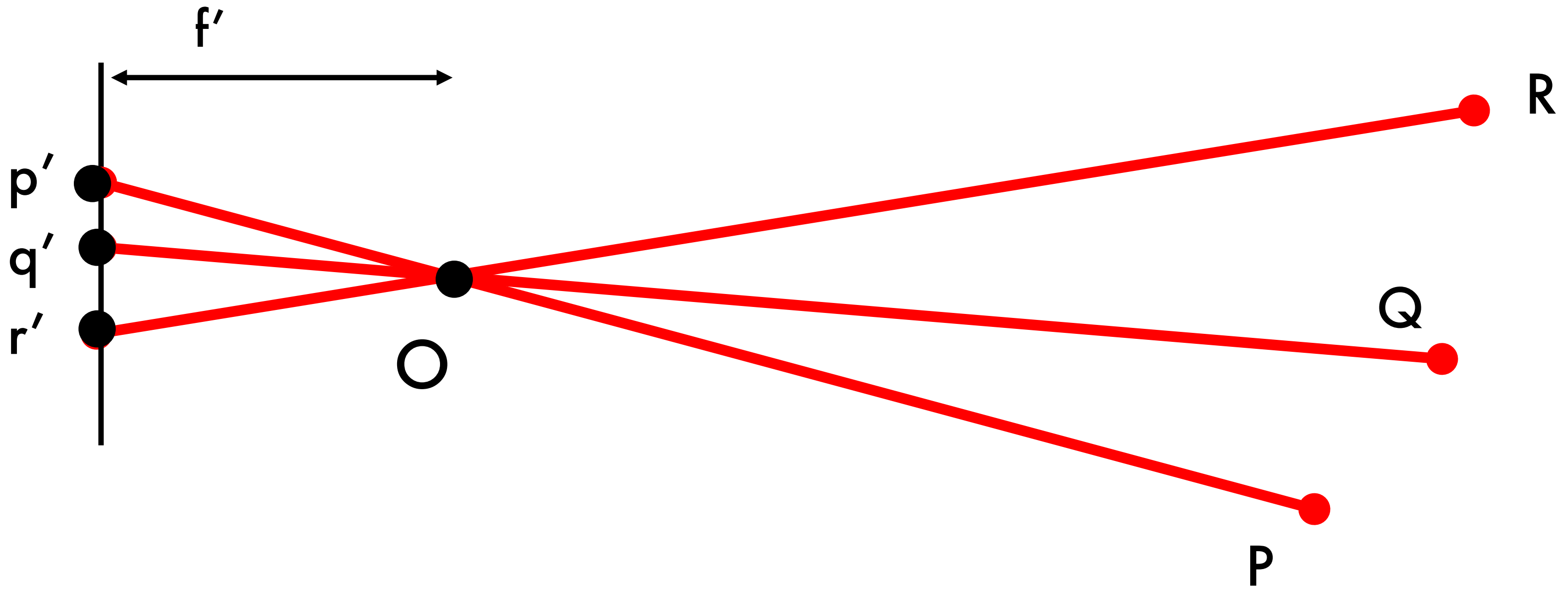
$$P' = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\text{M}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$P' = M\,P$$

$$\Re^4 \xrightarrow{\text{H}} \Re^3$$

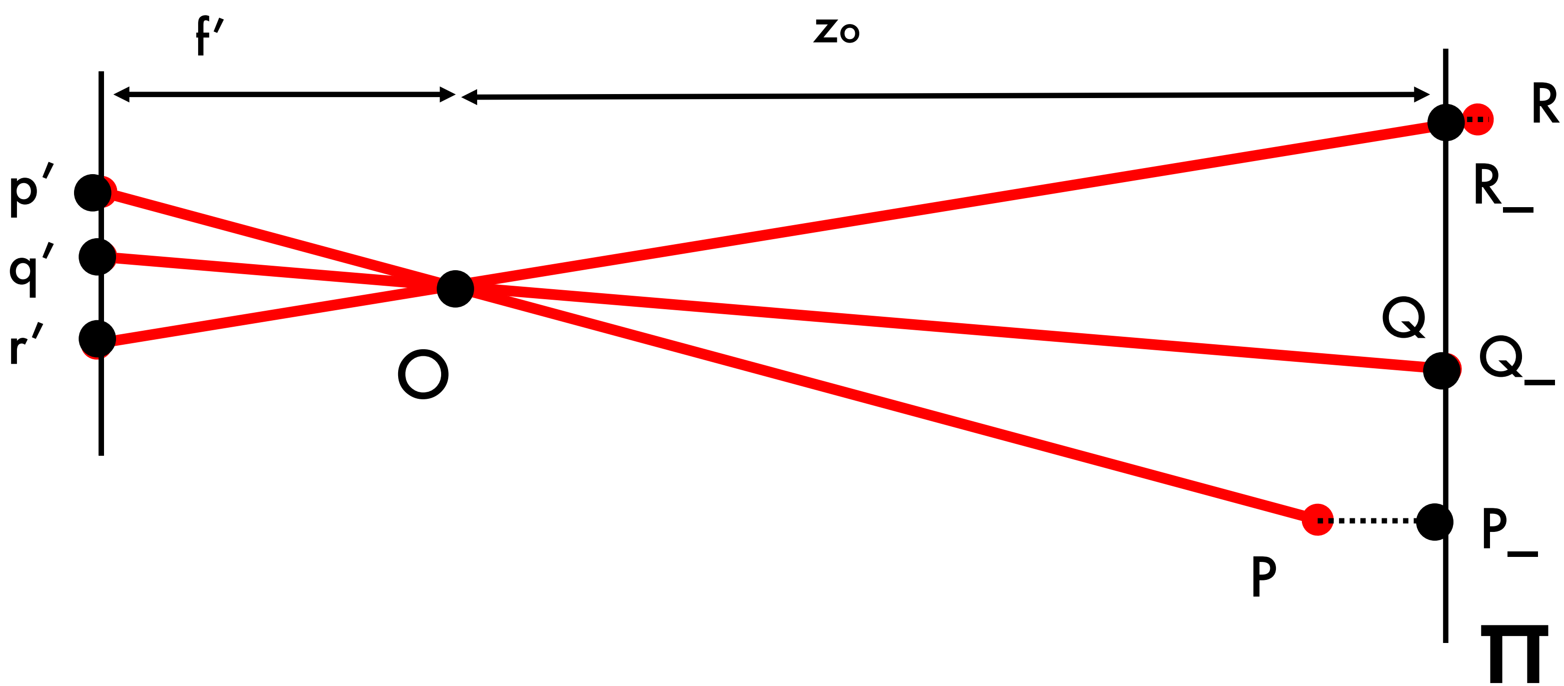
$$P_i' = \begin{bmatrix} \frac{x}{z} \\ \frac{y}{z} \end{bmatrix}$$

