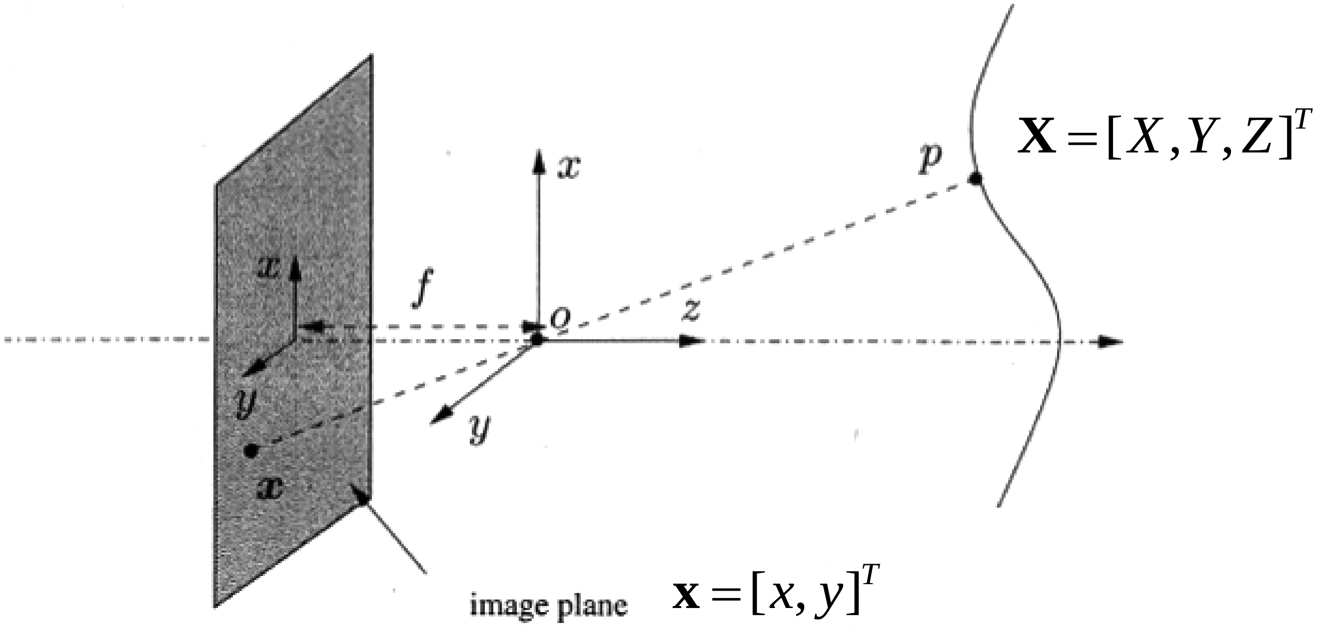
Image formation, camera model

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

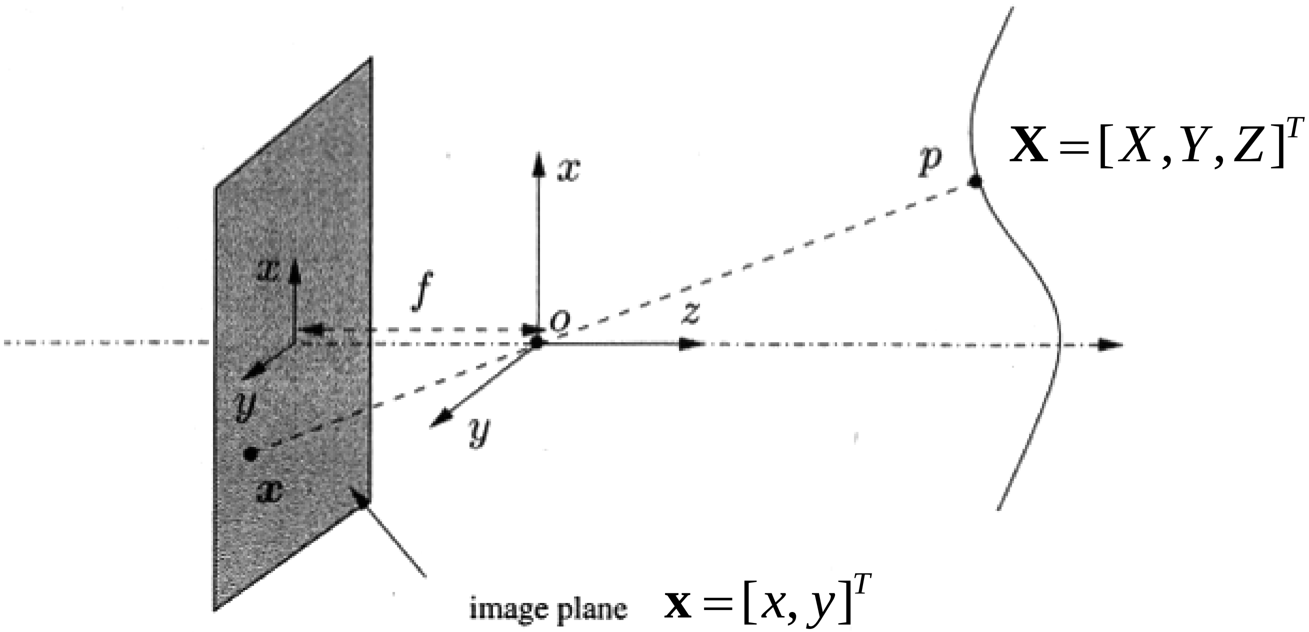


$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ z = \lambda Z \end{cases}$$

