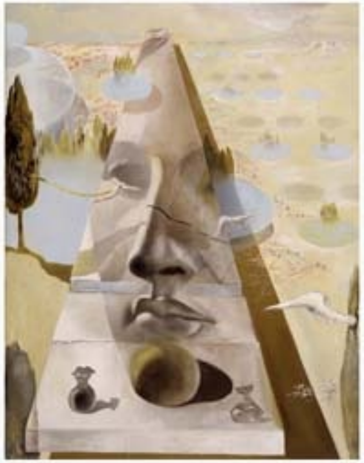


EECS 442 – Computer vision

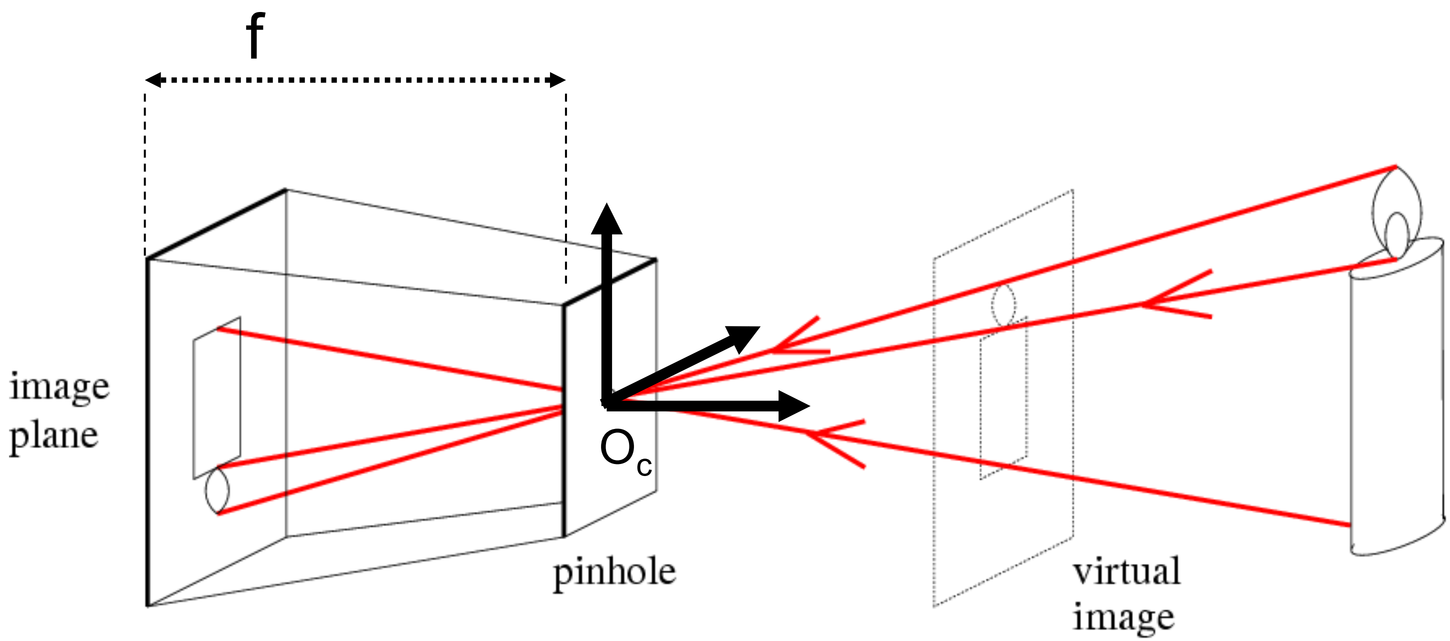
Camera Calibration

- Review camera parameters
- Camera calibration problem
- Example

Reading: [FP] Chapter 3
[HZ] Chapter 7

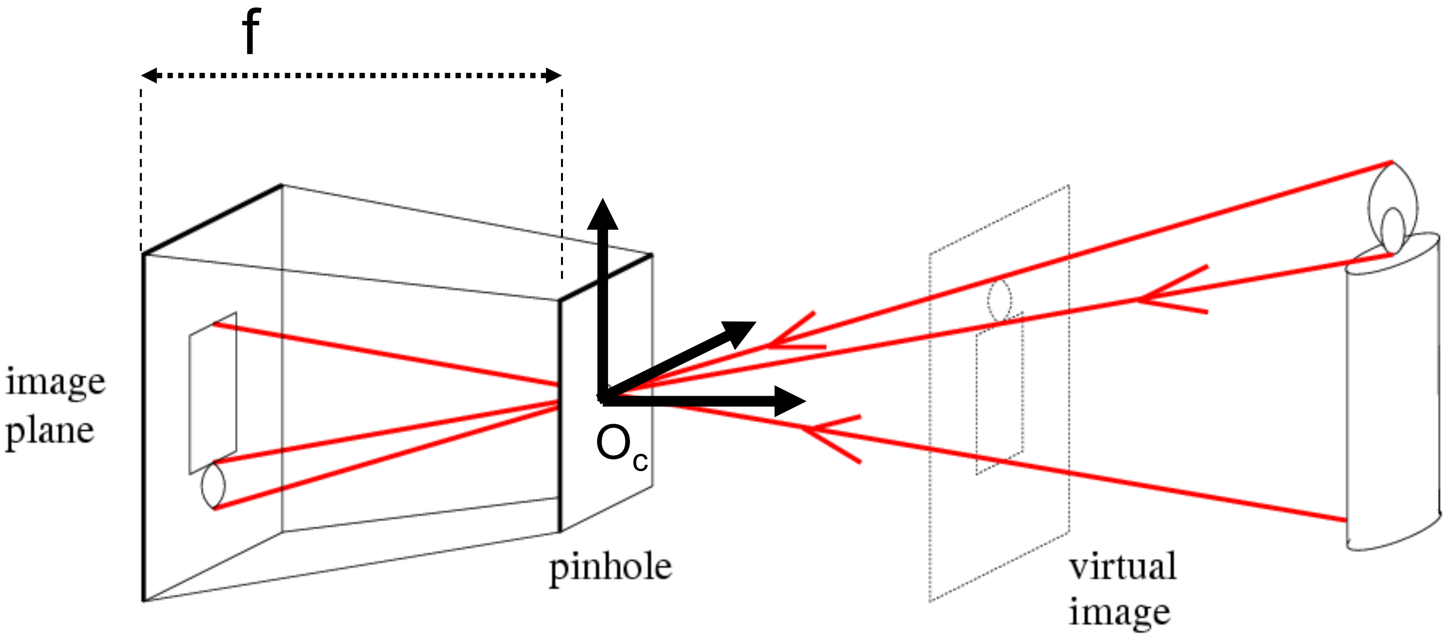
Some slides in this lecture are courtesy to Profs. J. Ponce & F-F Li

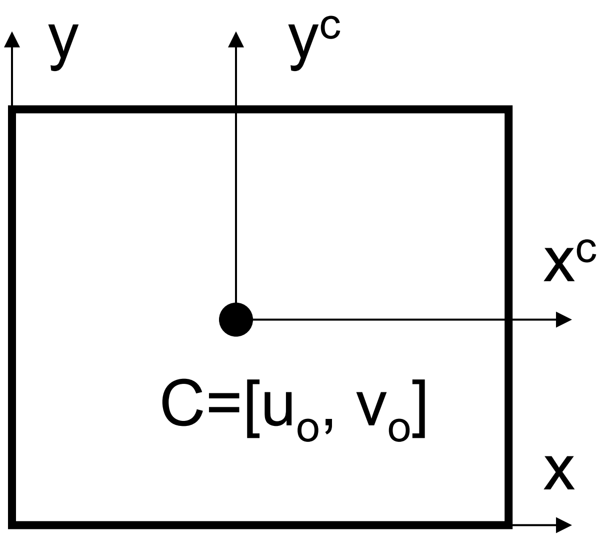
Projective camera



f = focal length

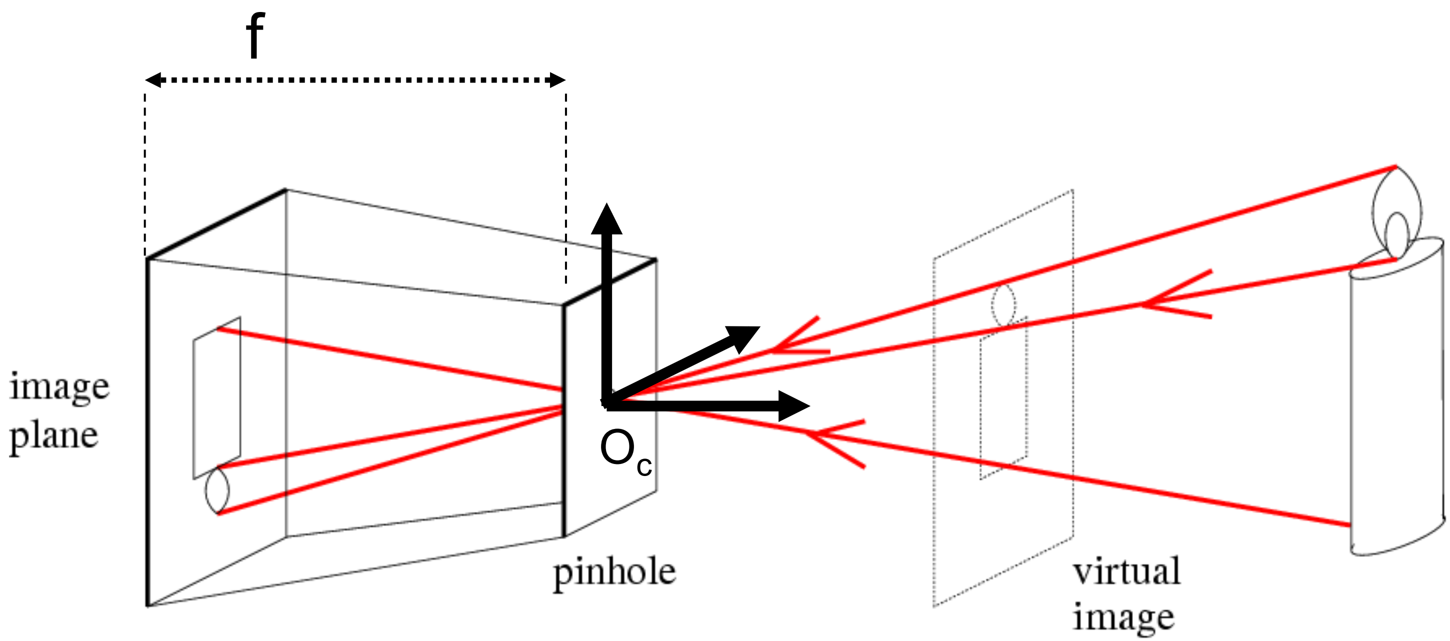
Projective camera





f = focal length
u_o, v_o = offset

Projective camera



Units: k,l [pixel/m]

f [m]

Non-square pixels

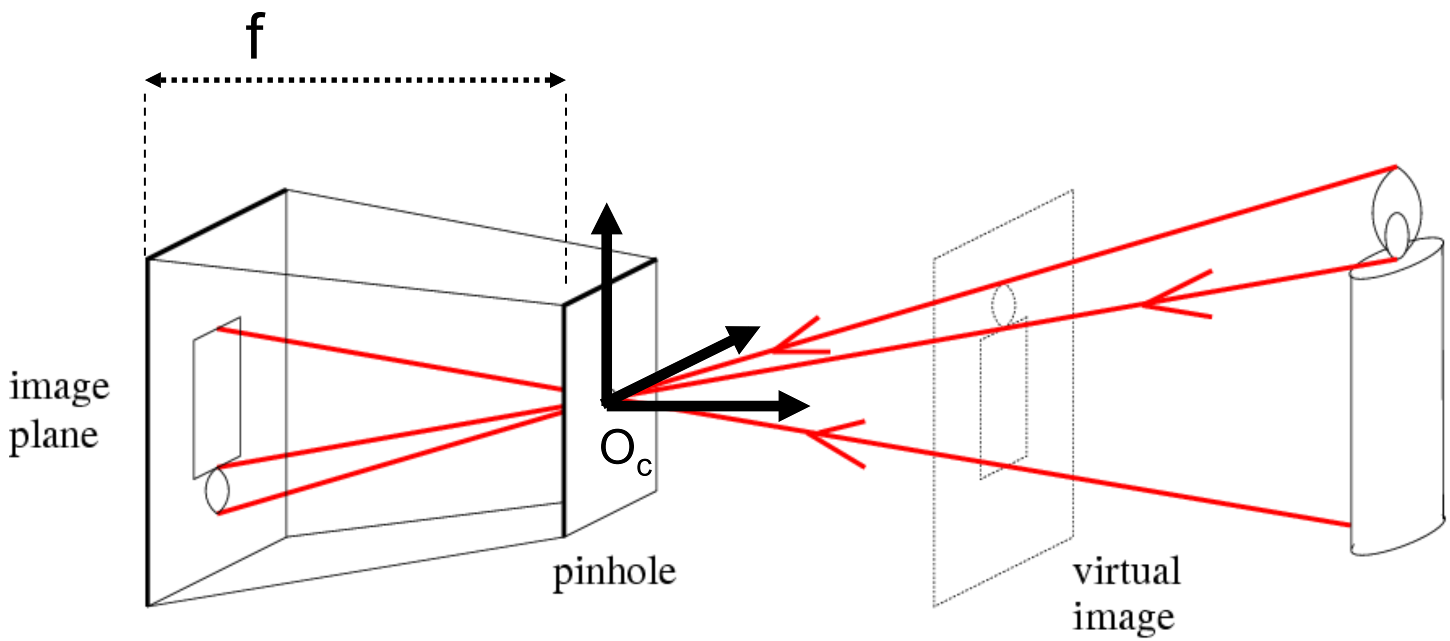
α, β [pixel]

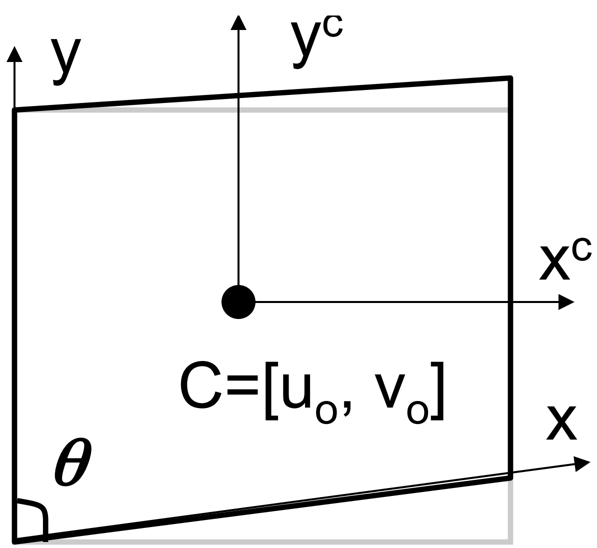
f = focal length

u_o, v_o = offset

α, β $\rightarrow$ non-square pixels

Projective camera





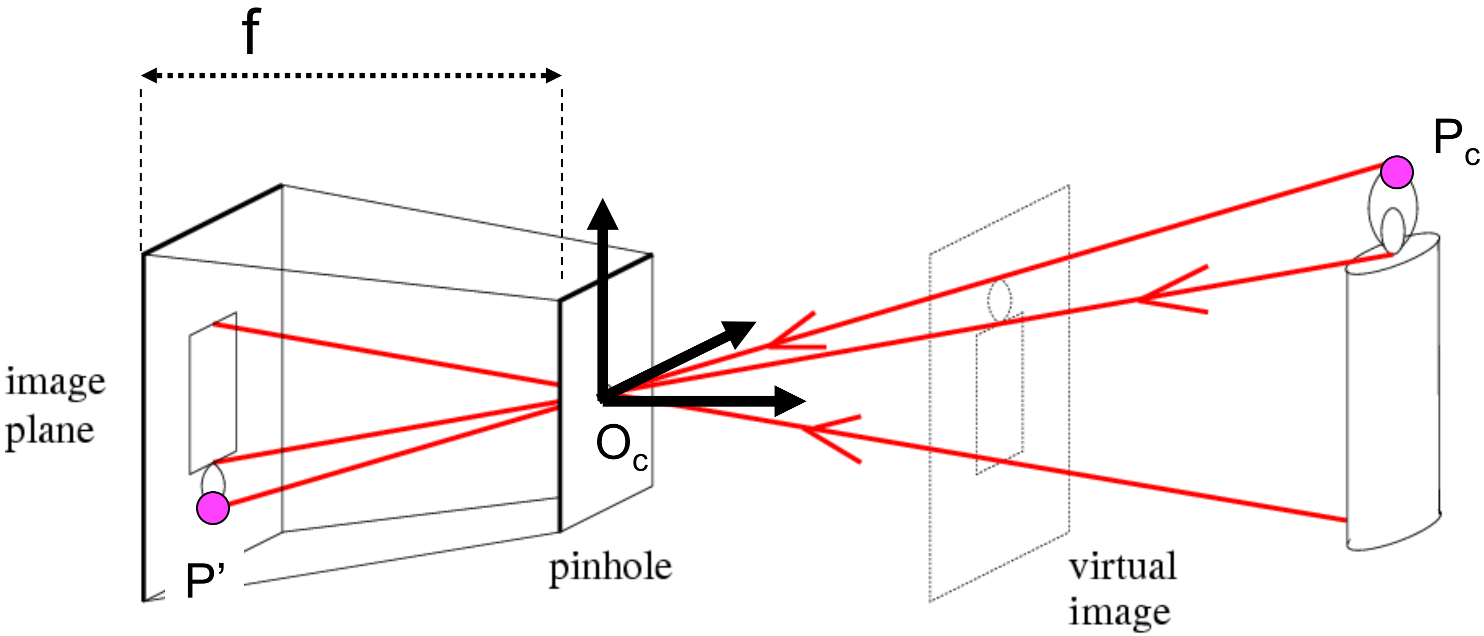
f = focal length

u_o, v_o = offset

α, β → non-square pixels

θ = skew angle

Projective camera



$$P' = \begin{bmatrix} \alpha & s & u_o & 0 \\ 0 & \beta & v_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