Projective camera

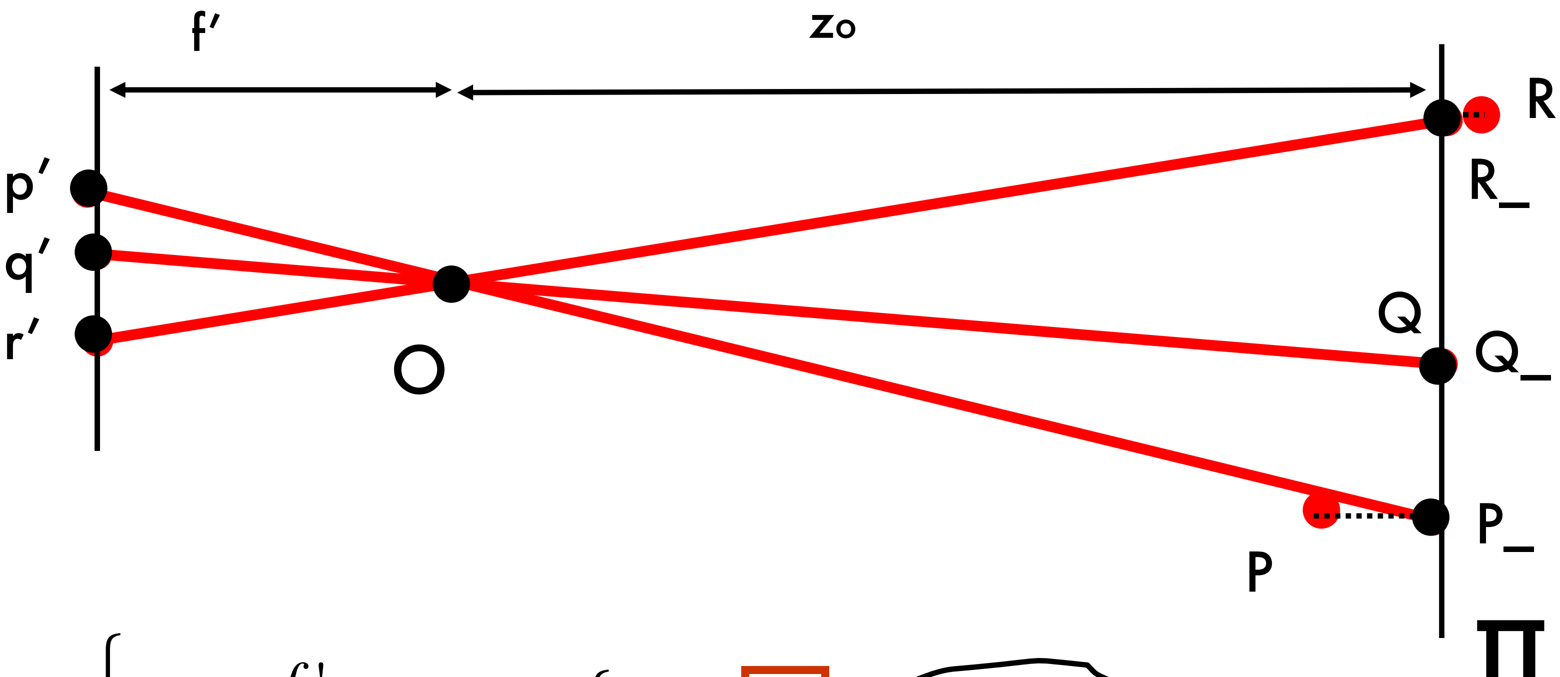


Weak perspective projection

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



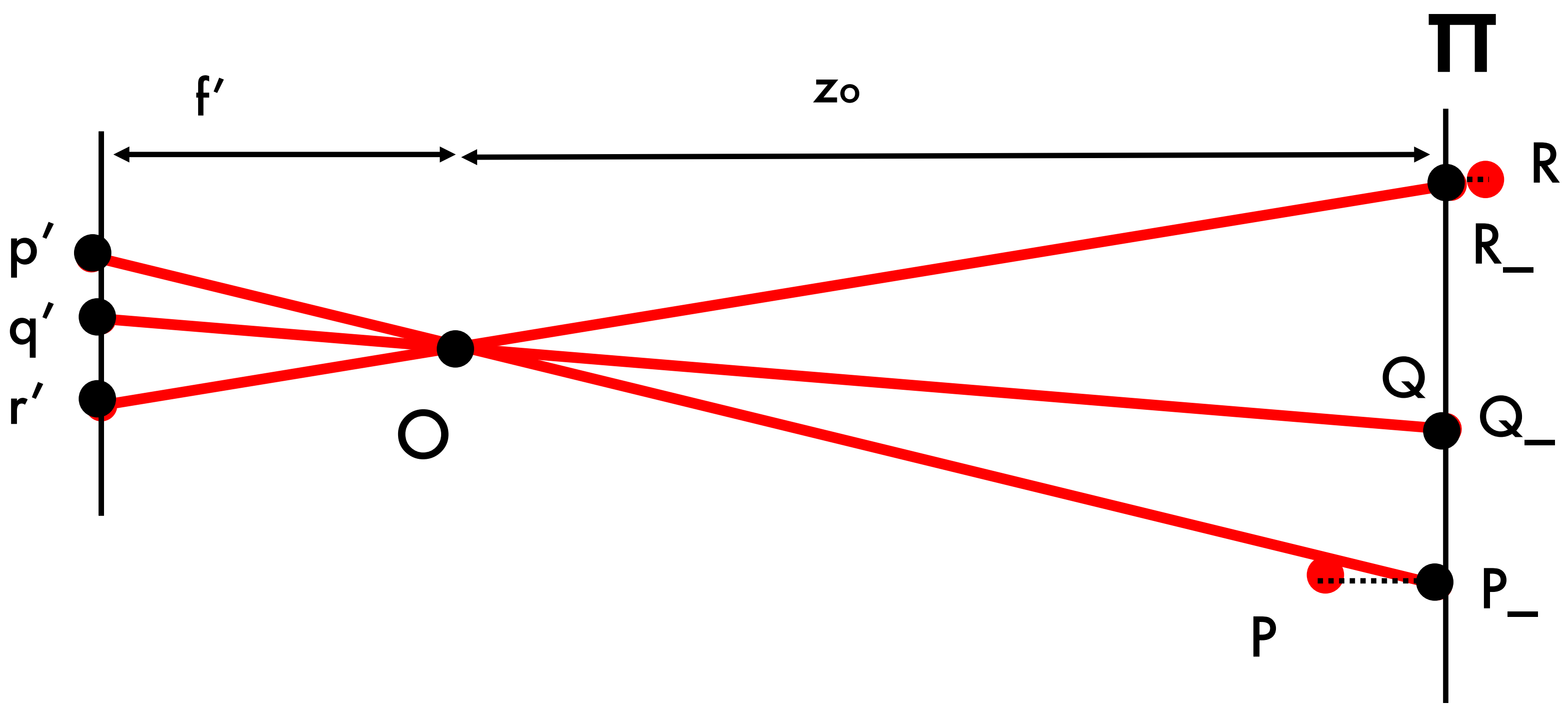
Weak perspective projection



$$\begin{cases} x' = \dfrac{f'}{z}x \\ y' = \dfrac{f'}{z}y \end{cases} \rightarrow \begin{cases} x' = \boxed{\dfrac{f'}{z_0}}x \\ y' = \dfrac{f'}{z_0}y \end{cases}$$

Magnification m

Weak perspective projection



Projective (perspective) Weak perspective

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

$$P' = M\,P_w = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_W = \begin{bmatrix} \mathbf{m}_1 P_W \\ \mathbf{m}_2 P_W \\ \mathbf{m}_3 P_W \end{bmatrix} \qquad M = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{v} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

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

Perspective

$$P' = M\,P_w = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_W = \begin{bmatrix} \mathbf{m}_1 P_W \\ \mathbf{m}_2 P_W \\ 1 \end{bmatrix} \qquad M = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} = \begin{bmatrix} & \mathbf{m}_1 & & \\ & \mathbf{m}_2 & & \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

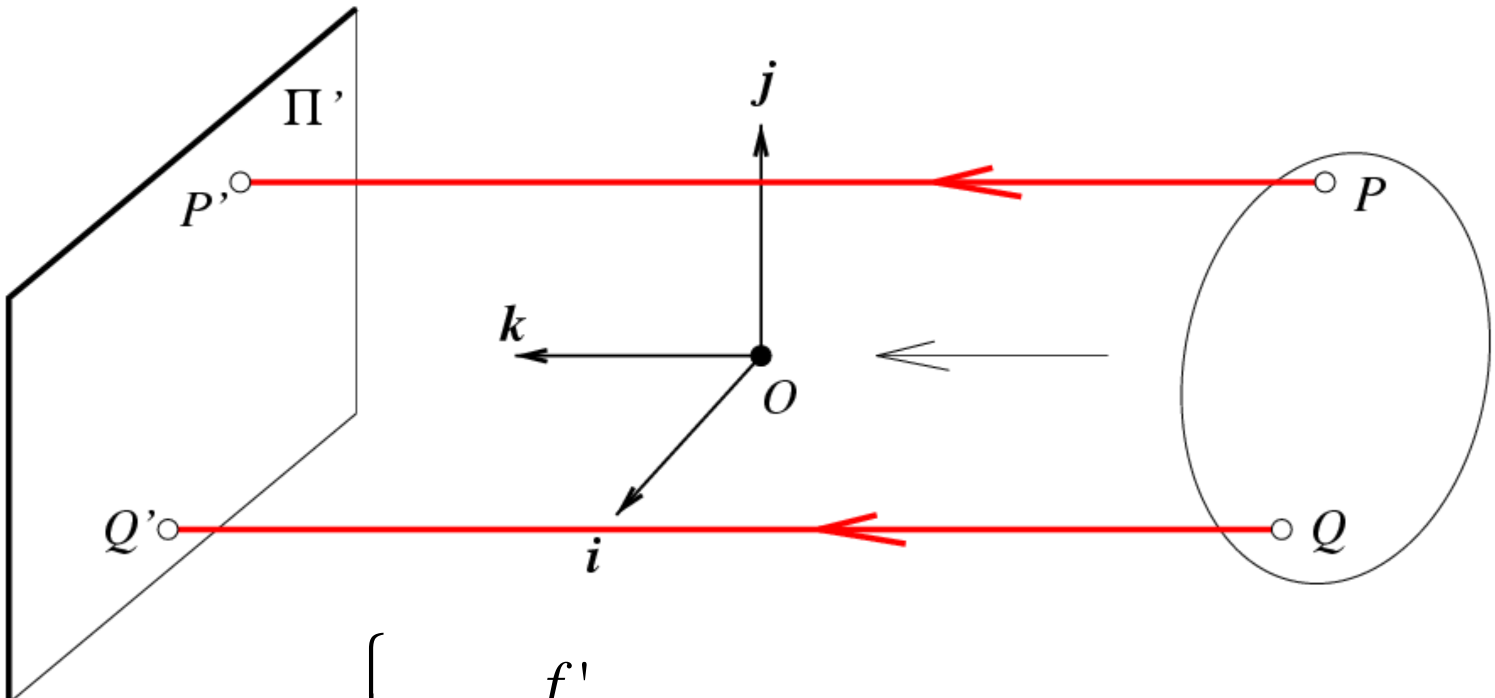
$$\xrightarrow{\mathbf{E}} (\mathbf{m}_1 P_w, \mathbf{m}_2 P_w)$$

magnification

Weak perspective

Orthographic (affine) projection

Distance from center of projection to image plane is infinite



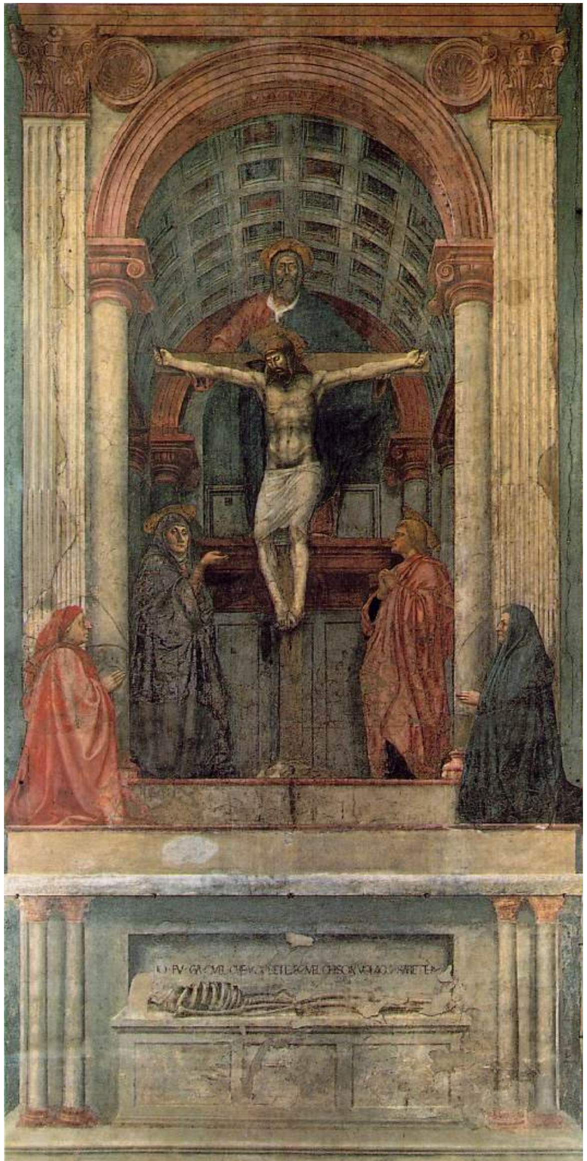
$$\begin{cases} x' = \dfrac{f'}{z} x \\[2em] y' = \dfrac{f'}{z} y \end{cases} \rightarrow \begin{cases} x' = x \\ y' = y \end{cases}$$

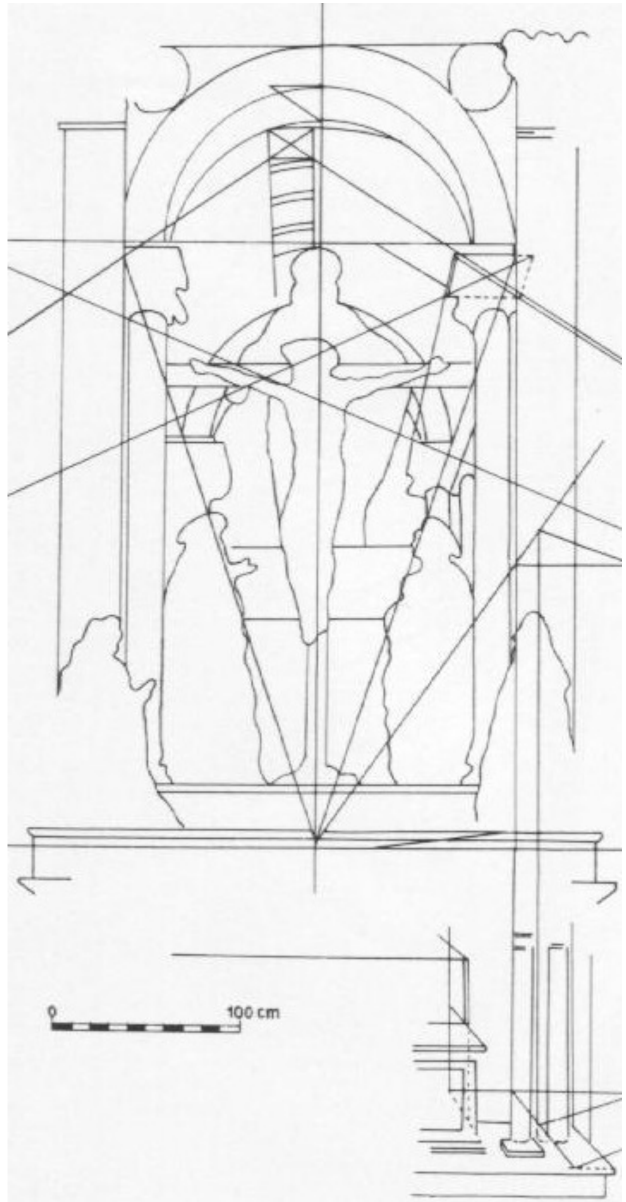
Pros and Cons of These Models

- 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

Credit slide S. Lazebnik

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

Why is this important?

Estimate camera parameters such pose or focal length from images!

