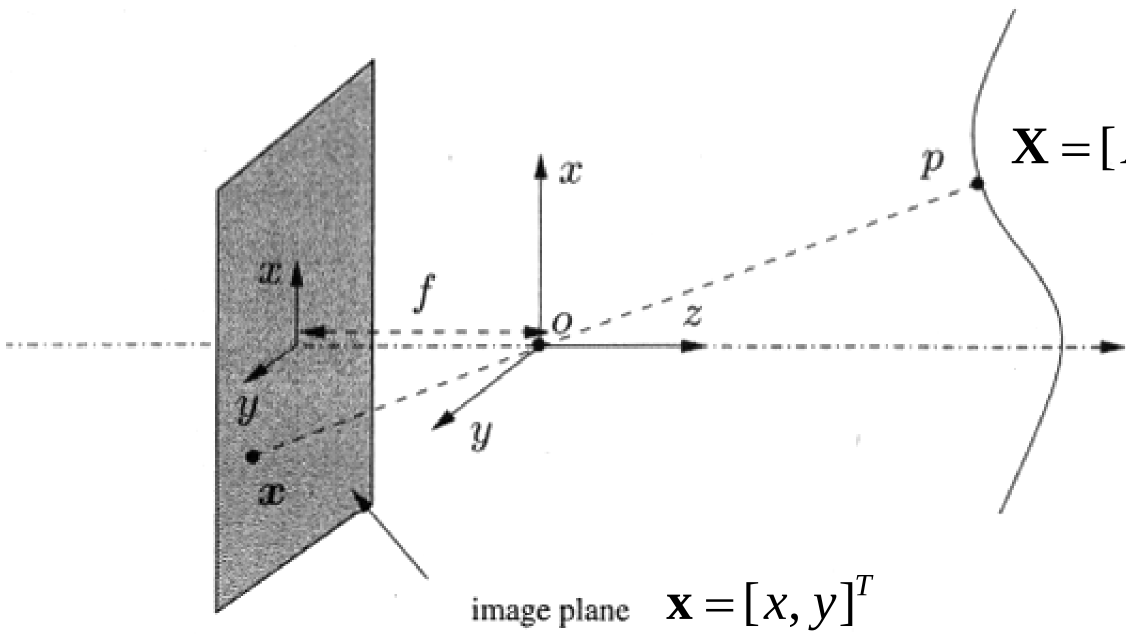
Camera model & image formation

SINA – 07/08

Image formation, camera model

Consider a pinhole camera, force all rays to go through the optical center



$\mathbf{X} = [X, Y, Z]^T$

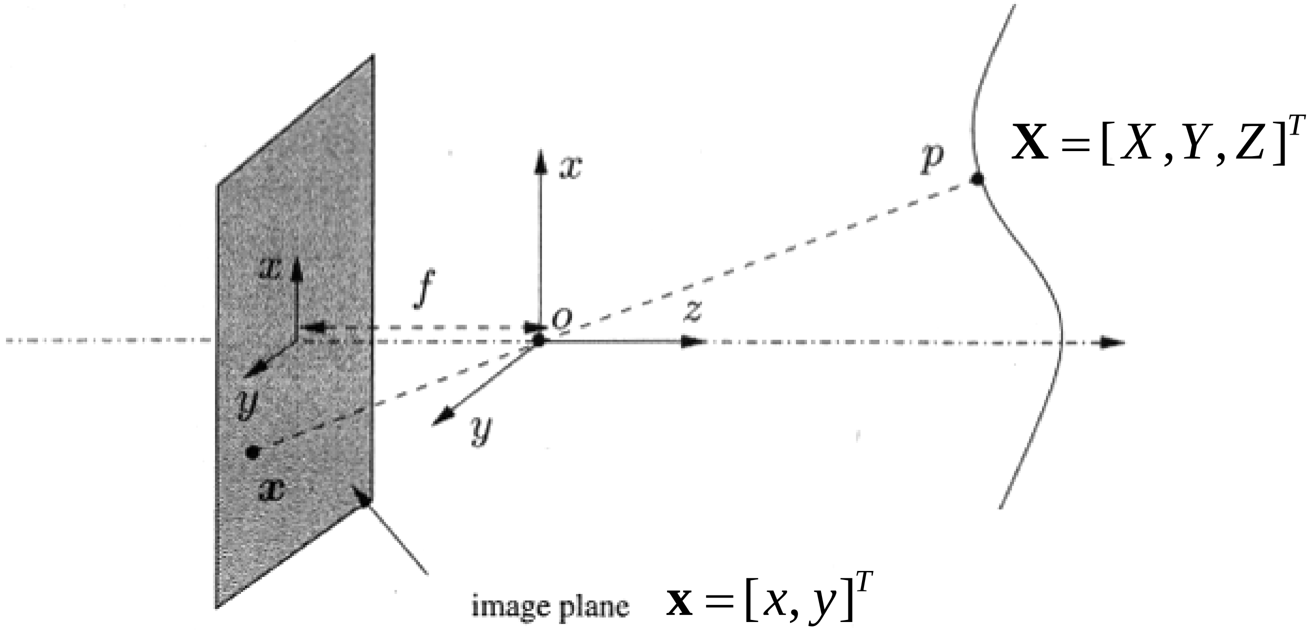
$\mathbf{x} = [x, y]^T$

$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ \mathrm{z} = \lambda Z \end{cases}$$