f = focal length

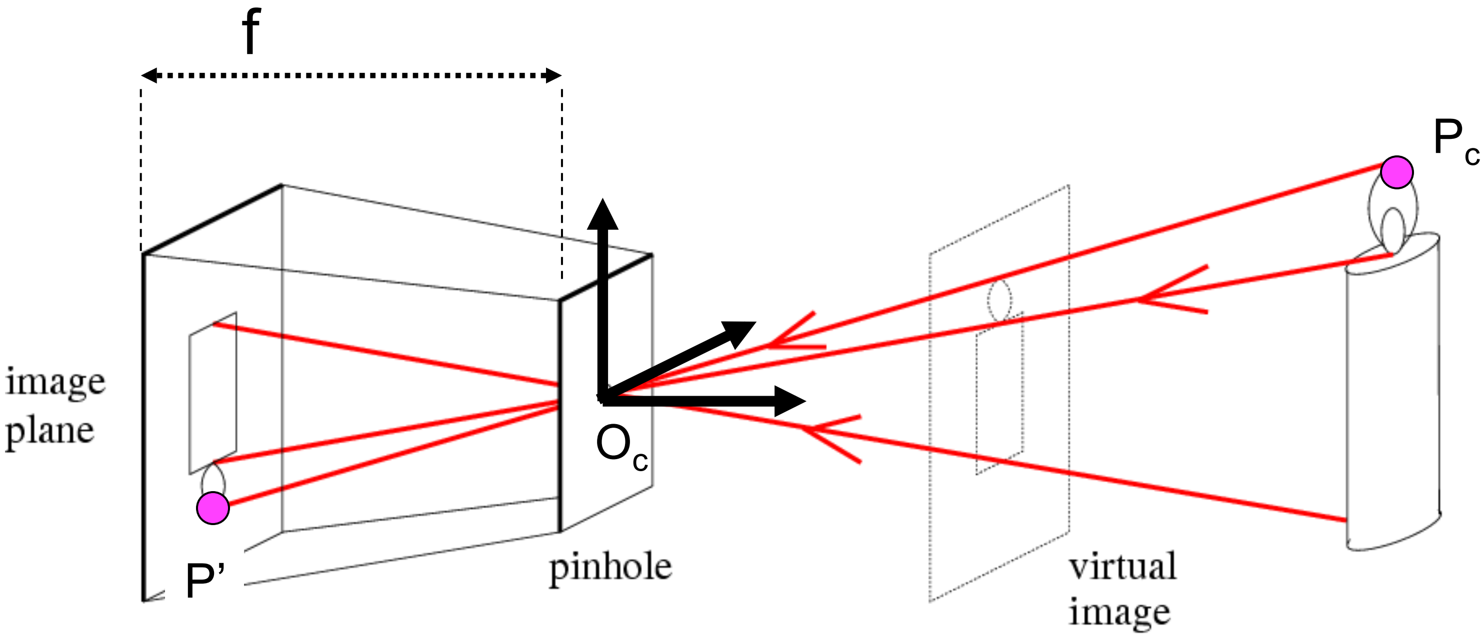
u_o, v_o = offset

α, β → non-square pixels

θ = skew angle

K has 5 degrees of freedom!

Projective camera



$$P' = \begin{bmatrix} \alpha & -\alpha\cot\theta & u_o & 0 \\ 0 & \dfrac{\beta}{\sin\theta} & v_o & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

f = focal length

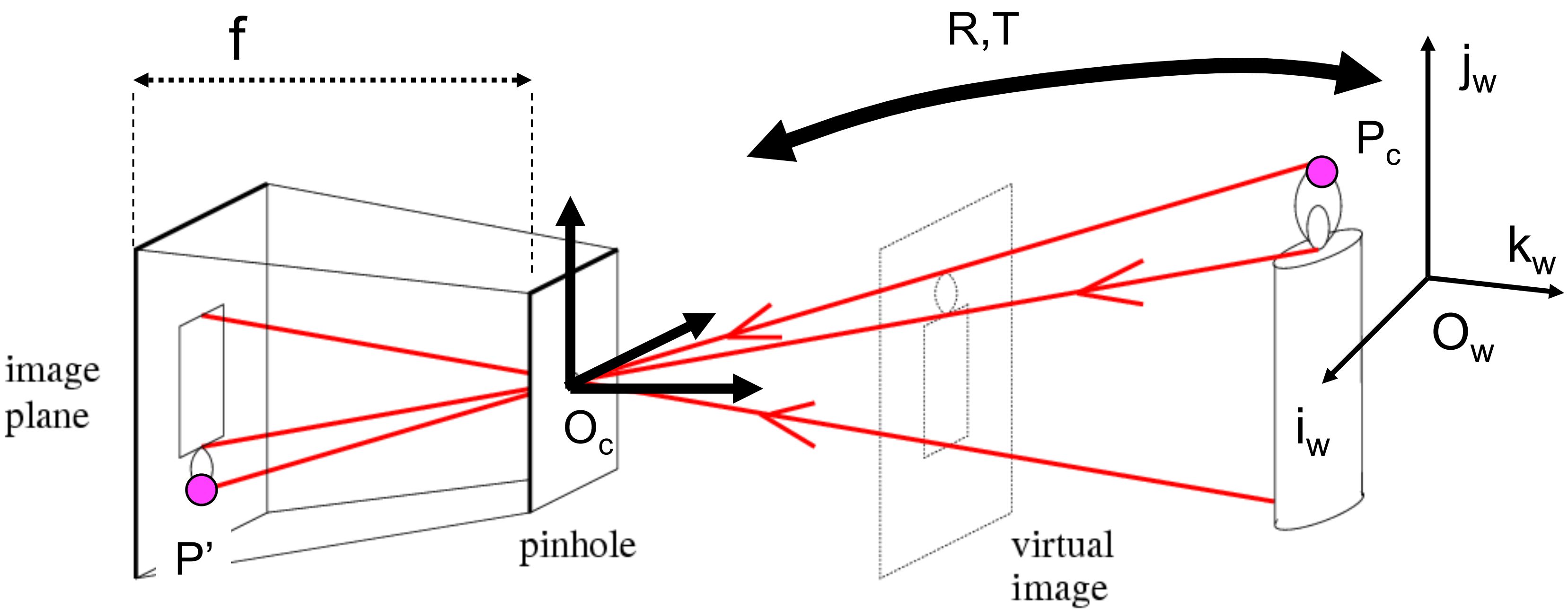
u_o, v_o = offset

$\alpha, \beta \rightarrow$ non-square pixels

θ = skew angle

K has 5 degrees of freedom!

Projective camera



$$P = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4\times 4} P_w$$

$$T = -R\,O_c$$

f = focal length

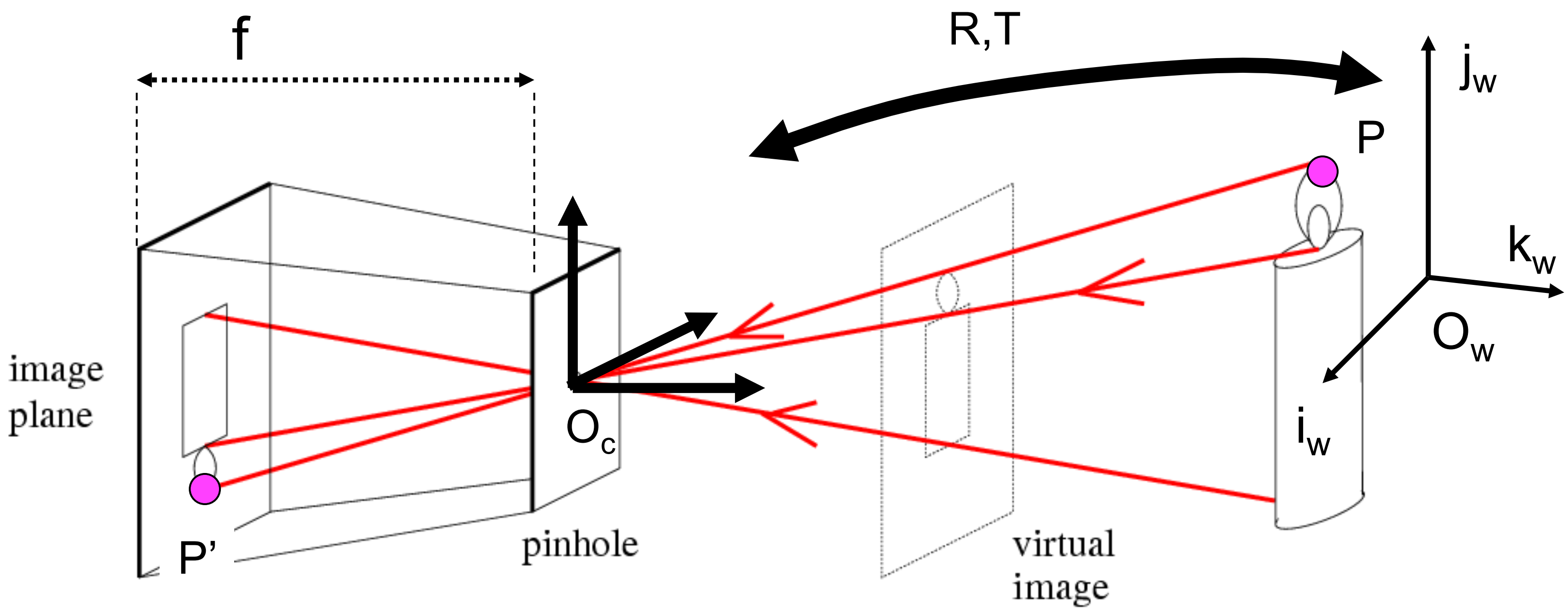
u_o, v_o = offset

$\alpha, \beta \rightarrow$ non-square pixels

θ = skew angle

R,T = rotation, translation

Projective camera



$$P' = M\,P_w$$

$$= \underbrace{K}_{\text{Internal parameters}}\underbrace{\begin{bmatrix} R & T \end{bmatrix}}_{\text{External parameters}} P_w$$

f = focal length

u_o, v_o = offset

$\alpha, \beta \rightarrow$ non-square pixels

θ = skew angle

R,T = rotation, translation

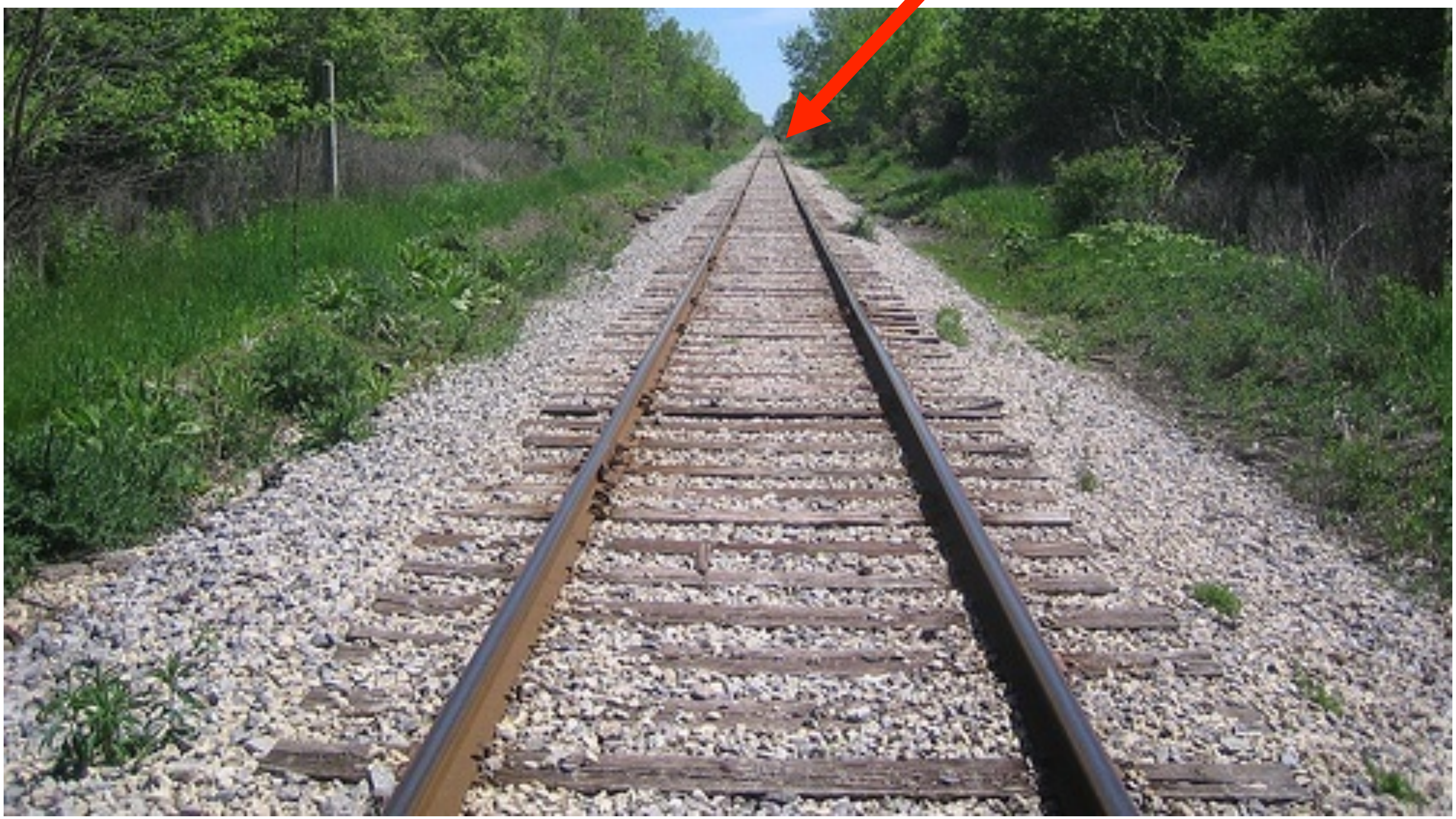
Properties of Projection

- Points project to points
- Lines project to lines
- Distant objects look smaller

Properties of Projection

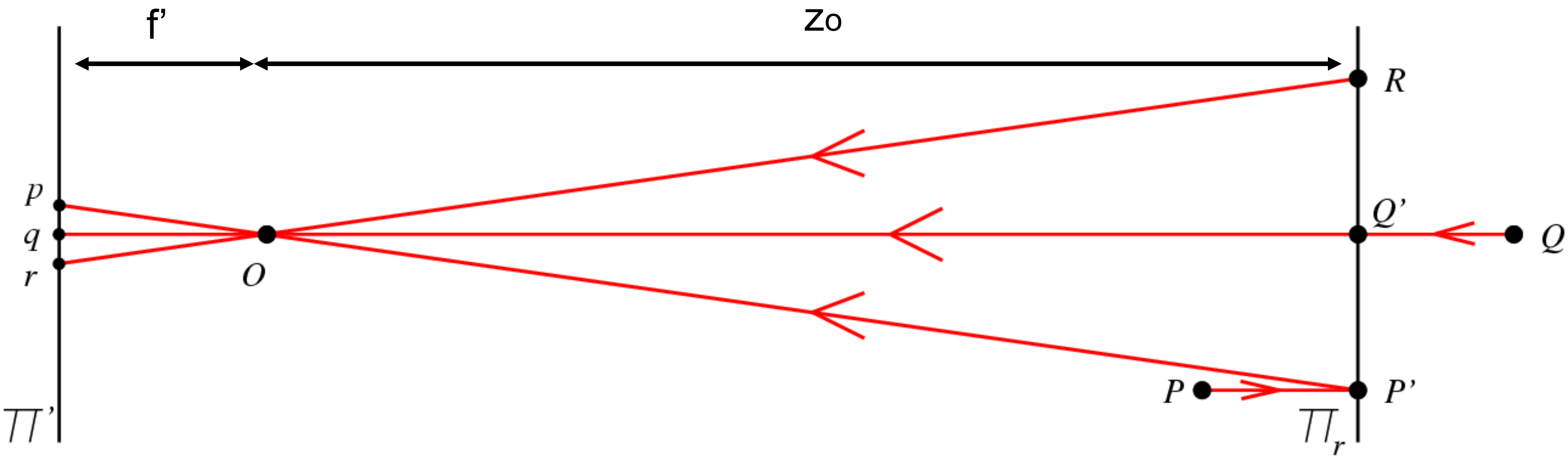
- Angles are not preserved
- Parallel lines meet!

