TU Delft 3Dgeoinfo

GEO1016
Photogrammetry and 3D Computer Vision

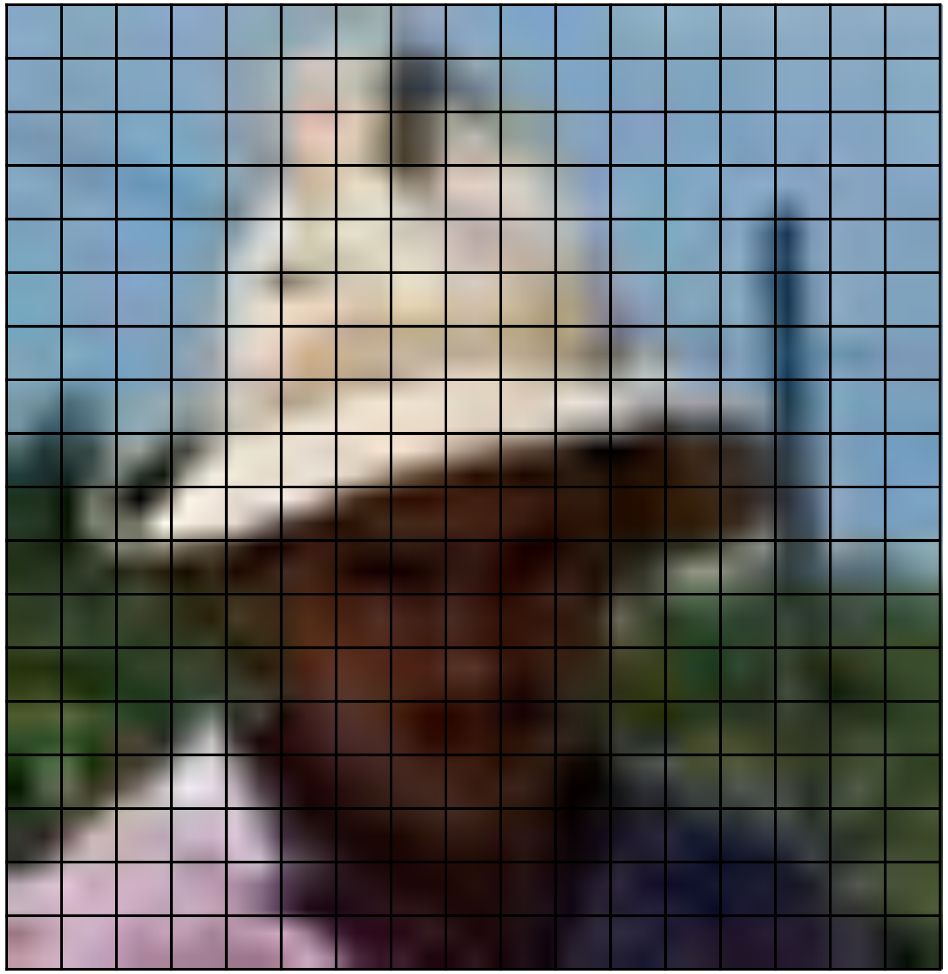
Lecture 3
Camera Calibration

Liangliang Nan

Today's Agenda

- Review of Lecture 2: Camera models
- Camera calibration
  - General idea
  - Calibration
    - Find correspondences
    - Solve for projection matrix
    - Extract the parameters
- Assignment 1: Camera calibration

