Lecture 8: Camera Calibration

Professor Fei-Fei Li

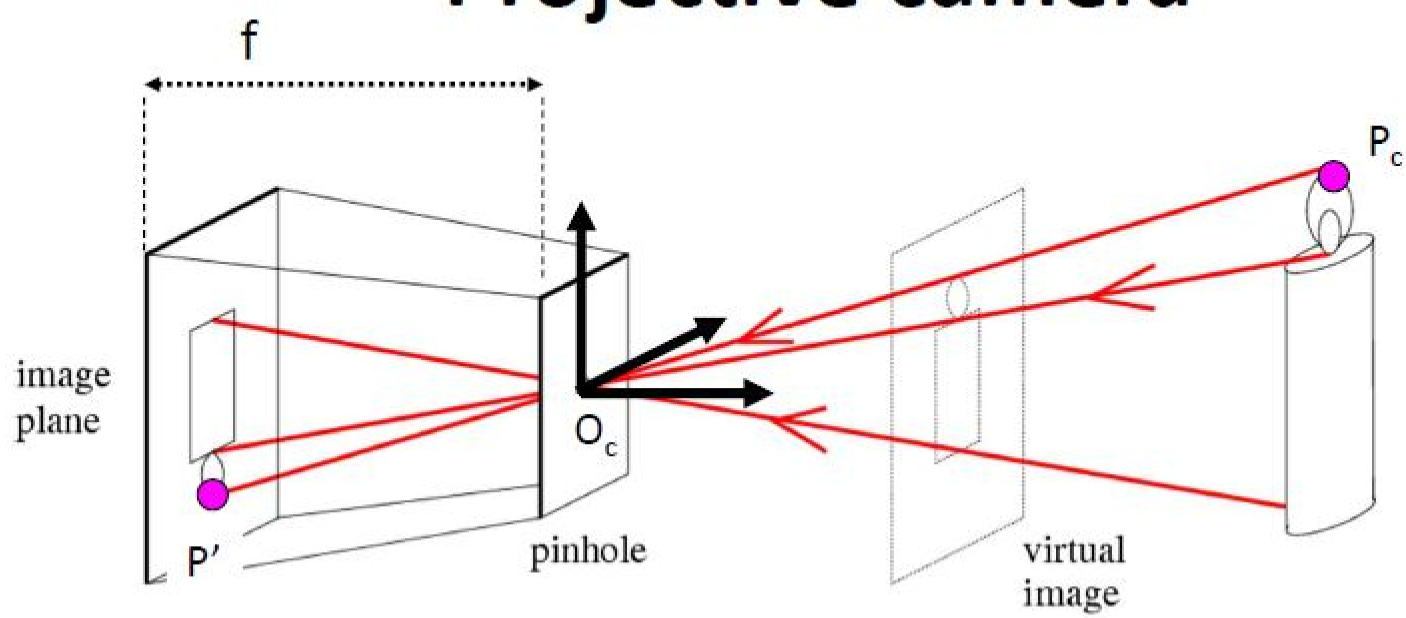
Stanford Vision Lab

What we will learn today?

- Review camera parameters
- Affine camera model
- Camera calibration
- Vanishing points and lines

Reading:
- [FP] Chapter 3
- [HZ] Chapter 7, 8.6

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

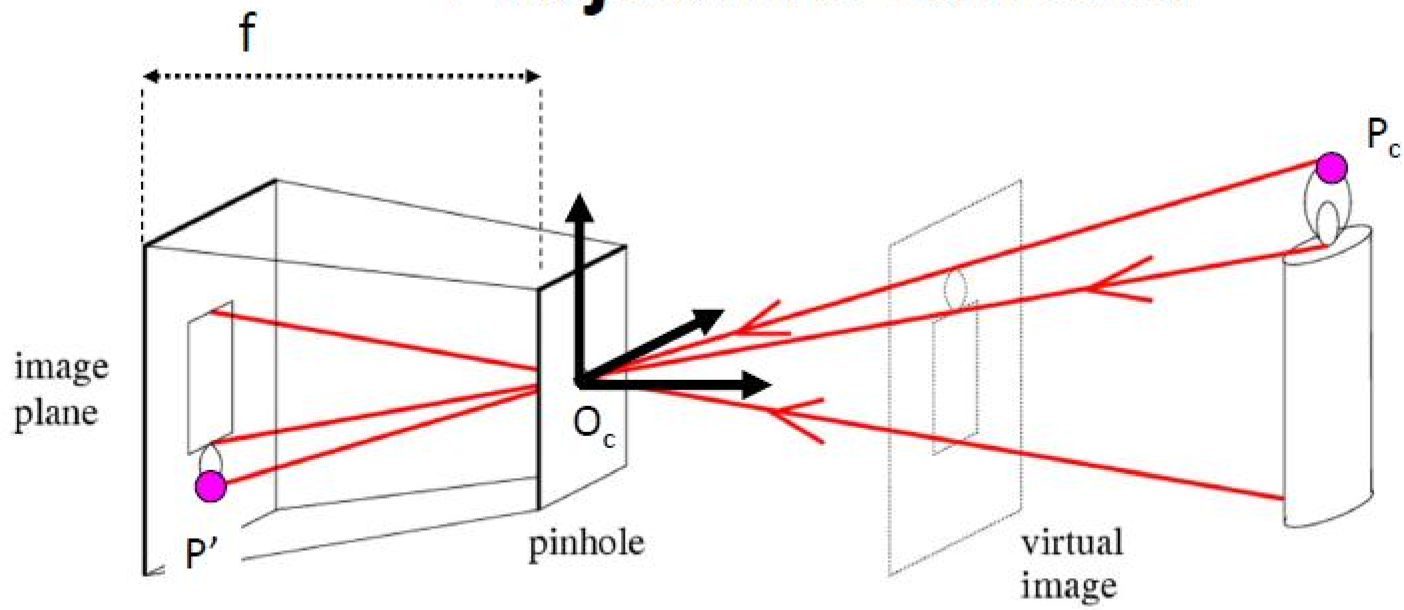
u_o, v_o = offset

$\boldsymbol{\alpha}, \boldsymbol{\beta} \rightarrow$ non-square pixels

θ = skew angle

K has 5 degrees of freedom!

Projective camera



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

K has 5 degrees of freedom!