Vanishing point



Weak perspective projection

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

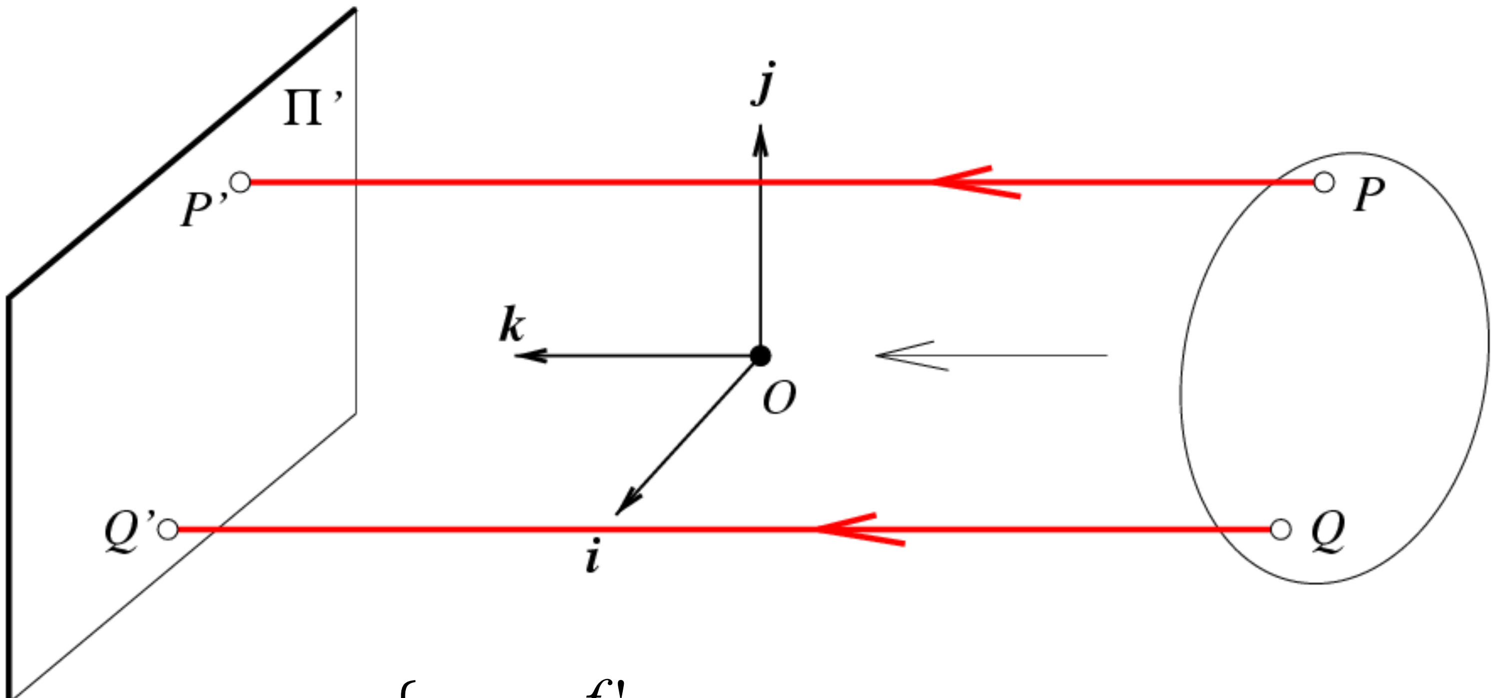


$$\begin{cases} x' = -\dfrac{f'}{z}x \\ y' = -\dfrac{f'}{z}y \end{cases} \rightarrow \begin{cases} x' = -\boxed{\dfrac{f'}{z_0}}x \\ y' = -\dfrac{f'}{z_0}y \end{cases}$$

Magnification m

Orthographic (affine) projection

Distance from center of projection to image plane is infinite



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

Pros and Cons of These Models

- Weak perspective much simpler math.
  - Accurate when object is small and distant.
  - Most useful for recognition.
- Pinhole perspective much more accurate for scenes.
  - Used in structure from motion.

Projective camera

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

Projective camera

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

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

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

Goal of calibration

Estimate intrinsic and extrinsic parameters from 1 or multiple images

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

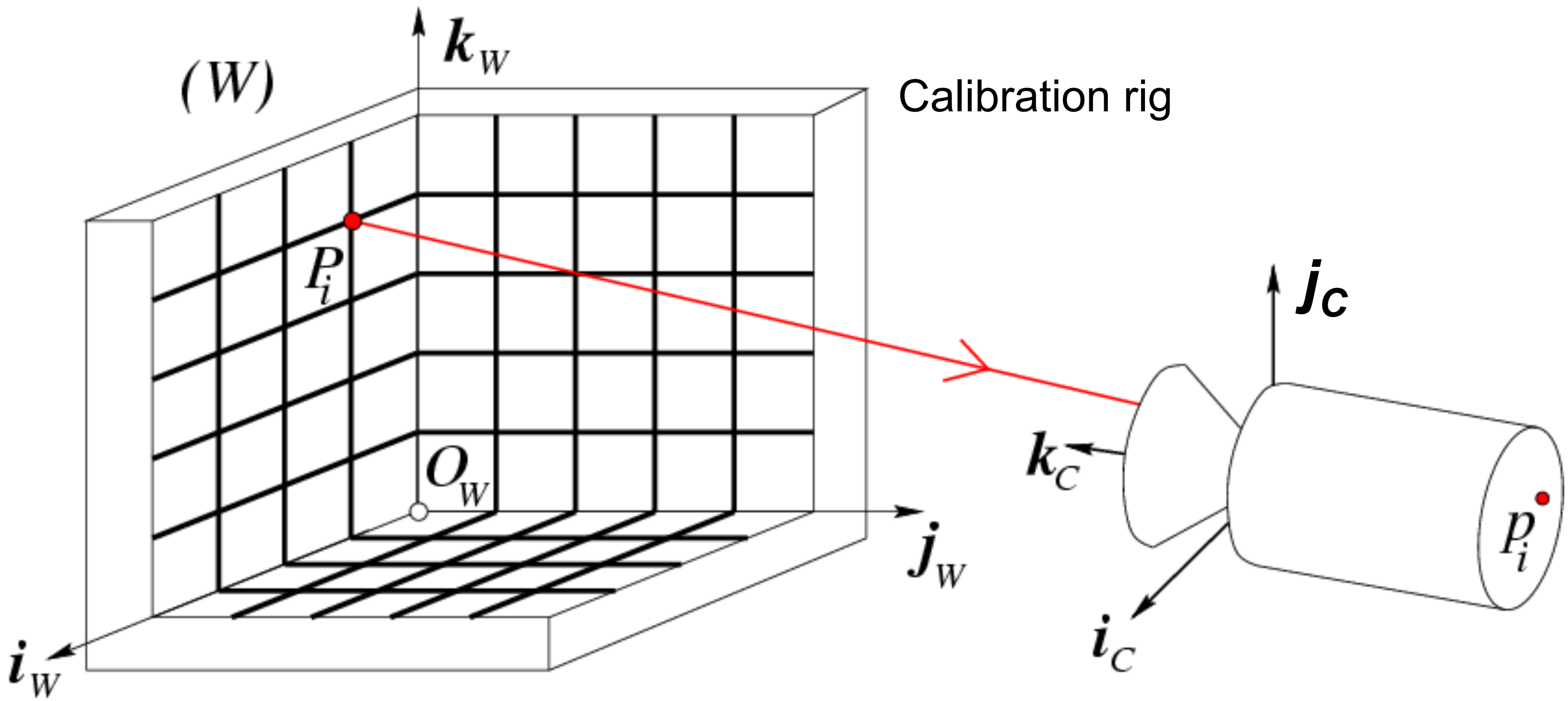
$$\mathcal{M} = \begin{pmatrix} \alpha \boldsymbol{r}_1^T - \alpha \cot\theta \boldsymbol{r}_2^T + u_0 \boldsymbol{r}_3^T & \alpha t_x - \alpha \cot\theta t_y + u_0 t_z \\ \dfrac{\beta}{\sin\theta} \boldsymbol{r}_2^T + v_0 \boldsymbol{r}_3^T & \dfrac{\beta}{\sin\theta} t_y + v_0 t_z \\ \boldsymbol{r}_3^T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

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

Change notation:
$P = P_w$

$p = P'$

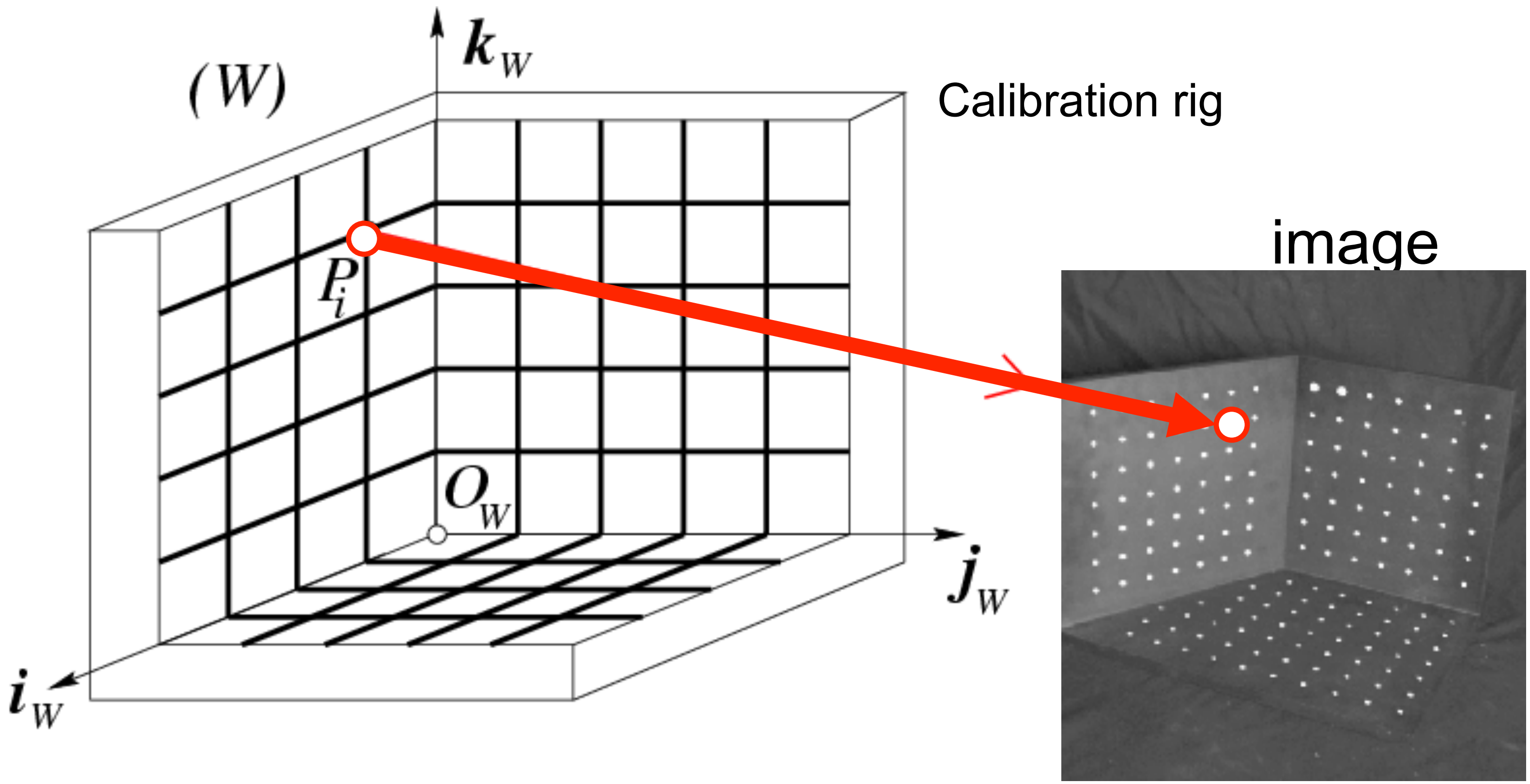
Calibration Problem



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

Goal: compute intrinsic and extrinsic parameters

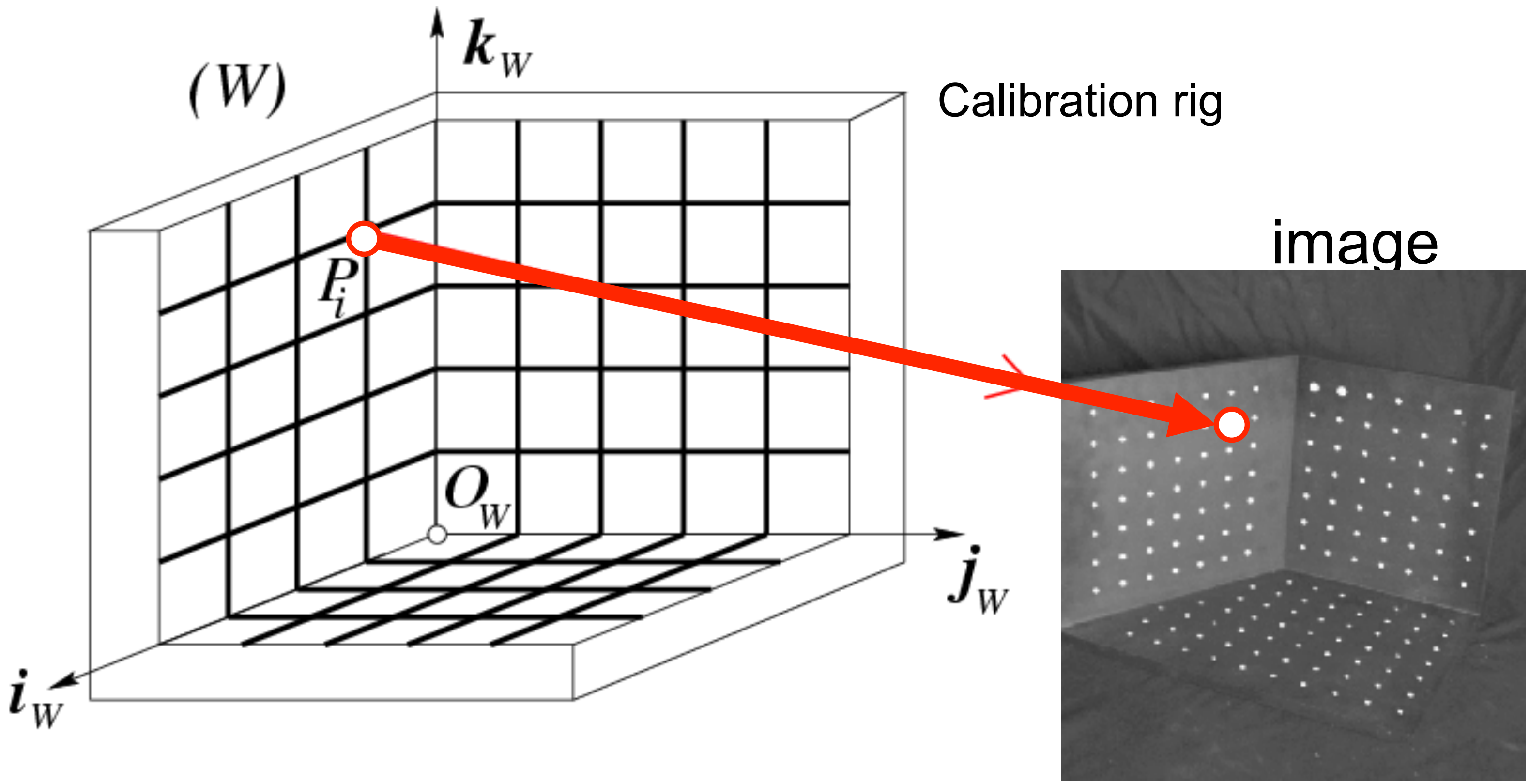
Calibration Problem



- $P_1 \ldots P_n$ 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

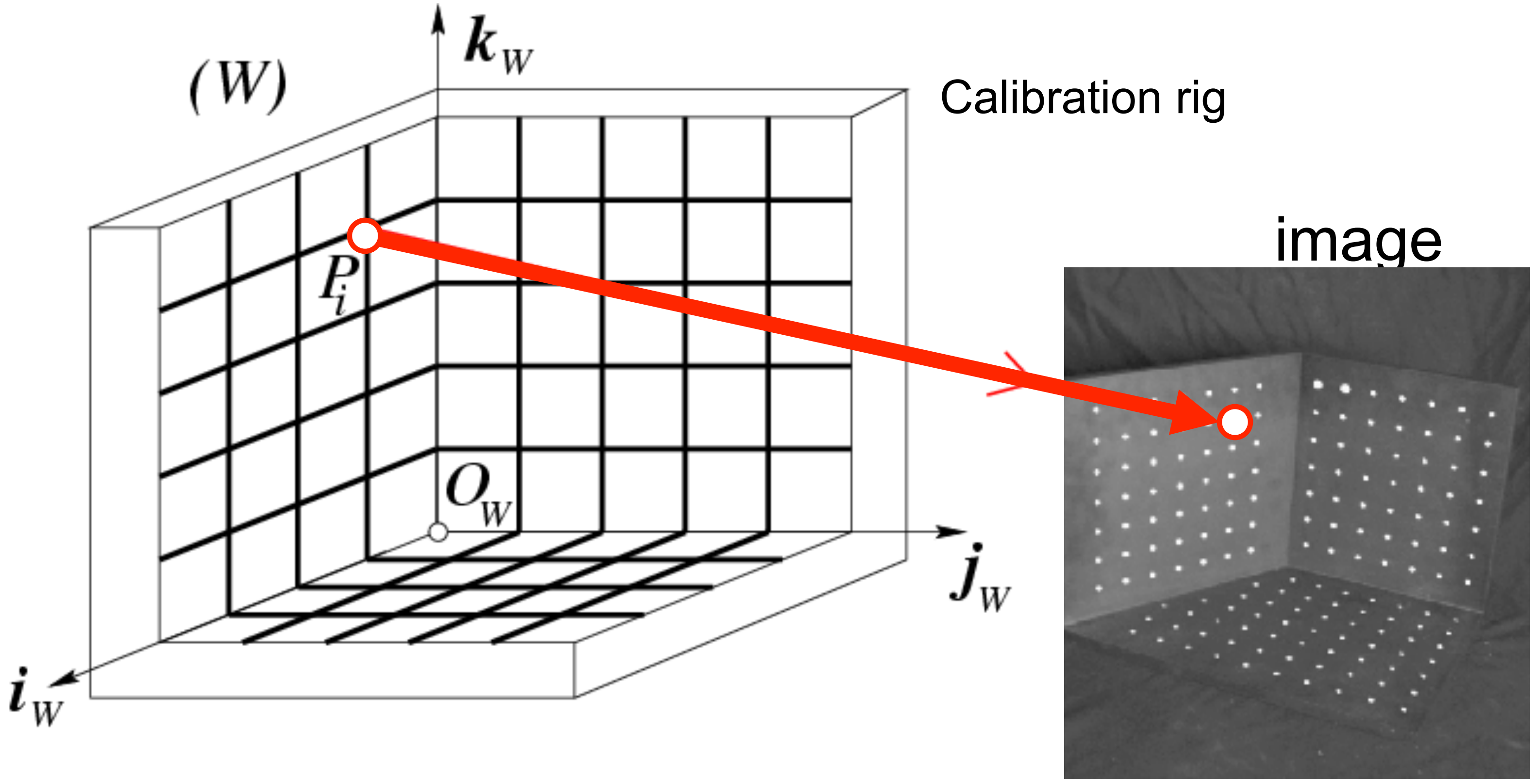
Calibration Problem



How many correspondences do we need?