See: An Invitation to 3-D Vision, Ma, Soatto, Kosecka, Sastry, Forsyth and Ponce, Computer Vision a Modern Approach

Image formation, camera model

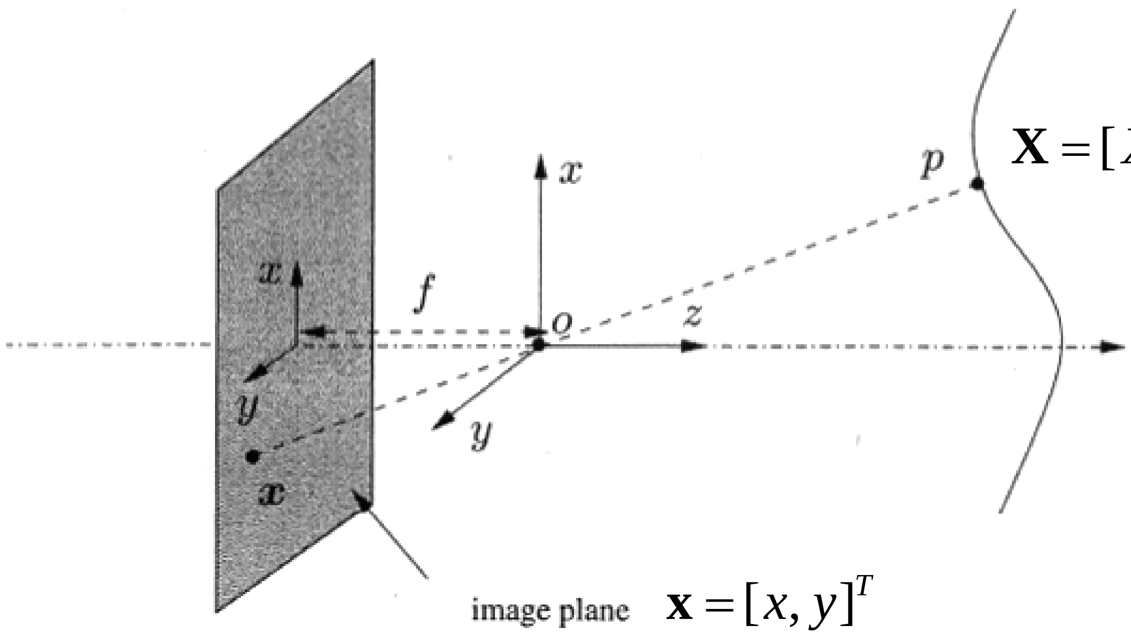
Consider a pinhole camera, force all rays to go through the optical center



$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ z = \lambda Z \end{cases} \Rightarrow \begin{cases} x = \lambda X \\ y = \lambda Y \\ z = -f \Rightarrow \lambda = -f / Z \end{cases}$$