Images

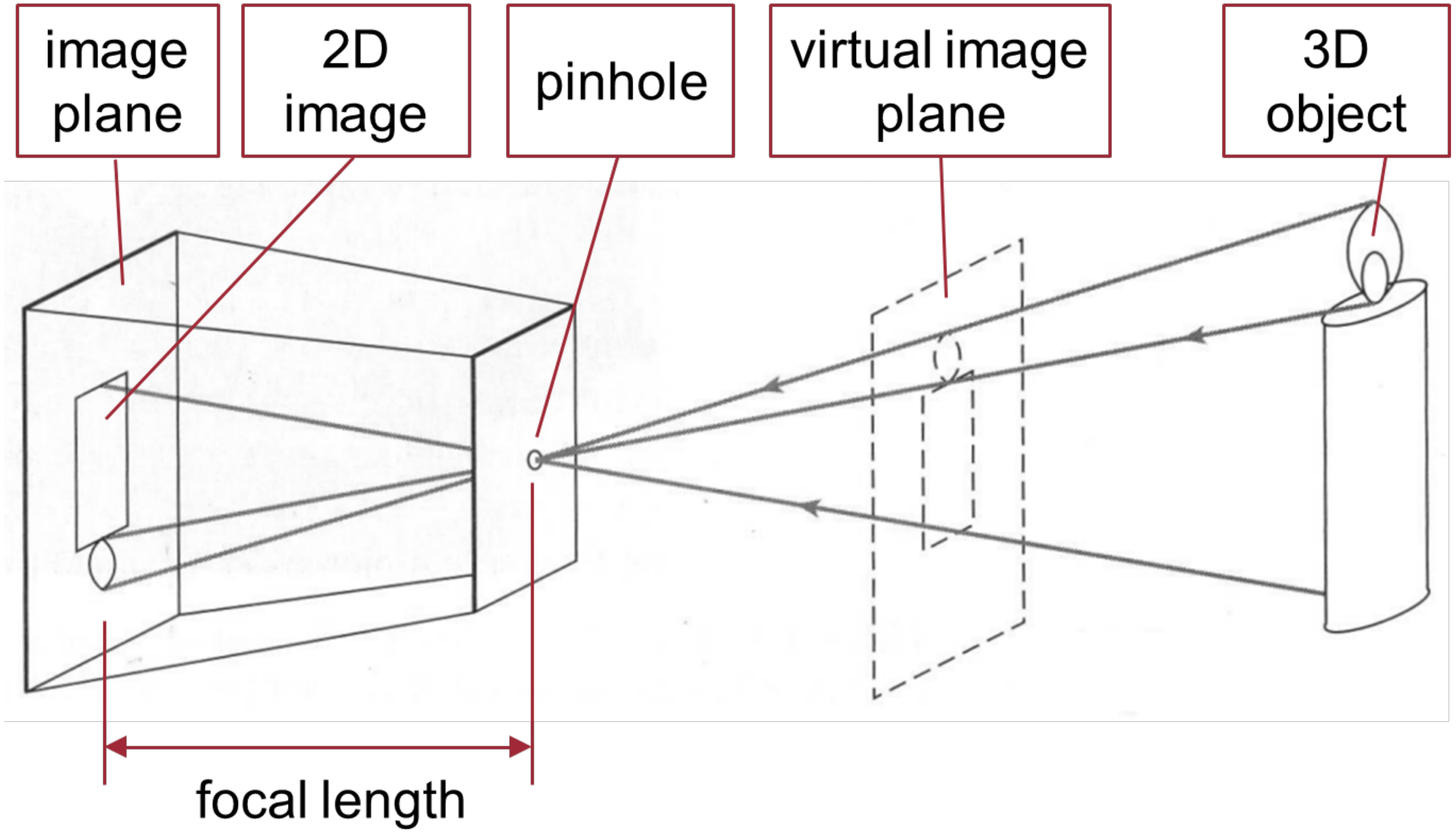


A color image: R, G, B channels

$$f(x,y) = \begin{bmatrix} r(x,y) \\ g(x,y) \\ b(x,y) \end{bmatrix}$$

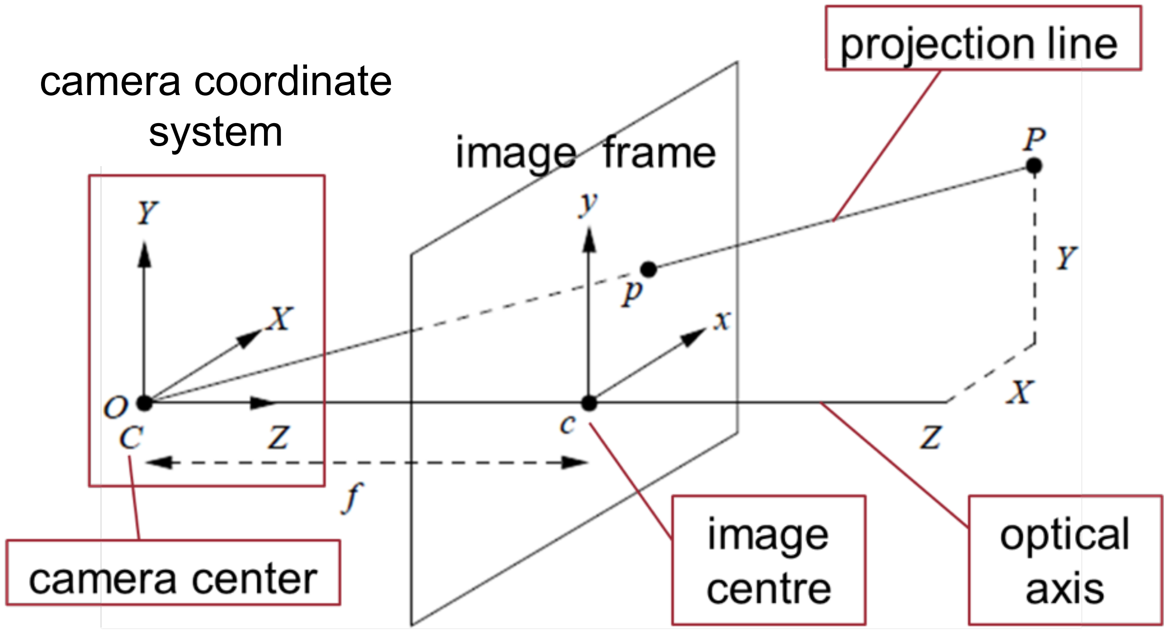
"vector-valued" function

Pinhole camera model



Pinhole camera model

- 3D point $\boldsymbol{P} = (X, Y, Z)^T$ projected to 2D image $\boldsymbol{p} = (x, y)^T$



$$x = f\frac{X}{Z}, \quad y = f\frac{Y}{Z}$$

Perspective projection model

- Camera sensor's pixels not exactly square $x = kf\frac{X}{Z}, \quad y = lf\frac{Y}{Z}$

Perspective projection model

- Camera sensor's pixels not exactly square $x = kf\frac{X}{Z}, \quad y = lf\frac{Y}{Z}$
- Image center or principal point c may not be at origin

$$x = f_x\frac{X}{Z} + c_x, \quad y = f_y\frac{Y}{Z} + c_y$$

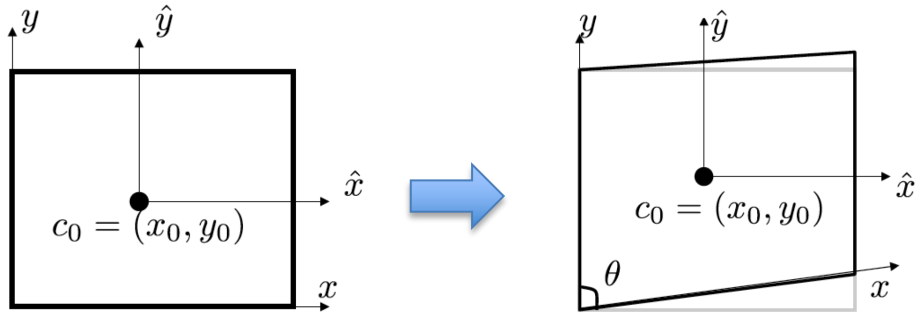
Perspective projection model

- Camera sensor's pixels not exactly square $x = kf\frac{X}{Z}, \quad y = lf\frac{Y}{Z}$
- Image center or principal point c may not be at origin

$$x = f_x\frac{X}{Z} + c_x, \quad y = f_y\frac{Y}{Z} + c_y$$

- Distortion (image frame may not be exactly rectangular)

$$x = f_x\frac{X}{Z} - f_x \cot\theta\frac{Y}{Z} + c_x, \quad y = \frac{f_y}{\sin\theta}\frac{Y}{Z} + c_y$$



Perspective projection model

$$x = f_x \frac{X}{Z} - f_x \cot\theta \frac{Y}{Z} + c_x, \quad y = \frac{f_y}{\sin\theta} \frac{Y}{Z} + c_y$$

- Intrinsic parameters
- Intrinsic parameter matrix

$$\tilde{\mathrm{x}} = \frac{1}{Z}\mathbf{K}\mathbf{X}, \quad \mathbf{K} = \begin{bmatrix} f_x & -f_x \cot\theta & c_x \\ 0 & \frac{f_y}{\sin\theta} & c_y \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow \mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Intrinsic parameter matrix

Perspective projection model

- Camera motion
  - Camera coordinate system is not aligned with world coordinate system
  - Camera can move and rotate

$$ {}^{C}\mathbf{X} = {}^{C}_{W}\mathbf{R}\,{}^{W}\mathbf{X} + {}^{C}_{W}\mathbf{T} $$

Intrinsic/Extrinsic Parameters

- The complete transformation

$$
\begin{aligned}
\mathbf{P}' &= \mathcal{M}\mathbf{P}_w \\
&= \underbrace{\boxed{\mathcal{K}}}_{\text{Internal (intrinsic) parameters}} \underbrace{\boxed{\begin{bmatrix} \mathcal{R} & \mathbf{T} \end{bmatrix}}}_{\text{External (extrinsic) parameters}} \mathbf{P}_w
\end{aligned}
$$

Today's Agenda

- Review of Lecture 2: Camera models
- Camera calibration
  - General idea
  - Camera calibration
    - Find correspondences
    - Solve for projection matrix
    - Extract the parameters
- Assignment 1: Camera calibration

General Idea

- Precise 3D -> 2D projection
  - Given 3D scene, camera location and orientation
  - Intrinsic parameters unknow

$$\begin{aligned}\mathbf{P}' &= \mathcal{M}\mathbf{P}_w \\ &= \boxed{\mathcal{K}}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w\end{aligned}$$

General Idea

- Precise 3D -> 2D projection
  - Given 3D scene, camera location and orientation
  - Intrinsic parameters unknow

General Idea

- Precise 3D -> 2D projection
  - Given 3D scene, camera location and orientation
  - Recover intrinsic parameters
- Camera calibration
  - Recovering K
  - Recovering R and T

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \boxed{\mathcal{K}} \left[\boxed{\mathcal{R} \quad \mathbf{T}}\right] \mathbf{P}_w$$

Internal (intrinsic) parameters

External (extrinsic) parameters

General Idea

- How many parameters to recover?