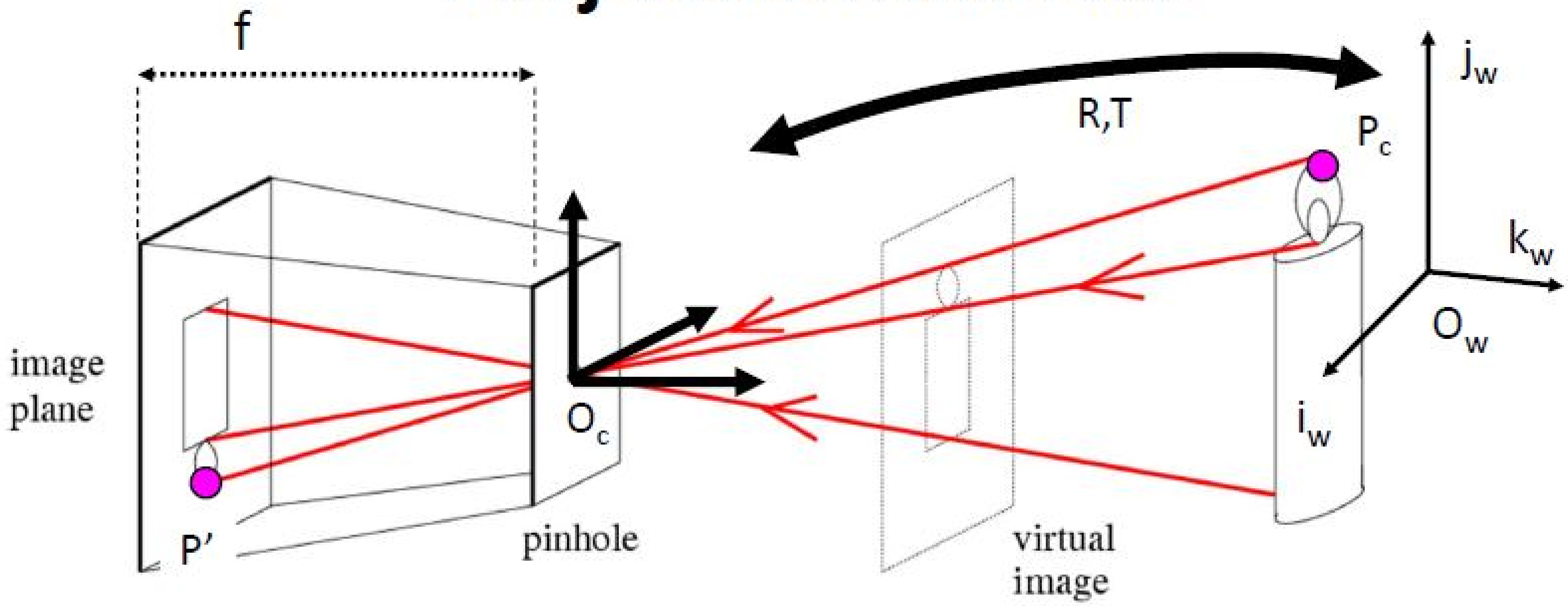
f = focal length

u_o, v_o = offset

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

θ = skew angle

Projective camera



$$P = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4\times4} P_w$$

$$T = -R\widetilde{O}_c$$

??? $\widetilde{O}_c$???

f = focal length

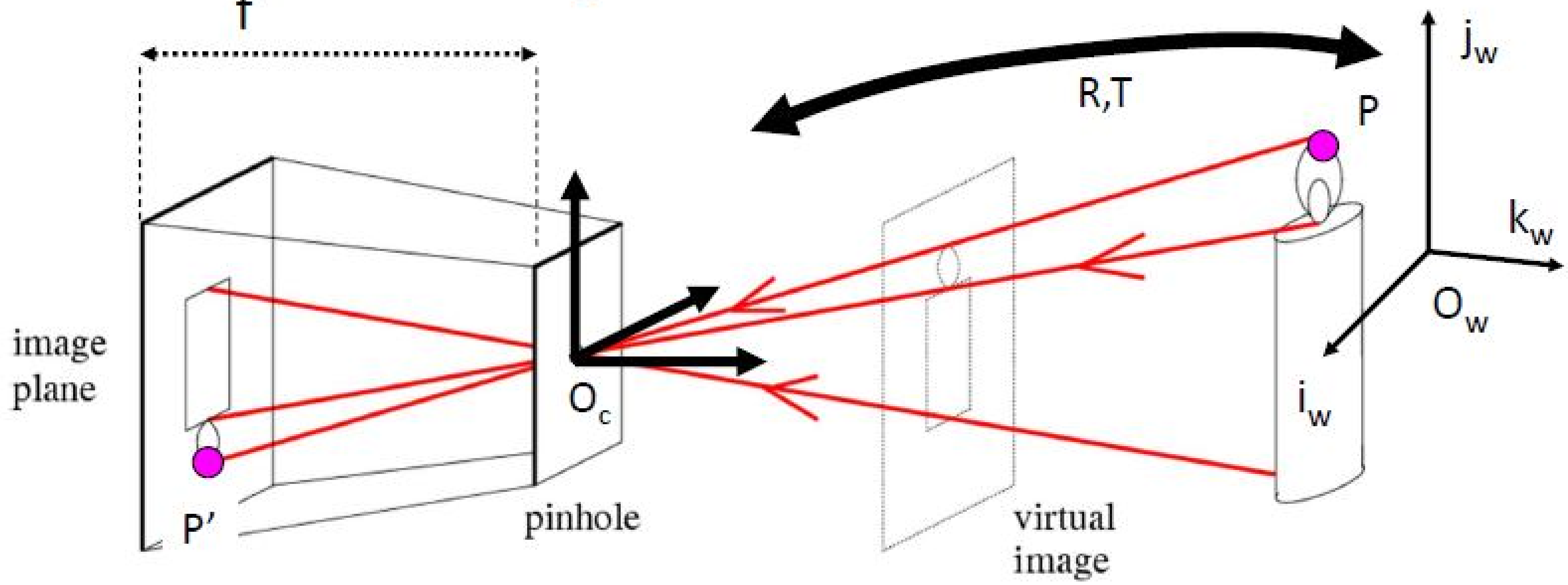
u_o, v_o = offset

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

θ = skew angle

R,T = rotation, translation

Projective camera



$$P' = M\,P_w$$

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

f = focal length

u_o, v_o = offset

$\boldsymbol{\alpha}, \boldsymbol{\beta} \rightarrow$ non-square pixels

θ = skew angle

R,T = rotation, translation

Goal of calibration

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

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

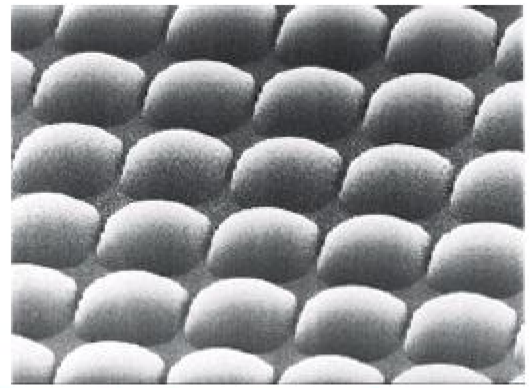
$$K = \begin{bmatrix} \boldsymbol{\alpha} & -\boldsymbol{\alpha}\cot\boldsymbol{\theta} & u_o \\ 0 & \dfrac{\boldsymbol{\beta}}{\sin\boldsymbol{\theta}} & v_o \\ 0 & 0 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} \qquad T = \begin{bmatrix}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

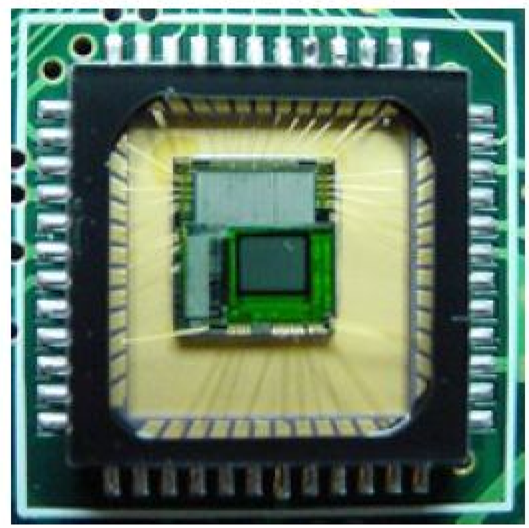
Estimate intrinsic and extrinsic parameters
from 1 or multiple images

How Cameras Produce Images

- Basic process:
  - photons hit a detector
  - the detector becomes charged
  - the charge is read out as brightness

- Sensor types:
  - CCD (charge-coupled device)
    - high sensitivity
    - high power
    - cannot be individually addressed
    - blooming
  - CMOS
    - most common
    - simple to fabricate (cheap)
    - lower sensitivity, lower power
    - can be individually addressed





Images are two-dimensional patterns of brightness values.