Projective camera

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

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

$$K = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_o \\ 0 & \dfrac{\beta}{\sin\theta} & v_o \\ 0 & 0 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} \qquad T = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

Goal of calibration

$$P' = M\,P_w = \boxed{K}\begin{bmatrix} \boxed{R \quad T} \end{bmatrix} P_w$$

Internal parameters

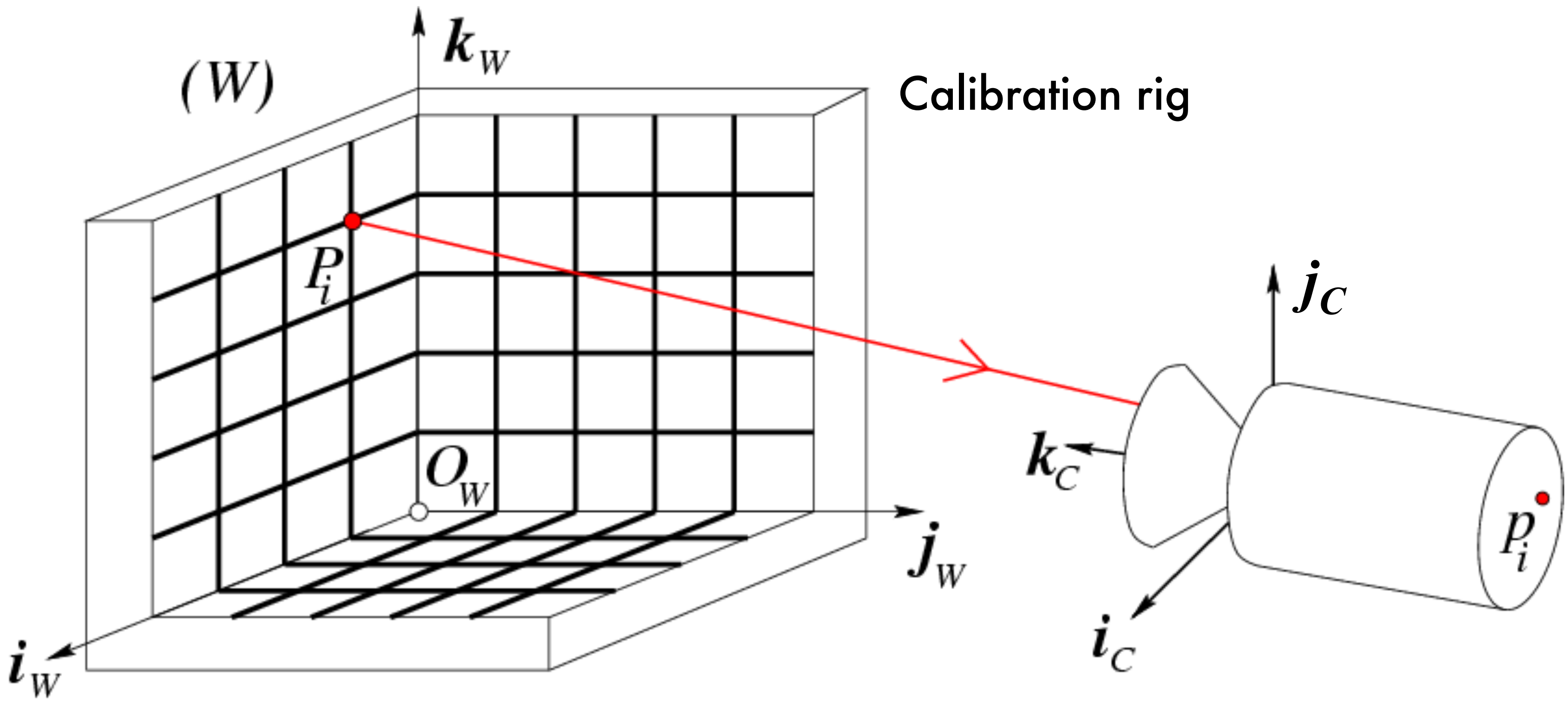
External parameters

Estimate intrinsic and extrinsic parameters from 1 or multiple images

Change notation:
$P = P_w$

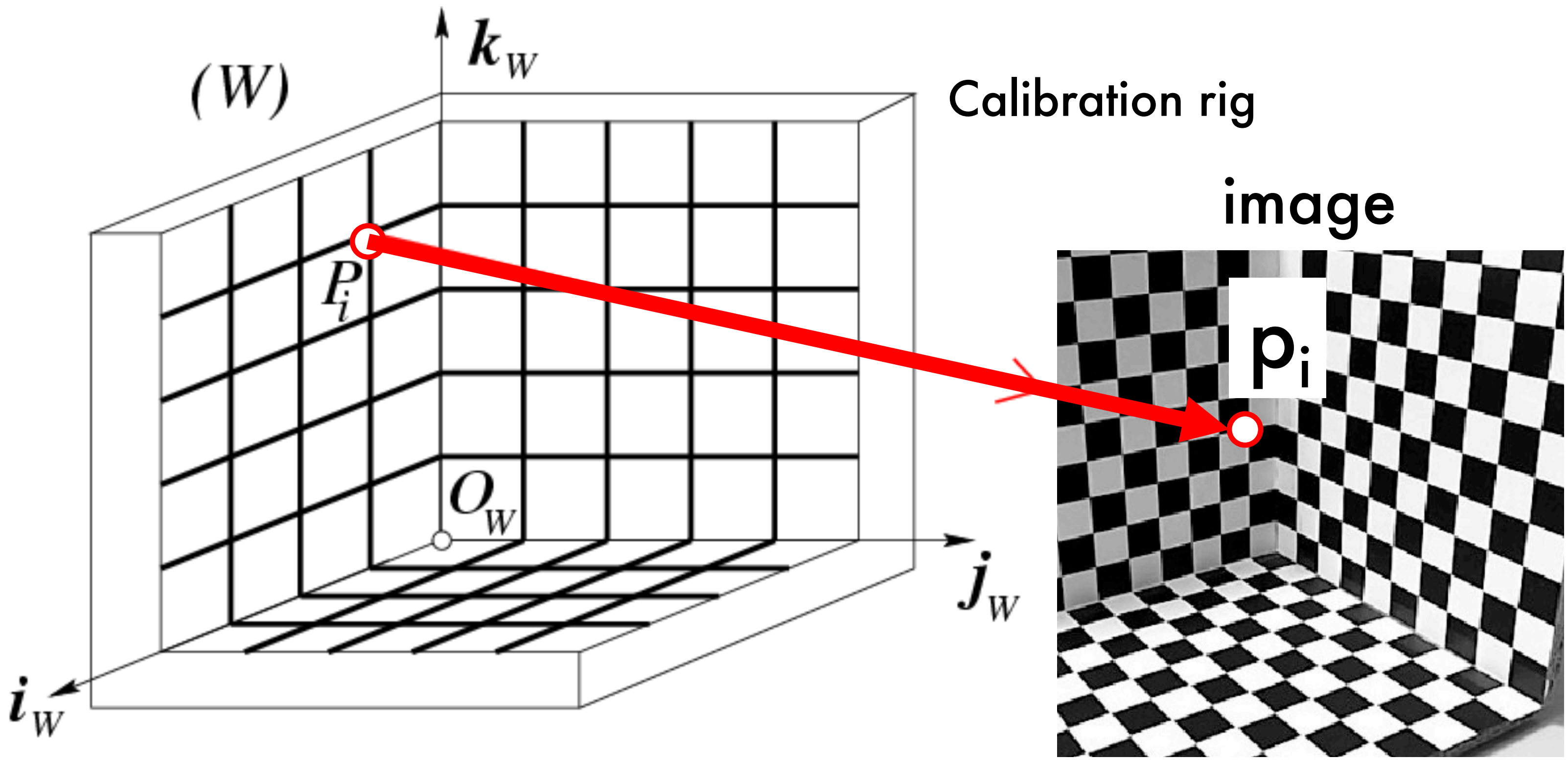
$p = P'$

Calibration Problem



- $P_1 \ldots P_n$ with known positions in $[O_w, i_w, j_w, k_w]$

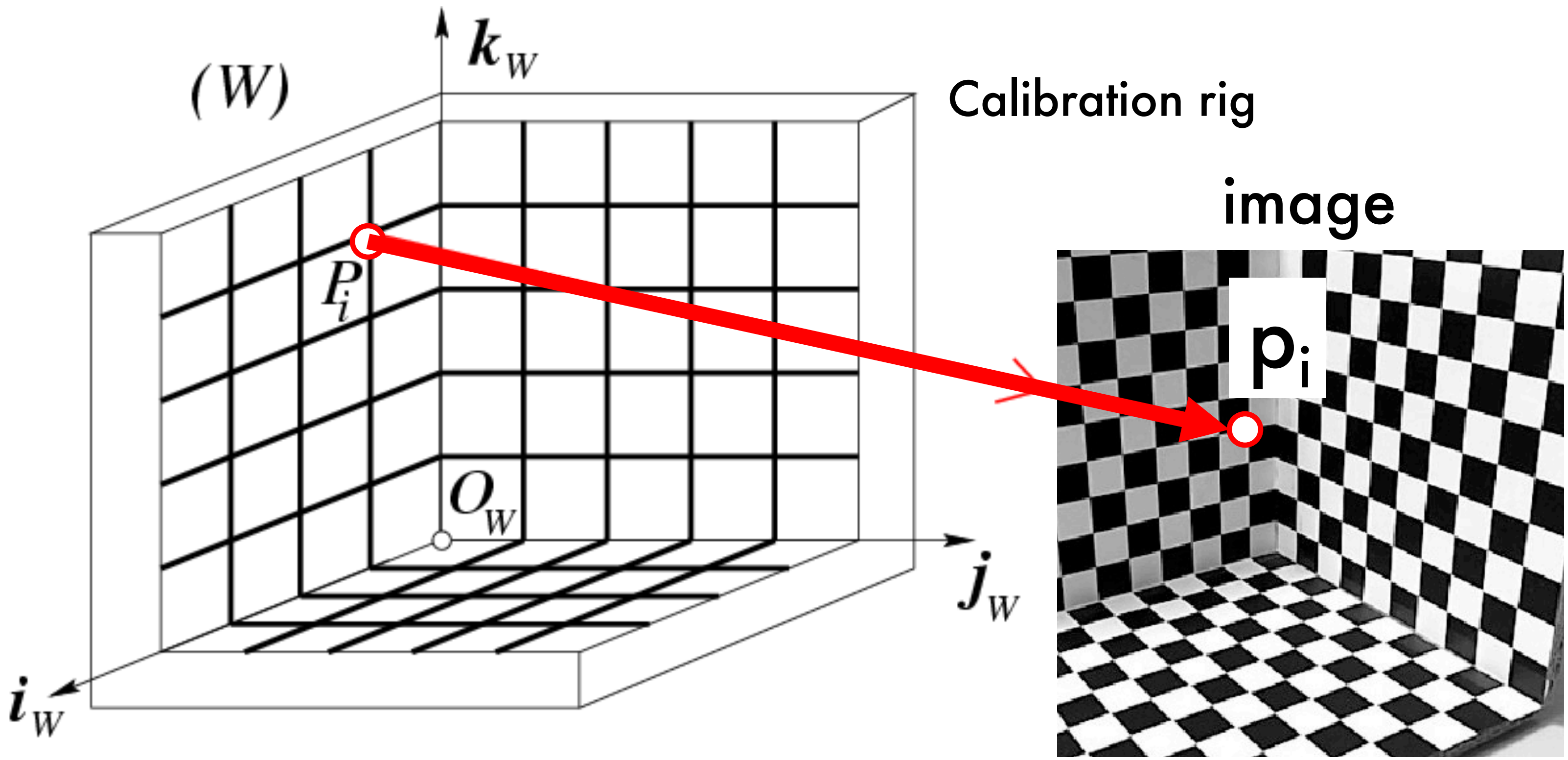
Calibration Problem



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

Goal: compute intrinsic and extrinsic parameters

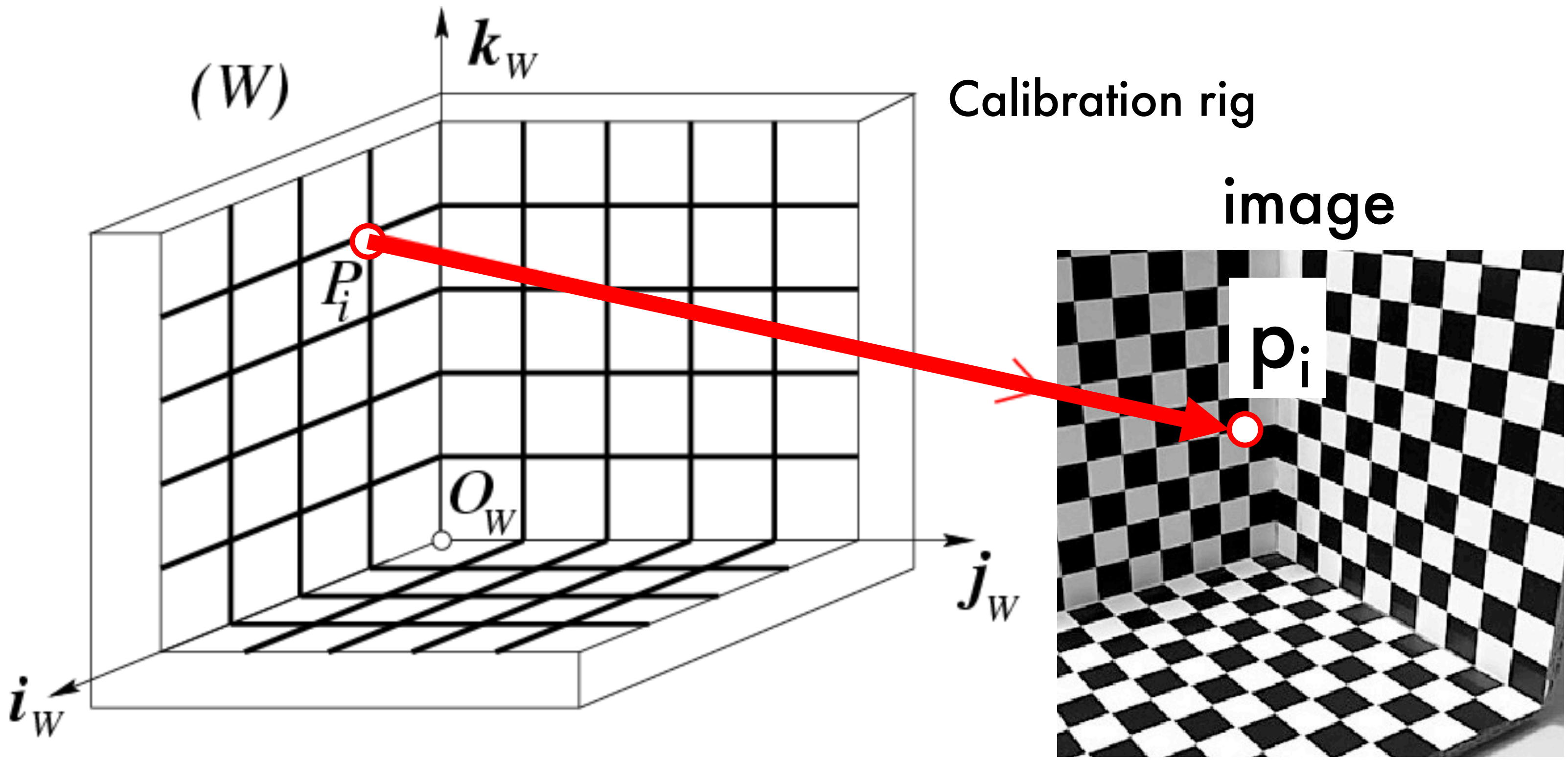
Calibration Problem



How many correspondences do we need?

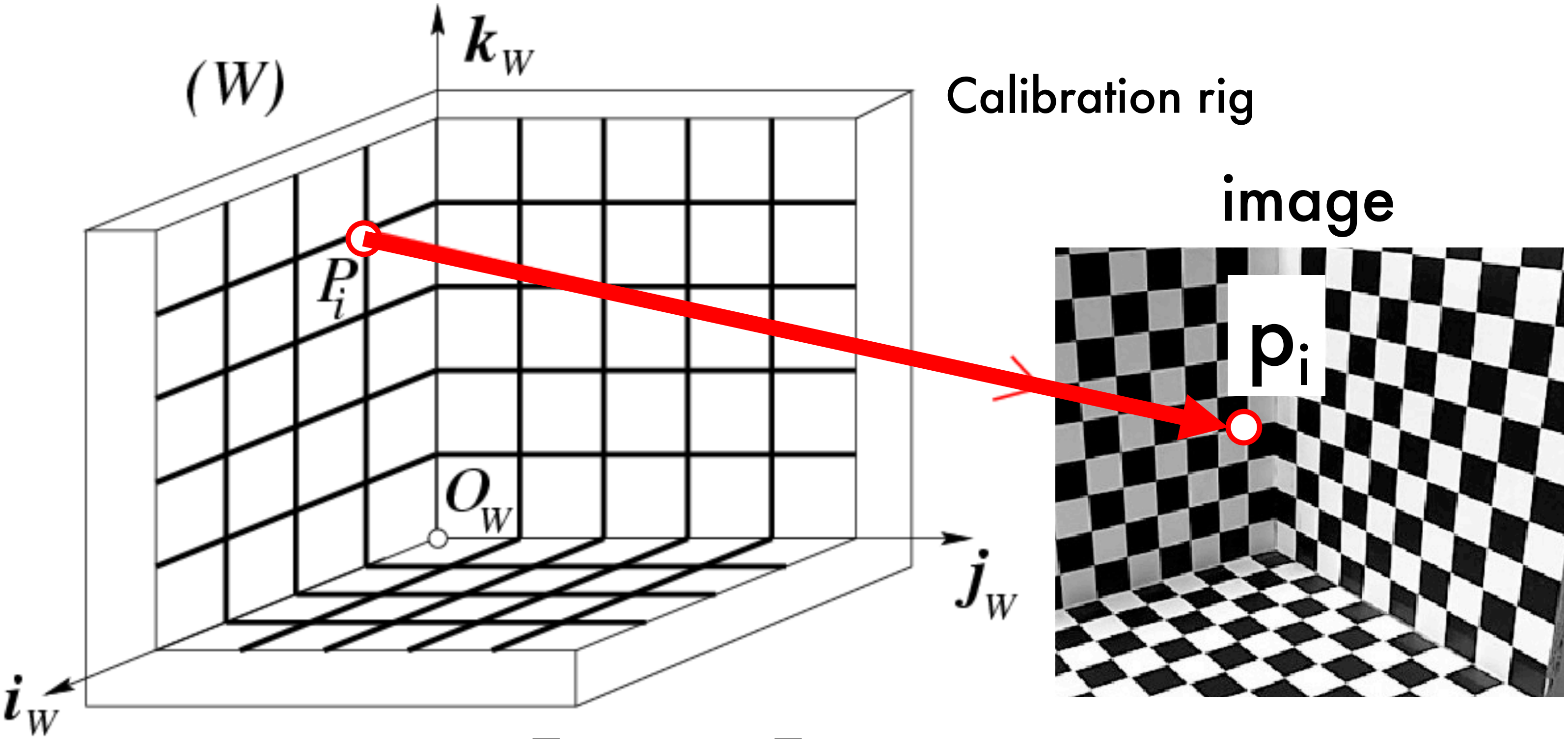
- M has 11 unknowns
- We need 11 equations
- 6 correspondences would do it

Calibration Problem



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

Calibration Problem



$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{m}_1 \, \mathrm{P_i}}{\mathbf{m}_3 \, \mathrm{P_i}} \\ \dfrac{\mathbf{m}_2 \, \mathrm{P_i}}{\mathbf{m}_3 \, \mathrm{P_i}} \end{bmatrix} = M \; P_i \qquad \mathrm{M} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

in pixels

[Eq. 1]

Calibration Problem

[Eq. 1]