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \boxed{\mathcal{K}}\,\left[\boxed{\mathcal{R} \quad \mathbf{T}}\right]\mathbf{P}_w$$

Internal (intrinsic) parameters

External (extrinsic) parameters

General Idea

- How many parameters to recover?

$$\begin{aligned}\mathbf{P}' &= \mathcal{M}\mathbf{P}_w \\ &= \boxed{\mathcal{K}}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w\end{aligned}$$

Internal (intrinsic) parameters

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

General Idea

- How many parameters to recover?

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \mathcal{K}\begin{bmatrix}\boxed{\mathcal{R} \quad \mathbf{T}}\end{bmatrix}\mathbf{P}_w$$

External (extrinsic) parameters

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

General Idea

- How many parameters to recover?
  - 5 intrinsic parameters
    - 2 for focal length
    - 2 for offset (image center)
    - 1 for skewness
  - 6 extrinsic parameters
    - 3 for rotation
    - 3 for translation

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad \mathrm{R} = \begin{bmatrix} \mathbf{r}_1^{\mathrm{T}} \\ \mathbf{r}_2^{\mathrm{T}} \\ \mathbf{r}_3^{\mathrm{T}} \end{bmatrix} \quad \mathrm{T} = \begin{bmatrix} \mathrm{t}_x \\ \mathrm{t}_y \\ \mathrm{t}_z \end{bmatrix}$$

General Idea

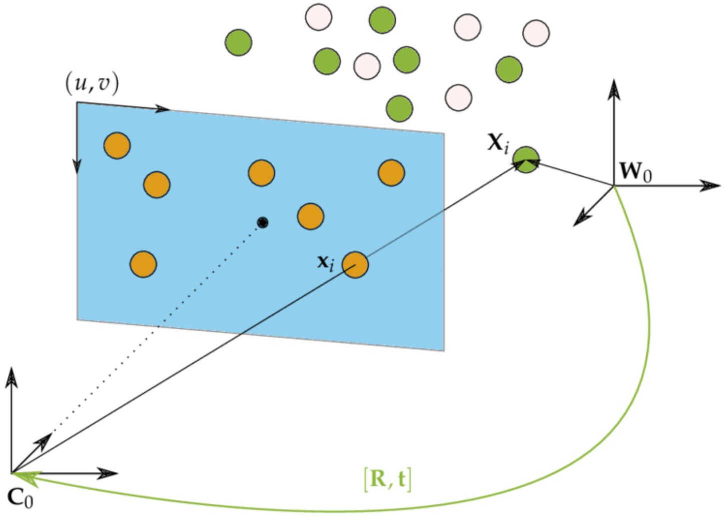
- How many parameters to recover?
- What information to use?

General Idea

- How many parameters to recover?
- What information to use?

  - Corresponding 3D-2D point pairs

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \mathcal{K}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w$$



General Idea

- How many parameters to recover?
- What information to use?
  - Corresponding 3D-2D point pairs
    - How many pairs do we need?

11 unknow parameters

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad \mathrm{R} = \begin{bmatrix} \mathbf{r}_1^{\mathrm{T}} \\ \mathbf{r}_2^{\mathrm{T}} \\ \mathbf{r}_3^{\mathrm{T}} \end{bmatrix} \quad \mathrm{T} = \begin{bmatrix} \mathrm{t_x} \\ \mathrm{t_y} \\ \mathrm{t_z} \end{bmatrix}$$

General Idea

- How many parameters to recover?
- What information to use?
  - Corresponding 3D-2D point pairs
    - How many pairs do we need?

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = MP_i = \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix}$$

General Idea

- How many parameters to recover?
- What information to use?
  - Corresponding 3D-2D point pairs
    - How many pairs do we need?
      - Each 3D-2D point pair -> 2 constraints
      - 11 unknown -> 6 point correspondence
      - Use more to handle noisy data

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = MP_i = \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix}$$

$$u_i(m_3 P_i) - m_1 P_i = 0$$
$$v_i(m_3 P_i) - m_2 P_i = 0$$

General Idea

- How many parameters to recover?
- What information to use?
  - Corresponding 3D-2D point pairs

$$
\begin{aligned}
u_1(m_3P_1) - m_1P_1 &= 0 \\
v_1(m_3P_1) - m_2P_1 &= 0 \\
&\vdots \\
u_n(m_3P_n) - m_1P_n &= 0 \\
v_n(m_3P_n) - m_2P_n &= 0
\end{aligned}
\quad \Rightarrow \quad
\begin{bmatrix}
\mathbf{P}_1^T & 0^T & -u_1\mathbf{P}_1^T \\
0^T & \mathbf{P}_1^T & -v_1\mathbf{P}_1^T \\
 & \vdots & \\
\mathbf{P}_n^T & 0^T & -u_n\mathbf{P}_n^T \\
0^T & \mathbf{P}_n^T & -v_n\mathbf{P}_n^T
\end{bmatrix}
\begin{bmatrix}
\mathbf{m}_1^T \\
\mathbf{m}_2^T \\
\mathbf{m}_3^T
\end{bmatrix}
= P\mathbf{m} = 0
$$

General Idea

- How many parameters to recover?
- What information to use?
- How to solve it?

$$\begin{bmatrix} \mathbf{P}_1^T & 0^T & -u_1\mathbf{P}_1^T \\ 0^T & \mathbf{P}_1^T & -v_1\mathbf{P}_1^T \\ & \vdots & \\ \mathbf{P}_n^T & 0^T & -u_n\mathbf{P}_n^T \\ 0^T & \mathbf{P}_n^T & -v_n\mathbf{P}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix} = P\mathbf{m} = 0$$

General Idea

- How many parameters to recover?
- What information to use?
- How to solve it?
  - – *m = 0* always a trivial solution
  - – *k * m* (*k* is non-zero) is also a solution

$$\begin{bmatrix} \mathbf{P}_1^T & 0^T & -u_1\mathbf{P}_1^T \\ 0^T & \mathbf{P}_1^T & -v_1\mathbf{P}_1^T \\ & \vdots & \\ \mathbf{P}_n^T & 0^T & -u_n\mathbf{P}_n^T \\ 0^T & \mathbf{P}_n^T & -v_n\mathbf{P}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix} = P\mathbf{m} = 0$$