Image formation, camera model

Consider a pinhole camera, force all rays to go through the optical center



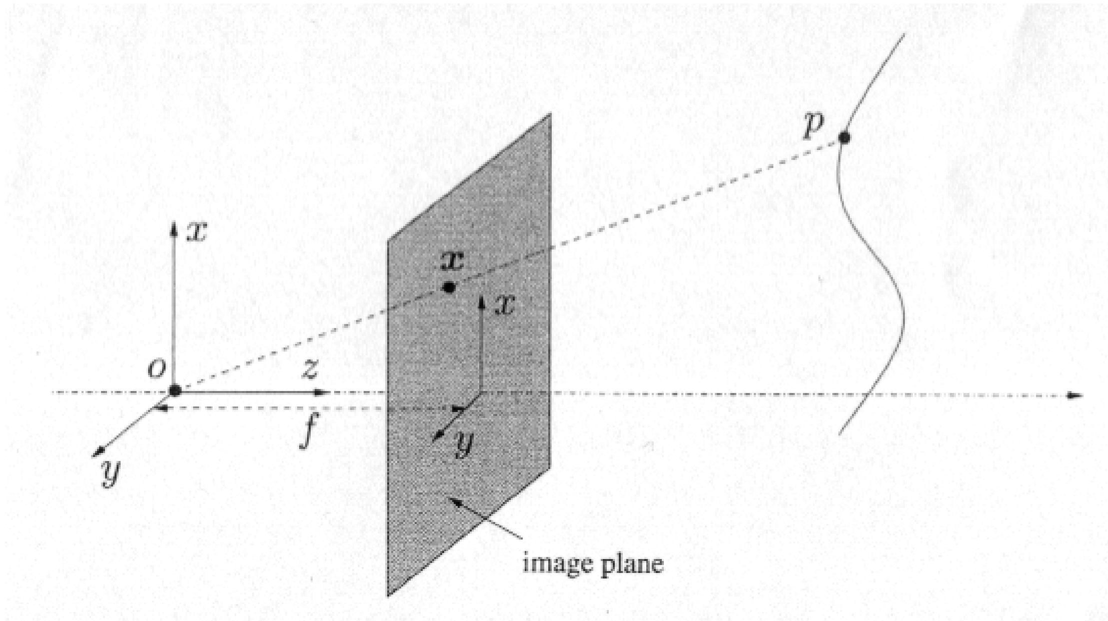
$$\mathbf{X} = [X, Y, Z]^T$$

$$\mathbf{x} = [x, y]^T$$

$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ \mathrm{z} \ = \lambda Z \end{cases} \Rightarrow \begin{cases} x = \lambda X \\ y = \lambda Y \\ \mathrm{z} \ = -f \Rightarrow \lambda = -f \ / \ Z \end{cases} \Rightarrow x = -f \, \frac{X}{Z}, y = -f \, \frac{Y}{Z}$$

ideal pinhole camera model

Often we flip the image (-x,-y) → (x,y), which is equivalent to placing the image plane in front of the optical center:



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

Note: any point on the line through *o* and *p* projects onto the same coordinates *(x,y)*

- Consider a generic point *p* with coordinates $\mathbf{X}_0$=[$X_{0,}$ Y_0 Z_0] relative to the world reference frame
- The coordinates **X**=[X, Y Z] of *p* relative to the camera frame are given by the rigid body transformation:

$$\mathbf{X} = \mathbf{R} \cdot \mathbf{X_0} + \mathbf{T}$$

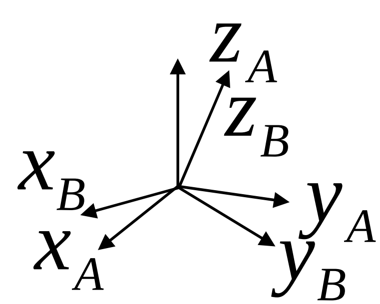
$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

homogeneous representation

Representing rotations

- Representing rotations and translations between coordinate frames of reference

$$ {}^{A}v = [?]\, {}^{B}v $$



$$ {}^{A}v = [\, {}^{A}x_B \mid {}^{A}y_B \mid {}^{A}z_B \,]\, {}^{B}v = {}^{A}R_B \, {}^{B}v \qquad B \rightarrow A $$

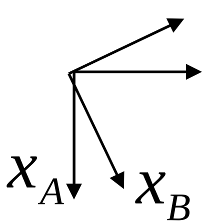
$$ {}^{A}x_B = {}^{A}R_B \, {}^{B}x_B = {}^{A}R_B [1,0,0]^T $$