See: Forsyth and Ponce, Computer Vision a Modern Approach

Image formation, camera model

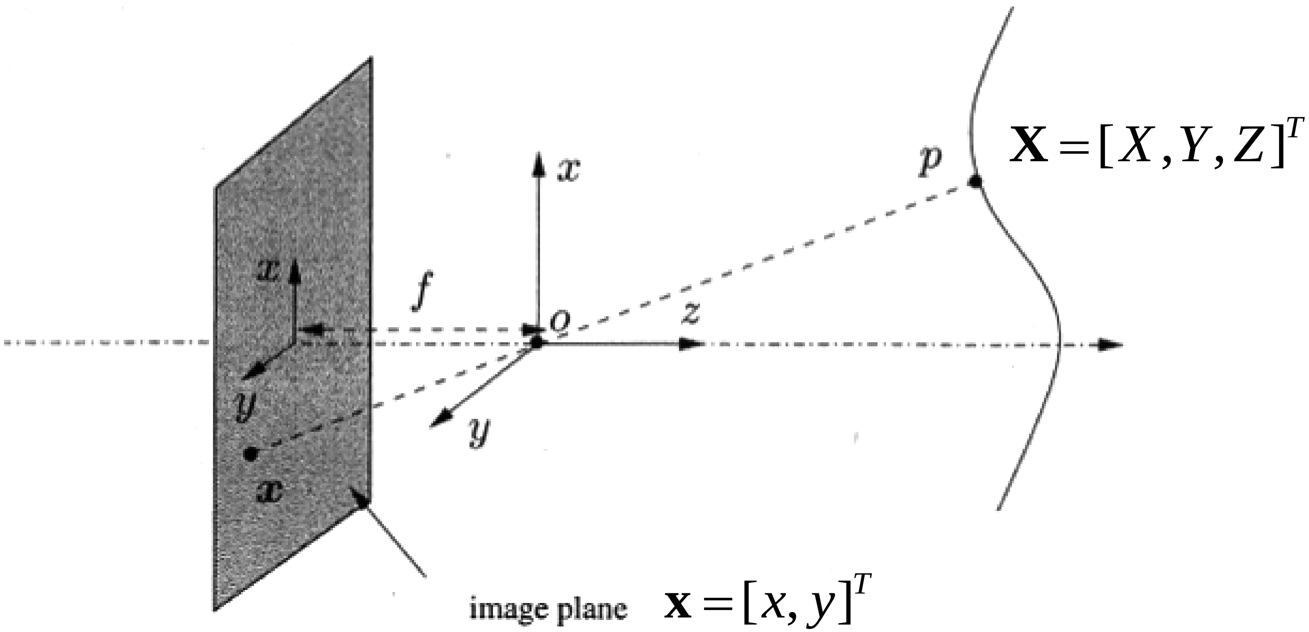
Consider a pinhole camera, force all rays to go through the optical center