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \\ \dfrac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \end{bmatrix}$$

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

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

[Eqs. 2]

Calibration Problem

$$
\begin{cases}
u_1(\mathbf{m}_3\, P_1) - \mathbf{m}_1\, P_1 = 0 \\
v_1(\mathbf{m}_3\, P_1) - \mathbf{m}_2\, P_1 = 0 \\
\vdots \\
u_i(\mathbf{m}_3\, P_i) - \mathbf{m}_1\, P_i = 0 \\
v_i(\mathbf{m}_3\, P_i) - \mathbf{m}_2\, P_i = 0 \\
\vdots \\
u_n(\mathbf{m}_3\, P_n) - \mathbf{m}_1\, P_n = 0 \\
v_n(\mathbf{m}_3\, P_n) - \mathbf{m}_2\, P_n = 0
\end{cases}
$$

[Eqs. 3]

Block Matrix Multiplication

$$\mathrm{A} = \begin{bmatrix} \mathrm{A}_{11} & \mathrm{A}_{12} \\ \mathrm{A}_{21} & \mathrm{A}_{22} \end{bmatrix} \qquad \mathrm{B} = \begin{bmatrix} \mathrm{B}_{11} & \mathrm{B}_{12} \\ \mathrm{B}_{21} & \mathrm{B}_{22} \end{bmatrix}$$

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

$$\begin{cases} -u_1(\mathbf{m}_3\, P_1) + \mathbf{m}_1\, P_1 = 0 \\ -v_1(\mathbf{m}_3\, P_1) + \mathbf{m}_2\, P_1 = 0 \\ \vdots \\ -u_n(\mathbf{m}_3\, P_n) + \mathbf{m}_1\, P_n = 0 \\ -v_n(\mathbf{m}_3\, P_n) + \mathbf{m}_2\, P_n = 0 \end{cases} \longrightarrow \boxed{\underbrace{\mathbf{P}}_{\text{known}}\, \underbrace{\mathbf{m}}_{\text{unknown}} = 0}$$

[Eq. 4]

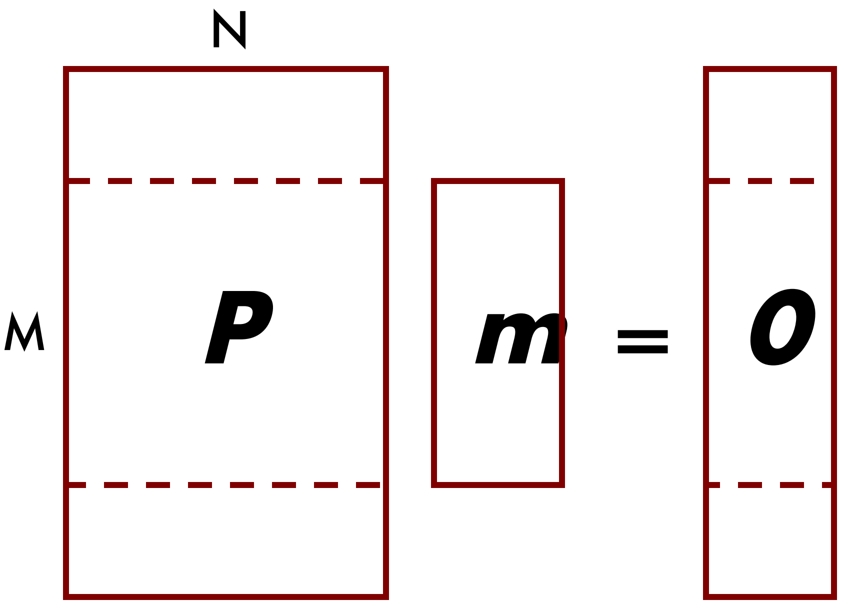
Homogenous linear system

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

$$\boldsymbol{m} \overset{\text{def}}{=} \begin{pmatrix} \overset{4\times 1}{\mathbf{m}_1^{\mathrm{T}}} \\ \mathbf{m}_2^{\mathrm{T}} \\ \mathbf{m}_3^{\mathrm{T}} \end{pmatrix}_{12\times 1}$$