Rotation matrix

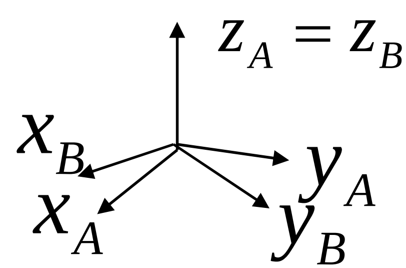
$$ {}^{A}R_{B}({}^{A}R_{B})^{T} = I \Leftrightarrow ({}^{A}R_{B})^{T} = ({}^{A}R_{B})^{-1} = {}^{B}R_{A} $$

Orthogonal matrix

Example: rotation along the Z axis $\begin{bmatrix} \cos\vartheta & -\sin\vartheta & 0 \\ \sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix}$



$$ {}^{A}x_{B} = \begin{bmatrix} \cos\vartheta \\ \sin\vartheta \\ 0 \end{bmatrix} $$



Rigid body transformations

$$\left\| p(t) - q(t) \right\| = \left\| p(0) - q(0) \right\| = \text{constant}$$

- Given that the object is:

$$O \subset \mathbb{R}^3$$

- The motion of the body is represented by a family of mappings:

$$g(t) : O \rightarrow \mathbb{R}^3$$

- A rigid displacement of the body is:

$$g : O \rightarrow \mathbb{R}^3$$

Action on points and vectors

$$g_*(v) = g(q) - g(p)$$

Where:

$$v = q - p$$

Note the difference between points and vectors (although both are represented as 3-tuples of numbers). A vector has magnitude and direction and doesn't belong to a body (free vector).

Then…

$$g : \mathbb{R}^3 \to \mathbb{R}^3$$

is a rigid body transformation if:

$$\|g(p) - g(q)\| = \|p - q\| \text{ for all points } p, q \in \mathbb{R}^3$$

Length is preserved

$$g_*(v \times w) = g_*(v) \times g_*(w) \text{ for all vectors } v, w \in \mathbb{R}^3$$

The cross product is preserved

$\downarrow$

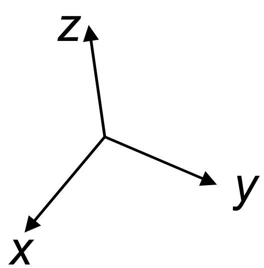
The inner product is also preserved, thus:

$$v^T w = g_*(v)^T g_*(w)$$

I.e. orthogonal vectors remain orthogonal

Some more requirements

- Right handed coordinate systems:



$$z = x \times y$$

- If a coordinate system is attached to a rigid body undergoing rigid motion:

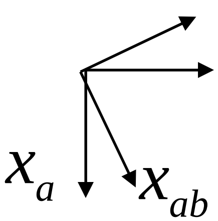
$$v_1, v_2, v_3 \text{ attached in } p \text{ then by effect of } g$$

$$g_*(v_1), g_*(v_2), g_*(v_3) \text{ are attached in g}(p)$$

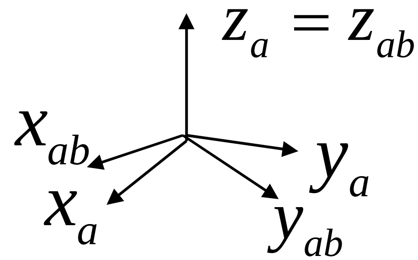
Rotation matrix (planar case)

Example: rotation along the Z axis

$$\begin{bmatrix} \cos\vartheta & -\sin\vartheta & 0 \\ \sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



$$x_{ab} = \begin{bmatrix} \cos\vartheta \\ \sin\vartheta \\ 0 \end{bmatrix}$$



The group of rotations SO(3)

- The set of 3x3 matrices with these properties is denoted:

$SO(3)$ which means Special Orthogonal of size 3

- That is:

$$SO(3) = \{R \in \mathbb{R}^{3\times 3} : RR^T = I, \det R = +1\}$$

Orthogonal Special

SO(3) is a group under matrix multiplication

1. Closure

$$R_1, R_2 \in SO(3) \Rightarrow R_1 R_2 \in SO(3)$$

2. Identity

$$I \text{ is the identity element } IR = R \ \forall R$$

3. Inverse

$$RR^T = R^T R = I, R^T \in SO(3)$$

4. Associativity

$$(R_1 R_2) R_3 = R_1 (R_2 R_3)$$

More simple rotations

Example: rotation along the Y axis

$$\begin{bmatrix} \cos\vartheta & 0 & \sin\vartheta \\ 0 & 1 & 0 \\ -\sin\vartheta & 0 & \cos\vartheta \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\vartheta & -\sin\vartheta \\ 0 & \sin\vartheta & \cos\vartheta \end{bmatrix}$$