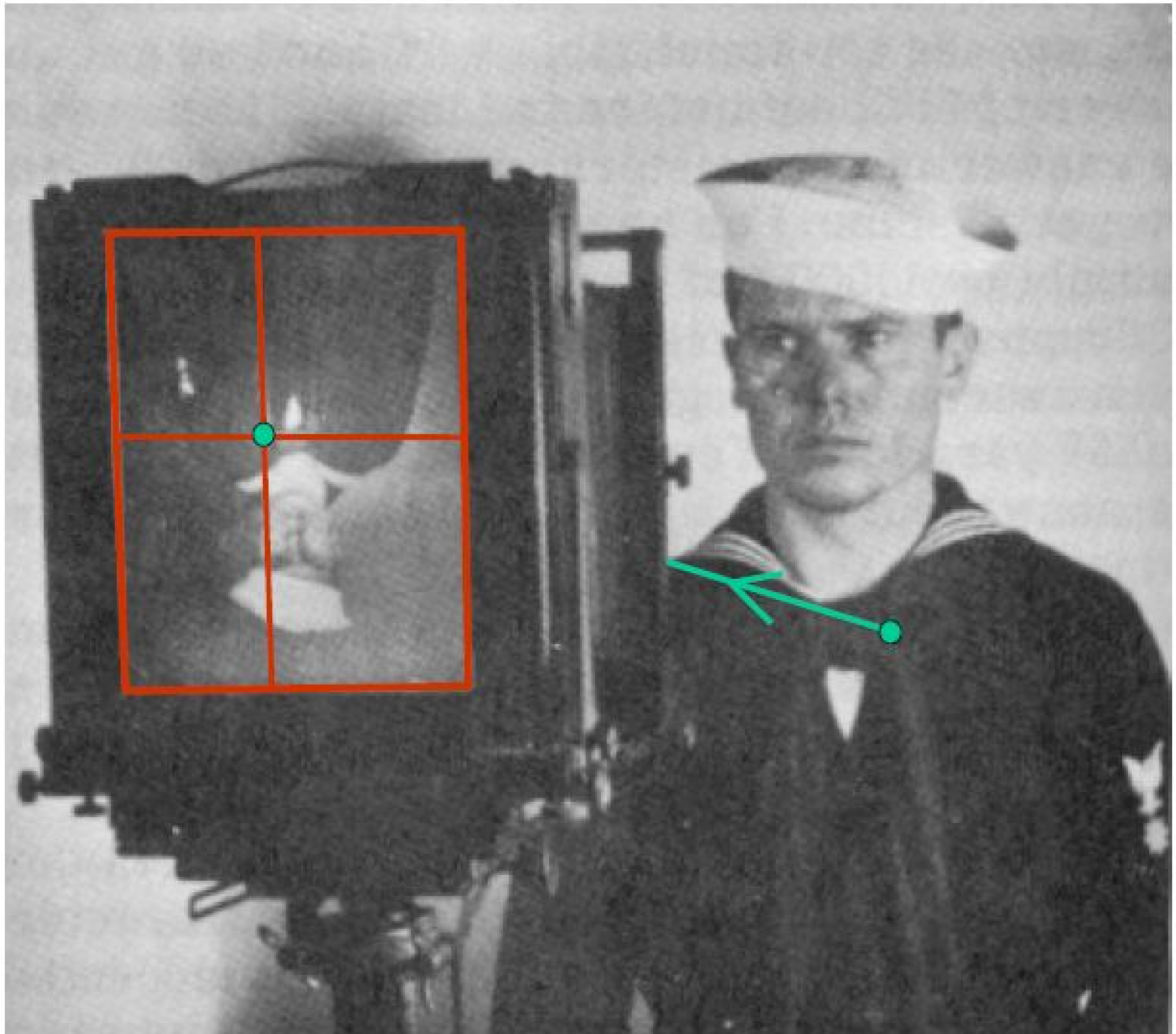
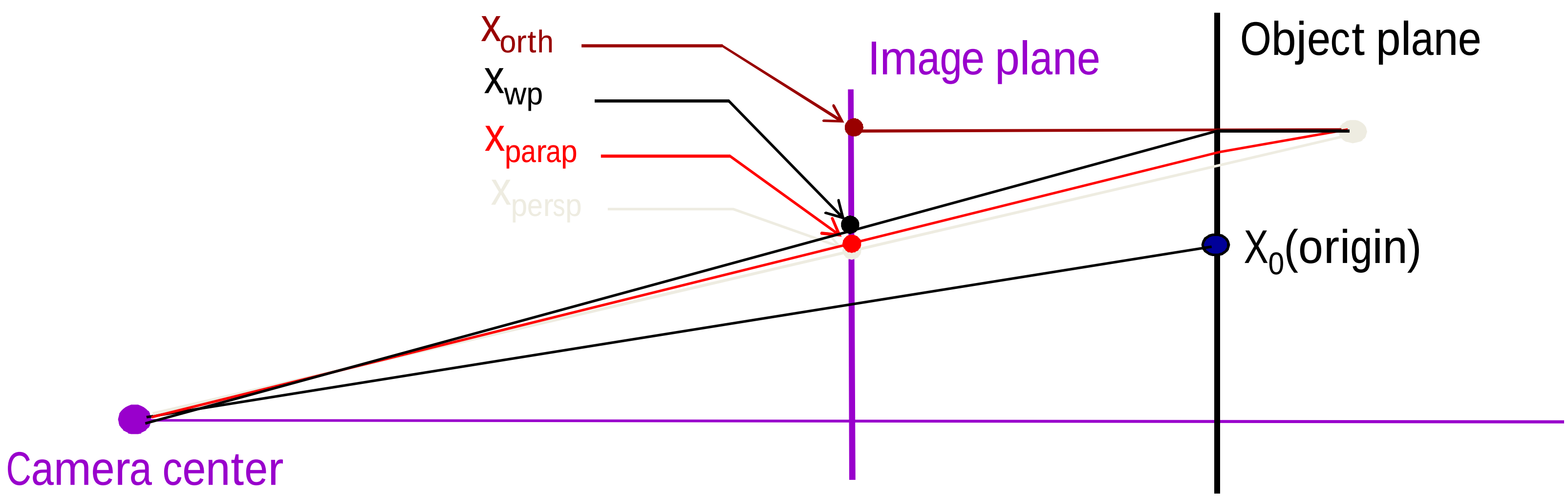


Figure from US Navy Manual of Basic Optics and Optical Instruments, prepared by Bureau of Naval Personnel. Reprinted by Dover Publications, Inc., 1969.

They are formed by the projection of 3D objects.