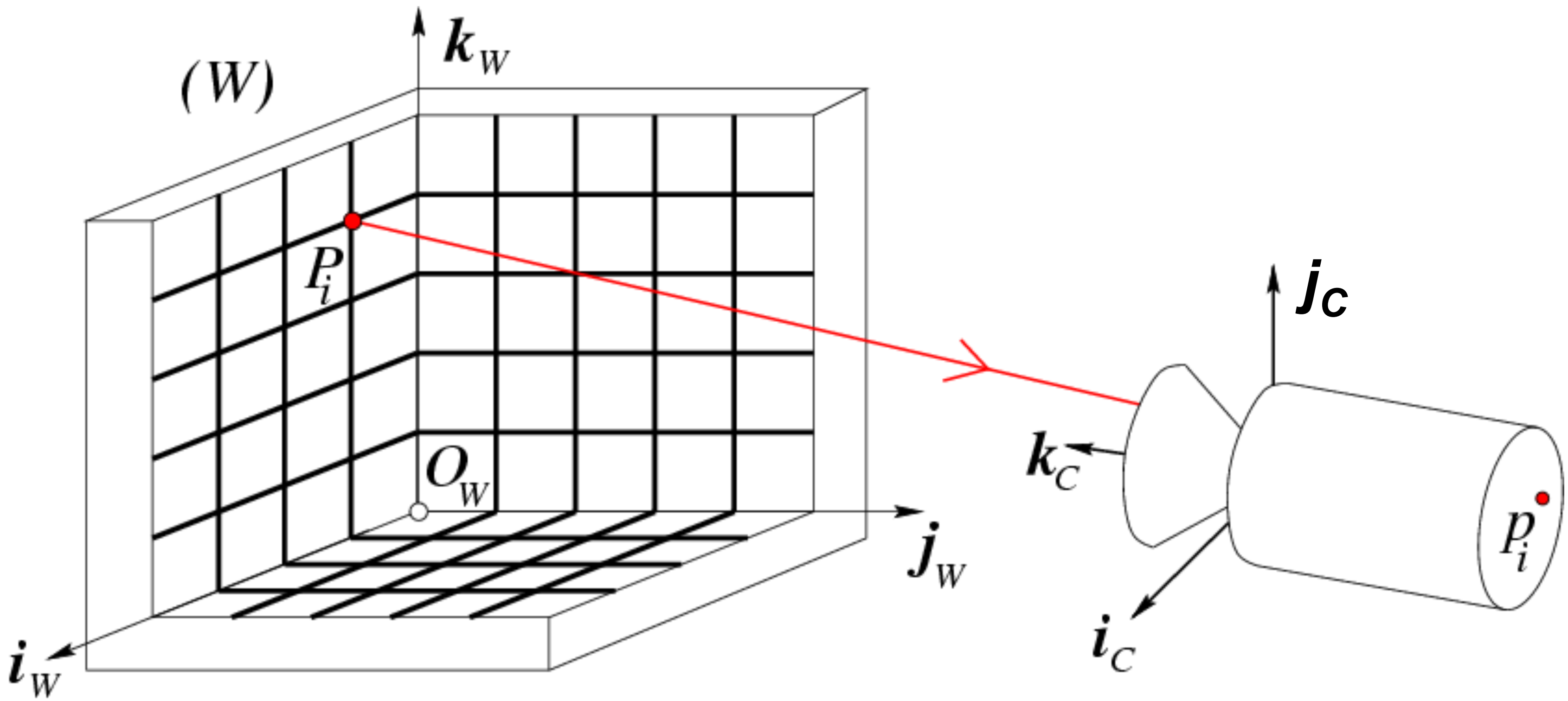
- M has 11 unknown
- We need 11 equations
- 6 correspondences would do it

Calibration Problem



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

Calibration Problem



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

in pixels

Calibration Problem

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

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

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

Calibration Problem

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

Block Matrix Multiplication

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

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

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

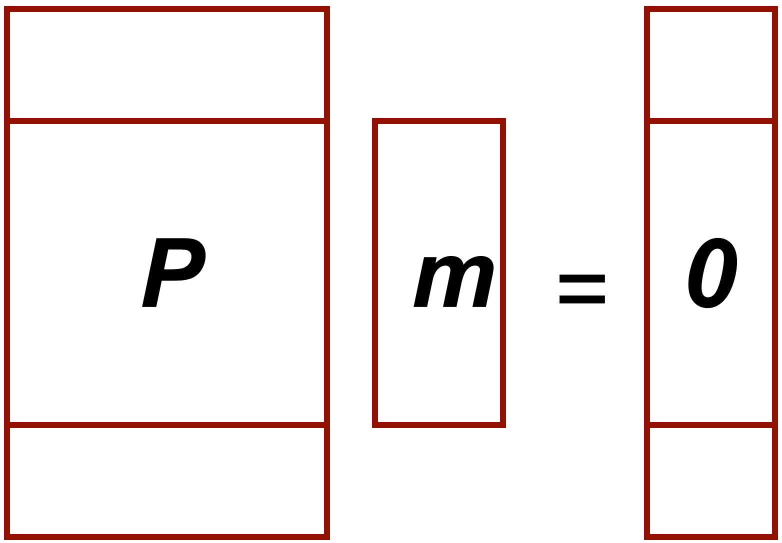
Homogenous linear system

$$\mathcal{P} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{P}_1^T & \mathbf{0}^T & -u_1 \overbrace{\boldsymbol{P}_1^T}^{1\text{x}4} \\ \mathbf{0}^T & \boldsymbol{P}_1^T & -v_1 \boldsymbol{P}_1^T \\ \cdots & \cdots & \cdots \\ \boldsymbol{P}_n^T & \mathbf{0}^T & -u_n \boldsymbol{P}_n^T \\ \mathbf{0}^T & \boldsymbol{P}_n^T & -v_n \boldsymbol{P}_n^T \end{pmatrix}_{2n \text{ x } 12}$$

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

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

$$\mathcal{P}\boldsymbol{m} = 0$$

- How do we solve this homogenous linear system?
- Using DLT (Direct Linear Transformation) algorithm via SVD decomposition

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

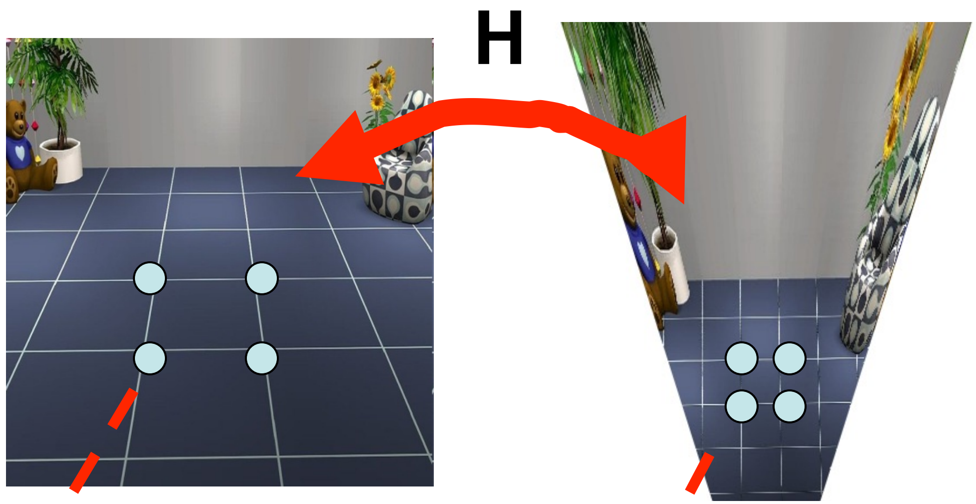
$$A = U\,\Sigma\,V^{-1} \qquad \Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \cdot & \\ & & & \sigma_N \end{bmatrix}$$

U, V = orthogonal matrix

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

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

DLT algorithm (Direct Linear Transformation)



$\mathbf{x}_i$ $\mathbf{x}'_i$

$$\mathbf{x}'_i = \mathbf{H}\,\mathbf{x}_i \longrightarrow \underbrace{\mathbf{A}_i}_{\text{Function of measurements}} \overbrace{\mathbf{h}}^{\text{unknown}} = 0$$

Calibration Problem

$$\boxed{\mathcal{P}}\boldsymbol{m} = 0$$

SVD decomposition of P

$$\boxed{U_{2n\times12}\ D_{12\times12}\ V^T_{12\times12}}$$

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

$$\downarrow$$

M

$M\,P_i \rightarrow p_i$

Why? See pag 593 of AZ

Clarification about SVM

•Thanks to Pat O' Keefe!

$$P_{m\times n} = \boxed{U}_{m\times n} \; D_{n\times n} \; \boxed{V}^{T}_{n\times n}$$

Has n orthogonal columns

Orthogonal matrix

- This is one of the possible SVD decompositions
- This is typically used for efficiency
- The classic SVD is actually:

$$P_{m\times n} = \boxed{U}_{m\times m} \; D_{\;m\times n} \; \boxed{V}^{T}_{n\times n}$$

orthogonal

Orthogonal

Extracting camera parameters

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

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

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

Estimated values

Intrinsic

$$\rho = \frac{\pm 1}{|\mathbf{a}_3|} \qquad \mathrm{u}_o = \rho^2(\mathbf{a}_1 \cdot \mathbf{a}_2)$$

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

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

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 $\mathcal{M}$.

- A necessary and sufficient condition for $\mathcal{M}$ to be a perspective projection matrix is that $\mathrm{Det}(\mathcal{A}) \neq 0$.
- A necessary and sufficient condition for $\mathcal{M}$ to be a zero-skew perspective projection matrix is that $\mathrm{Det}(\mathcal{A}) \neq 0$ and

$$(\boldsymbol{a}_1 \times \boldsymbol{a}_3) \cdot (\boldsymbol{a}_2 \times \boldsymbol{a}_3) = 0.$$

- A necessary and sufficient condition for $\mathcal{M}$ to be a perspective projection matrix with zero skew and unit aspect-ratio is that $\mathrm{Det}(\mathcal{A}) \neq 0$ and

$$\begin{cases} (\boldsymbol{a}_1 \times \boldsymbol{a}_3) \cdot (\boldsymbol{a}_2 \times \boldsymbol{a}_3) = 0, \\ (\boldsymbol{a}_1 \times \boldsymbol{a}_3) \cdot (\boldsymbol{a}_1 \times \boldsymbol{a}_3) = (\boldsymbol{a}_2 \times \boldsymbol{a}_3) \cdot (\boldsymbol{a}_2 \times \boldsymbol{a}_3). \end{cases}$$

Extracting camera parameters

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

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

Estimated values

Intrinsic

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

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

$\rightarrow$ f

Extracting camera parameters

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

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

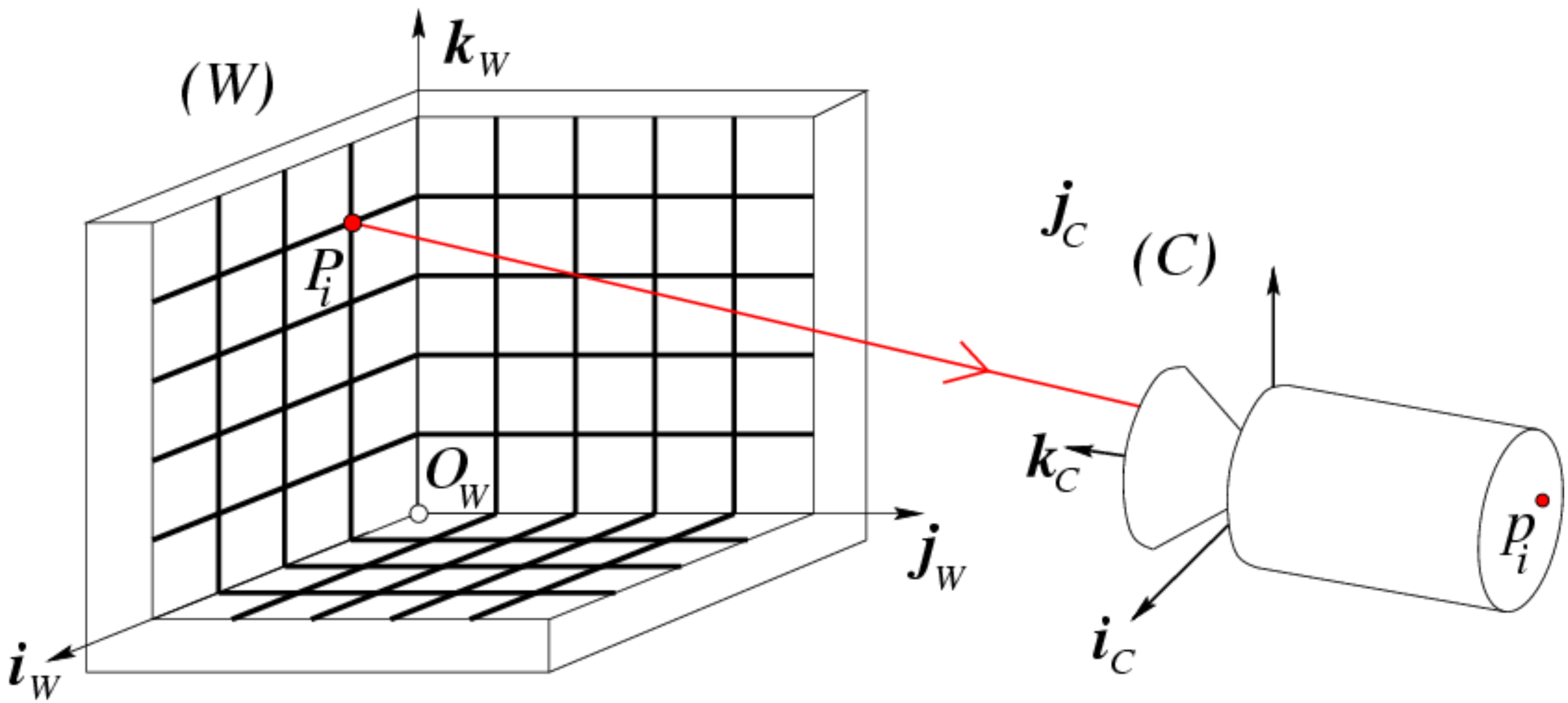
Estimated values

Extrinsic

$$\mathbf{r}_1 = \frac{(\mathbf{a}_2 \times \mathbf{a}_3)}{|\mathbf{a}_2 \times \mathbf{a}_3|} \qquad \mathbf{r}_3 = \frac{\pm 1}{|\mathbf{a}_3|}$$

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

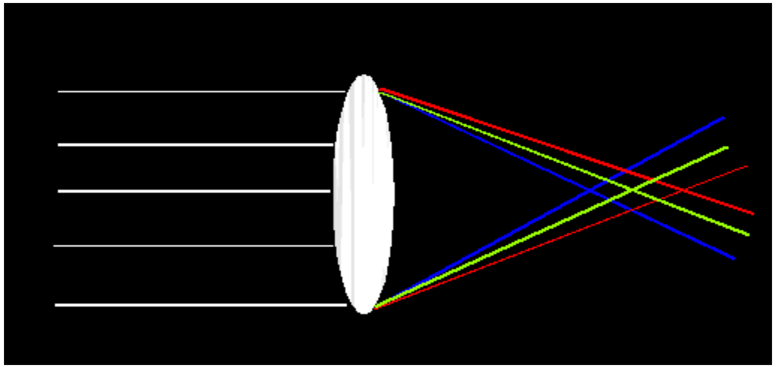
Degenerate cases

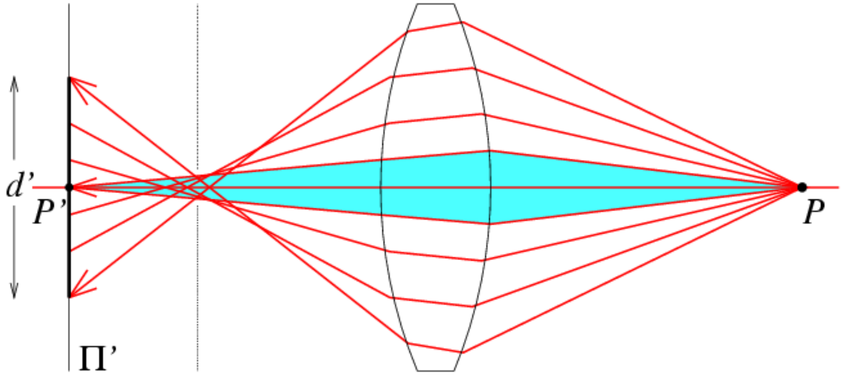


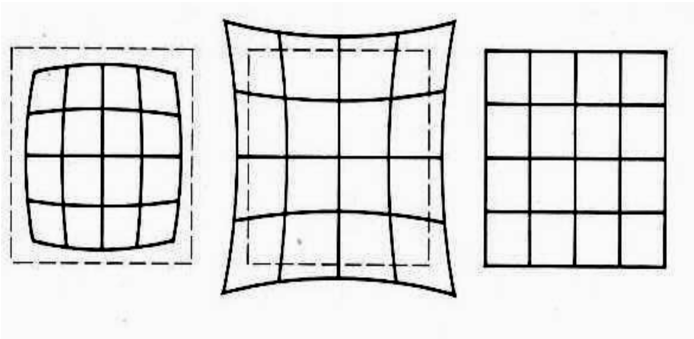
- P_i's cannot lie on the same plane!
- Points cannot lie on the intersection curve of two quadric surfaces