Example: rotation along the X axis

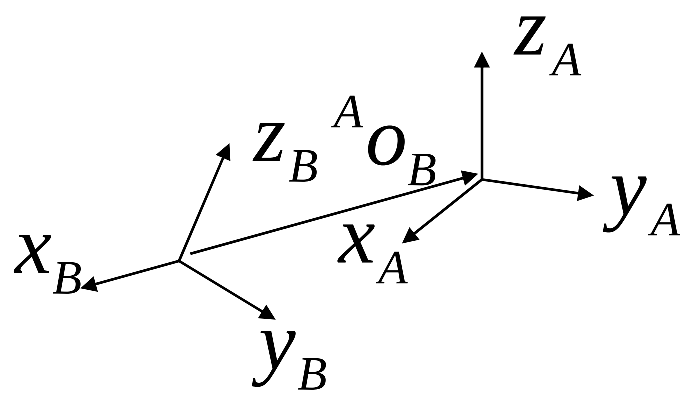
Representing 3D rotations

- Sequences of elementary rotations
  - Euler angles: z, y, z or z, x, z
  - Roll, pitch, yaw angles: z, y, x
  - Vector (axis of rotation) and angle

Roto-translation

- Rotation combined with translation

$$^{A}v = {}^{A}R_{B}\,{}^{B}v + {}^{A}o_{B}$$



Homogeneous representation

- To make things uniform

$$^{A}v = {}^{A}R_{B}\, {}^{B}v + {}^{A}o_{B}$$

$$\begin{bmatrix} ^{A}v \\ 1 \end{bmatrix} = \begin{bmatrix} ^{A}R_{B} & ^{A}o_{B} \\ 0 & 1 \end{bmatrix} . \begin{bmatrix} ^{B}v \\ 1 \end{bmatrix}$$

$$^{A}v = {}^{A}T_{B}\, {}^{B}v \qquad \dim(v) = 4$$

Clearly

$$ {}^{A}v = {}^{A}T_{B}\ {}^{B}T_{C}\ {}^{C}v \quad C \rightarrow A $$

$$ \begin{bmatrix} {}^{A}R_{B} & {}^{A}o_{B} \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} {}^{A}R_{B}^{T} & -{}^{A}R_{B}^{T}\ {}^{A}o_{B} \\ 0 & 1 \end{bmatrix} $$

$$ {}^{A}T_{B}{}^{-1} = {}^{B}T_{A} $$

- Composition of transforms
- Inverse of a rototranslation

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z \begin{bmatrix} x \\ y \end{bmatrix} = f \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \end{bmatrix} = f\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \end{bmatrix} = f\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$Z \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \end{bmatrix} = f\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$Z \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

replace Z with an arbitrary positive scalar

consider a point in the world reference frame

$$\lambda \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \end{bmatrix} = f\begin{bmatrix} X \\ Y \end{bmatrix} \Rightarrow Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$Z \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