Hierarchy of cameras

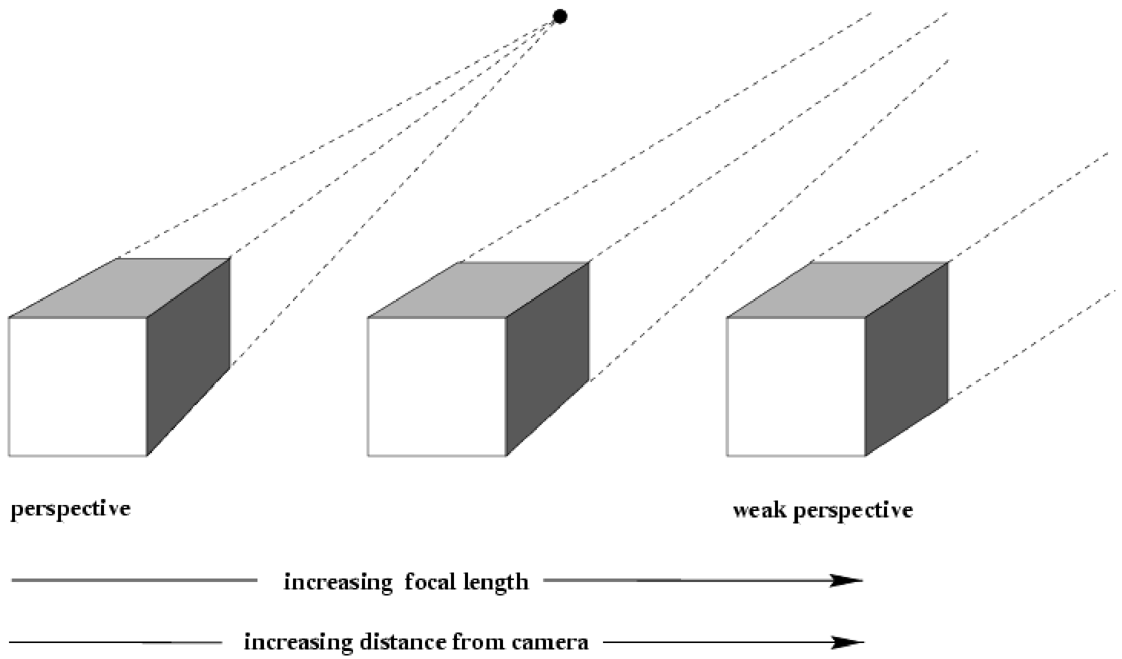


Examples of camera projections

perspective

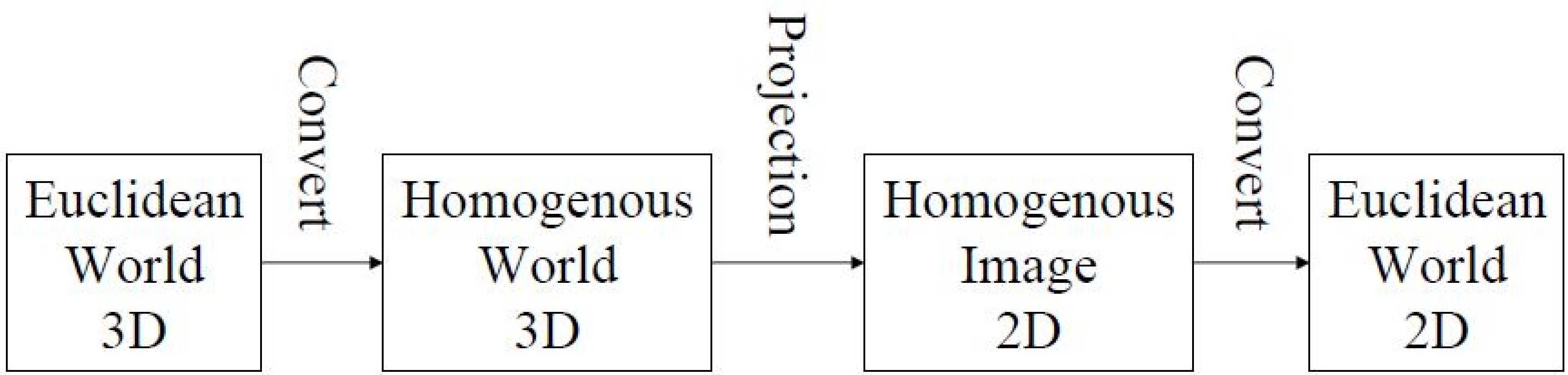
Orthographic
(parallel)

Affine cameras



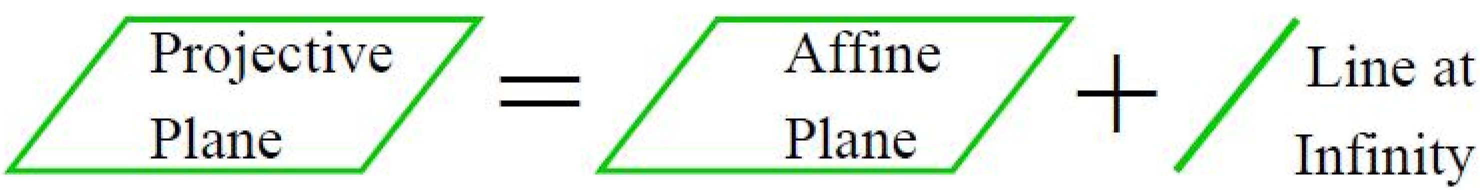
Homogenous coordinates

- Our usual coordinate system is called a Euclidean or affine coordinate system
- Rotations, translations and projection in Homogenous coordinates can be expressed linearly as matrix multiplies



Projective Geometry

- Axioms of Projective Plane
  1. Every two distinct points define a line
  2. Every two distinct lines define a point (intersect at a point)
  3. There exists three points, A,B,C such that C does not lie on the line defined by A and B.
- Different than Euclidean (affine) geometry
- Projective plane is “bigger” than affine plane – includes “line at infinity”



Homogenous coordinates

A way to represent points in a projective space

1. Add an extra coordinate

   e.g., (x,y) -> (x,y,1)=(u,v,w)

2. Impose equivalence relation such that (λ not 0)

   $$(u,v,w) \approx \lambda^*(u,v,w)$$

   i.e., $(x,y,1) \approx (\lambda x, \lambda y, \lambda)$

3. "Point at infinity" – zero for last coordinate

   e.g., (x,y,0)

- Why do this?
  - Possible to represent points "at infinity"
    - Where parallel lines intersect
    - Where parallel planes intersect
  - Possible to write the action of a perspective camera as a matrix