Homogeneous M x 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 $|\mathbf{P}\ \mathbf{m}|^2$

under the constraint $|\mathbf{m}|^2 = 1$

Calibration Problem

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

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

Calibration Problem

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

SVD decomposition of P

$$\boxed{\mathrm{U}_{2n\times 12}\ \mathrm{D}_{12\times 12}\ \mathrm{V}^{T}{}_{12\times 12}}$$

Last column of V gives $\boldsymbol{m}$

Why? See pag 592 of HZ

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

$$\boldsymbol{m} \longrightarrow M$$

Extracting camera parameters

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

Extracting camera parameters

See [FP], Sec. 1.3.1

$$\frac{M}{\rho} = \begin{pmatrix} \underbrace{\begin{matrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + u_0 \boldsymbol{r}_3^T \\ \frac{\beta}{\sin\theta} \boldsymbol{r}_2^T + v_0 \boldsymbol{r}_3^T \\ \boldsymbol{r}_3^T \end{matrix}}_{A} & \underbrace{\begin{matrix} \alpha t_x - \alpha \cot\theta t_y + u_0 t_z \\ \frac{\beta}{\sin\theta} t_y + v_0 t_z \\ t_z \end{matrix}}_{\mathbf{b}} \end{pmatrix} = \mathrm{K}\begin{bmatrix} \mathrm{R} & \mathrm{T} \end{bmatrix}$$

$$\mathrm{K} = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_o \\ 0 & \frac{\beta}{\sin\theta} & v_o \\ 0 & 0 & 1 \end{bmatrix}$$

Box 1

$$\mathrm{A} = \begin{bmatrix} \mathbf{a}_1^{\mathrm{T}} \\ \mathbf{a}_2^{\mathrm{T}} \\ \mathbf{a}_3^{\mathrm{T}} \end{bmatrix} \qquad \mathrm{b} = \begin{bmatrix} \mathrm{b}_1 \\ \mathrm{b}_2 \\ \mathrm{b}_3 \end{bmatrix}$$

Estimated values

Intrinsic

$$\rho = \frac{\pm 1}{|\mathbf{a}_3|} \qquad u_o = \rho^2 (\mathbf{a}_1 \cdot \mathbf{a}_3)$$

$$\mathrm{v}_o = \rho^2 (\mathbf{a}_2 \cdot \mathbf{a}_3)$$

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

Extracting camera parameters

See [FP], Sec. 1.3.1

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

A $\qquad$ $\mathbf{b}$

$$\mathrm{K} = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_o \\ 0 & \frac{\beta}{\sin\theta} & v_o \\ 0 & 0 & 1 \end{bmatrix}$$

Box 1

$$\mathrm{A} = \begin{bmatrix} \mathbf{a}_1^{\mathrm{T}} \\ \mathbf{a}_2^{\mathrm{T}} \\ \mathbf{a}_3^{\mathrm{T}} \end{bmatrix} \qquad \mathrm{b} = \begin{bmatrix} \mathrm{b}_1 \\ \mathrm{b}_2 \\ \mathrm{b}_3 \end{bmatrix}$$

Estimated values

Intrinsic

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

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

Theorem (Faugeras, 1993)

Let $\mathcal{M} = (\mathcal{A} \quad \boldsymbol{b})$ be a 3×4 matrix and let $\boldsymbol{a}_i^T$ $(i = 1, 2, 3)$ denote the rows of the matrix $\mathcal{A}$ formed by the three leftmost columns of $\mathcal{M}$.

Extracting camera parameters

See [FP], Sec. 1.3.1

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

$$\underbrace{}_{\mathcal{A}} \qquad \underbrace{}_{\mathbf{b}}$$

$$\mathrm{K} = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_o \\ 0 & \frac{\beta}{\sin\theta} & v_o \\ 0 & 0 & 1 \end{bmatrix}$$

Box 1

$$\mathrm{A} = \begin{bmatrix} \mathbf{a}_1^{\mathrm{T}} \\ \mathbf{a}_2^{\mathrm{T}} \\ \mathbf{a}_3^{\mathrm{T}} \end{bmatrix} \qquad \mathrm{b} = \begin{bmatrix} \mathrm{b}_1 \\ \mathrm{b}_2 \\ \mathrm{b}_3 \end{bmatrix}$$

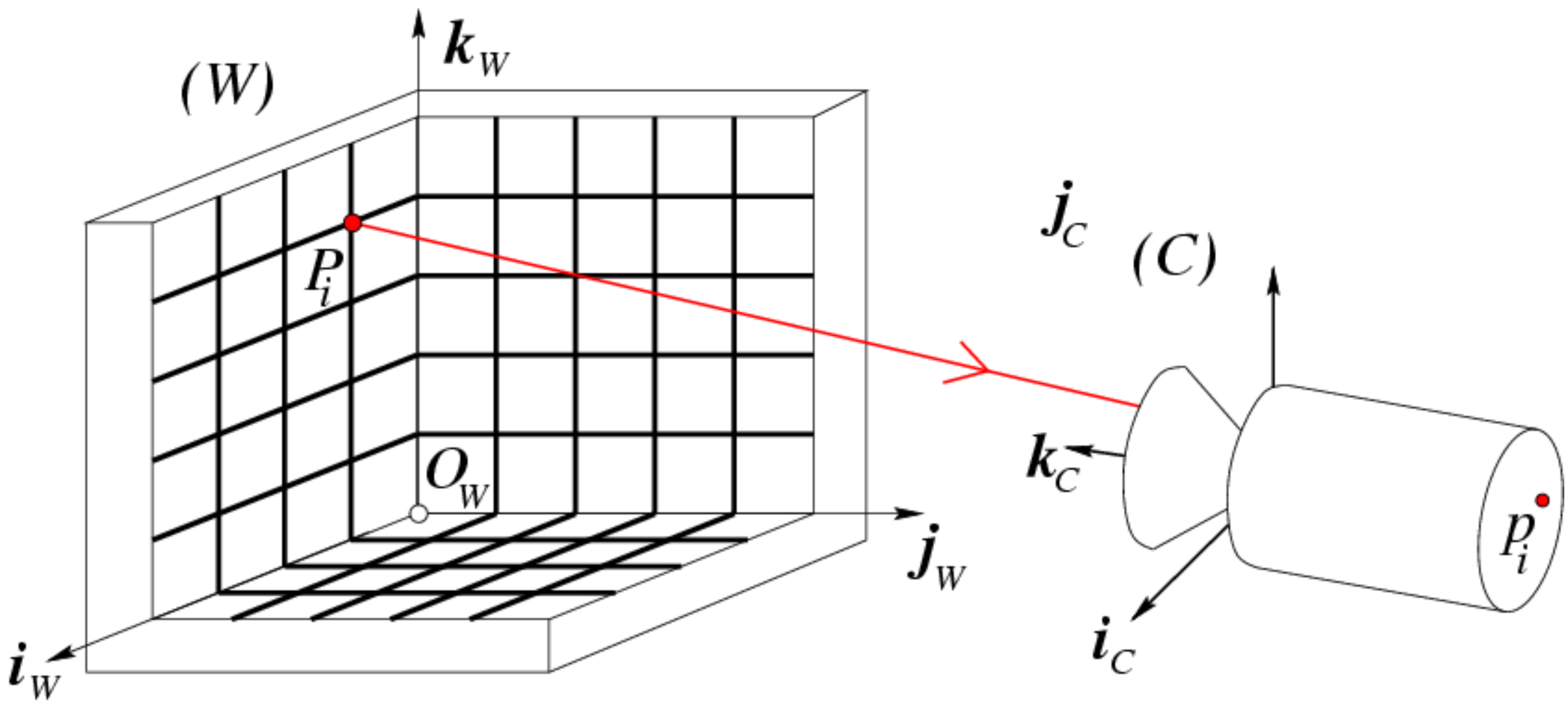
Estimated values

Extrinsic

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

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

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

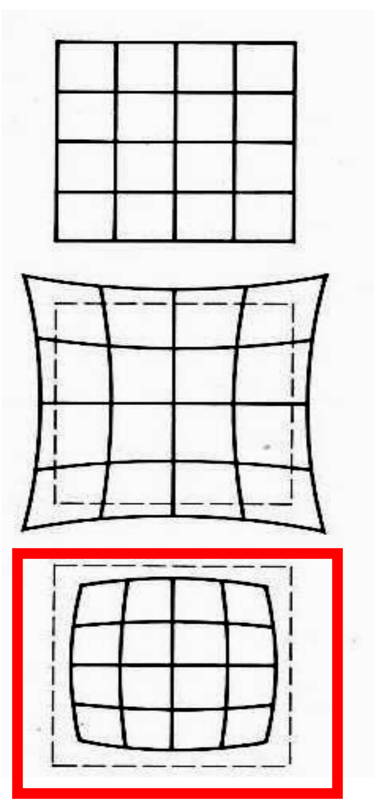
- Recap of projective cameras
- Camera calibration problem
- Camera calibration with radial distortion
- Example

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

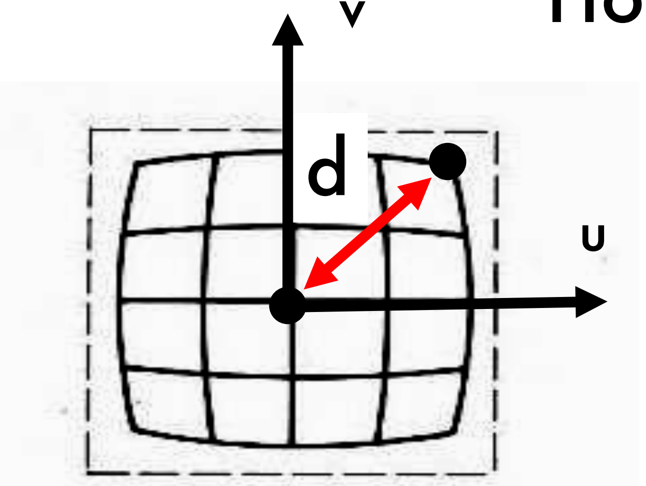
Pin cushion

Barrel

Radial Distortion

Image magnification decreases with distance from the optical center

How do we model that?



$$\underbrace{\begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{S_\lambda} M\, P_i \rightarrow \begin{bmatrix} u_i \\ v_i \end{bmatrix} = p_i$$

$$\lambda = \underbrace{1 \pm \sum_{p=1}^{3} \overbrace{\boxed{\kappa_p}}^{\text{Distortion coefficient}} d^{2p}}_{\text{Polynomial function}}$$

[Eq. 5]

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

To model radial behavior

[Eq. 6]

Radial Distortion

$$\underbrace{\begin{bmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathrm{M}}_{\mathrm{Q}} \mathrm{P_i} \rightarrow \begin{bmatrix} \mathrm{u_i} \\ \mathrm{v_i} \end{bmatrix} = \mathrm{p_i} \qquad \mathrm{Q} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

$$\mathrm{p_i} = \begin{bmatrix} \mathrm{u_i} \\ \mathrm{v_i} \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \mathrm{P_i}}{\mathbf{q}_3 \mathrm{P_i}} \\ \dfrac{\mathbf{q}_2 \mathrm{P_i}}{\mathbf{q}_3 \mathrm{P_i}} \end{bmatrix} \rightarrow \begin{cases} u_i \mathbf{q}_3 P_i = \mathbf{q}_1 P_i \\ v_i \mathbf{q}_3 P_i = \mathbf{q}_2 P_i \end{cases}$$

Is this a linear system of equations?

No! why?