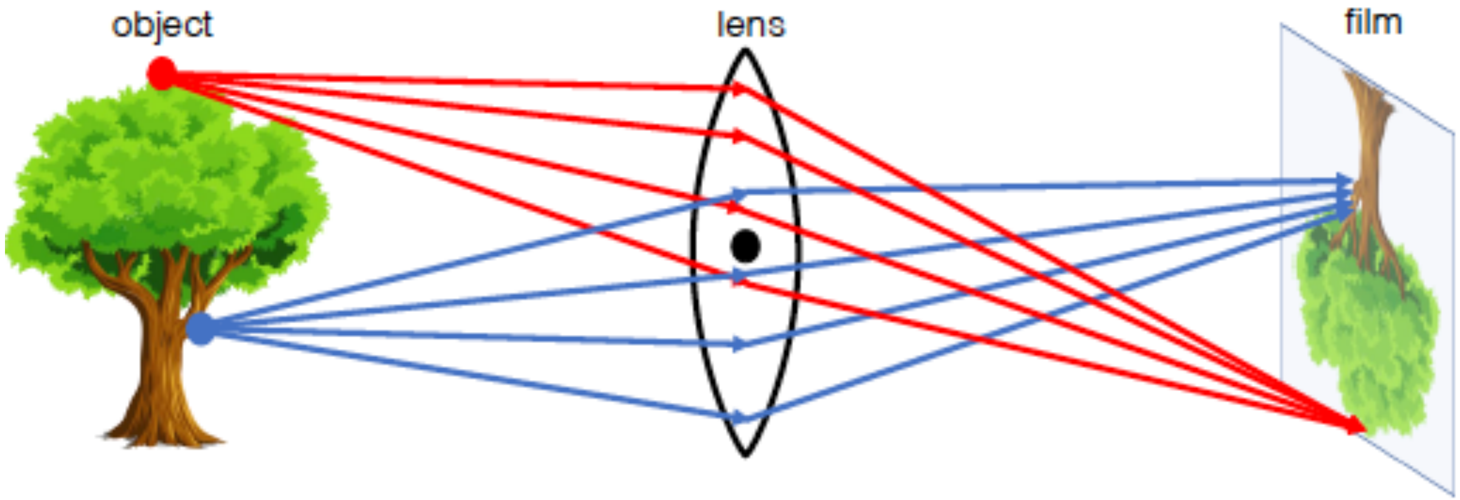
General Idea

- How many parameters to recover?
- What information to use?
- How to solve it?
  - *m = 0* always a trivial solution
  - *k * m* (*k* is non-zero) is also a solution
  - Constrained optimization

$$\begin{bmatrix} \mathbf{P}_1^T & 0^T & -u_1\mathbf{P}_1^T \\ 0^T & \mathbf{P}_1^T & -v_1\mathbf{P}_1^T \\ & \vdots & \\ \mathbf{P}_n^T & 0^T & -u_n\mathbf{P}_n^T \\ 0^T & \mathbf{P}_n^T & -v_n\mathbf{P}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix} = P\mathbf{m} = 0 \Rightarrow \begin{array}{ll} \underset{\mathbf{m}}{\text{minimize}} & \|P\mathbf{m}\|^2 \\ \text{subject to} & \|\mathbf{m}\|^2 = 1 \end{array}$$

Camera calibration

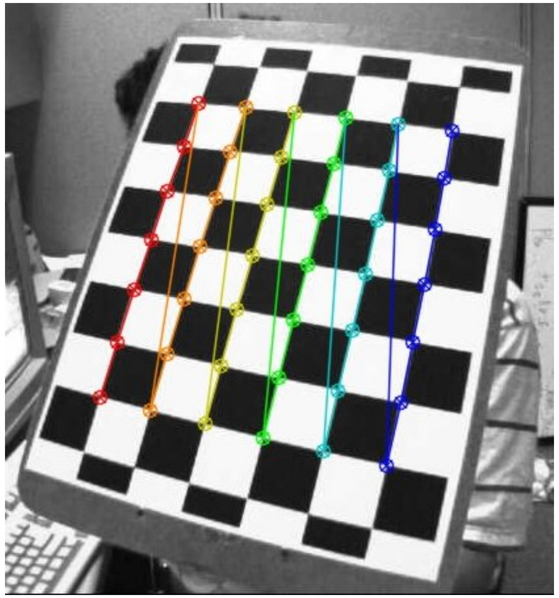
- How to find the corresponding point?
  - >= 6 3D-2D point pairs
    - 3D points with known 3D coordinates
    - Corresponding image points with known 2D coordinates



Camera calibration

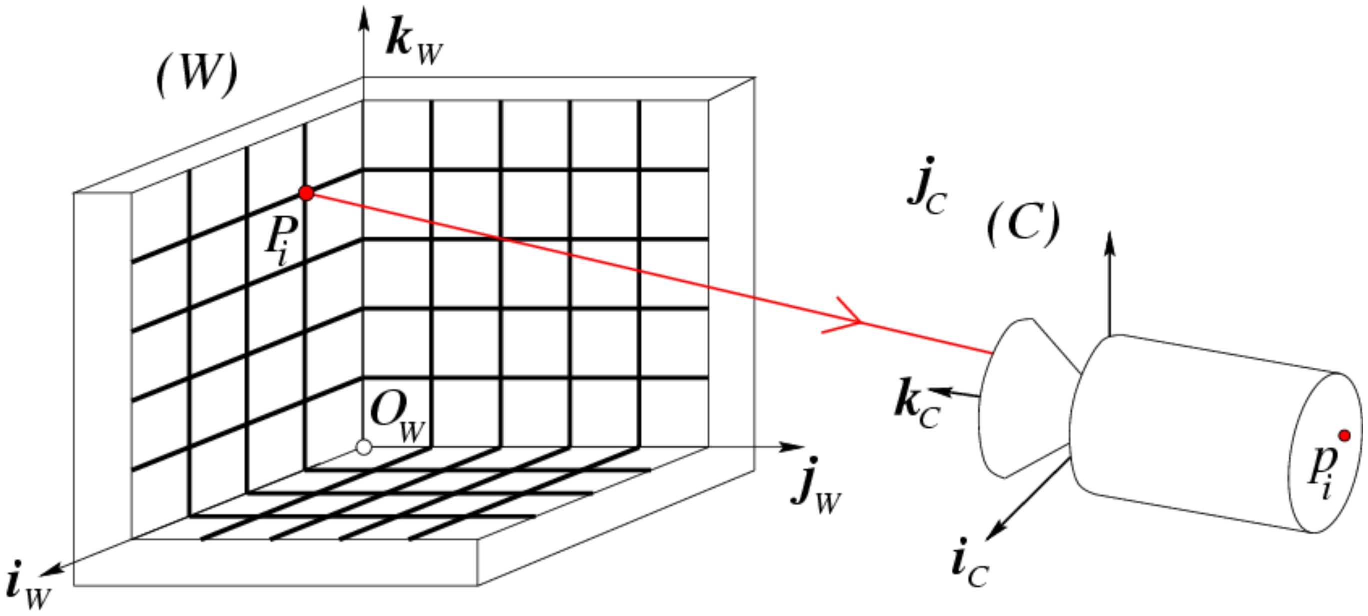
- How to find the corresponding point?
  - >= 6 3D-2D point pairs
    - 3D points with known 3D coordinates
    - Corresponding image points with known 2D coordinates