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

Image formation, camera model

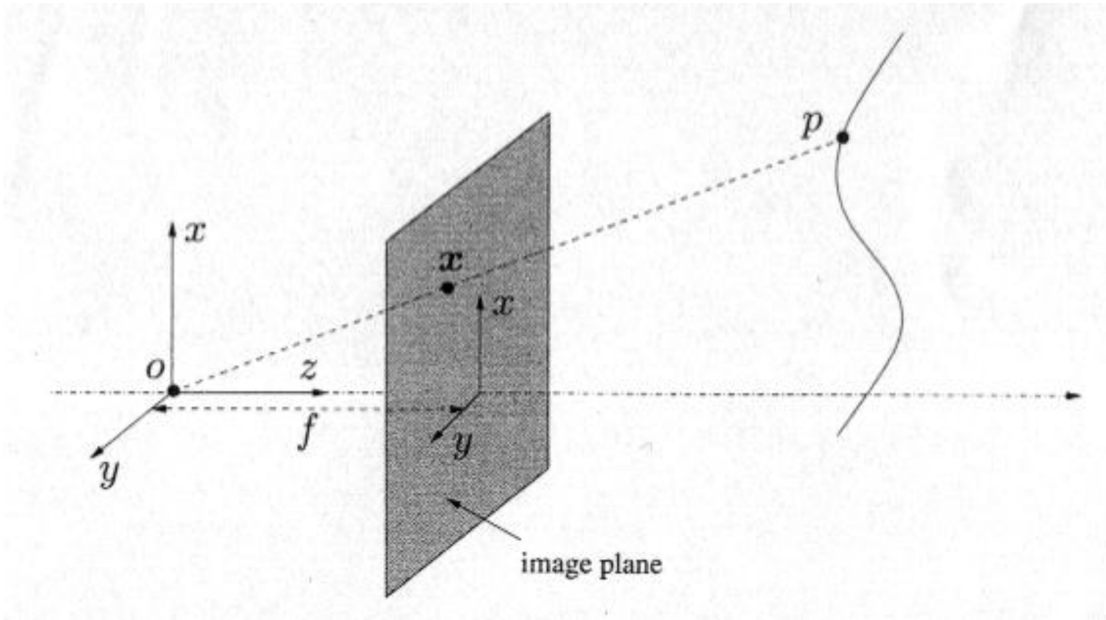
Consider a pinhole camera, force all rays to go through the optical center



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

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

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

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

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

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

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

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

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

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

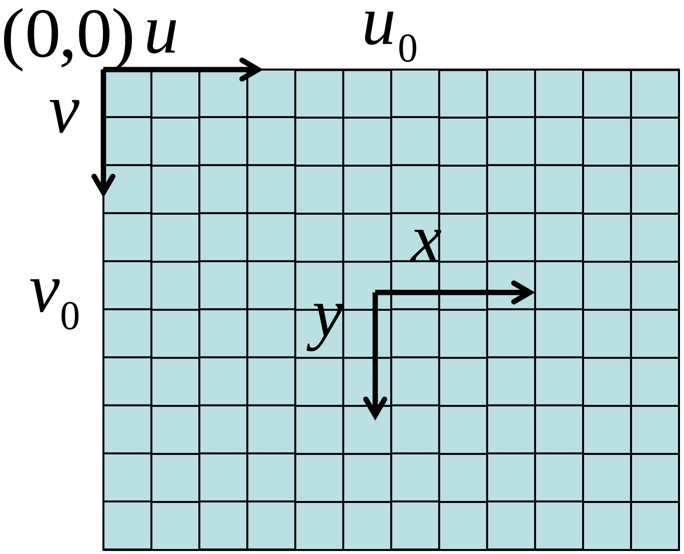
consider a point in the world reference frame

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

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

geometric model for *an ideal camera*

Intrinsic parameters



Pixel size:

$$\frac{1}{k} \times \frac{1}{l} \quad \text{units: [pixels/m]}$$

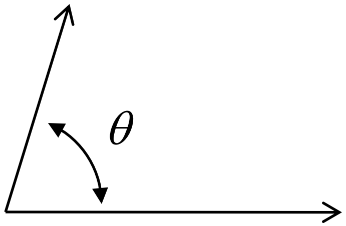
$$u = k \cdot x + u_0$$

$$v = l \cdot x + v_0$$

$$\mathbf{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k & 0 & u_0 \\ 0 & l & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