[Eqs.7]

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1\ \mathrm{P}_i}{\mathbf{q}_3\ \mathrm{P}_i} \\ \dfrac{\mathbf{q}_2\ \mathrm{P}_i}{\mathbf{q}_3\ \mathrm{P}_i} \end{bmatrix} \longrightarrow X = f(Q) \quad \text{[Eq .8]}$$

i=1…n

X: measurements

Q: parameters

f( ) is the nonlinear mapping

-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

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \, \mathrm{P}_i}{\mathbf{q}_3 \, \mathrm{P}_i} \\[2ex] \dfrac{\mathbf{q}_2 \, \mathrm{P}_i}{\mathbf{q}_3 \, \mathrm{P}_i} \end{bmatrix} \longrightarrow X = f(Q) \quad \text{[Eq .8]}$$

i=1…n

measurements

parameters

f( ) is the nonlinear mapping

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

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \, \mathrm{P_i}}{\mathbf{q}_3 \, \mathrm{P_i}} \\ \dfrac{\mathbf{q}_2 \, \mathrm{P_i}}{\mathbf{q}_3 \, \mathrm{P_i}} \end{bmatrix} \longrightarrow X = f(Q) \quad \text{[Eq .8]}$$

i=1…n

measurements

parameters

f( ) is the nonlinear mapping

Typical assumptions:

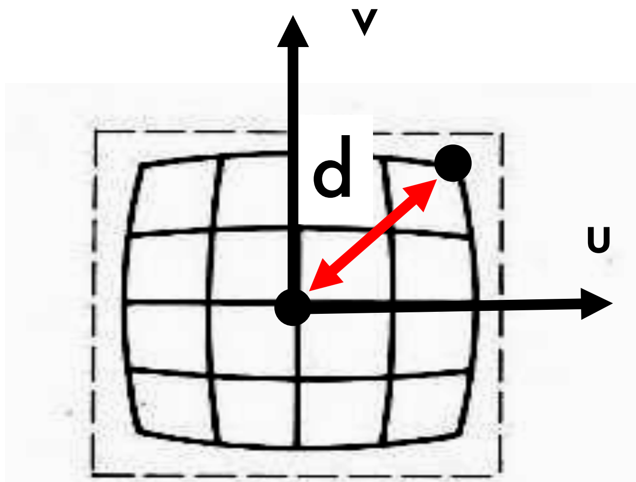
- zero-skew, square pixel
- u_o, v_o = known center of the image

Radial Distortion

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

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...

$$\mathrm{p_i} = \begin{bmatrix} \mathrm{u_i} \\ \mathrm{v_i} \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \dfrac{\mathbf{m}_1 \, \mathrm{P_i}}{\mathbf{m}_3 \, \mathrm{P_i}} \\ \dfrac{\mathbf{m}_2 \, \mathrm{P_i}}{\mathbf{m}_3 \, \mathrm{P_i}} \end{bmatrix} \rightarrow \frac{u_i}{v_i} = \frac{\dfrac{(\mathbf{m}_1 \, P_i)}{(\mathbf{m}_3 \, P_i)}}{\dfrac{(\mathbf{m}_2 \, P_i)}{(\mathbf{m}_3 \, P_i)}} = \frac{\mathbf{m}_1 \, P_i}{\mathbf{m}_2 \, P_i}$$

[Eq .9]

[Eq .10]

$$\begin{cases} v_1(\mathbf{m}_1 \, P_1) - u_1(\mathbf{m}_2 \, P_1) = 0 \\ v_i(\mathbf{m}_1 \, P_i) - u_i(\mathbf{m}_2 \, P_i) = 0 \\ \vdots \\ v_n(\mathbf{m}_1 \, P_n) - u_n(\mathbf{m}_2 \, P_n) = 0 \end{cases}$$

[Eq .11]

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

Get $\mathbf{m}_1$ and $\mathbf{m}_2$ by SVD

$$\mathbf{L} \stackrel{\text{def}}{=} \begin{pmatrix} v_1 \mathrm{P}_1^T & -u_1 \mathrm{P}_1^T \\ v_2 \mathrm{P}_2^T & -u_2 \mathrm{P}_2^T \\ \vdots & \vdots \\ v_n \mathrm{P}_n^T & -u_n \mathrm{P}_n^T \end{pmatrix}$$

$$\mathbf{n} = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \end{bmatrix}$$

Radial Distortion

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

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \dfrac{\mathbf{m}_1 P_i}{\mathbf{m}_3 P_i} \\ \dfrac{\mathbf{m}_2 P_i}{\mathbf{m}_3 P_i} \end{bmatrix}$$

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

There are some degenerate configurations for which m_1 and m_2 cannot be computed

Lecture 3
Camera Calibration

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

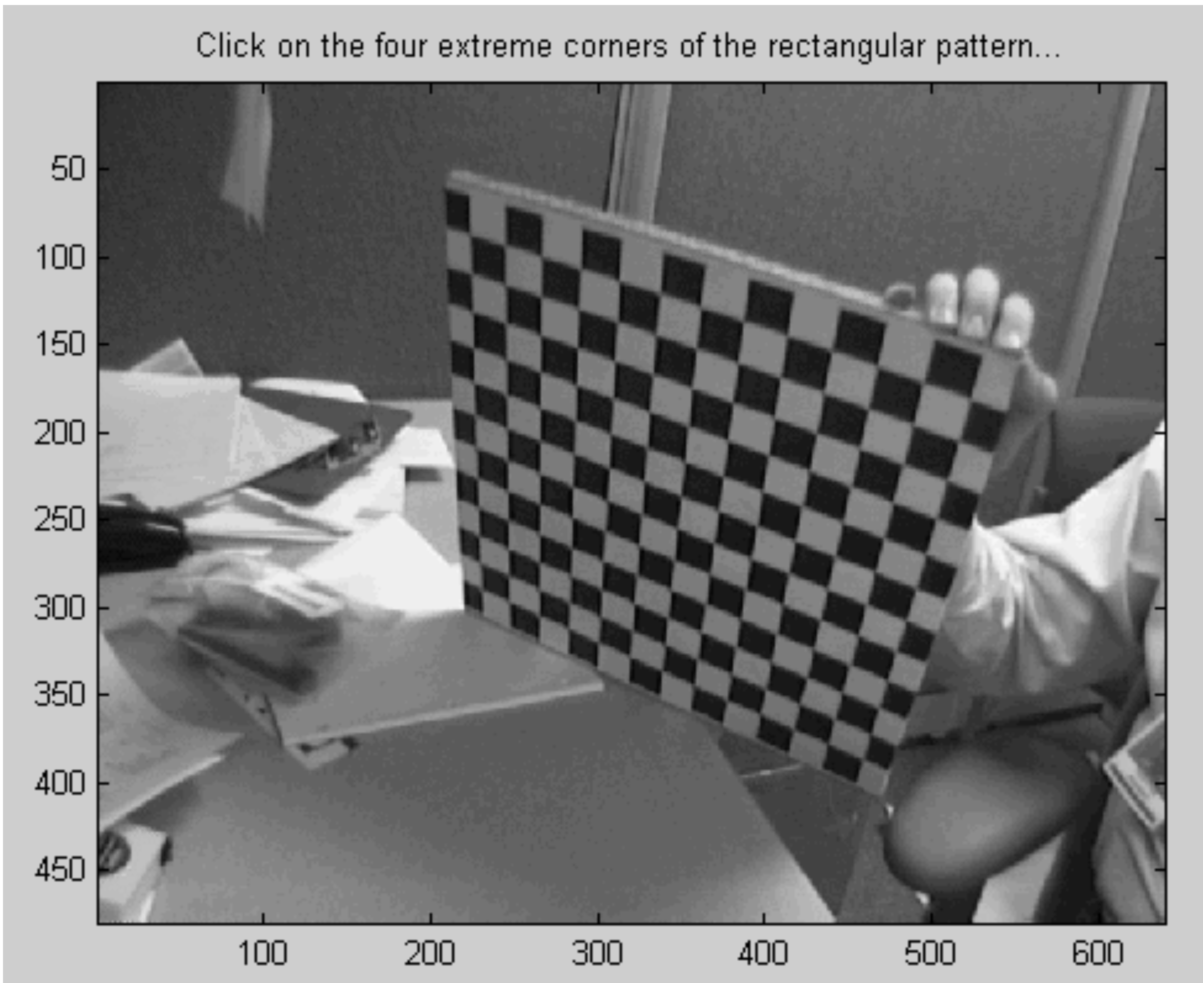
Reading: [FP] Chapter 1 "Geometric Camera Calibration"
[HZ] Chapter 7 "Computation of Camera Matrix P"

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

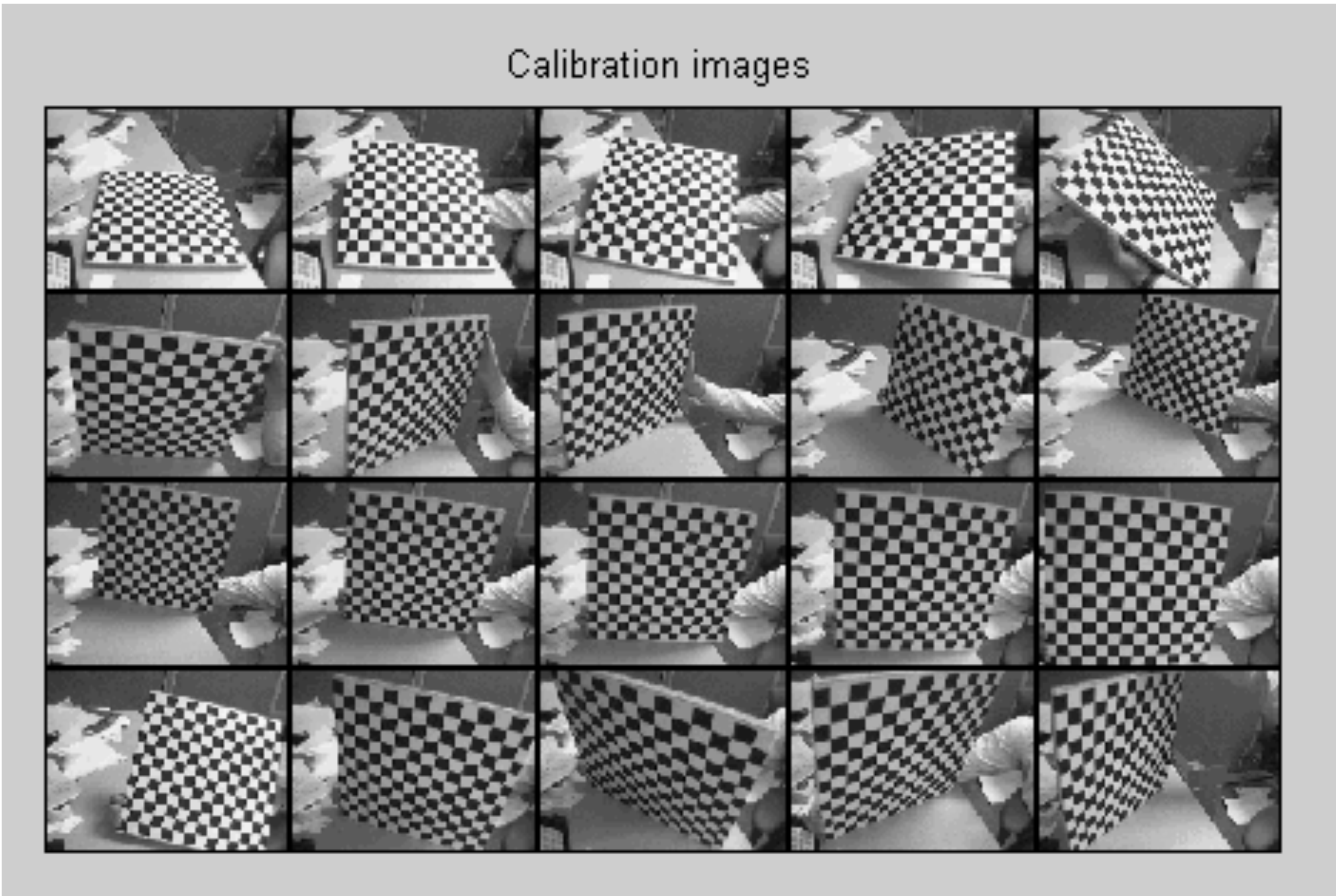
Calibration Procedure

Camera Calibration Toolbox for Matlab
J. Bouguet – [1998-2000]