Euclidean -> Homogenous-> Euclidean

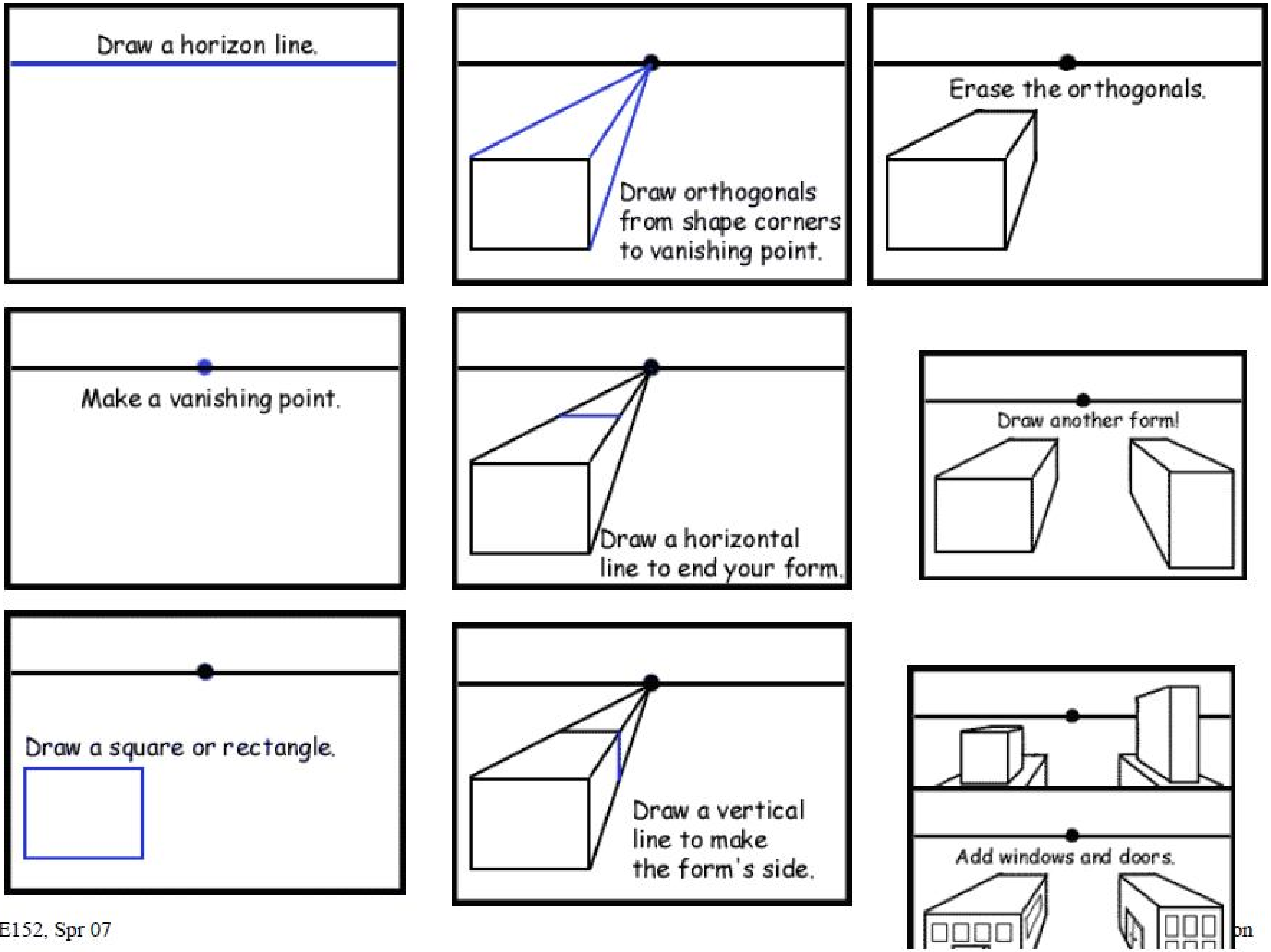
In 2-D

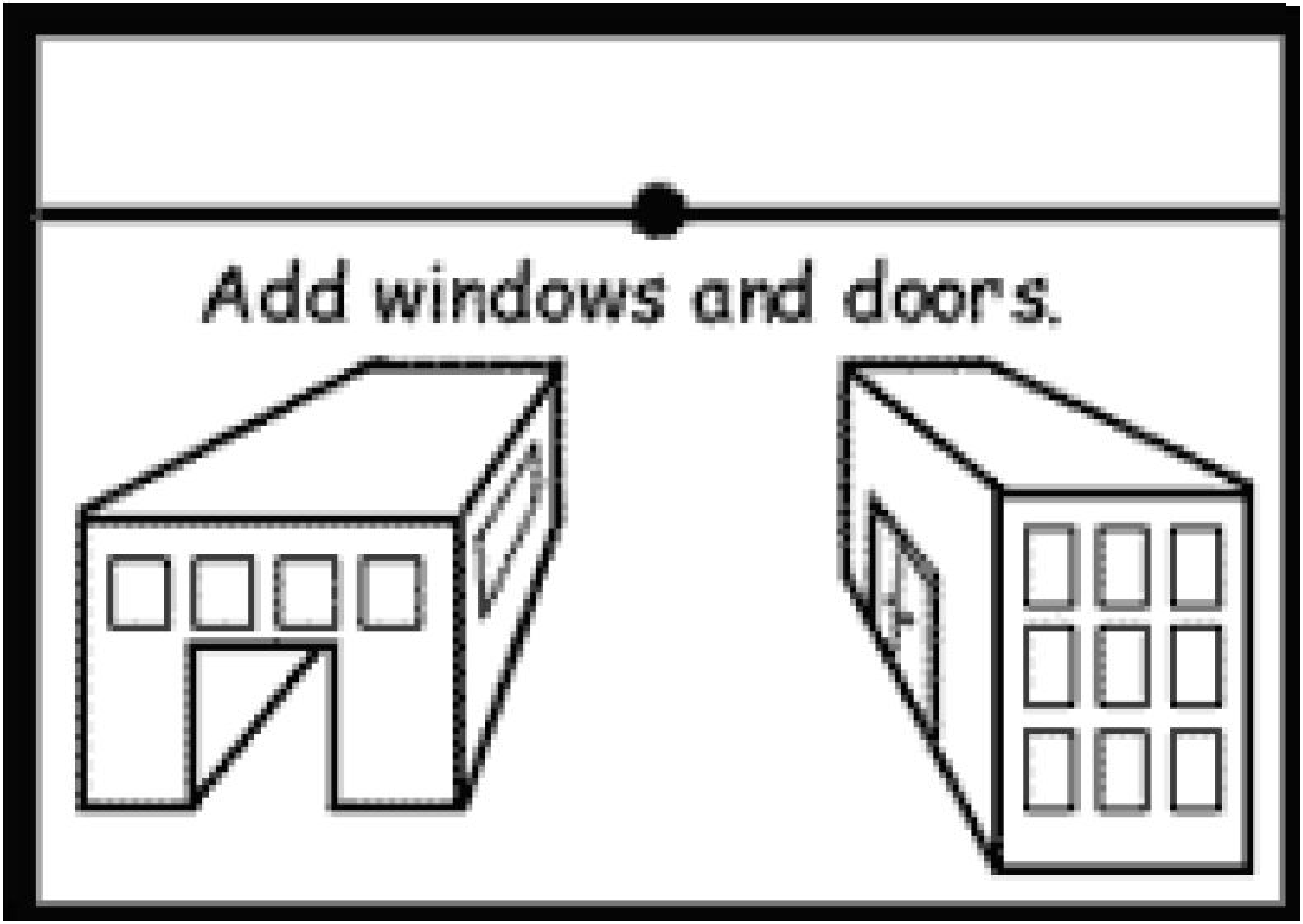
- Euclidean -> Homogenous: (x, y) -> k (x,y,1)
- Homogenous -> Euclidean: (u,v,w) -> (u/w, v/w)

In 3-D

- Euclidean -> Homogenous: (x, y, z) -> k (x,y,z,1)
- Homogenous -> Euclidean: (x, y, z, w) -> (x/w, y/w, z/w)

Projective geometry provides an elegant means for handling these different situations in a unified way and **homogenous coordinates** are a way to represent entities (points & lines) in projective spaces.


Draw a horizon line.
Make a vanishing point.
Draw a square or rectangle.
Draw orthogonals from shape corners to vanishing point.
Draw a horizontal line to end your form.
Draw a vertical line to make the form's side.
Erase the orthogonals.
Draw another form!
Add windows and doors.



http://www.sanford-artedventures.com/create/tech_1pt_perspective.html

Weak Perspective Projection

If the relative distance δz (scene depth) between two points of a 3D object along the optical axis is much smaller than the average distance $\bar{z}$ ($\delta z < \frac{\bar{z}}{20}$),

then

$$
\begin{aligned}
u &= f\frac{x}{z} \approx \frac{fx}{\bar{z}} \\
v &= f\frac{y}{z} \approx \frac{fy}{\bar{z}}
\end{aligned}
$$

We have linear equations since all projections have the same scaling factor.

Orthographic Projection

As a special case of the weak perspective projection, when $\frac{f}{z}$ factor equals 1, we have $u = x$ and $v = y$, i.e., the lins (rays) of projection are parallell to the optical axis. This leads to the sizes of image and the object are the same. This is called orthgraphic projection.