Taking lens distortions into account

- Chromatic Aberration
- Spherical aberration
- Radial Distortion

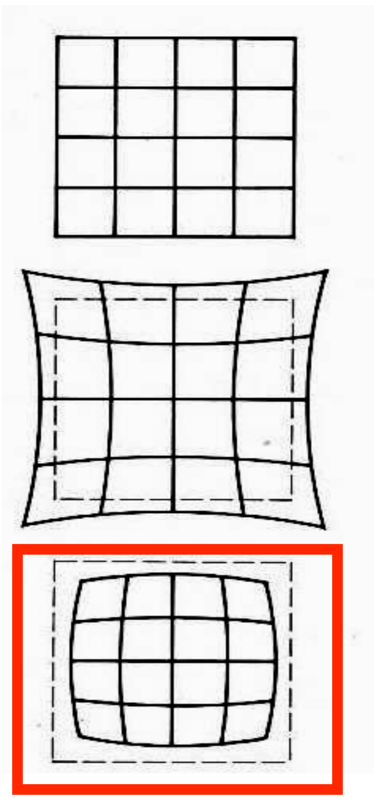






Radial Distortion

– Caused by imperfect lenses
– Deviations are most noticeable for rays that pass through the edge of the lens

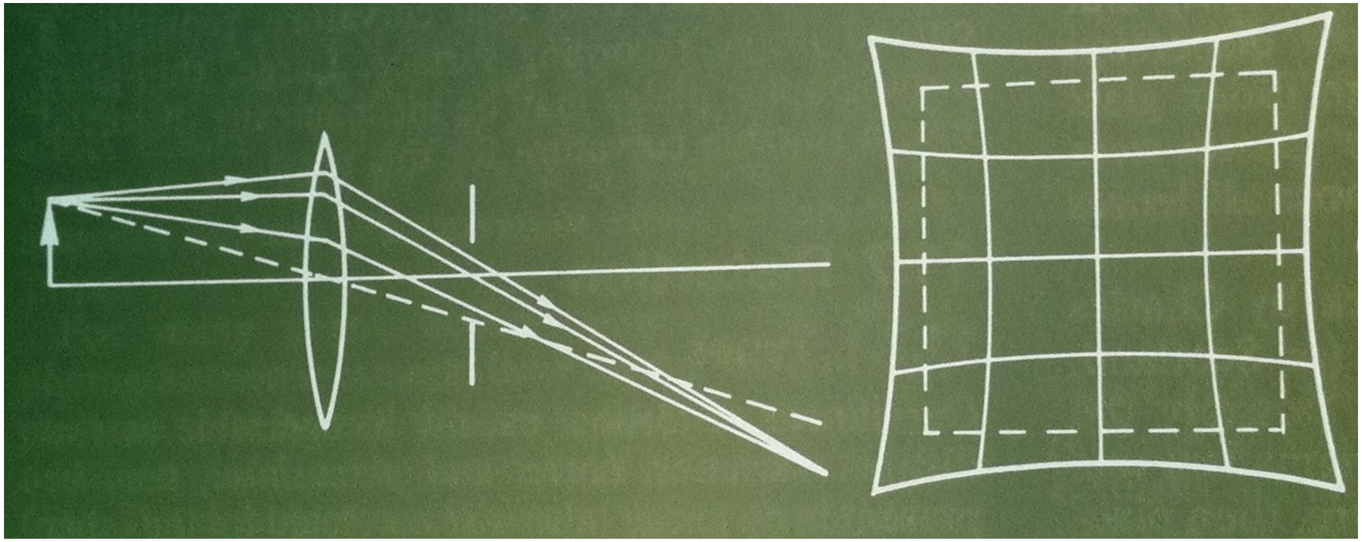


No distortion

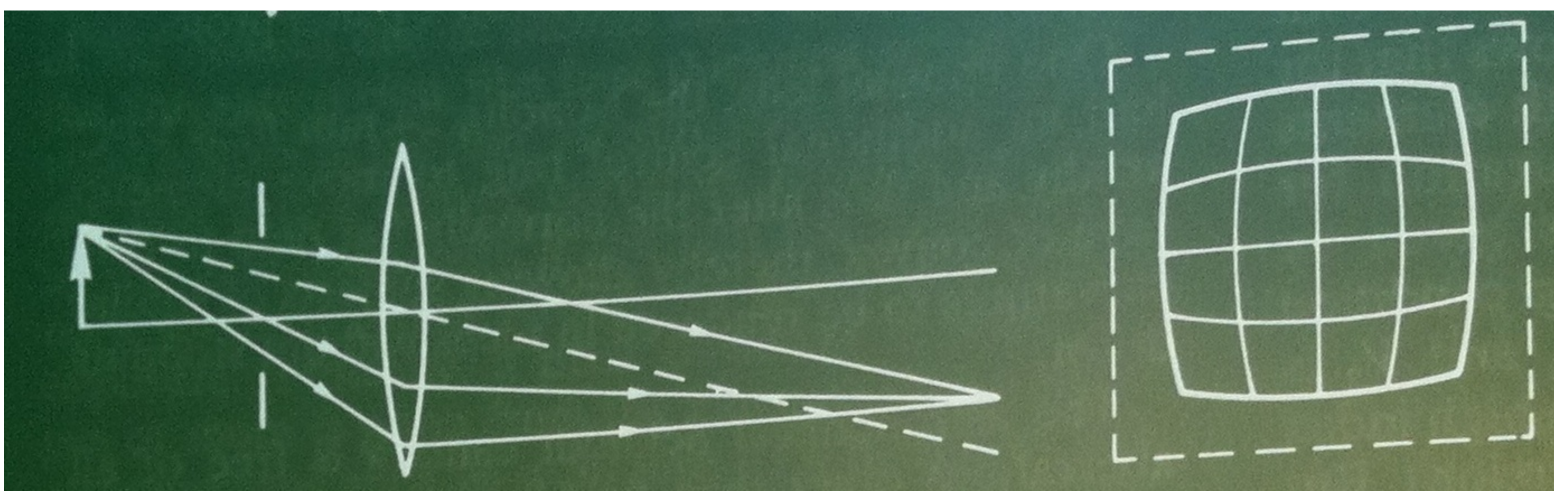
Pin cushion

Barrel

Issues with lenses: Radial Distortion

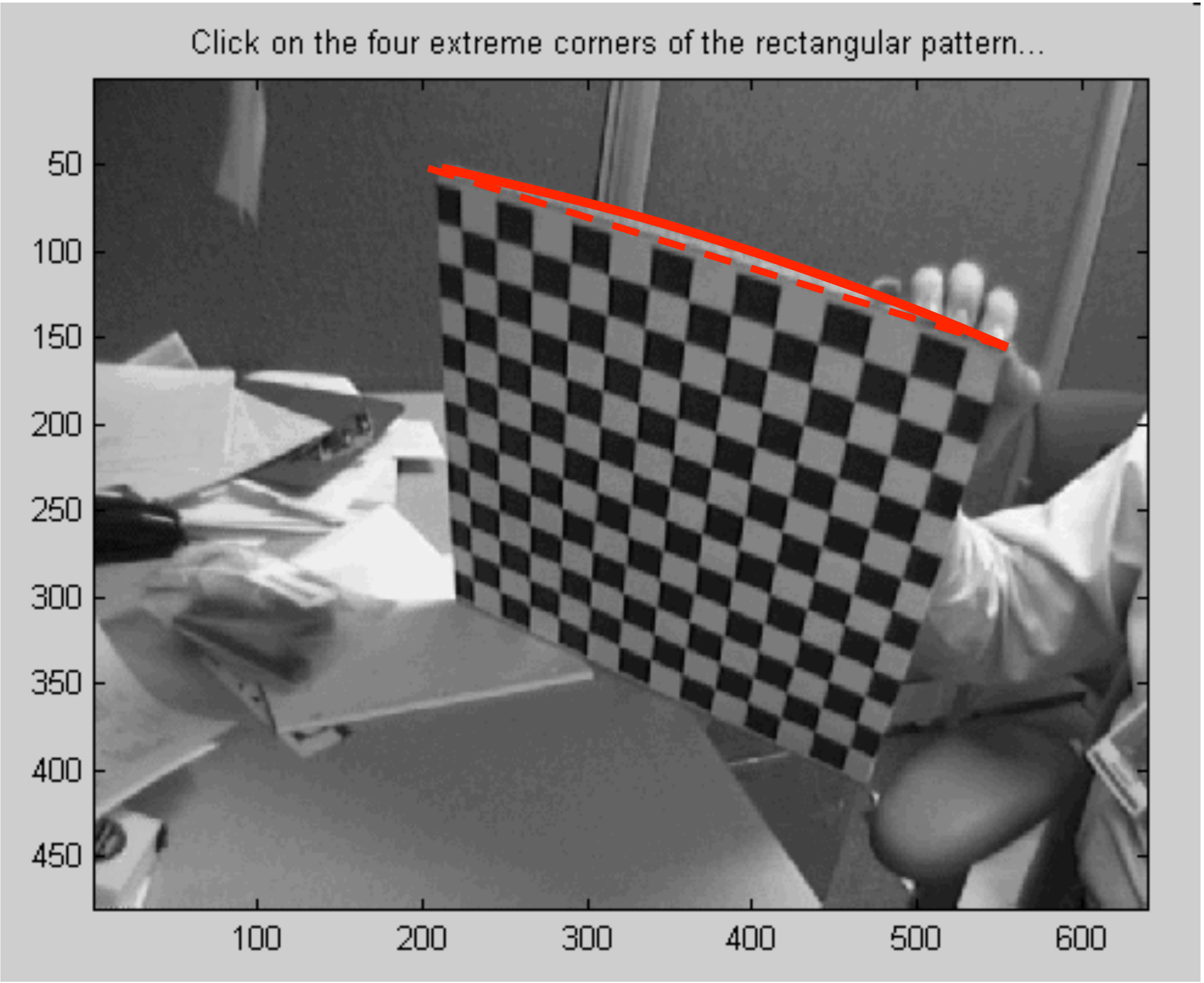


Pin cushion



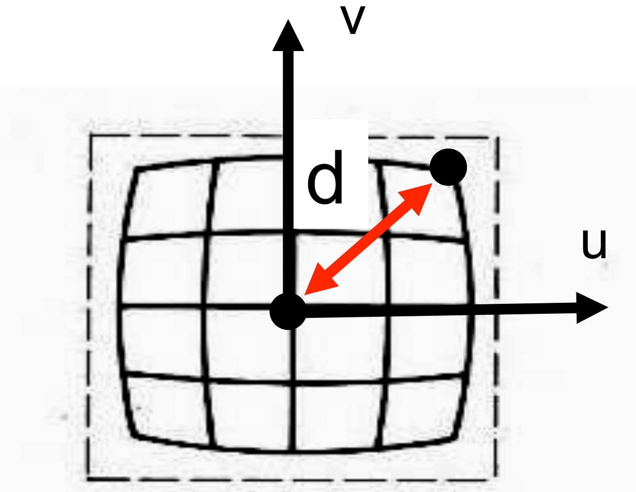
Barrel (fisheye lens)

Radial Distortion



Radial Distortion

Image magnification in(de)creases with distance from the optical center



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

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

To model radial behavior

Distortion coefficient

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

Radial Distortion

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

This is not linear system of equations

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

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \mathrm{P}_\mathrm{i}}{\mathbf{q}_3 \mathrm{P}_\mathrm{i}} \\ \dfrac{\mathbf{q}_2 \mathrm{P}_\mathrm{i}}{\mathbf{q}_3 \mathrm{P}_\mathrm{i}} \end{bmatrix} \longrightarrow X = f(P)$$

measurement

parameter

f( ) is nonlinear

-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
- Newton requires the computation of J, H
- Levenberg-Marquardt doesn't require the computation of H

General Calibration Problem

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \, \mathrm{P}_\mathrm{i}}{\mathbf{q}_3 \, \mathrm{P}_\mathrm{i}} \\ \dfrac{\mathbf{q}_2 \, \mathrm{P}_\mathrm{i}}{\mathbf{q}_3 \, \mathrm{P}_\mathrm{i}} \end{bmatrix} \longrightarrow X = f(P)$$

measurement

parameter

f( ) is nonlinear

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

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 \, \mathrm{P_i}}{\mathbf{q}_3 \, \mathrm{P_i}} \\ \dfrac{\mathbf{q}_2 \, \mathrm{P_i}}{\mathbf{q}_3 \, \mathrm{P_i}} \end{bmatrix} \longrightarrow X = f(P)$$

measurement

parameter

f( ) is nonlinear

Typical assumptions:

- zero-skew, square pixel
- u_o, v_o = known center of the image
- no distortion

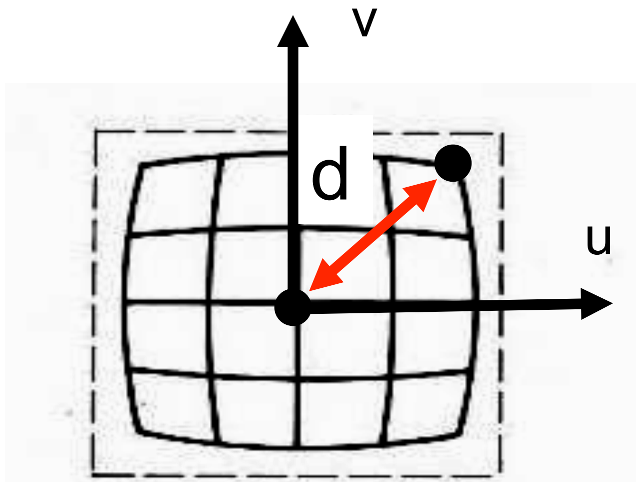
$\longrightarrow$ Just estimate f and R, T

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 estimate m_1 and m_2 and ignore the radial distortion?

Hint:



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

Radial Distortion

Estimating m_1 and m_2…

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

$$\frac{u_i}{v_i} = \frac{\dfrac{(\mathbf{m}_1\ P_i)}{(\mathbf{m}_3\ P_i)}}{\dfrac{(\mathbf{m}_2\ P_i\ )}{(\mathbf{m}_3\ P_i)}} = \frac{\mathbf{m}_1\ P_i}{\mathbf{m}_2\ P_i}$$

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

$$\mathrm{Q}\ \mathbf{n} = 0 \qquad \mathbf{n} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}$$

Tsai technique [87]

Radial Distortion

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

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

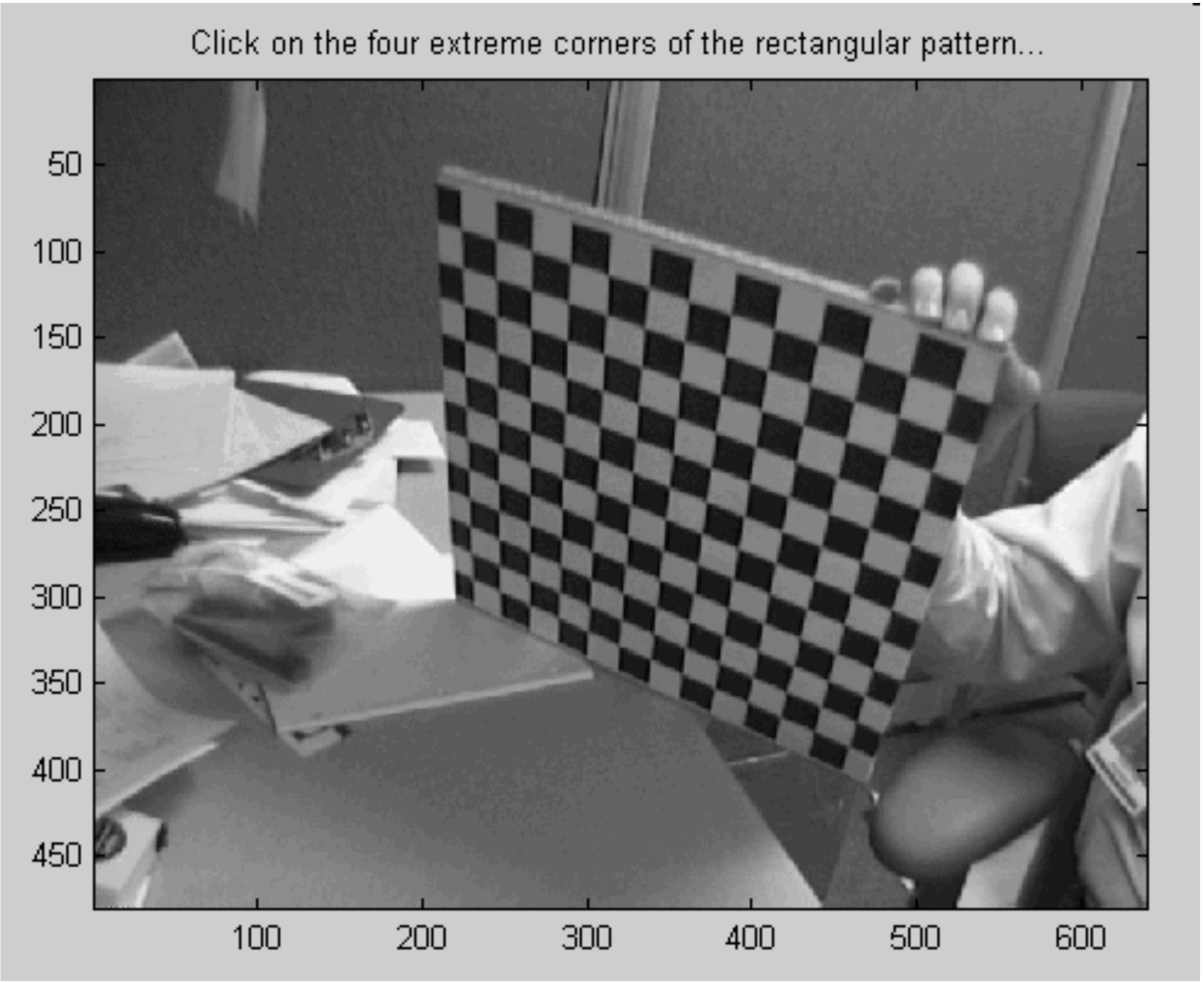
$\mathbf{m}_3$ is non linear function of $\mathbf{m}_1$, $\mathbf{m}_2$, λ