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

$$\mathbf{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k & s_\theta & u_0 \\ 0 & l & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



where s_θ is called *skew factor*

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

$$\mathbf{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k & s_\theta & u_0 \\ 0 & l & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where s_θ is called *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:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} k & s_\theta & u_0 \\ 0 & l & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

$$K = K_s \cdot K_f = \begin{bmatrix} kf & fs_\theta & u_0 \\ 0 & lf & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha & \gamma_\theta & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- To summarize:

extrinsic parameters

$$\lambda \mathbf{p} = K M_0 g \mathbf{X}_0 = M \mathbf{P}$$

$K = K_s K_f$ *intrinsic parameters*

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

Projection matrix: characterization

- The projection matrix can be written explicitly as a function of its five intrinsic parameters and the six extrinsic ones (we skip the details):

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

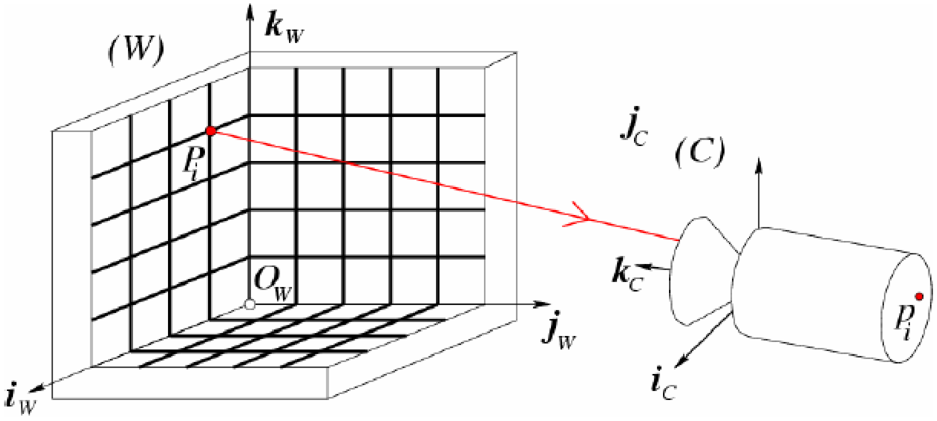
$\mathbf{r}_1^T, \mathbf{r}_2^T, \mathbf{r}_3^T$ denote the three rows of the matrix $\mathbf{R}$