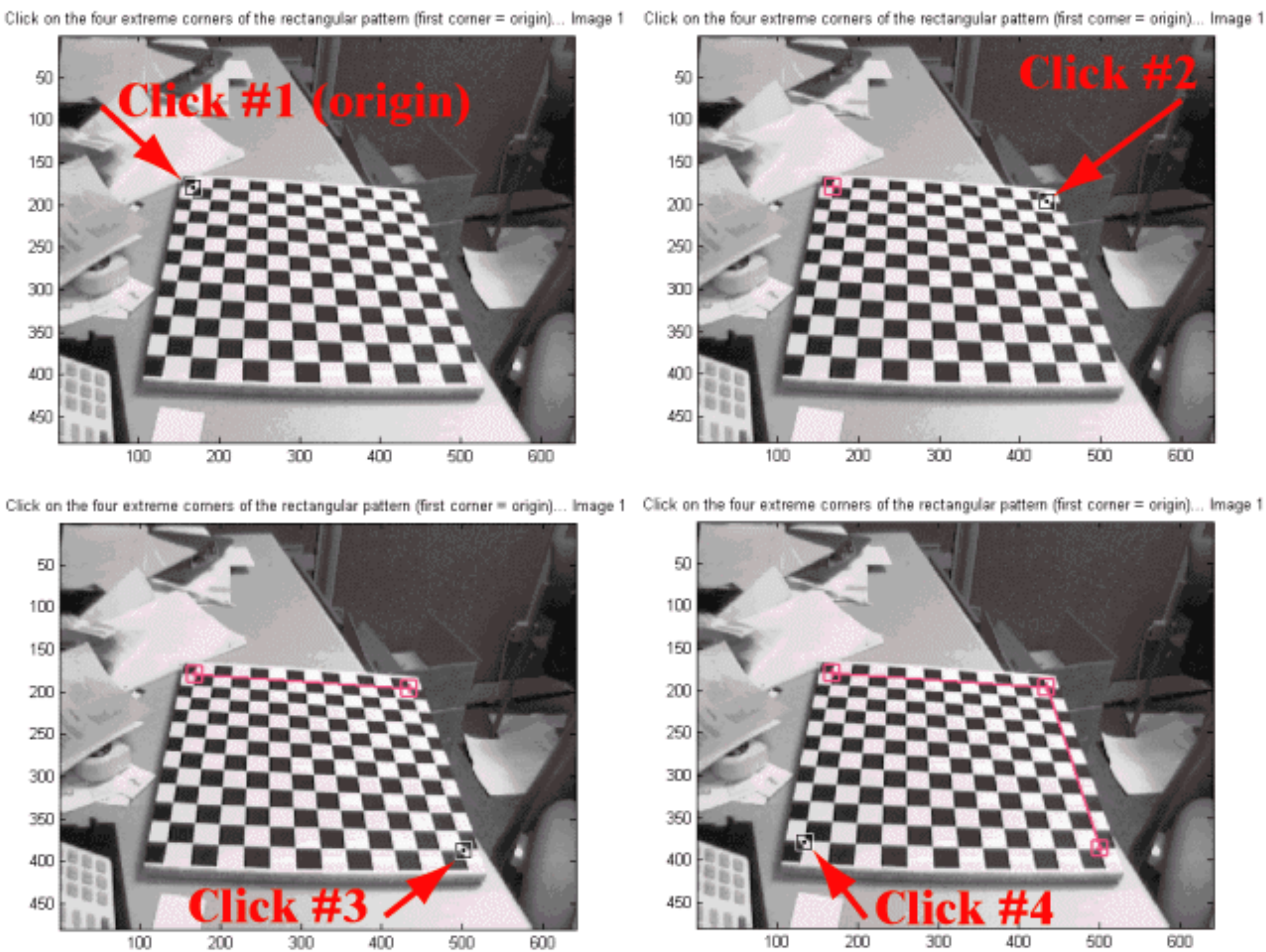
http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#examples