replace Z with an arbitrary positive scalar

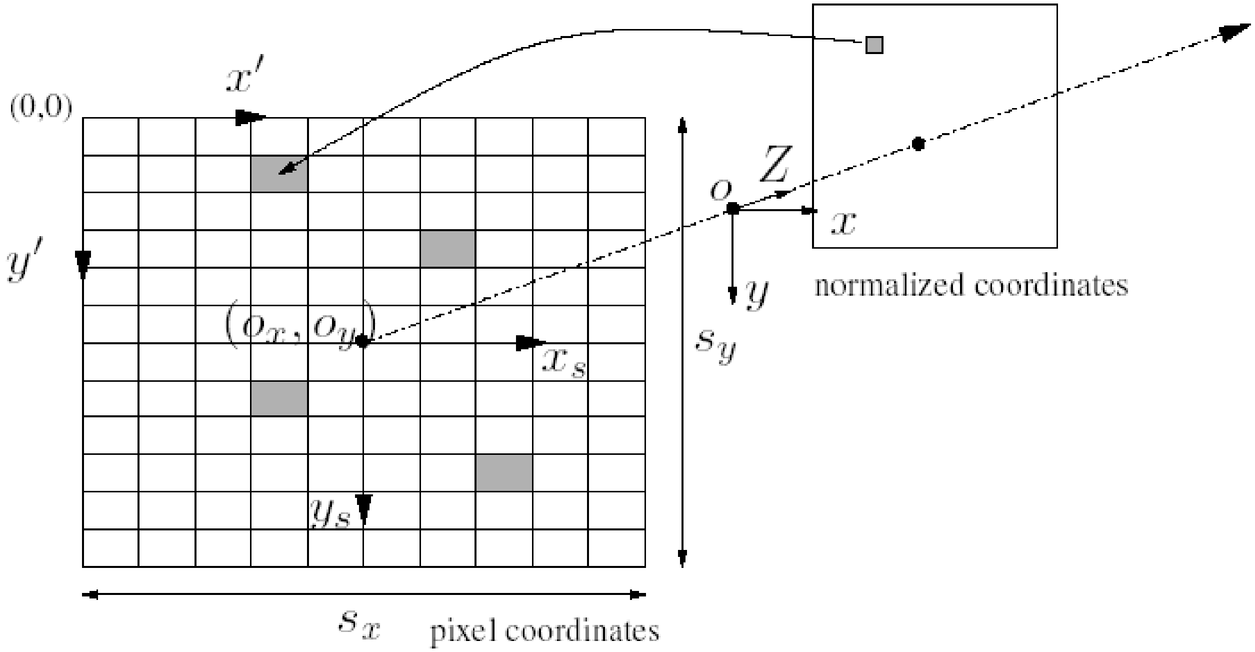
$$\lambda \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

consider a point in the world reference frame

$$\boxed{\lambda \cdot \mathbf{x} = K_f M_0 g \mathbf{X}_0}$$

geometric model for *an ideal camera*

Intrinsic parameters



$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

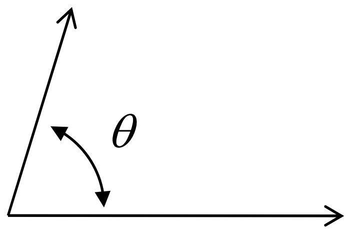
$$x' = x_s + o_x$$

$$y' = y_s + o_y$$

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- If pixels are not rectangular, a more general form of matrix is considered:

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



where s_θ is the *skew factor*

- If pixels are not rectangular, a more general form of matrix is considered:

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where s_θ is the *skew factor*

- A more realistic model of a transformation between homogeneous coordinates of a 3D point relative to the world reference frame and its image in terms of pixels:

$$\lambda \cdot \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

- If pixels are not rectangular, a more general form of matrix is considered:

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where s_θ is the *skew factor*

- A more realistic model of a transformation between homogeneous coordinates of a 3D point relative to the world reference frame and its image in terms of pixels:

$$\lambda \cdot \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

$$K = K_s \cdot K_f = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

- To summarize:

extrinsic parameters

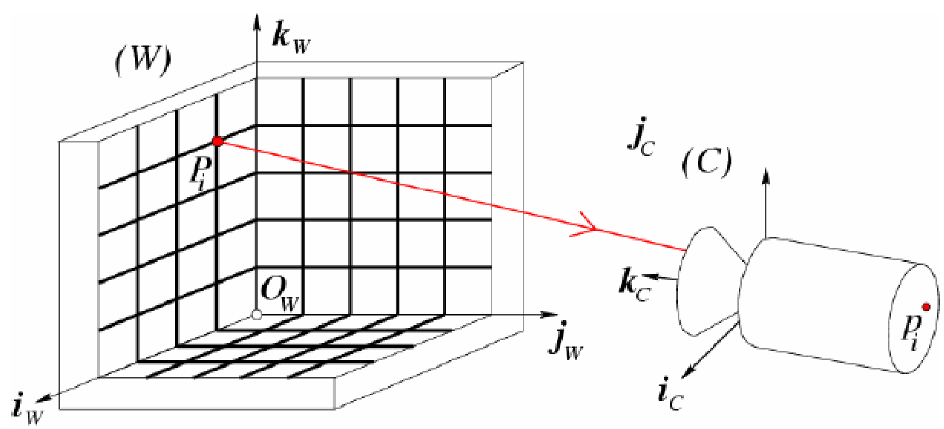
$$\boxed{\lambda \cdot \mathbf{x} = K M_0 g \mathbf{X}_0 = M \mathbf{X}_0}$$