tape measure



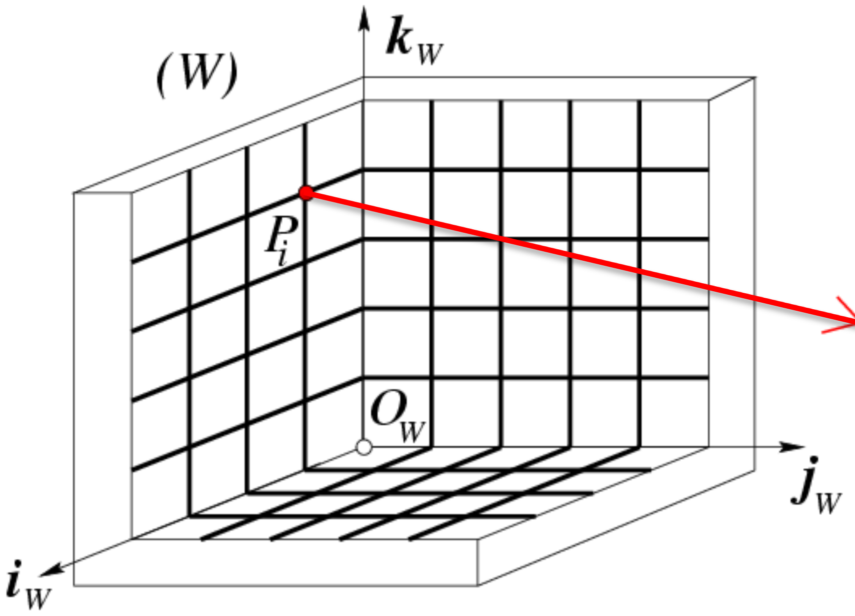
Corresponding points

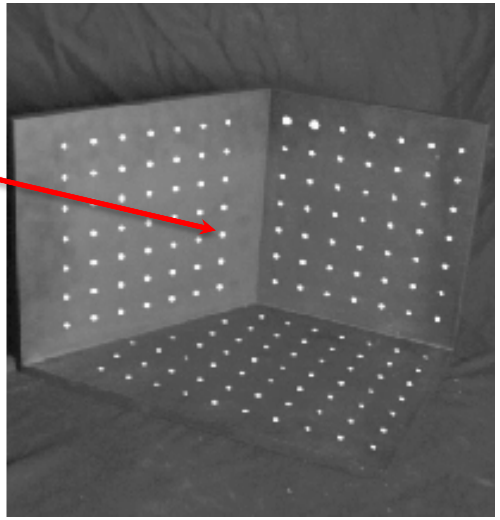
- Calibration rig - a special apparatus
  - P_1, … P_n with known positions in $[O_w, i_w, j_w, k_w]$



Corresponding points

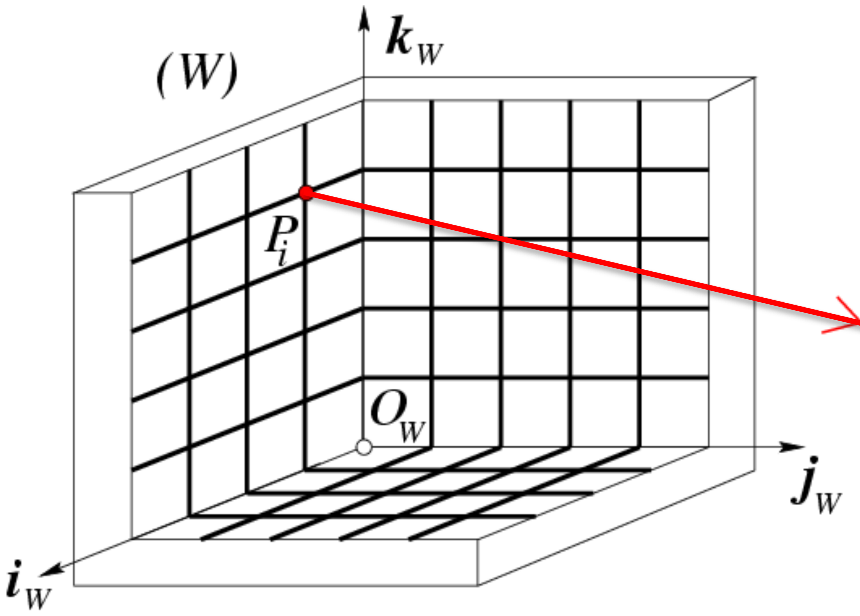
- Calibration rig - a special apparatus
  - P_1, … P_n with known positions in $[O_w, i_w, j_w, k_w]$
  - p_1, … p_n known positions in the image

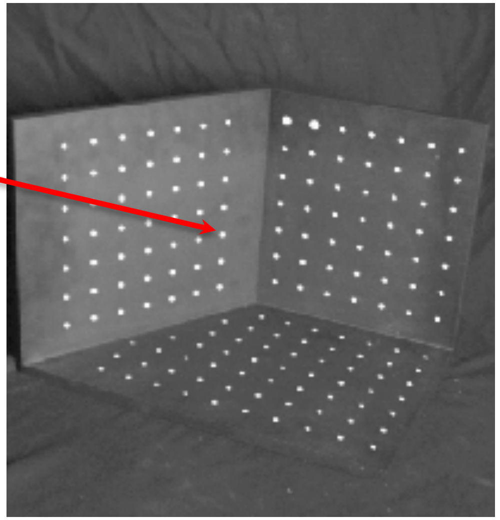




Corresponding points

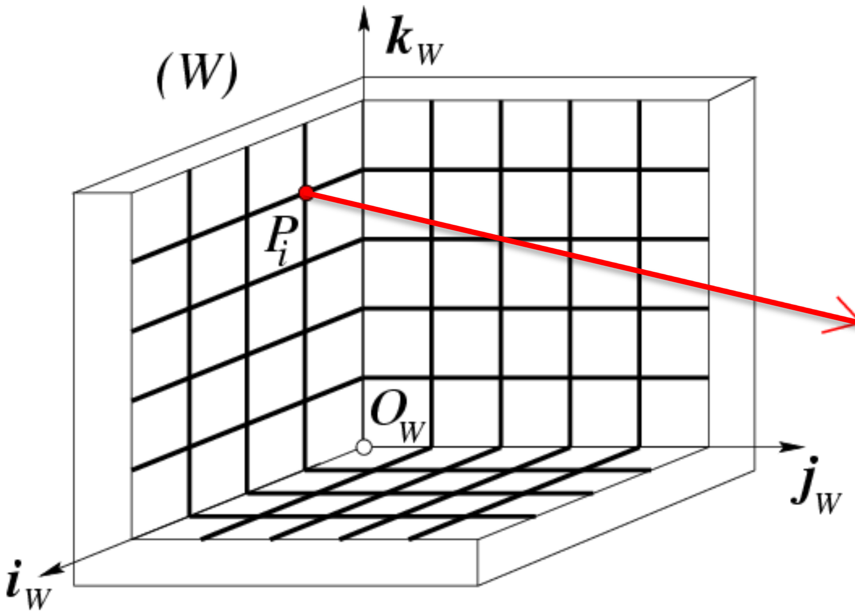
- Calibration rig - a special apparatus
  - P_1, … P_n with known positions in $[O_w, i_w, j_w, k_w]$
  - p_1, … p_n known positions in the image
  - >= 6 correspondences