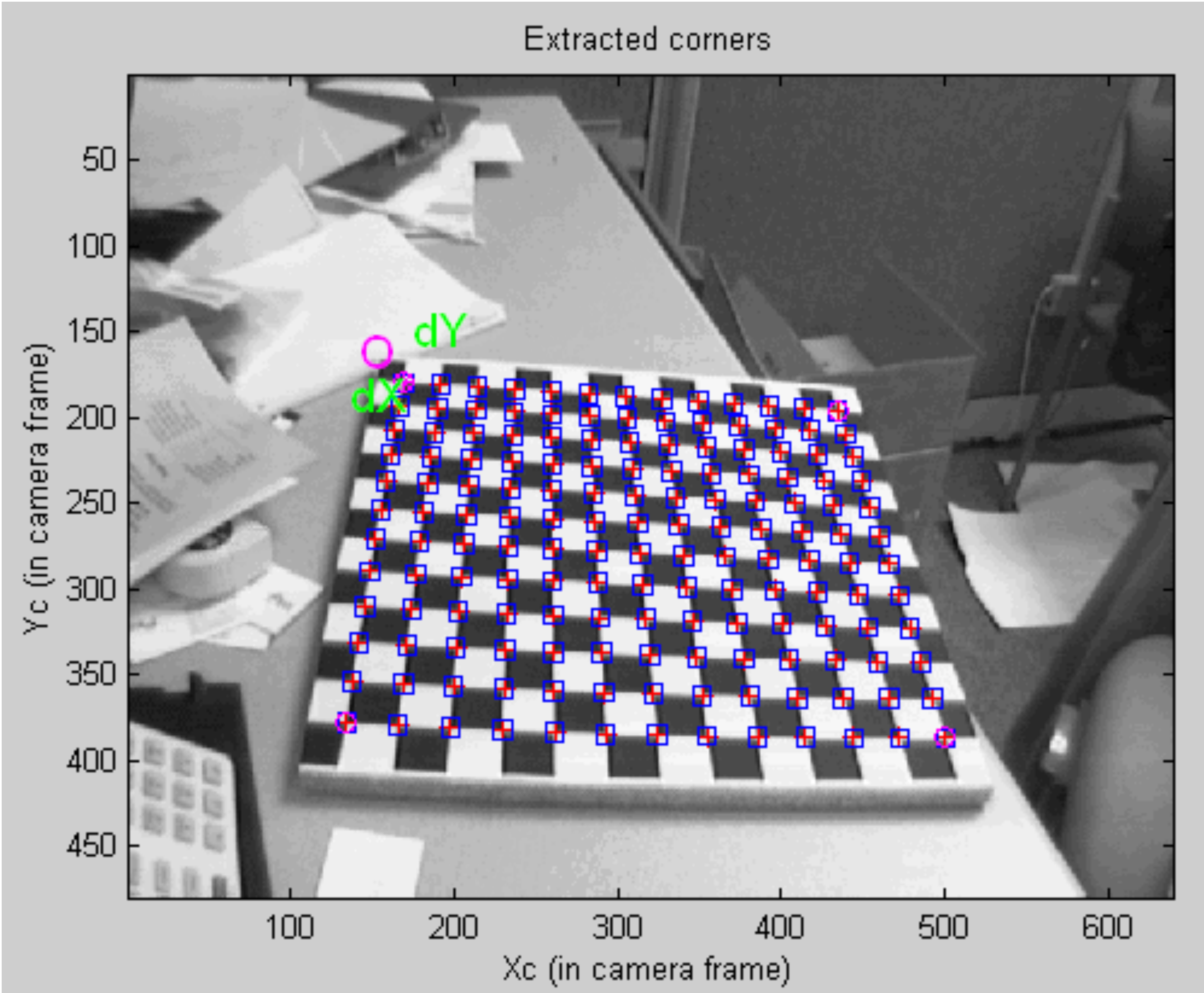
Calibration Procedure



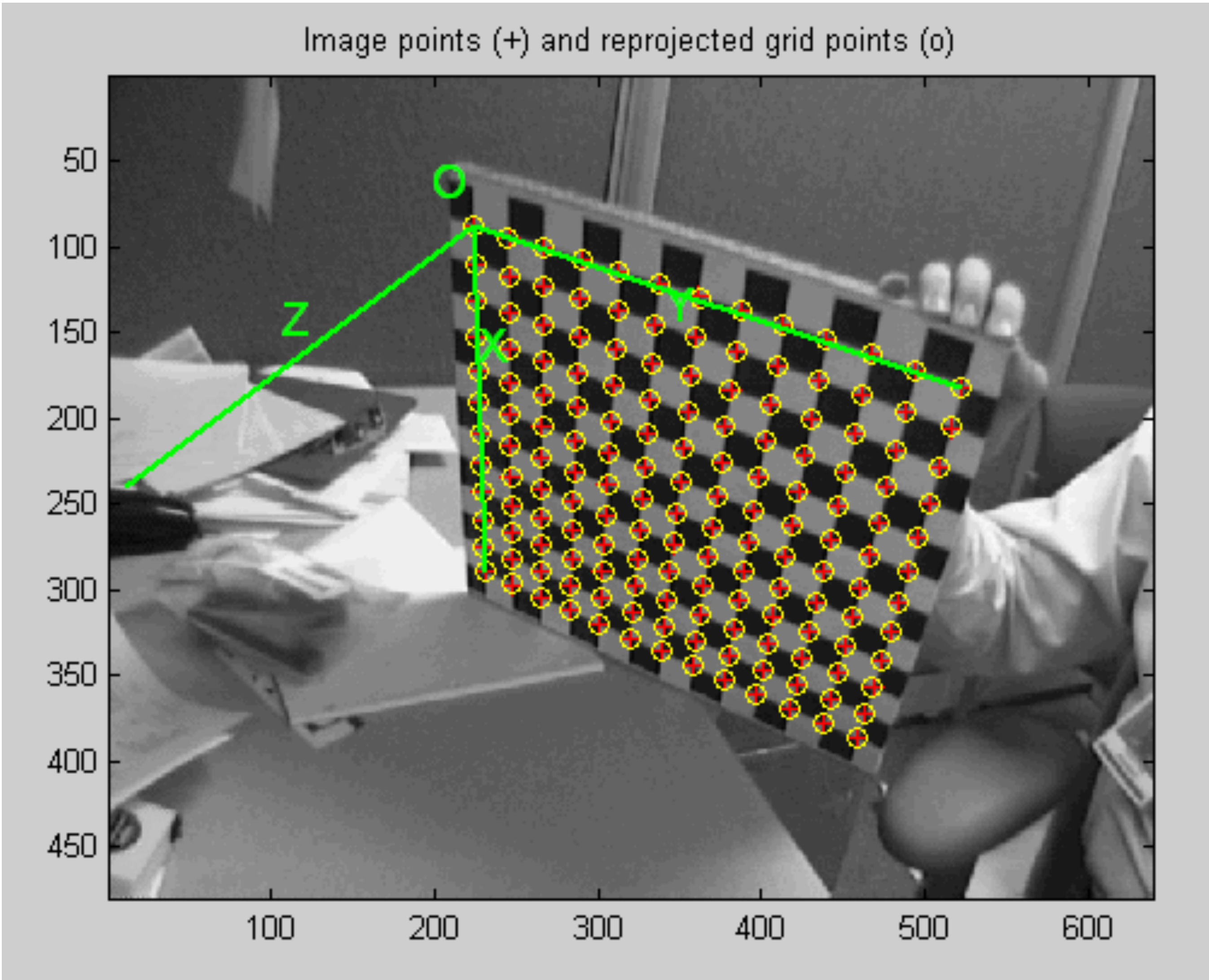
Calibration Procedure



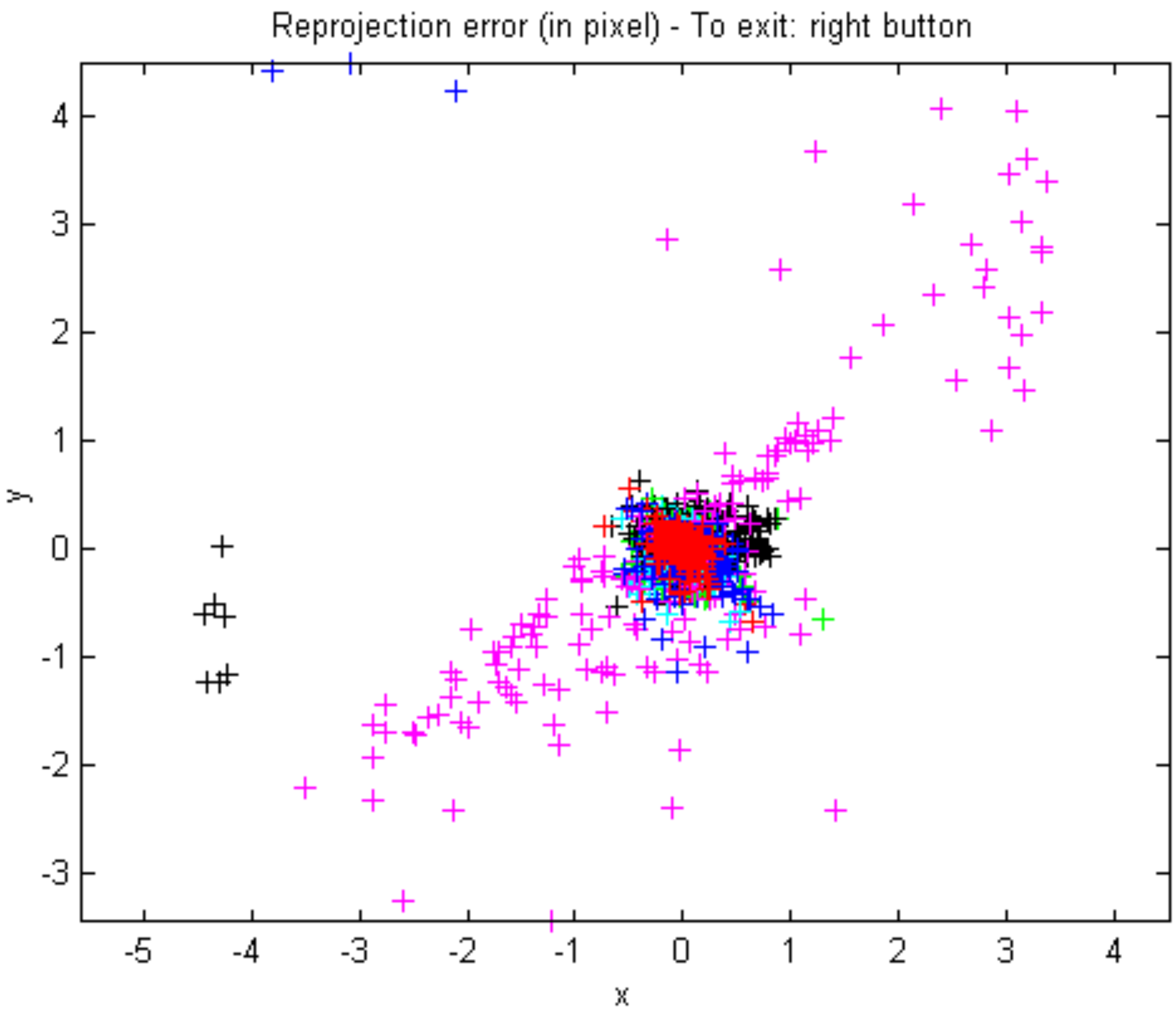
Calibration Procedure



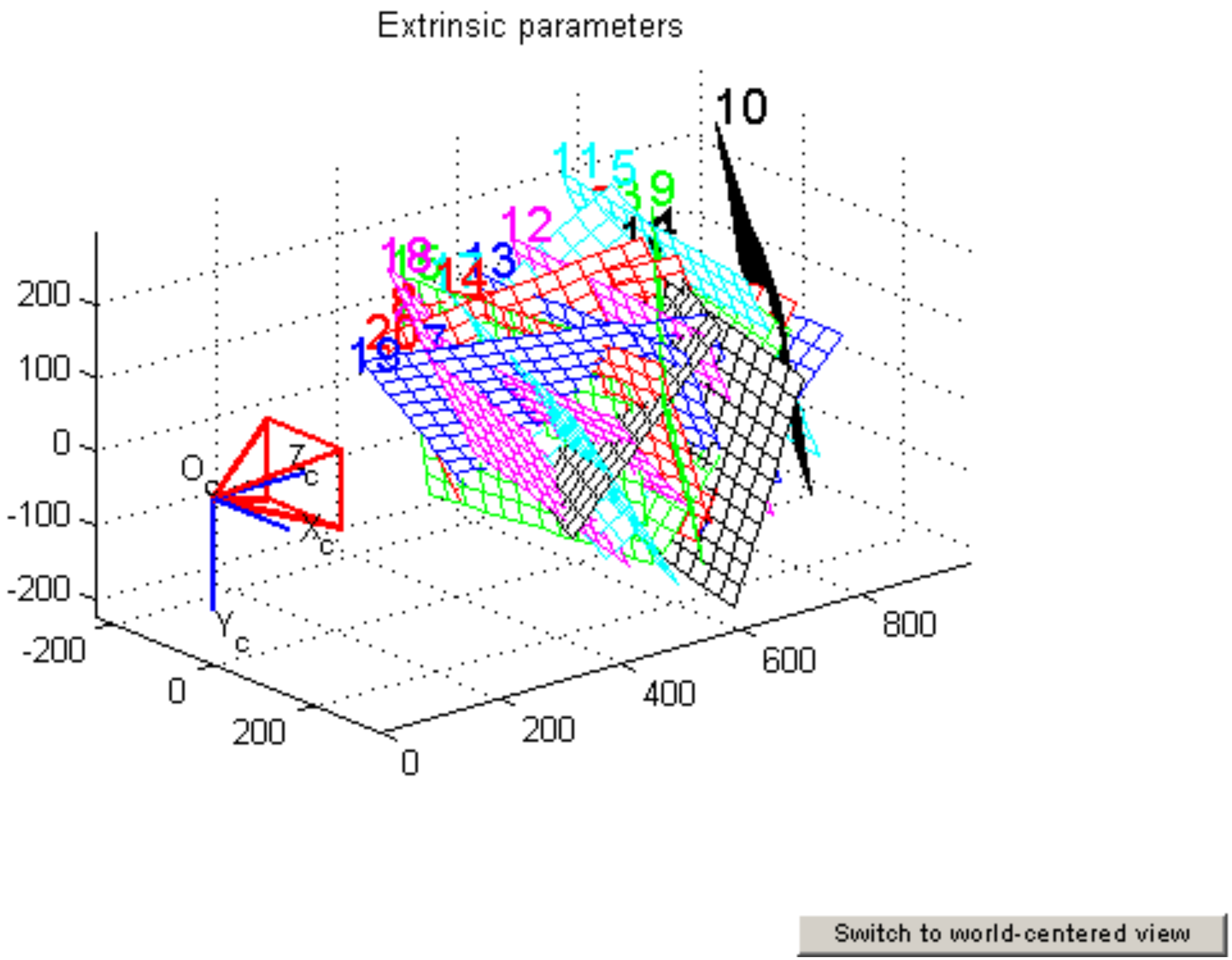
Calibration Procedure



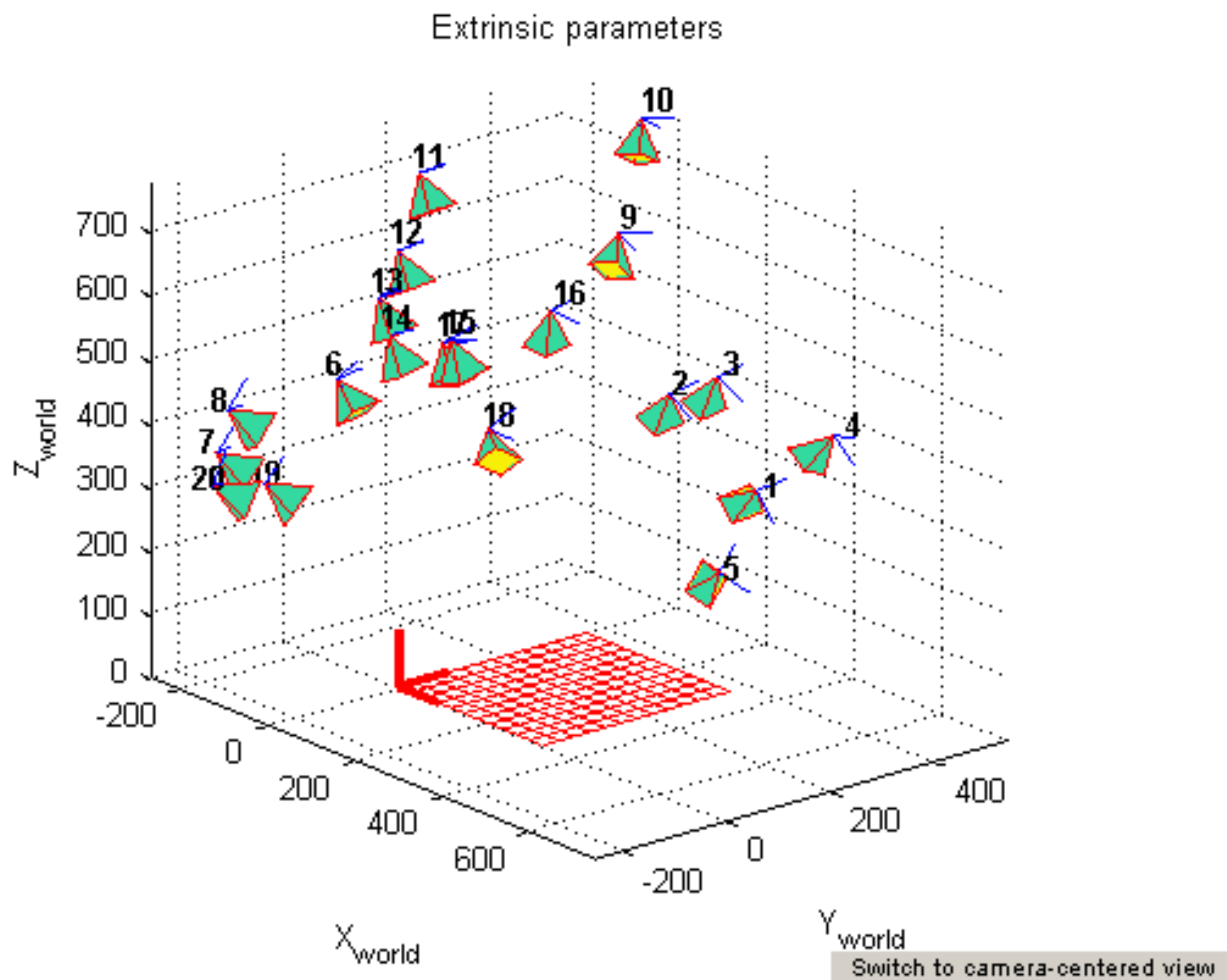
Calibration Procedure



Calibration Procedure



Calibration Procedure



Next lecture

- Single view reconstruction

Eigenvalues and Eigenvectors

Eigendecomposition

$$A = S\Lambda\, S^{-1} = S \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \cdot & \\ & & & \lambda_N \end{bmatrix} S^{-1}$$

Eigenvectors of A are columns of S

$$S = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_N \end{bmatrix}$$

Singular Value decomposition

$$A = U \Sigma V^{-1}$$

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \cdot & \\ & & & \sigma_N \end{bmatrix}$$

U, V = orthogonal matrix

$$\sigma_i = \sqrt{\lambda_i}$$

σ = singular value
λ = eigenvalue of $A^t A$