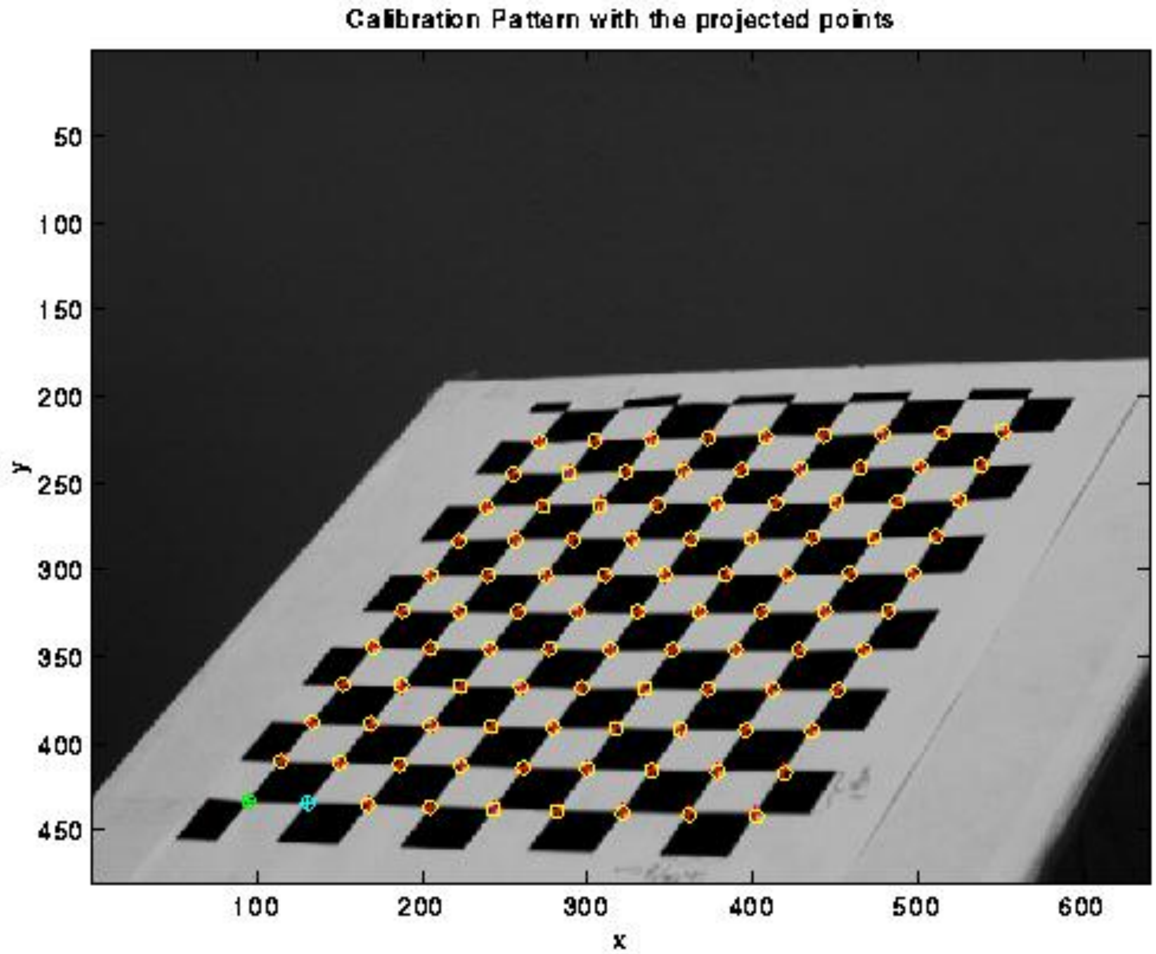
t_x, t_y, t_z are the coordinates of the vector $\mathbf{T}$

- *M* is a 3x4 matrix
- Given the structure of **R**, *M* has 11 degrees of freedom: 5 intrinsic parameters + 6 extrinsic ones (3 for rotation and 3 for translation)

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
- These points can be localized automatically or manually
- Goal: 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




Calibration Pattern with the projected points
50
100
150
200
250
300
350
400
450
y
100
200
300
400
500
600
x

Linear Approach

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \Leftrightarrow \lambda \mathbf{p} = M\mathbf{P}$$

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{13} \\ m_{21} & m_{22} & m_{23} & m_{23} \\ m_{31} & m_{32} & m_{33} & m_{31} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1^T \\ \mathbf{m}_2^T \\ \mathbf{m}_3^T \end{bmatrix}$$

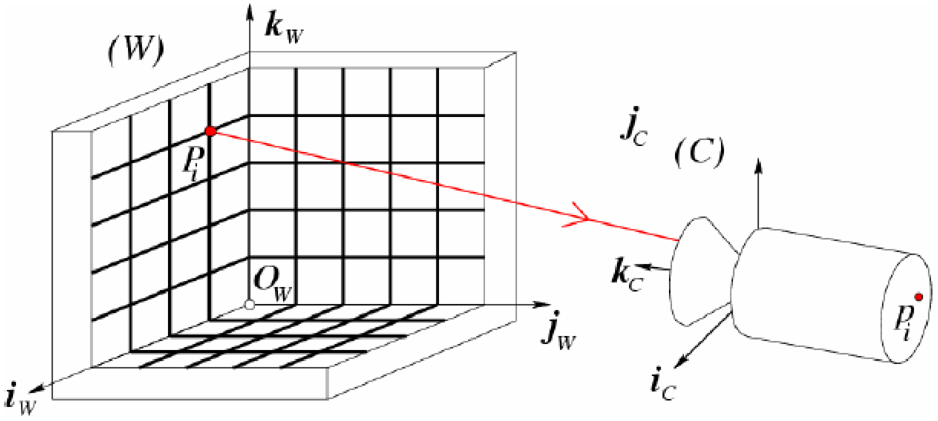
$$\mathbf{m}_i \overset{def}{=} \begin{bmatrix} m_{i,1} \\ m_{i,2} \\ m_{i,3} \end{bmatrix}$$