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

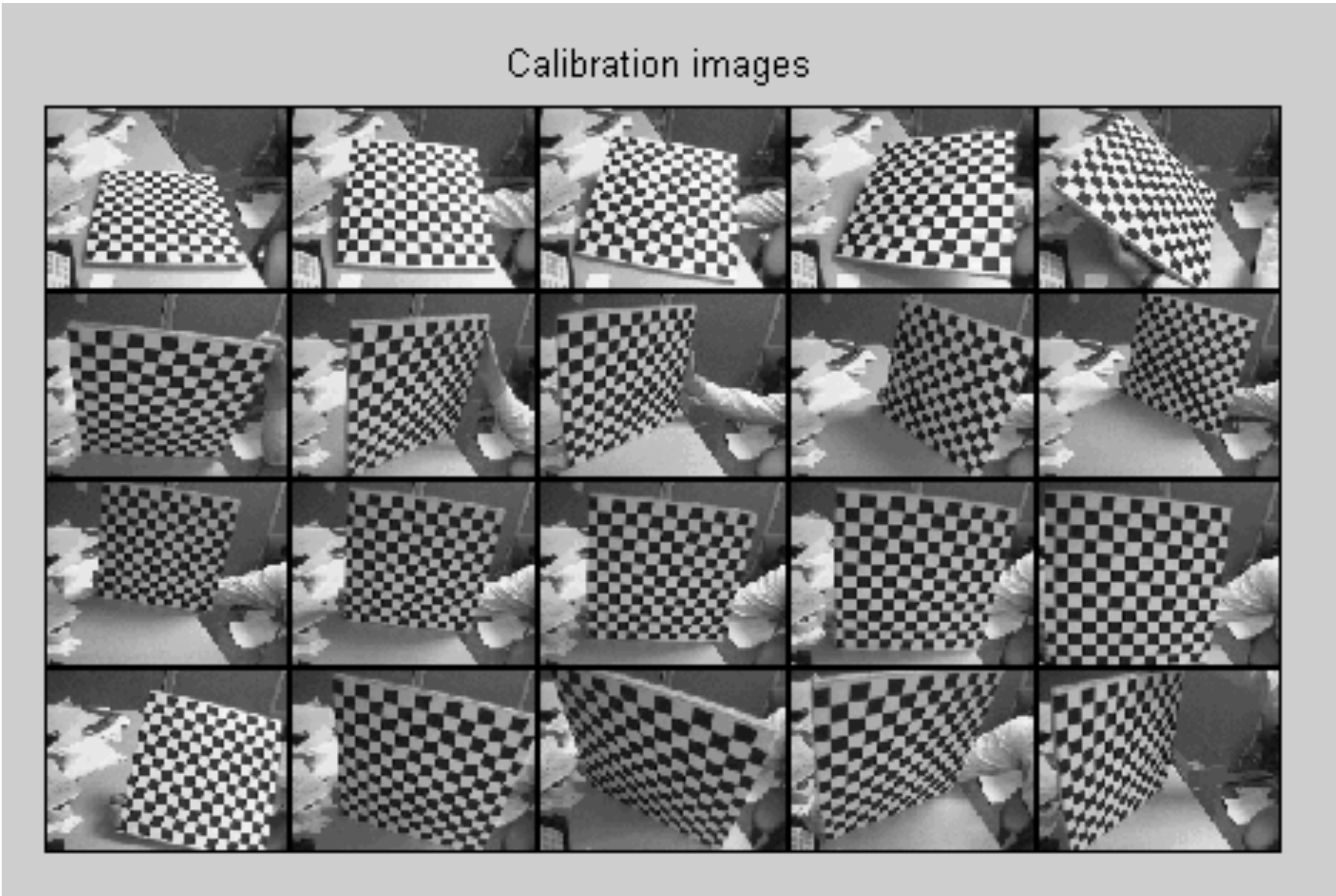
Calibration Procedure

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

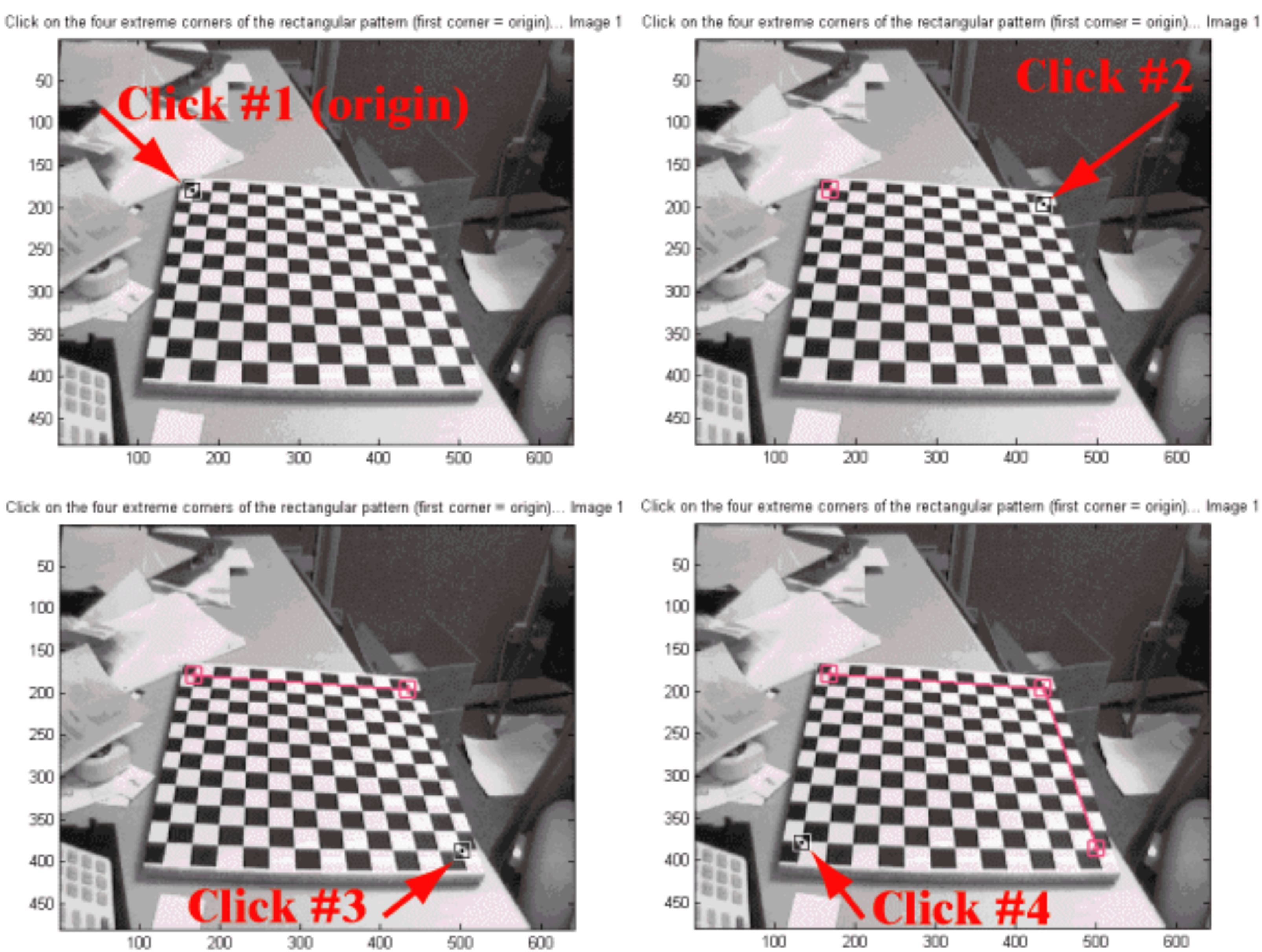
http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#examples