$K = K_s K_f$ *intrinsic parameters*

- Intrinsic and extrinsic parameters can be estimated with a general technique called "*camera calibration*" (see for example: *R.Y. Tsai 1986)*

Geometric Camera Calibration (introduction)

- We assume that the camera observes a set of features such as points or lines with known positions in a fixed world coordinate system
- Derive the intrinsic and extrinsic parameters of the camera
- Allow associating with any image point a well-defined ray passing through the point and the camera's optical center



Linear Approach

$$Z\mathbf{p} = M\mathbf{P}$$

$$\mathbf{m}_1^T, \mathbf{m}_2^T, \mathbf{m}_3^T \text{ rows of M}$$

Linear Approach

$$Z\mathbf{p} = M\mathbf{P} \Rightarrow Z = \mathbf{m}_3 \cdot \mathbf{P}$$

$$\underbrace{\mathbf{m}_1^T, \mathbf{m}_2^T, \mathbf{m}_3^T}_{\text{contain extrinsic and intrinsic parameters of the camera}} \text{ rows of M}$$

$$\begin{cases} x = \dfrac{\mathbf{m}_1 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}}, \\ y = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \end{cases}$$

contain extrinsic and intrinsic parameters of the camera

Linear Approach

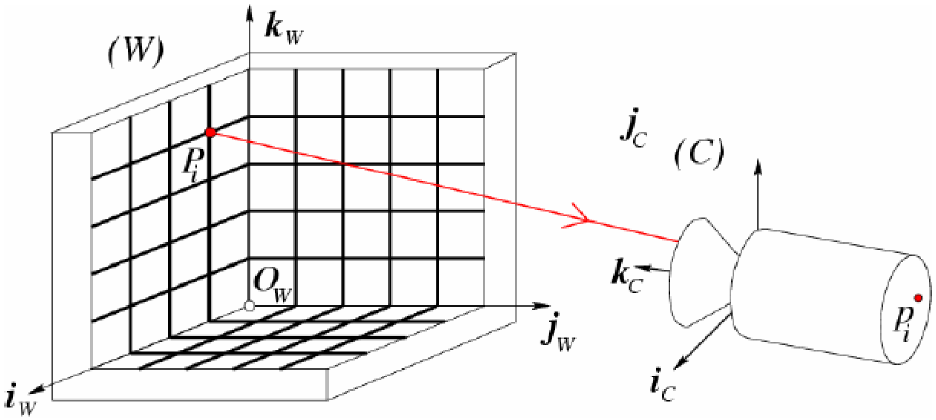
$$Z\mathbf{p} = M\mathbf{P} \Rightarrow Z = \mathbf{m}_3 \cdot \mathbf{P}$$

$$\underbrace{\mathbf{m}_1^T, \mathbf{m}_2^T, \mathbf{m}_3^T}_{} \text{ rows of M}$$

contain extrinsic and intrinsic parameters of the camera

$$\begin{cases} x = \dfrac{\mathbf{m}_1 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}}, \\ y = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \end{cases}$$

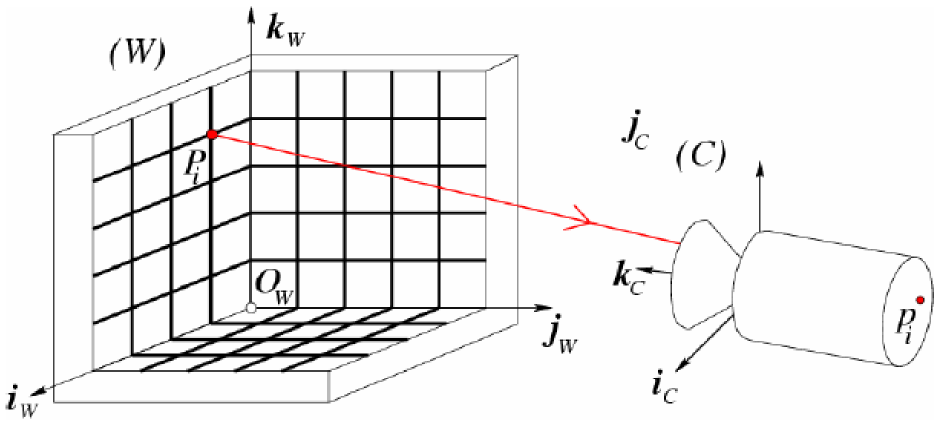
Linear Approach



Consider a set of n points with *known* position P_i, and projection x_i,y_i

$$\begin{cases} x_i = \dfrac{\mathbf{m}_1 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i}, \\ y_i = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \end{cases}$$