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

$$\lambda = \mathbf{m}_3 \mathbf{P} \Rightarrow \begin{cases} u = \dfrac{\mathbf{m_1} \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \\ v = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}}{\mathbf{m}_3 \cdot \mathbf{P}} \end{cases}$$

- *M* is the projection matrix it contains extrinsic and intrinsic parameters of the camera
- Algorithms for camera calibration usually consists in two steps:
  1. *Estimate M*
  2. *Reconstruct* the camera parameters from *M*

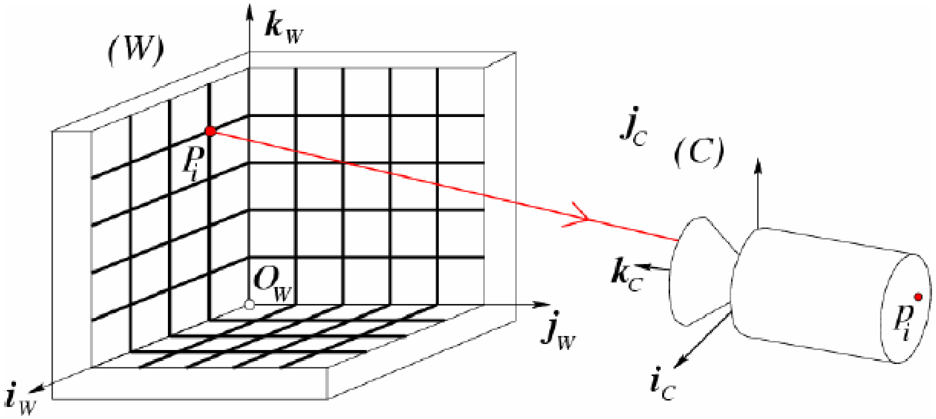
Linear Approach



Consider a set of *n* points with *known* position P_i, and projection u_i,v_i

$$\begin{cases} u_i = \dfrac{\mathbf{m}_1 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \\ v_i = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \end{cases}$$

Linear Approach



Consider a set of *n* points with *known* position P_i, and projection u_i,v_i

For each point *i* we get two equations

$$\begin{cases} u_i = \dfrac{\mathbf{m_1} \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \\ v_i = \dfrac{\mathbf{m}_2 \cdot \mathbf{P}_i}{\mathbf{m}_3 \cdot \mathbf{P}_i} \end{cases} \Rightarrow \begin{cases} (\mathbf{m_1} - u_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0 \\ (\mathbf{m}_2 - v_i \mathbf{m}_3) \cdot \mathbf{P}_i = 0 \end{cases}$$

We organize the equations in matrix form:

$$\begin{cases} u_i \mathbf{m}_3 \mathbf{P}_i = \mathbf{m}_1 \mathbf{P}_i \\ v_i \mathbf{m}_3 \mathbf{P}_i = \mathbf{m}_2 \mathbf{P}_i \end{cases}$$

$$\begin{cases} u_i m_{31} X_i + u_i m_{32} Y_i + u_i m_{33} Z_i + u_i m_{34} = m_{11} X_i + m_{12} Y_i + m_{13} Z_i + m_{14} \\ v_i m_{31} X_i + v_i m_{32} Y_i + v_i m_{33} Z_i + v_i m_{34} = m_{21} X_i + m_{22} Y_i + m_{23} Z_i + m_{24} \end{cases}$$

$$\begin{cases} -(m_{11} X_i + m_{12} Y_i + m_{13} Z_i + m_{14}) + u_i m_{31} X_i + u_i m_{32} Y_i + u_i m_{33} Z_i + u_i m_{34} = 0 \\ -(m_{21} X_i + m_{22} Y_i + m_{23} Z_i + m_{24}) + v_i m_{31} X_i + v_i m_{32} Y_i + v_i m_{33} Z_i + v_i m_{34} = 0 \end{cases}$$

in matrix form:

$$\begin{bmatrix} -X_i & -Y_i & -Z_i & -1 & 0 & 0 & 0 & 0 & u_i X_i & u_i Y_i & u_i Z_i & u_i \\ 0 & 0 & 0 & 0 & -X_i & -Y_i & -Z_i & -1 & v_i X_i & v_i Y_i & u_i Z_i & v_i \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ \ldots \\ m_{33} \\ m_{34} \end{bmatrix} = 0$$