Perspective projection geometry

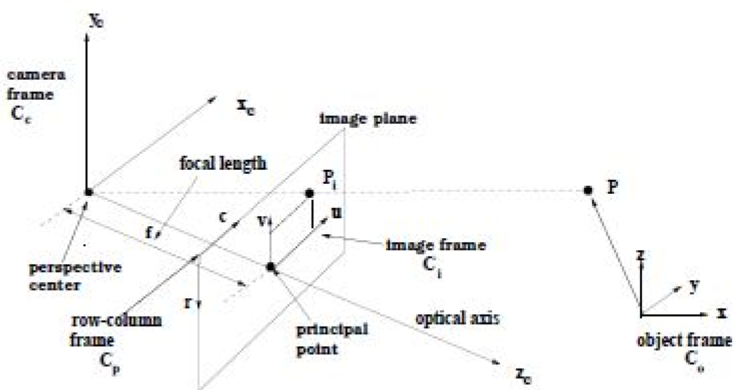


Figure 1: Perspective projection geometry

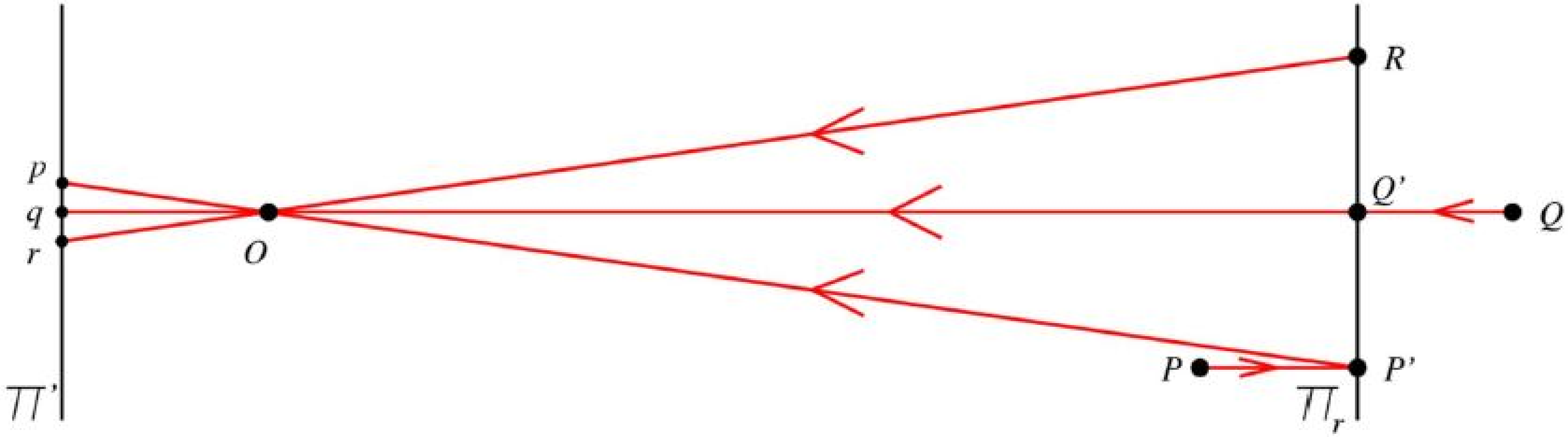
Affine Camera Model

A further simplification from weak perspective camera model is the affine camera model, which is often assumed by computer vision researchers due to its simplicity. The affine camera model assumes that the object frame is located on the centroid of the object being observed. As a result, we have $\bar{z}_c \approx t_z$, the affine perspective projection matrix is

$$
P_{affine} = \begin{pmatrix} s_x f r_1 & s_x f t_x + c_0 t_z \\ s_y f r_2 & s_y f t_y + r_0 t_z \\ 0 & t_z \end{pmatrix} \tag{11}
$$

Affine camera model represents the first order approximation of the full perspective projection camera model. It still only gives an approximation and is no longer useful when the object is close to

Weak perspective projection

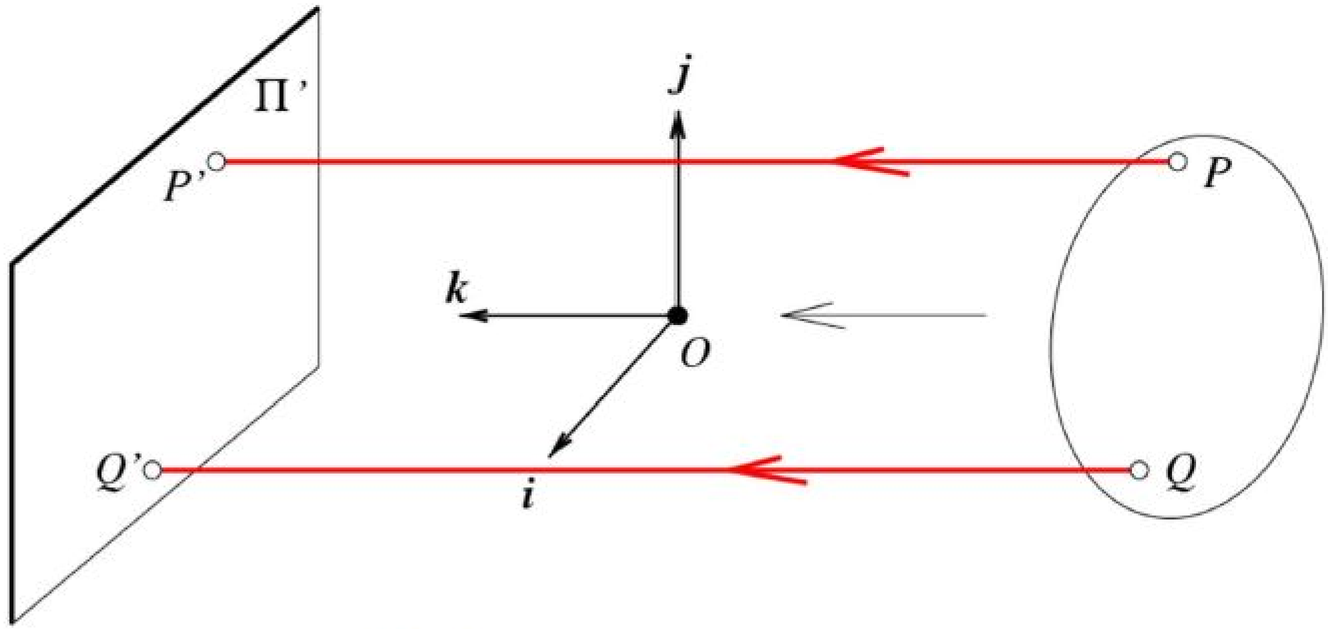


$$\boxed{\begin{cases} x' = -mx \\ y' = -my \end{cases}}$$

where $m = -\dfrac{f'}{z_0}$ = magnification

Relative scene depth is small compared to its distance from the camera

Orthographic (affine) projection



$$\begin{cases} x' = x \\ y' = y \end{cases}$$

Distance from center of projection to image plane is infinite

Affine cameras

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

Affine case

$$K = \begin{bmatrix} \alpha_x & s & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad M = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \boxed{0} & 1 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$$

Parallel projection matrix

Compared to

Projective case

$$K = \begin{bmatrix} \alpha_x & s & x_o \\ 0 & \alpha_y & y_o \\ 0 & 0 & 1 \end{bmatrix} \qquad M = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}$$

Remember....

Affinities:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ \boxed{0} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_a \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Projectivities:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ v & b \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_p \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\det(AB) = \det(A)\det(B). \quad \det(A^{-1}) = \frac{1}{\det(A)}.$$

System of linear equations, Homogeneous systems
http://en.wikipedia.org/wiki/System_of_linear_equations

Affine cameras

We can obtain a more compact formulation than: $P' = K\begin{bmatrix} R & T \end{bmatrix} P$

$$K = \begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathrm{M} = \mathrm{K}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \mathrm{R} & \mathrm{T} \\ 0 & 1 \end{bmatrix}$$

$$M = [3\times 3\,\text{affine}]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}[4\times 4\,\text{affine}] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

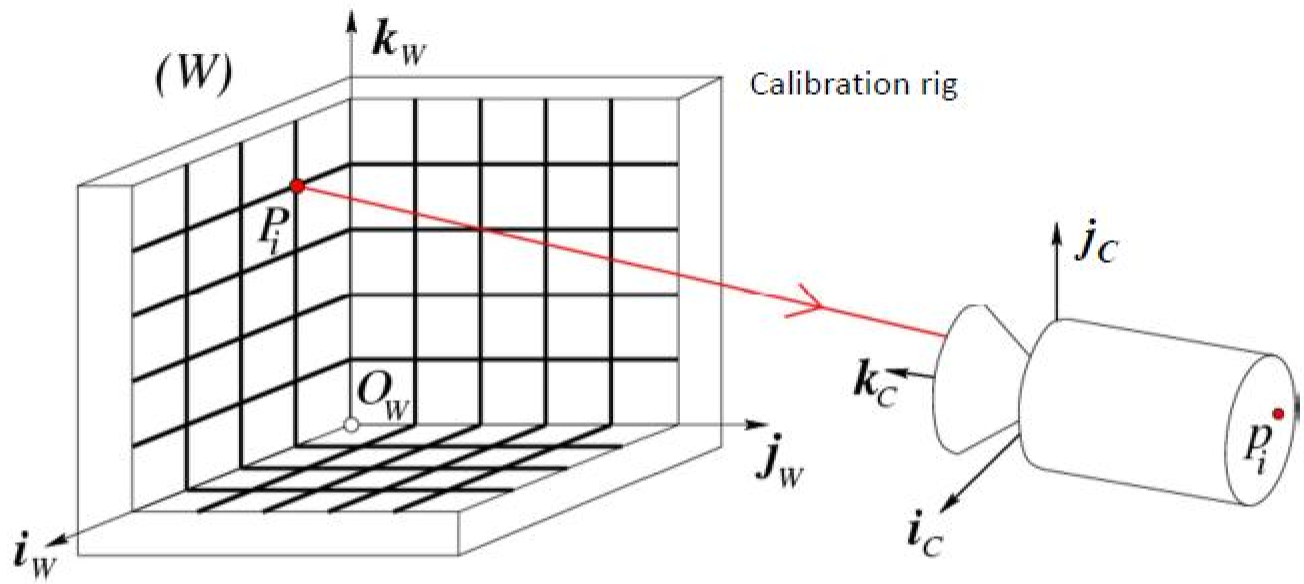
$$P' = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \mathbf{A}P + \mathbf{b} = M_{Euc}\begin{bmatrix} P \\ 1 \end{bmatrix}$$

$$\mathrm{M}_{\mathrm{Euc}} = \mathrm{M} = \begin{bmatrix} \mathrm{A} & \mathrm{b} \end{bmatrix}$$

Affine cameras

- 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.