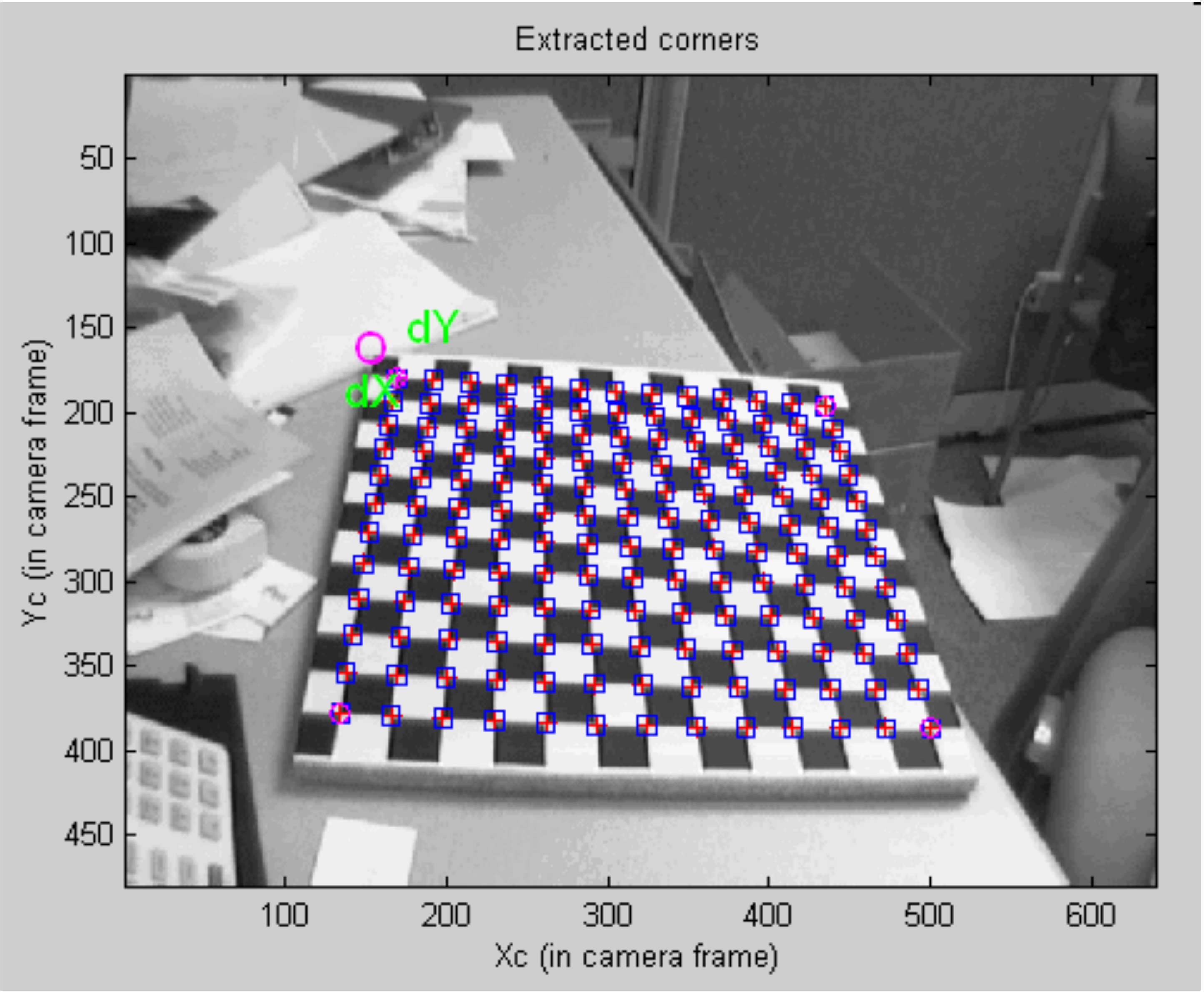
Calibration Procedure



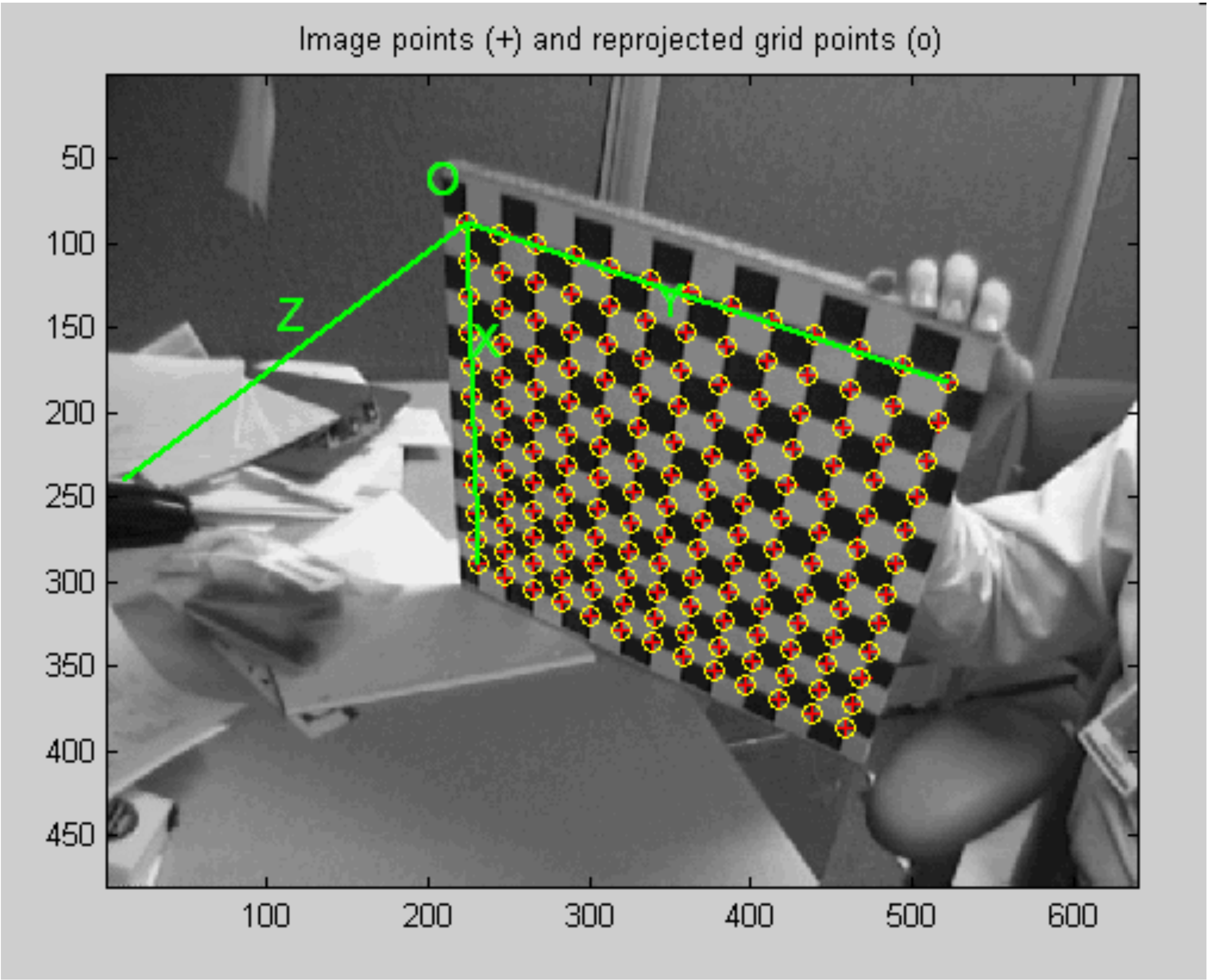
Calibration Procedure



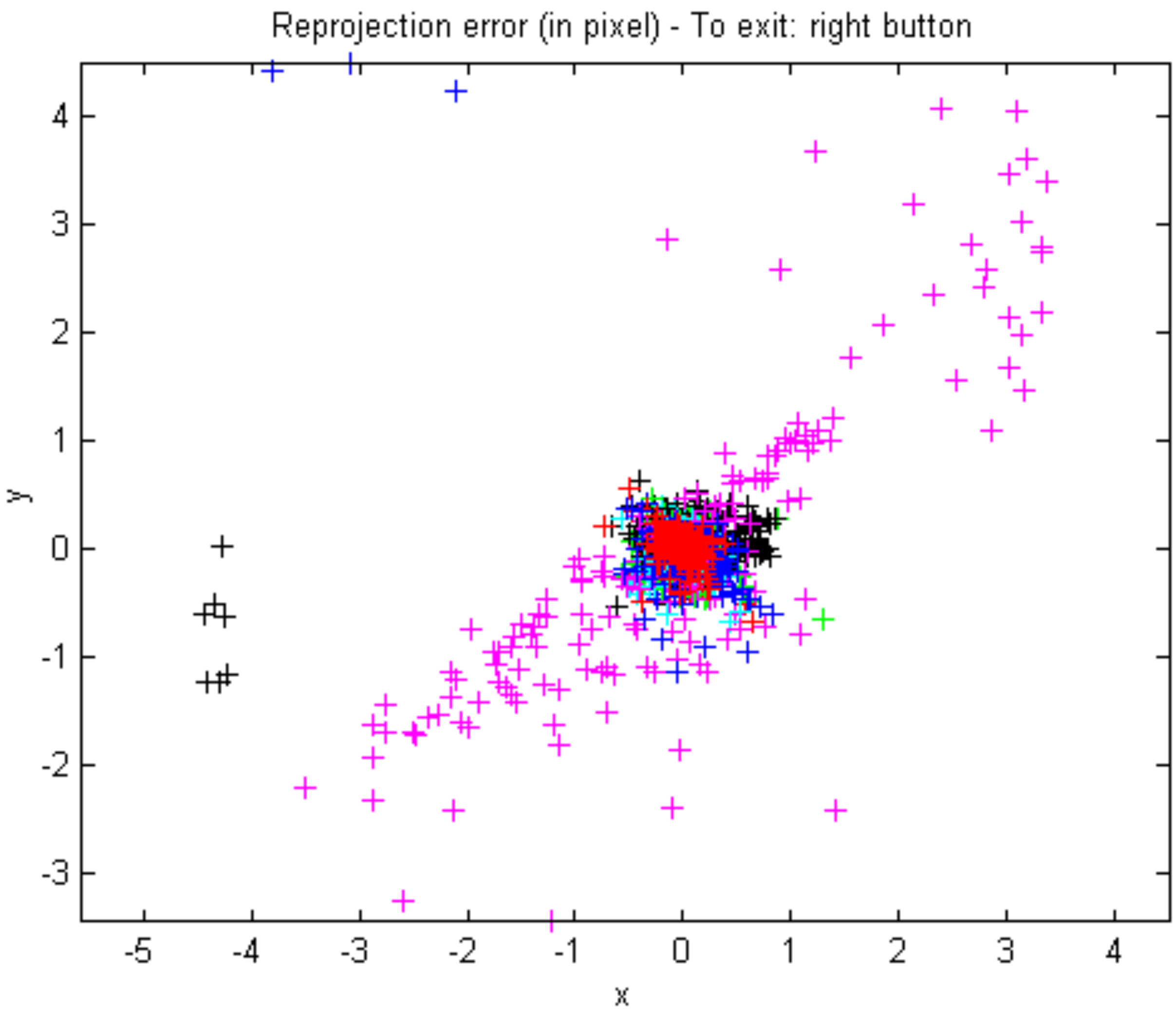
Calibration Procedure



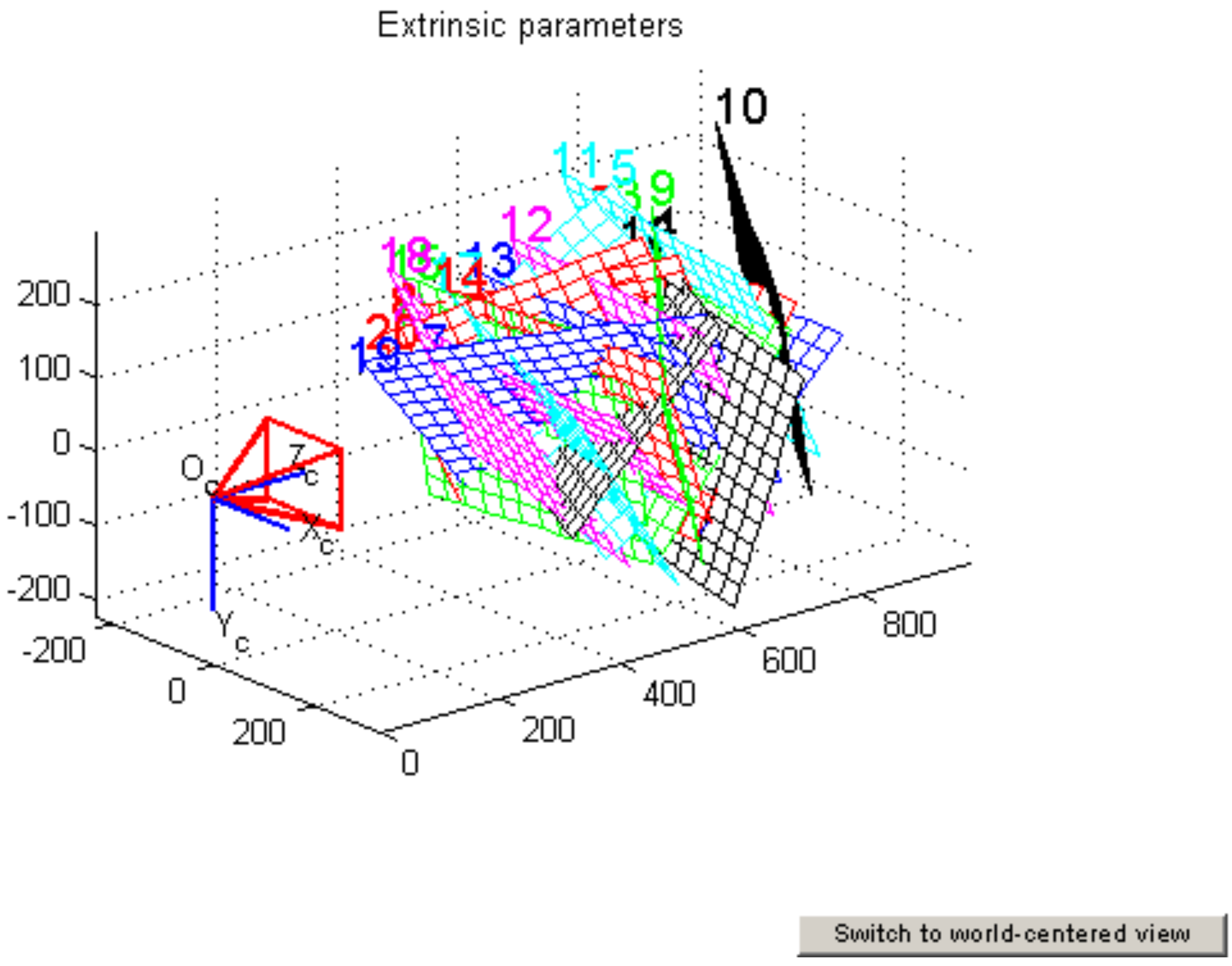
Calibration Procedure



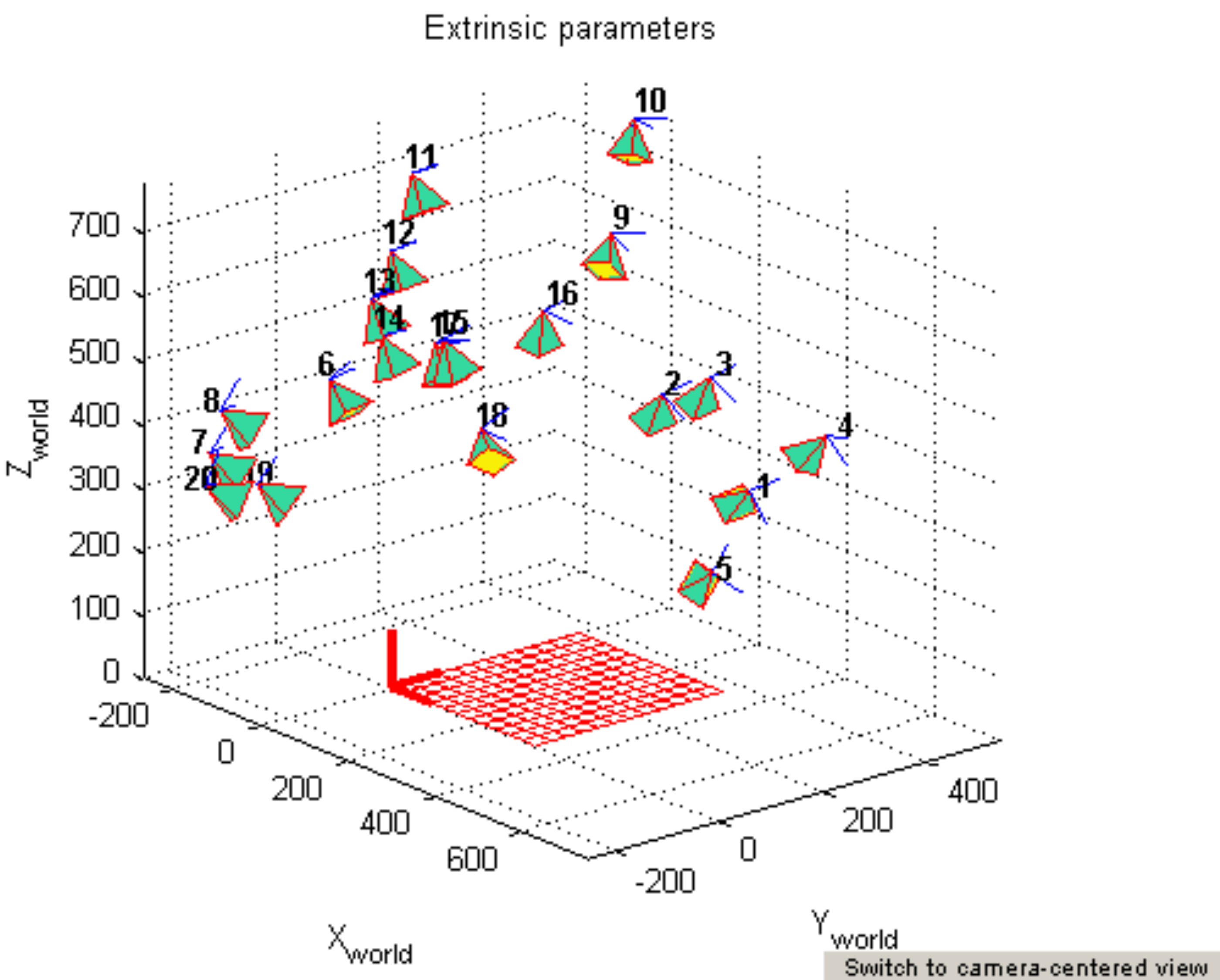
Calibration Procedure



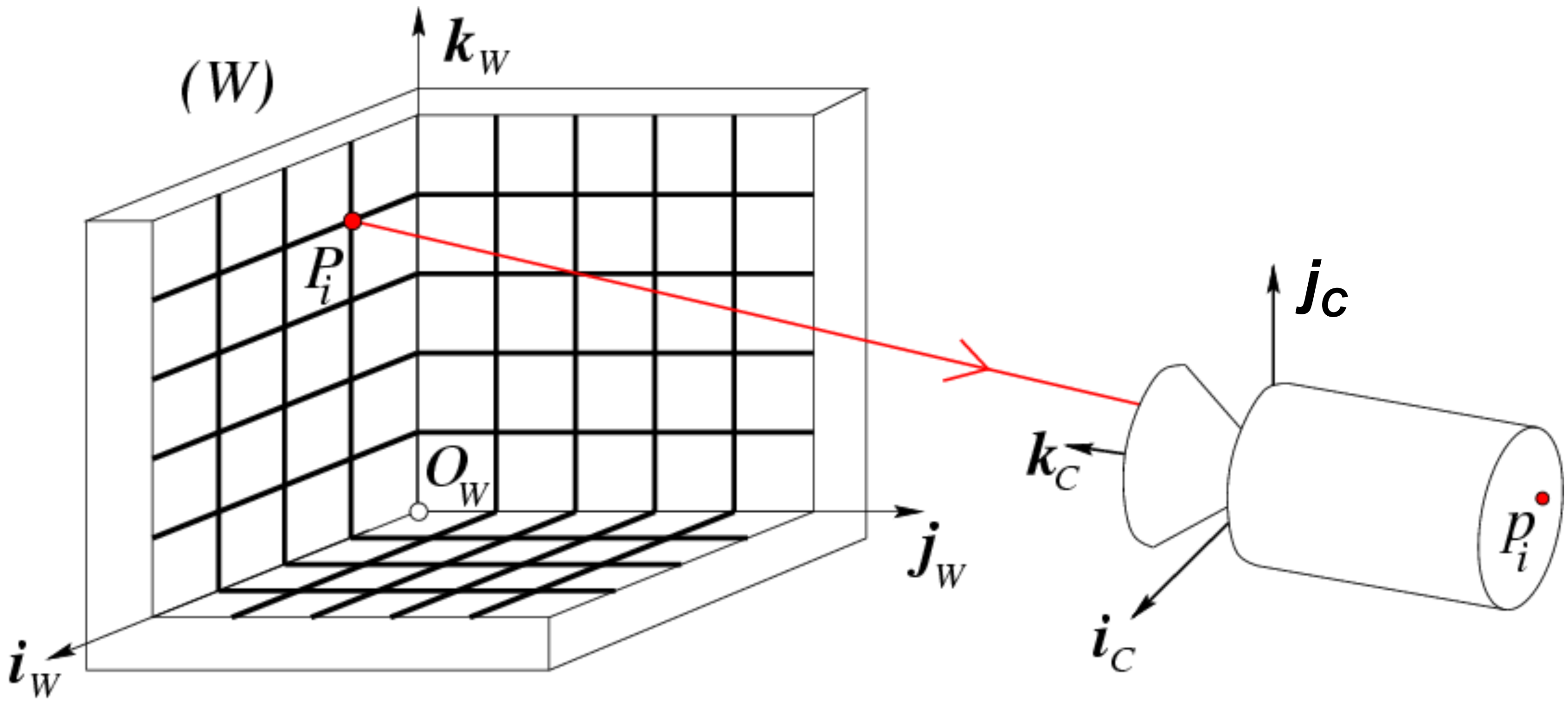
Calibration Procedure



Calibration Procedure



Calibration Problem



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

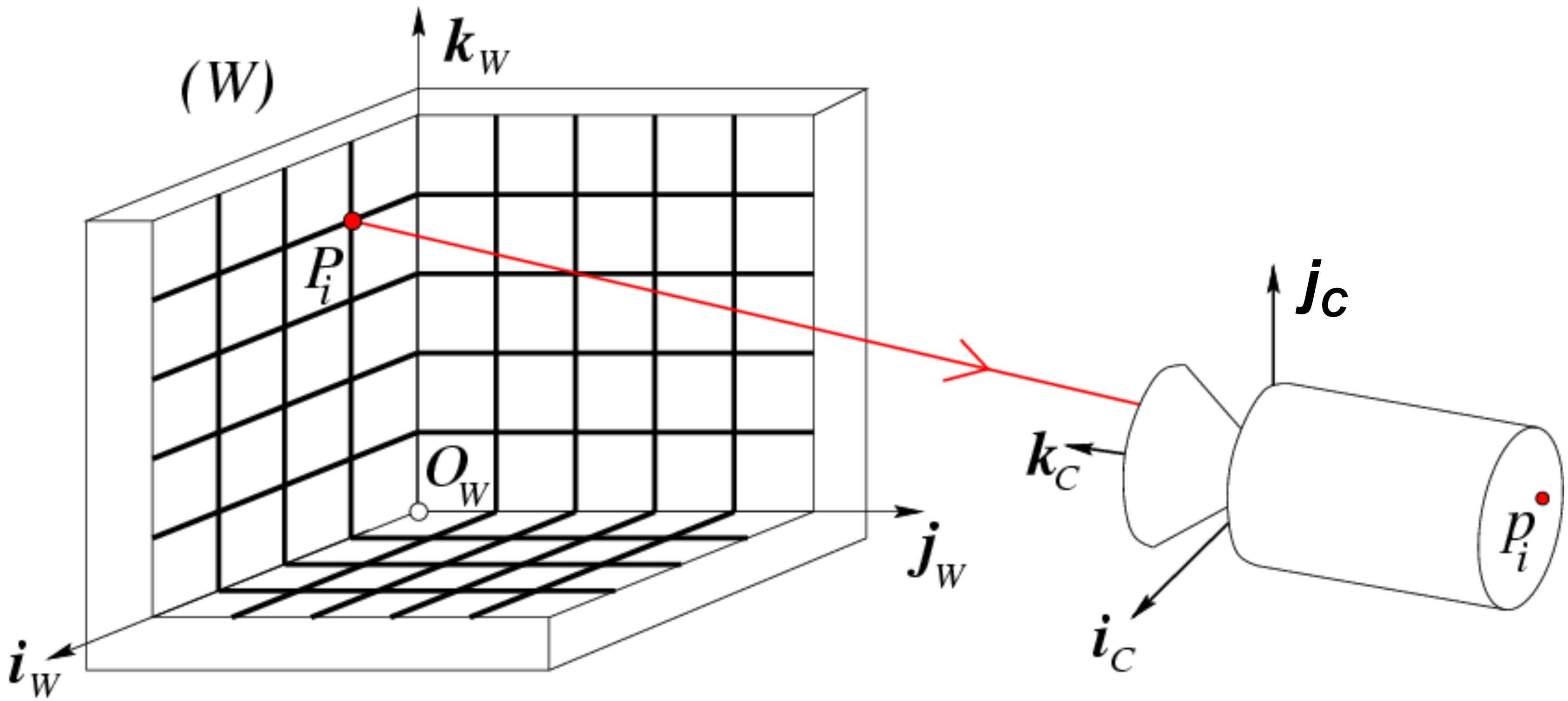
$$P_i \rightarrow M\,P_i \rightarrow p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

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

World ref. system

In pixels

Calibration Problem



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

$$P_i \rightarrow M\,P_i \rightarrow p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix}$$

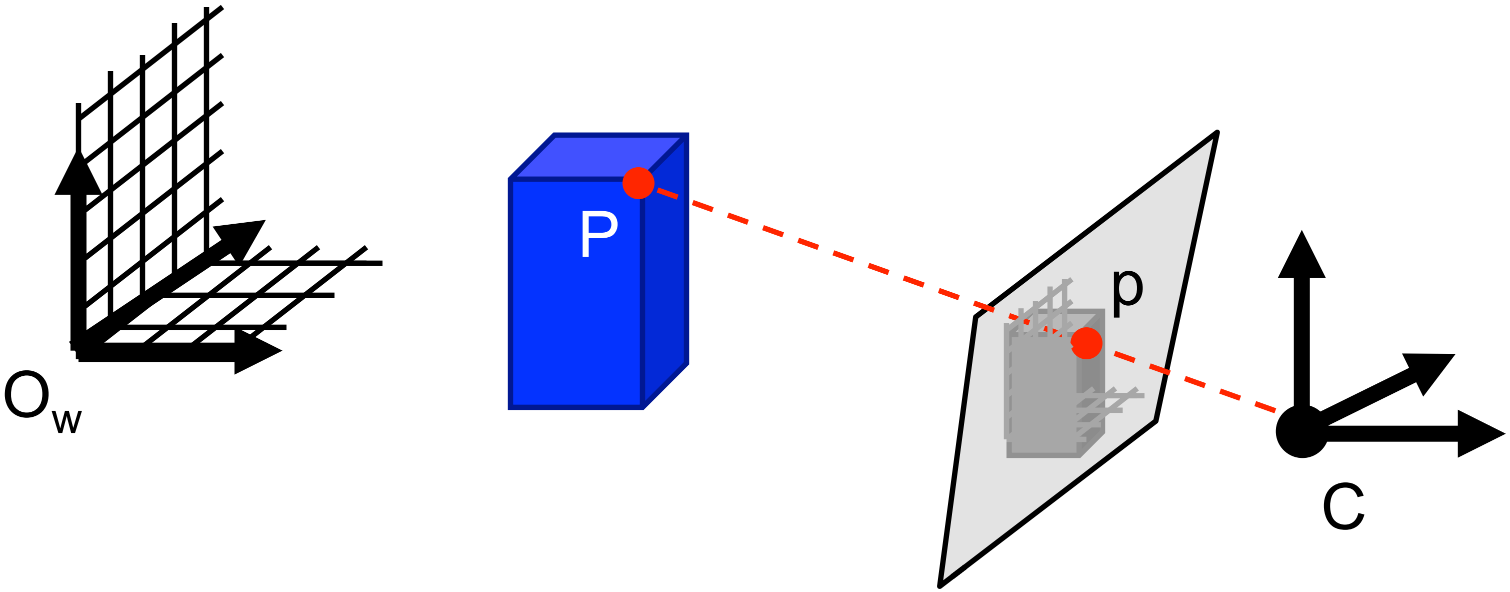
World ref. system

In pixels

11 unknown

Need at least 6 correspondences

Once the camera is calibrated...