Collecting *n* points leads to 2n equations:

$$\begin{bmatrix} -X_0 & -Y_0 & -Z_0 & -1 & 0 & 0 & 0 & 0 & u_0X_0 & u_0Y_0 & u_0Z_0 & u_0 \\ 0 & 0 & 0 & 0 & -X_0 & -Y_0 & -Z_0 & -1 & v_0X_0 & v_0Y_0 & u_0Z_0 & v_0 \\ -X_1 & -Y_1 & -Z_1 & -1 & 0 & 0 & 0 & 0 & u_1X_1 & u_1Y_1 & u_1Z_1 & u_1 \\ 0 & 0 & 0 & 0 & -X_1 & -Y_1 & -Z_1 & -1 & v_1X_1 & v_1Y_1 & v_1Z_1 & v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -X_{n-1} & -Y_{n-1} & -Z_{n-1} & -1 & 0 & 0 & 0 & 0 & u_{n-1}X_{n-1} & u_{n-1}Y_{n-1} & u_{n-1}Z_{n-1} & u_{n-1} \\ 0 & 0 & 0 & 0 & -X_{n-1} & -Y_{n-1} & -Z_{n-1} & -1 & v_{n-1}X_{n-1} & v_{n-1}Y_{n-1} & v_{n-1}Z_{n-1} & v_{n-1} \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ \ldots \\ m_{33} \\ m_{34} \end{bmatrix} = \mathbf{0}$$

$$\text{in compact form}:$$

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

- m is 12x1 (M has 12 coefficients)
- W is a 2nx12 matrix
- When *n* is large (>6) *homogeneous least-squares* can be used to determine **m** (and the projection matrix **M**), the solution is the eigenvector of W^TW such that **|m|**=1 (more on next slide)

Homogeneous Least-squares

- Consider the following problem

$$\begin{cases} u_{11}x_1 + u_{12}x_2 + \ldots + u_{1q}x_q = 0 \\ u_{21}x_1 + u_{22}x_2 + \ldots + u_{2q}x_q = 0 \\ \ldots \\ u_{p1}x_1 + u_{p2}x_2 + \ldots + u_{pq}x_q = 0 \end{cases} \Leftrightarrow \mathbf{U} \cdot \mathbf{x} = 0$$

- x=0 is always a solution (not interesting)
- To find a non trivial solution we set the additional constraint:

$$\left\| \mathbf{x} \right\| = 1$$

- The problem becomes:

$$E(\mathbf{x}) = |\mathbf{U}\mathbf{x}|^2 = \mathbf{x}^T \mathbf{U}^T \mathbf{U}\mathbf{x}$$

$$\min(E(\mathbf{x})) \quad \text{s.t.} |\mathbf{x}| = 1$$

- $U^T U$ is a symmetric positive semidefinite qxq matrix
- It can be diagonalized:

$$\mathbf{e}_i \quad i = 1, \dots, q$$

$$0 \leq \lambda_1 \leq \dots \leq \lambda_q$$

$$\mathbf{x} = u_1 \mathbf{e}_1 + \dots + u_q \mathbf{e}_q$$

$$u_1{}^2 + u_2{}^2 + \dots + u_q{}^2 = 1$$

$$E(\mathbf{x}) = \mathbf{x}^T U^T U \mathbf{x} = \lambda_1 {u_1}^2 + \ldots + \lambda_q {u_q}^2$$

$$E(\mathbf{e}_1) = {\mathbf{e}_1}^T U^T U \mathbf{e}_1 = \lambda_1$$

$$E(\mathbf{x}) - E(\mathbf{e}_1) = \mathbf{x}^T U^T U \mathbf{x} - {\mathbf{e}_1}^T U^T U \mathbf{e}_1 =$$

$$= \lambda_1 {u_1}^2 + \ldots + \lambda_q {u_q}^2 - \lambda_1 \geq \lambda_1 ({u_1}^2 + \ldots + {u_q}^2 - 1) = 0$$

$$E(\mathbf{x}) \geq E(\mathbf{e}_1) = \lambda_1$$

The unit vector $\mathbf{x}$ minimizing $E(\mathbf{x})$ is the eigenvector $\mathbf{e_1}$ associated with the minimum eigenvalue of $U^T U$.