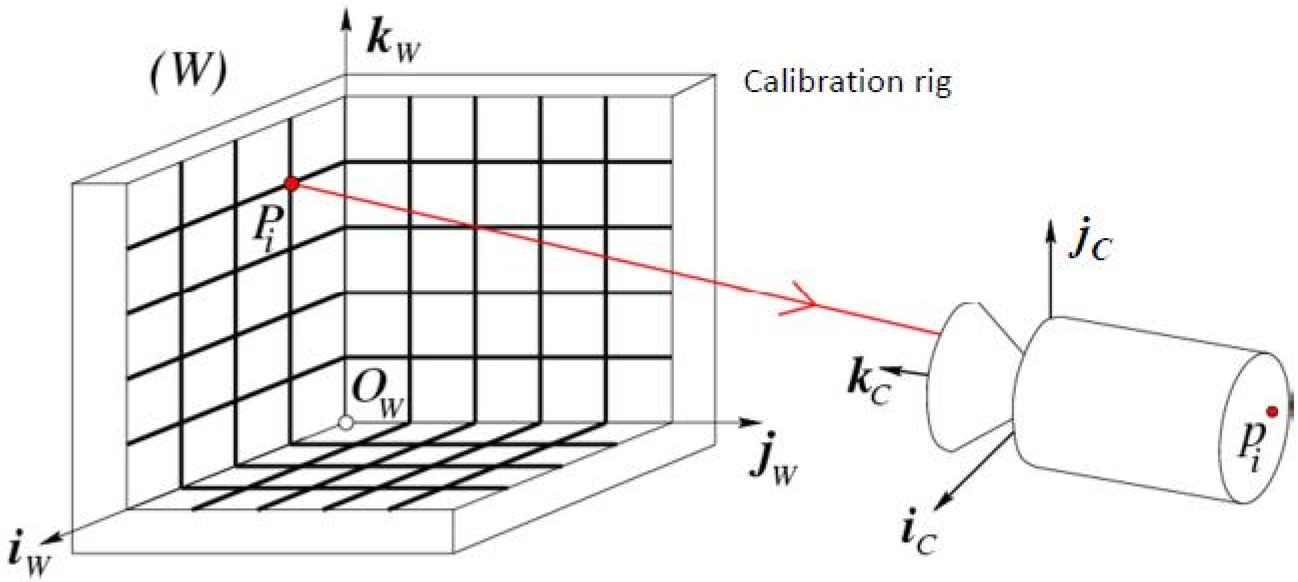
Calibration Problem



- $P_1 \ldots P_n$ 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

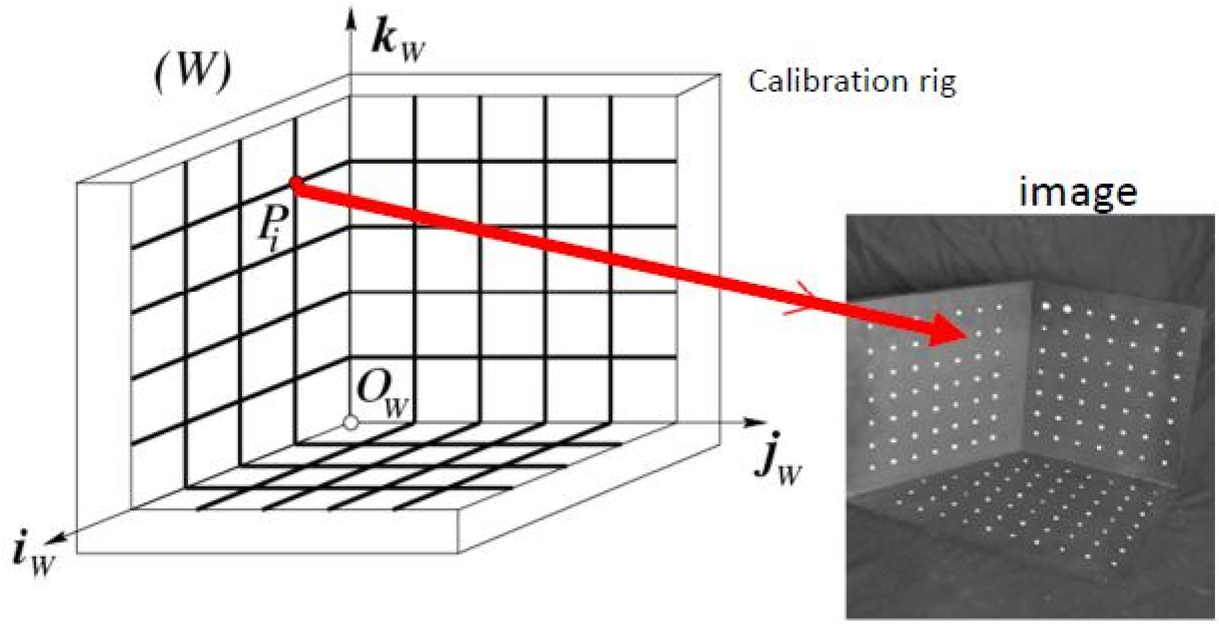
Calibration Problem



How many correspondences do we need?

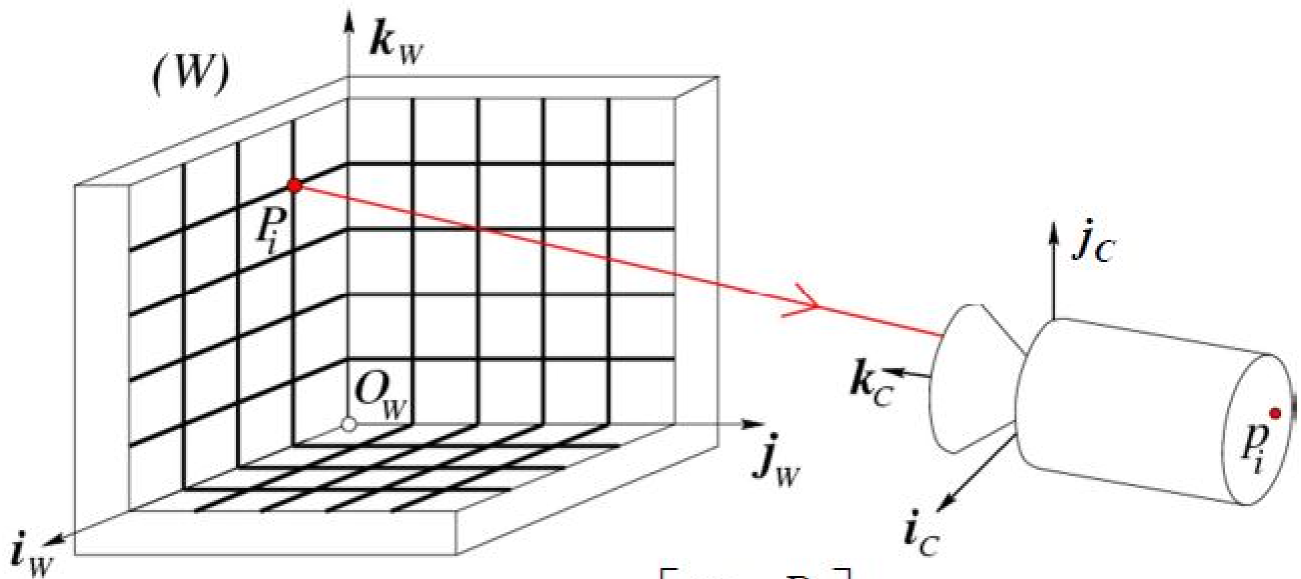
•M has 11 unknown • We need 11 equations • 6 correspondences would do it