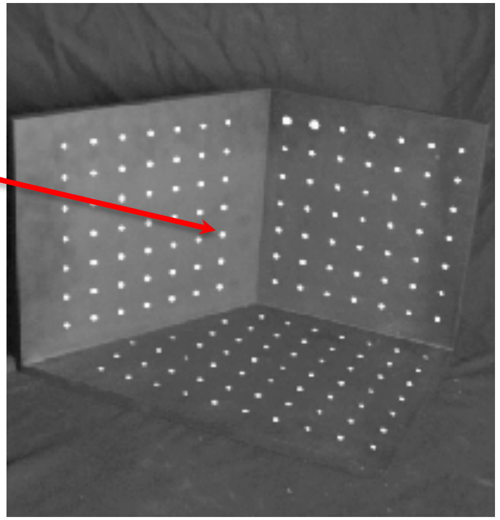




Corresponding points

- Calibration rig - a special apparatus
  - P_1, … P_n with known positions in $[O_w, i_w, j_w, k_w]$
  - p_1, … p_n known positions in the image
  - >= 6 correspondences
- Goal
  - Intrinsic parameters
  - Extrinsic parameters

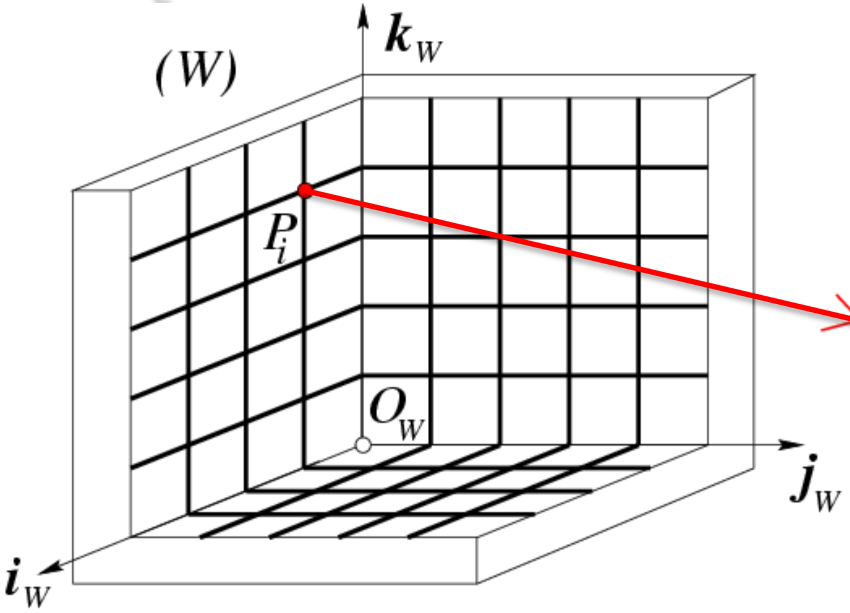


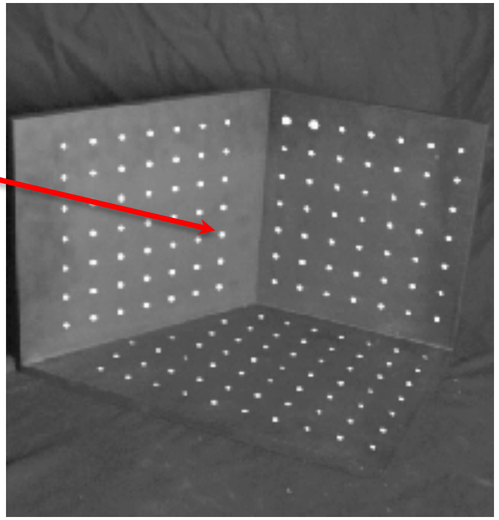


Calibration

- The equations

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \mathcal{K}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w$$





$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = MP_i = \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix}$$

$$u_i(m_3 P_i) - m_1 P_i = 0$$
$$v_i(m_3 P_i) - m_2 P_i = 0$$

Calibration

- The equations

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \frac{m_1 P_i}{m_3 P_i} \\ \frac{m_2 P_i}{m_3 P_i} \end{bmatrix}$$

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

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

Calibration

- The equations

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

Write it using matrix-vector product

Calibration

- The equations

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

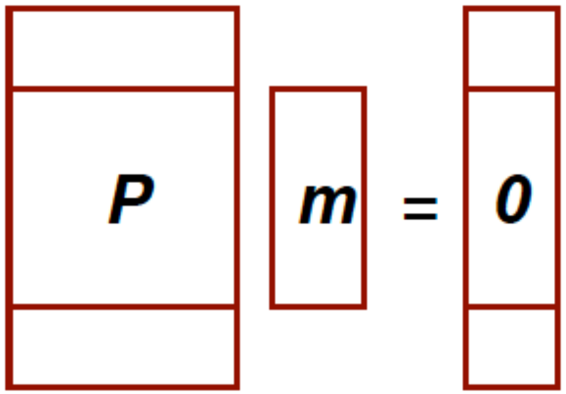
Homogenous linear system

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

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

Calibration

- The equations
  - 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 $|\mathbf{m}|^2 = 1$

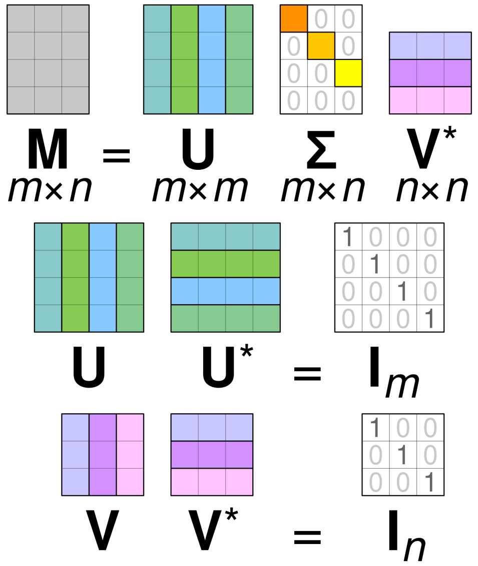
- Solve using SVD

SVD

- **S**ingular **V**alue **D**ecomposition
  - Generalization of the eigen-decomposition of a square matrix to any m by n matrix

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

U, V = orthogonal matrix



Calibration

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

SVD decomposition of P

$$\boxed{U_{2nx2n}\ D_{2nx12}\ V_{12x12}}$$

Last column of V gives ***m***

(Why? See page 593 of Hartley & Zisserman. Multiple View Geometry in Computer Vision)

Calibration

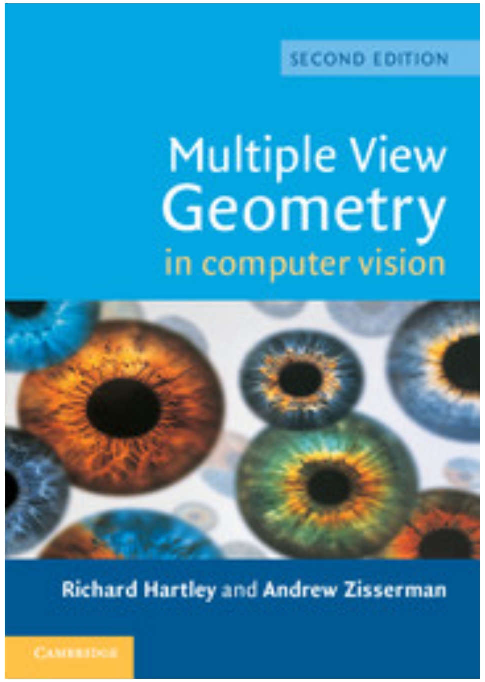
Objective

Given a matrix $\mathtt{A}$ with at least as many rows as columns, find $\mathbf{x}$ that minimizes $\|\mathtt{A}\mathbf{x}\|$ subject to $\|\mathbf{x}\| = 1$.