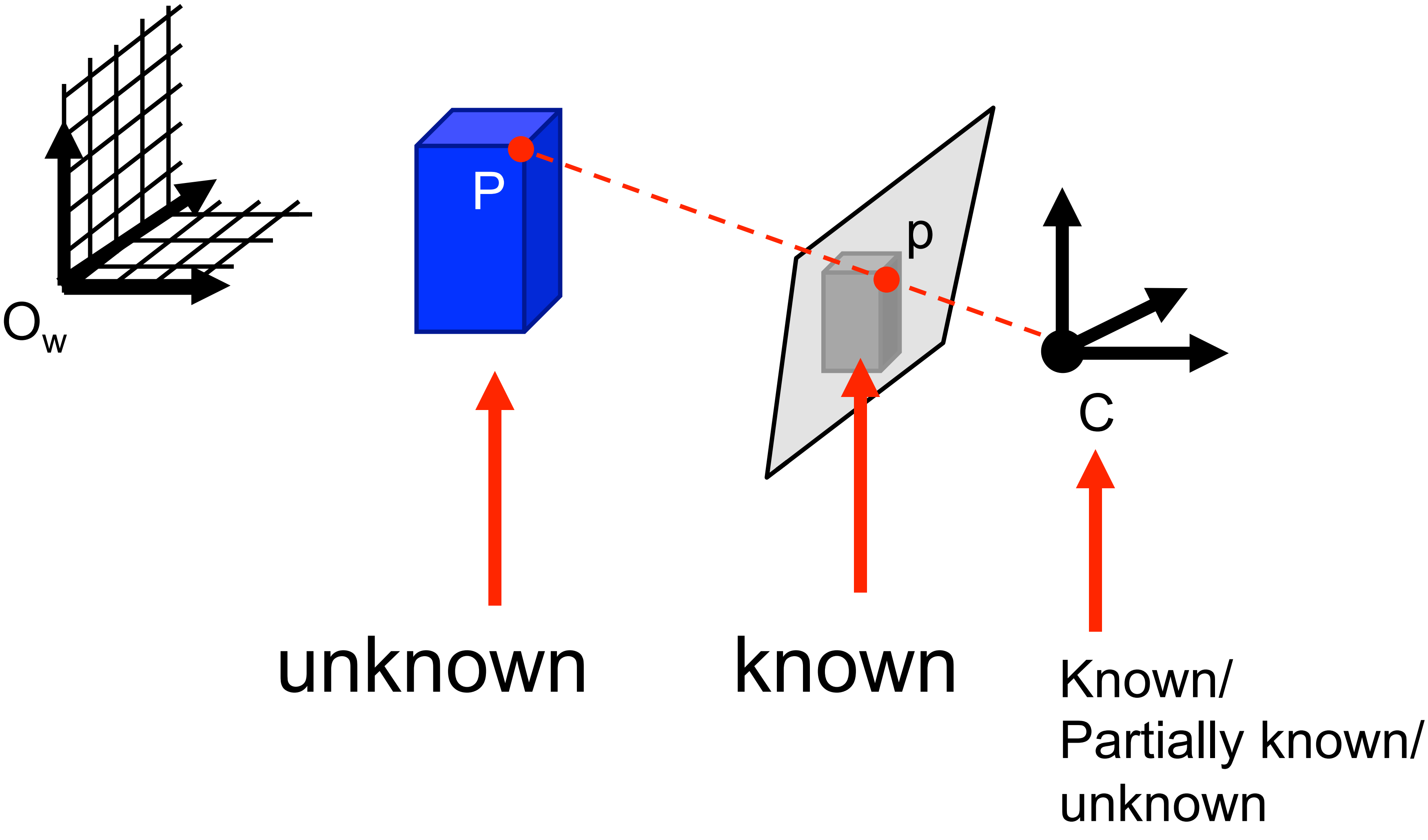
$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

-Internal parameters K are known
-R, T are known – but these can only relate C to the calibration rig

Can I estimate P from the measurement p from a single image?

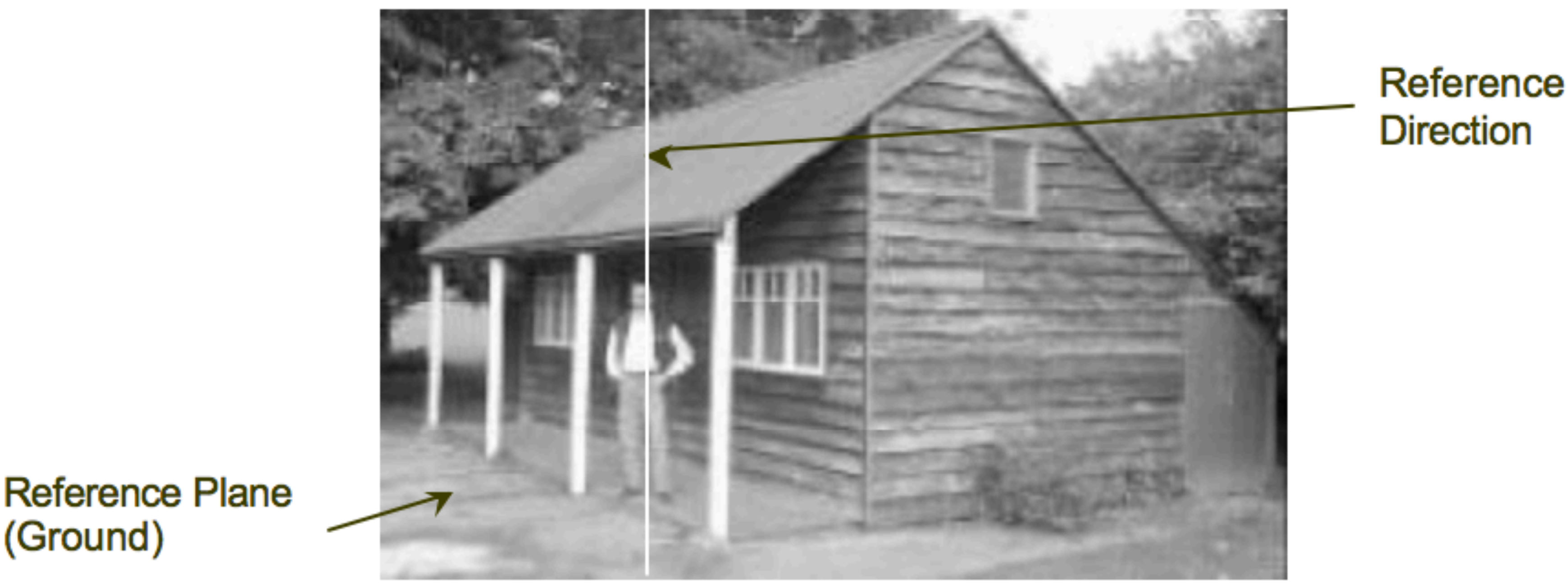
No - in general ☹ [P can be anywhere along the line defined by C and p]

Recovering structure from a single view

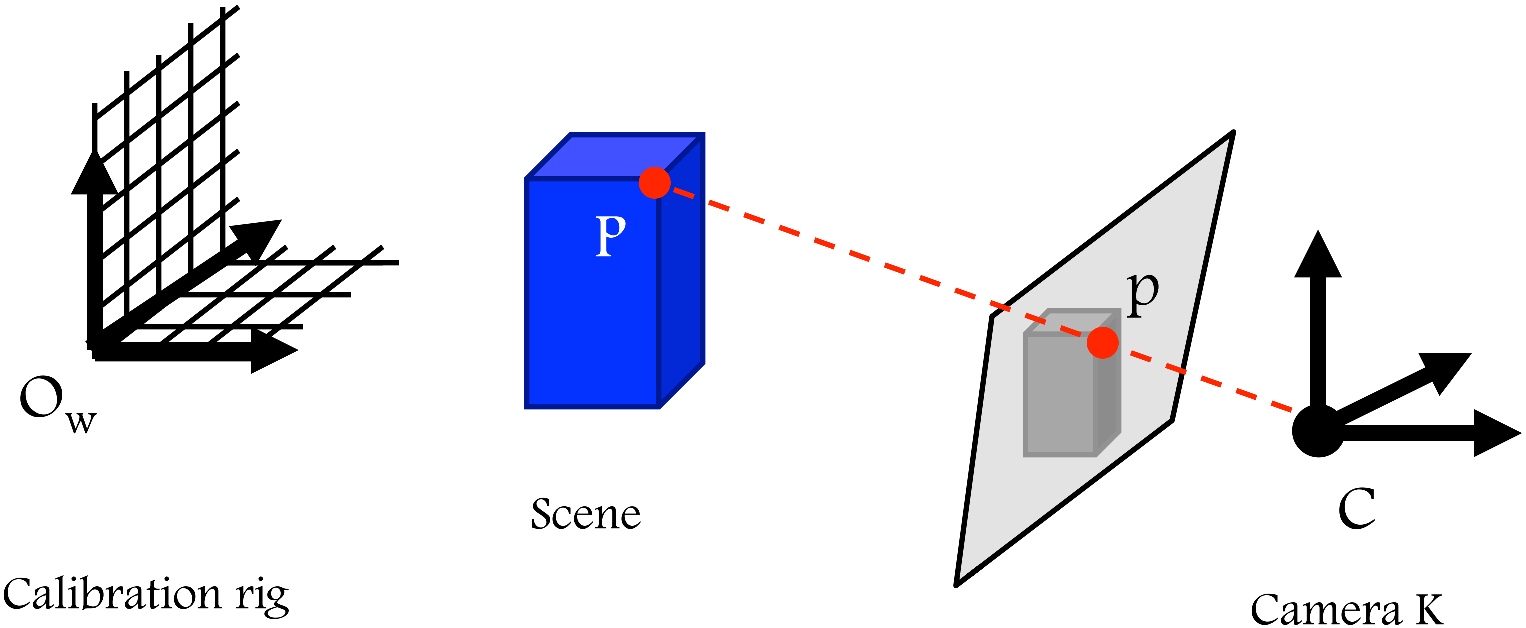


http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/hut/hutme.wrl

- Pick a reference plane in the scene
- Pick a reference direction
- Make an assumption about something in the scene

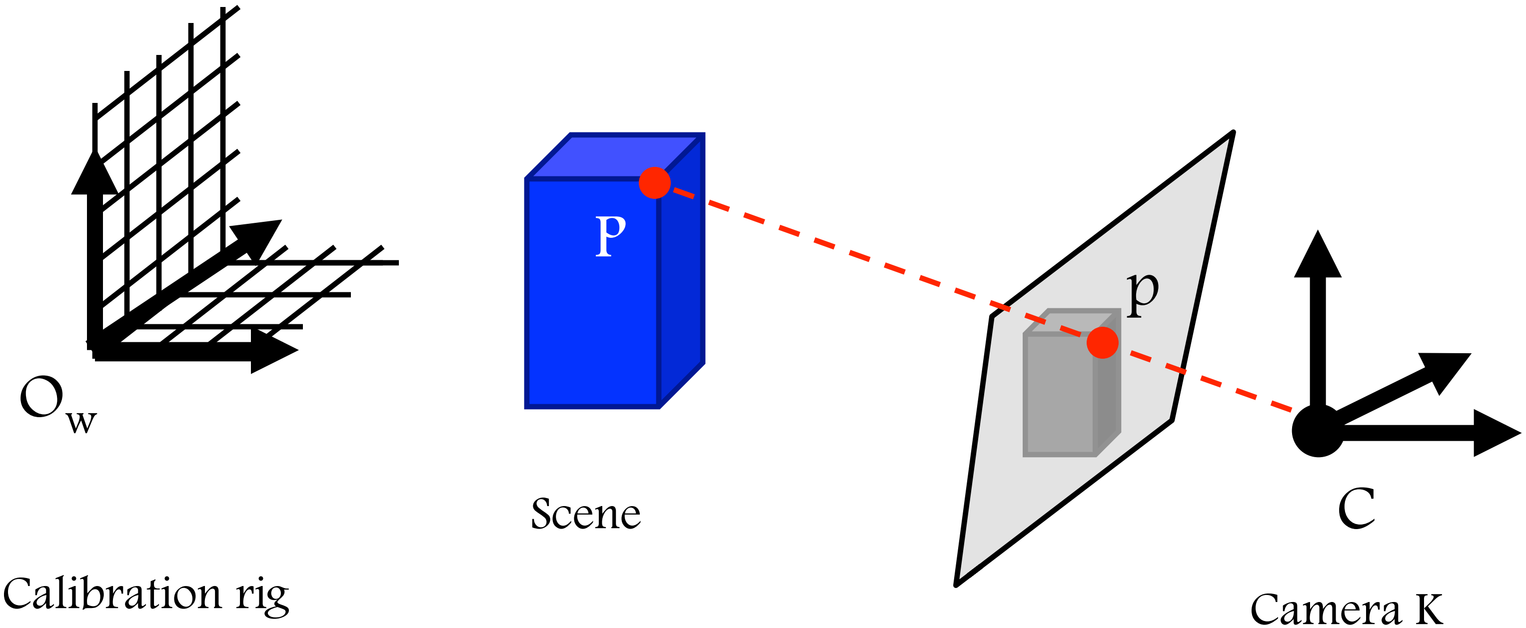


Recovering structure from a single view



From calibration rig	→ location/pose of the rig, K
From points and lines at infinity + orthogonal lines and planes	→ structure of the scene, K
Knowledge about scene (point correspondences, geometry of lines & planes, etc…	

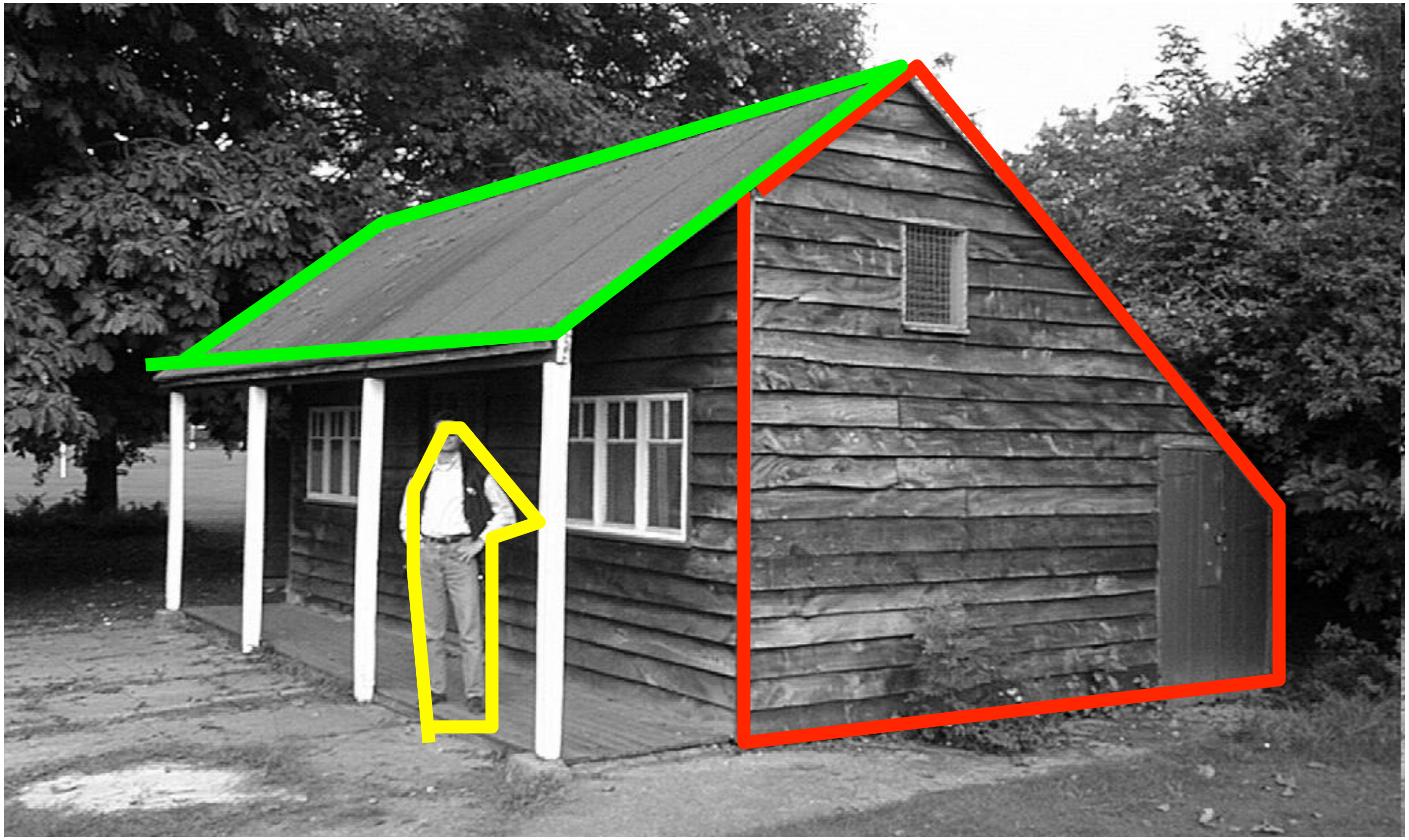
Recovering structure from a single view



Why is it so difficult?

Intrinsic ambiguity of the mapping from 3D to image (2D)

Single view reconstruction - drawbacks



Manually select:

- Vanishing points and lines;
- Planar surfaces;
- Occluding boundaries;
- Etc..

Automatic Photo Pop-up

Hoiem et al, 05

Automatic Photo Pop-up

Hoiem et al, 05…

Next Lecture Epipolar Geometry