The corresponding minimum value of E is λ_1

The problem can be solved using any technique for computing eigenvectors and eigenvalues. SVD in particular allows computing eigenvalues and eigenvector without constructing $U^T U$

Reconstruction of intrinsic and extrinsic parameters

Once the projection matrix *M* is estimated we can use its expression

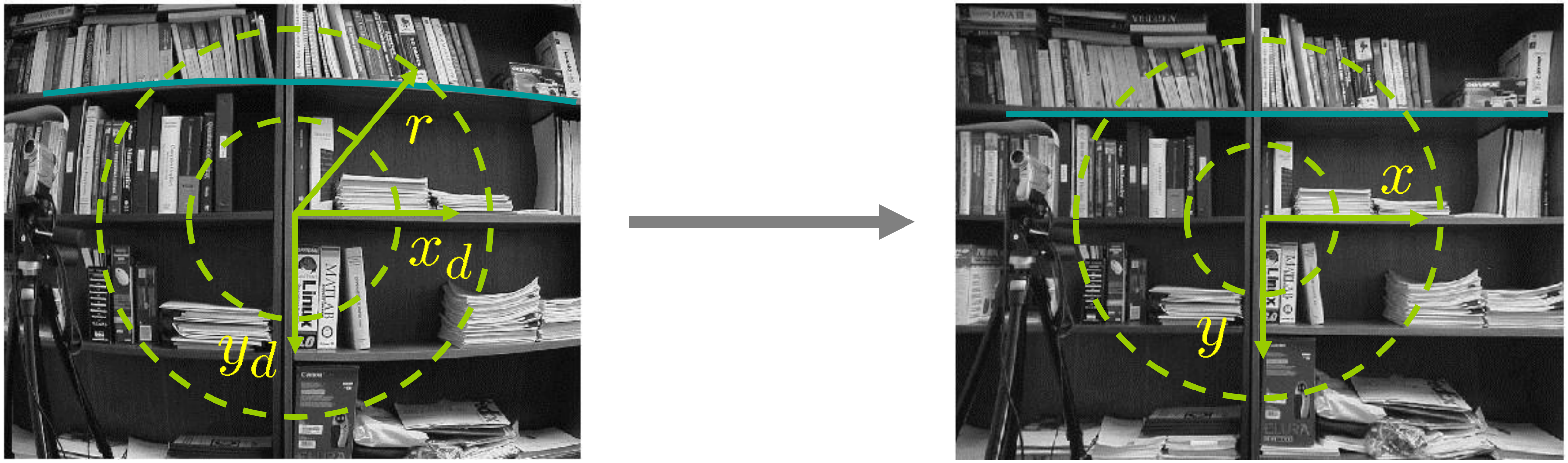
In a simple case in which theta=0, we get:

$$\begin{aligned}
\mathbf{r}_3 &= m_{34}\mathbf{m}_3 \\
u_0 &= (\alpha\mathbf{r}_1^T + u_0\mathbf{r}_3^T)\mathbf{r}^3 = m_{34}^2\mathbf{m}_1^T\mathbf{m}_3 \\
v_0 &= (\beta\mathbf{r}_2^T + v_0\mathbf{r}_3^T)\mathbf{r}^3 = m_{34}^2\mathbf{m}_2^T\mathbf{m}_3 \\
\alpha &= m_{34}^2|\mathbf{m}_1 \times \mathbf{m}^3| \\
\beta &= m_{34}^2|\mathbf{m}_2 \times \mathbf{m}^3|
\end{aligned}$$

$$\begin{aligned}
\mathbf{r}_1 &= \frac{m_{34}}{\alpha}(\mathbf{m}_1 - u_0\mathbf{m}_3) \\
\mathbf{r}_2 &= \frac{m_{34}}{\beta}(\mathbf{m}_2 - v_0\mathbf{m}_3) \\
t_z &= m_{34} \\
t_x &= \frac{m_{34}}{\alpha}(m_{14} - u_0) \\
t_y &= \frac{m_{34}}{\beta}(m_{24} - v_0)
\end{aligned}$$

See Forsyth & Ponce for details and skew-angle case.

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…