Solution

$\mathbf{x}$ is the last column of $\mathtt{V}$, where $\mathtt{A} = \mathtt{UDV}^{\mathsf{T}}$ is the SVD of $\mathtt{A}$.

Algorithm A5.4. *Least-squares solution of a homogeneous system of linear equations.*



Page 593 of Hartley & Zisserman. Multiple View Geometry in Computer Vision

Calibration

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w = \mathcal{K}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w$$

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

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

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

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

Estimated values

Intrinsic

$$\rho = \frac{\pm 1}{|\mathbf{a}_3|} \qquad u_0 = \rho^2\mathbf{a}_1^T\mathbf{a}_3$$

$$v_0 = \rho^2\mathbf{a}_2^T\mathbf{a}_3$$

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

Calibration

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \mathcal{K}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w$$

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

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

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

Estimated values

Intrinsic

$$\left.\begin{matrix}\alpha = \rho^2|\mathbf{a}_1 \times \mathbf{a}_3|\sin\theta \\ \beta = \rho^2|\mathbf{a}_2 \times \mathbf{a}_3|\sin\theta\end{matrix}\right\} \rightarrow f$$

Calibration

$$\mathbf{P}' = \mathcal{M}\mathbf{P}_w$$
$$= \mathcal{K}\begin{bmatrix}\mathcal{R} & \mathbf{T}\end{bmatrix}\mathbf{P}_w$$