Linear Approach



Consider a set of *n* points with *known* position P_i, and projection x_i,y_i

$$\begin{cases} x_i = \dfrac{\mathbf{m}_1 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i}, \\ y_i = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \end{cases} \Rightarrow \begin{cases} \left(\mathbf{m}_1 - x_i \mathbf{m}_3\right) \cdot \mathbf{P}_i = 0 \\ \left(\mathbf{m}_2 - x_i \mathbf{m}_3\right) \cdot \mathbf{P}_i = 0 \end{cases}$$

For each point *i* we get two equations

Linear Approach

Collecting *n* point yields to a system of *2n* homogeneus linear equations:

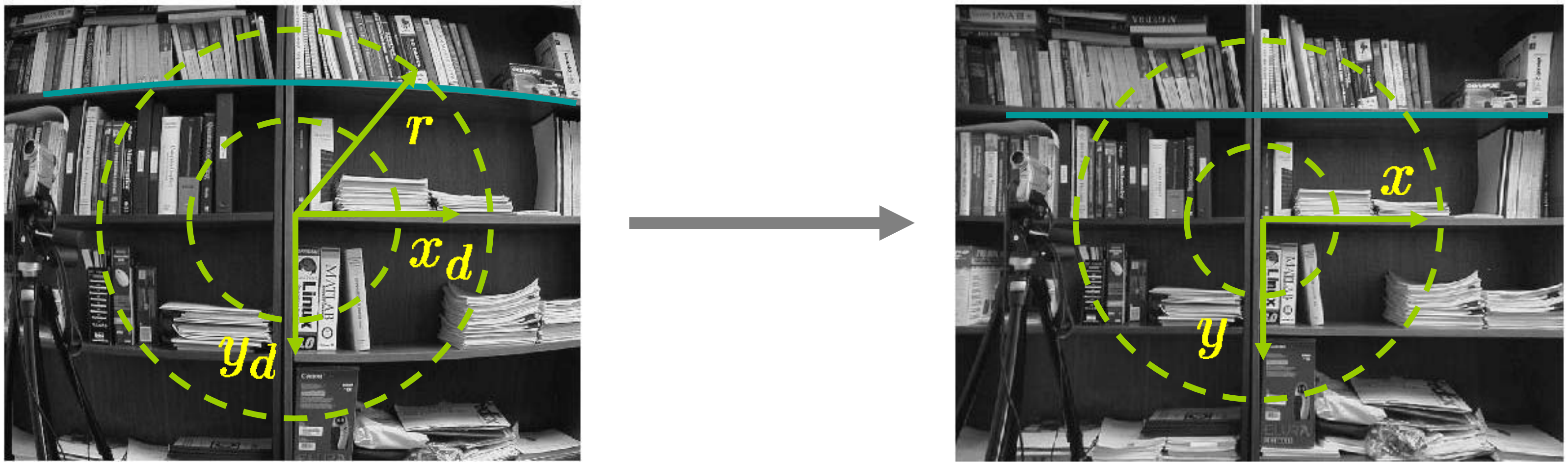
$$W\mathbf{m} = 0$$

$$W \triangleq \begin{bmatrix} \mathbf{P}_1^T & 0 & -x_1\mathbf{P}_1^T \\ 0 & \mathbf{P}_1^T & -y_1\mathbf{P}_1^T \\ \dots & \dots & \dots \\ \mathbf{P}_n^T & 0 & -x_n\mathbf{P}_n^T \\ 0 & \mathbf{P}_n^T & -y_n\mathbf{P}_n^T \end{bmatrix} \text{ and } \mathbf{m} \triangleq \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

When *n* is large (>6) *least-squares* can be used to determine **m** (and the projection matrix **M**)

Finally from **M** we extract intrinsic and extrinsic parameters of the camera

Radial distortion



$$x = x_d\left(1 + a_1 r^2 + a_2 r^4\right)$$
$$y = y_d\left(1 + a_1 r^2 + a_2 r^4\right)$$

Camera calibration becomes a non linear problem…