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

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

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

Estimated values

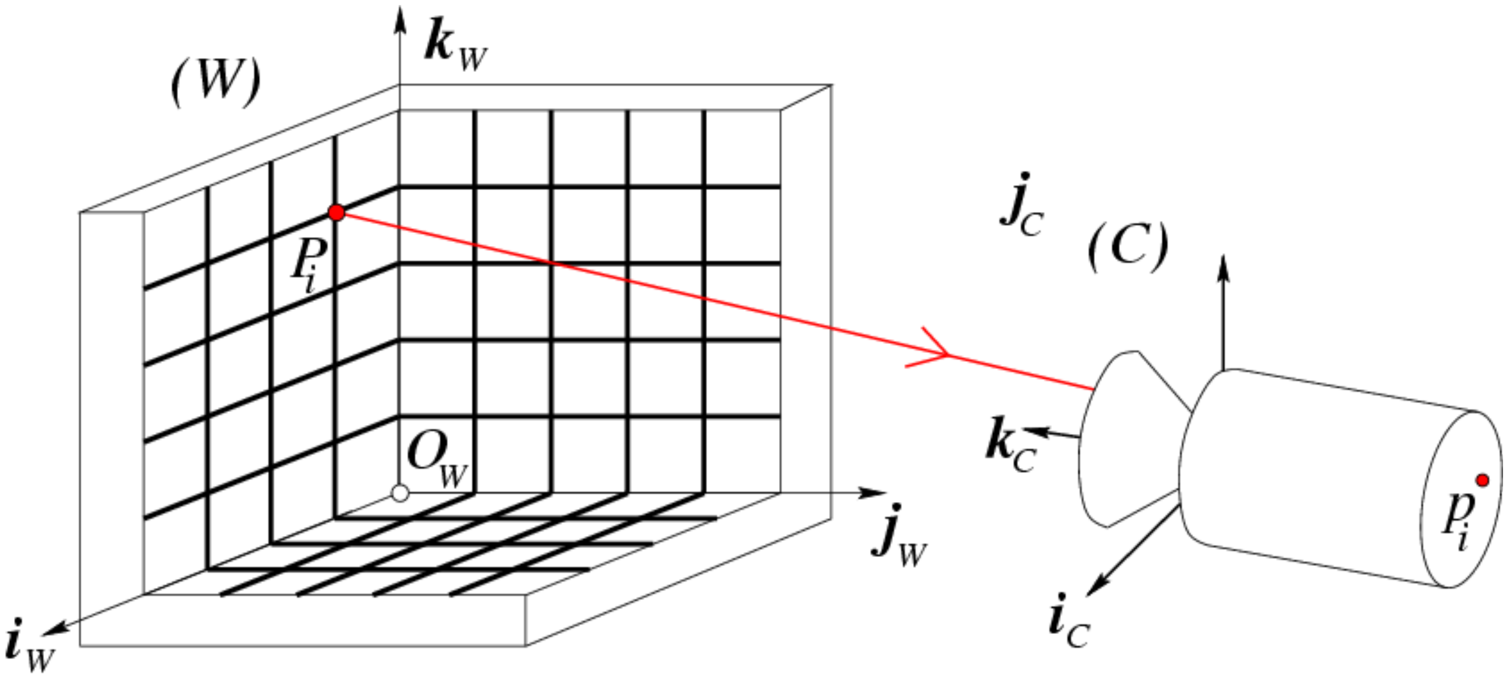
Extrinsic

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

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

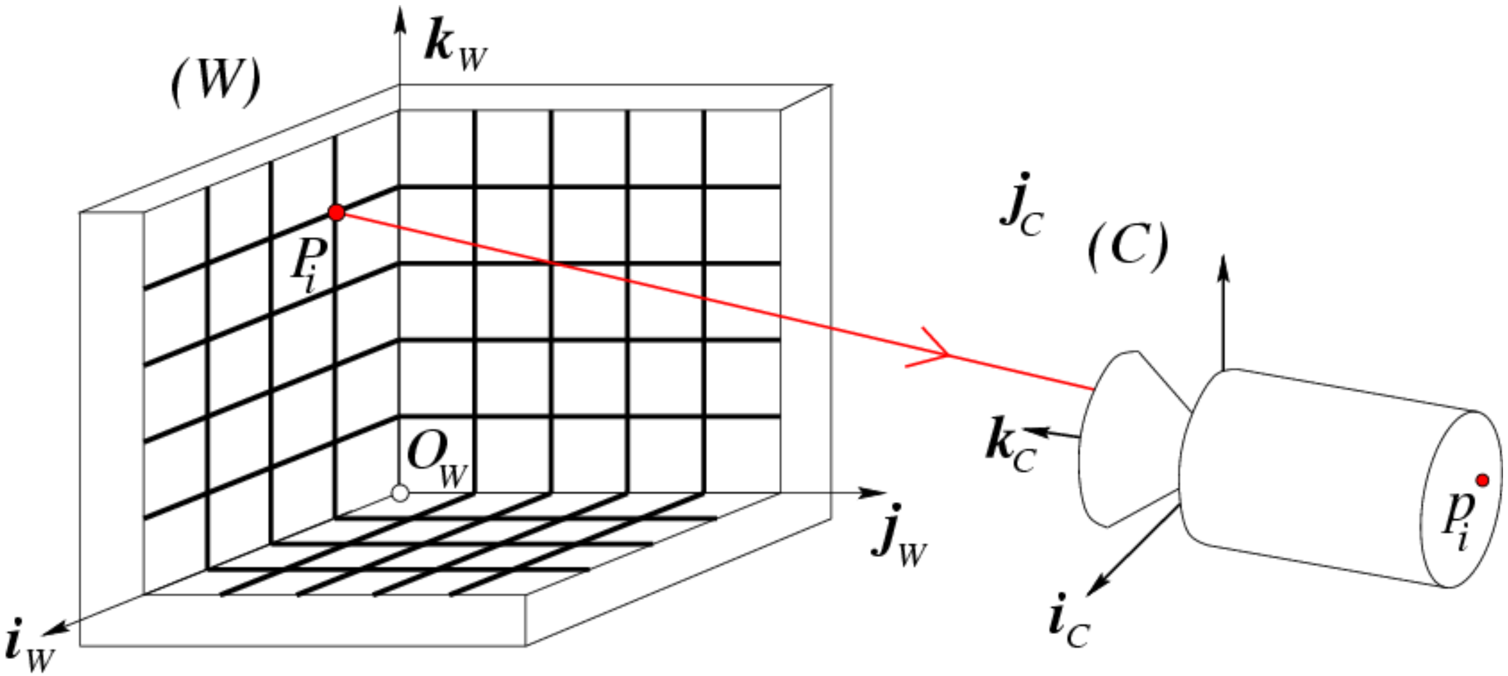
Calibration

- Always solvable?



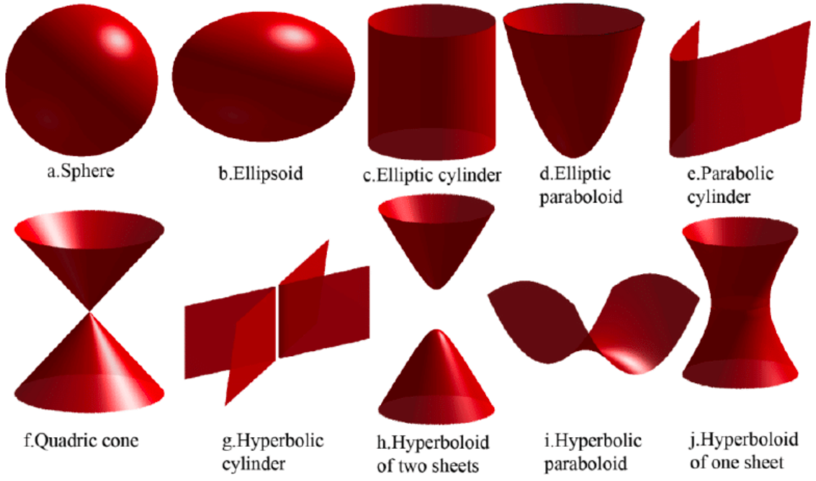
Calibration

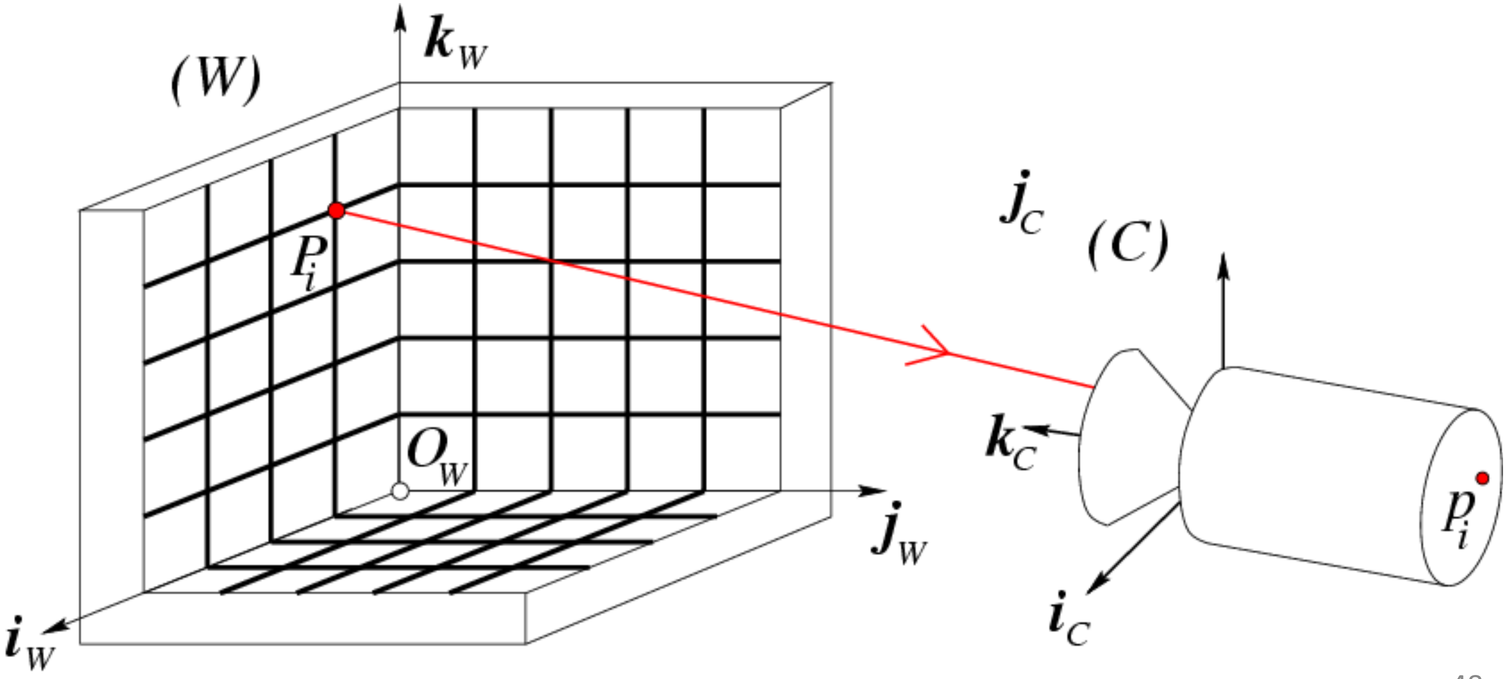
- Always solvable?
  - P_is cannot lie on the same plane
  - P_is cannot lie on the intersection curve of two quadric surfaces



Calibration

- Always solvable?
  - P_is cannot lie on the same plane
  - P_is cannot lie on the intersection curve of two quadric surfaces





Assignment 1: Camera calibration

Next Lecture

Two View Geometry