Calibration Problem



In practice: user may need to look at the image and select the n>=6 correspondences

Calibration Problem



$$p_i \rightarrow M\ P_i \quad \rightarrow \mathrm{p}_\mathrm{i} = \begin{bmatrix} \mathrm{u}_\mathrm{i} \\ \mathrm{v}_\mathrm{i} \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{m}_1 \cdot P_i}{\mathbf{m}_3 \cdot P_i} \\ \dfrac{\mathbf{m}_2 \cdot P_i}{\mathbf{m}_3 \cdot P_i} \end{bmatrix} \qquad \mathrm{M} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

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

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

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

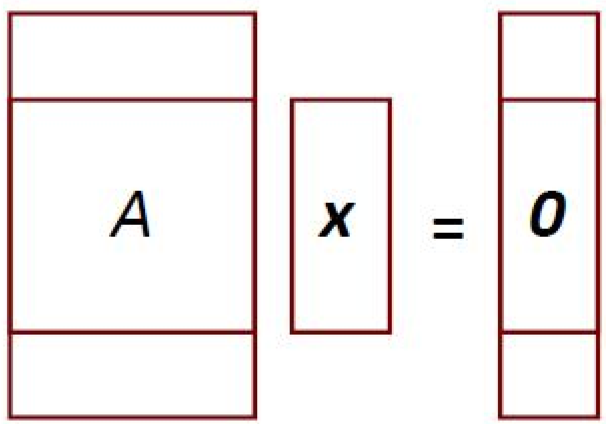
$$\begin{cases} -u_1(\mathbf{m}_3\ P_1) + \mathbf{m}_1\ P_1 = 0 \\ -v_1(\mathbf{m}_3\ P_1) + \mathbf{m}_2\ P_1 = 0 \\ \vdots \\ -u_n(\mathbf{m}_3\ P_n) + \mathbf{m}_1\ P_n = 0 \\ -v_n(\mathbf{m}_3\ P_n) + \mathbf{m}_2\ P_n = 0 \end{cases} \longrightarrow \boxed{\underbrace{\mathcal{P}}_{\text{known}}\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

Homogenous linear system

$$\mathcal{P} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{P}_1^T & \boldsymbol{0}^T & -u_1 \underbrace{\boldsymbol{P}_1^T}_{1\text{x}4} \\ \boldsymbol{0}^T & \boldsymbol{P}_1^T & -v_1 \boldsymbol{P}_1^T \\ \dots & \dots & \dots \\ \boldsymbol{P}_n^T & \boldsymbol{0}^T & -u_n \boldsymbol{P}_n^T \\ \boldsymbol{0}^T & \boldsymbol{P}_n^T & -v_n \boldsymbol{P}_n^T \end{pmatrix}_{2\text{n x }12} \qquad \boldsymbol{m} \stackrel{\text{def}}{=} \begin{pmatrix} \underbrace{\mathbf{m}_1^{\mathrm{T}}}_{4\text{x}1} \\ \mathbf{m}_2^{\mathrm{T}} \\ \mathbf{m}_3^{\mathrm{T}} \end{pmatrix}_{12\text{x}1}$$

Homogeneous M x N Linear Systems

M=number of equations
N=number of unknown



Rectangular system (M>N)

- 0 is always a solution
- To find non-zero solution

  Minimize $|Ax|^2$

  under the constraint $|x|^2 = 1$

Calibration Problem

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

How do we solve this homogenous linear system?

Singular Value Decomposition (SVD)

Calibration Problem

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

Compute SVD decomposition of P

$$\downarrow$$

$$\boxed{U_{2n\times 12}\ D_{12\times 12} V^{T}_{12\times 12}}$$

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

$$\downarrow$$

$$M$$

$$M\ P_i \rightarrow p_i$$

Why? See page 593 of Hartley & Zisserman

Extracting camera parameters

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

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

$$\mathrm{K} = \begin{bmatrix} \boldsymbol{\alpha} & -\boldsymbol{\alpha}\cot\boldsymbol{\theta} & \mathrm{u}_o \\ 0 & \frac{\boldsymbol{\beta}}{\sin\boldsymbol{\theta}} & \mathrm{v}_o \\ 0 & 0 & 1 \end{bmatrix}$$

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

Estimated values

Intrinsic

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

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

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

Theorem (Faugeras, 1993)

$$M = K\begin{bmatrix} R & T \end{bmatrix} = \begin{bmatrix} KR & KT \end{bmatrix} = \begin{bmatrix} A & b \end{bmatrix}$$

$$A = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix}$$

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 $\mathcal{M}$.

$$K = \begin{bmatrix} \alpha & s & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

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

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

$\boldsymbol{\alpha} = \text{f k};$

$\boldsymbol{\beta} = \text{f l}$

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

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

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

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

Estimated values

Intrinsic

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

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

$\longrightarrow$ f

Extracting camera parameters

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

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

Estimated values

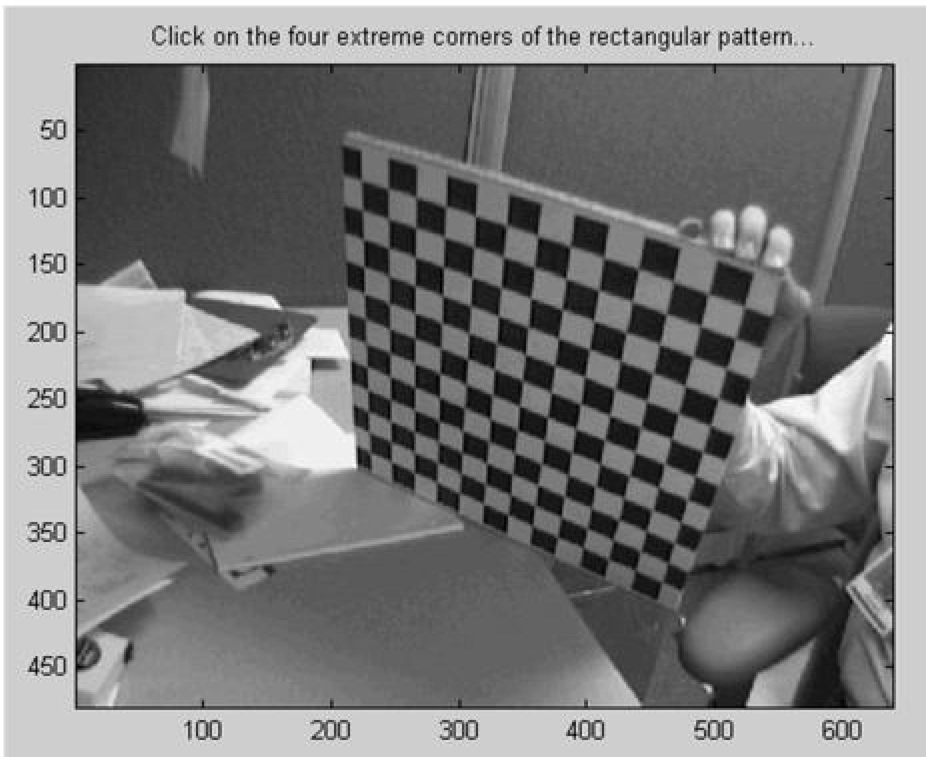
Extrinsic

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

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

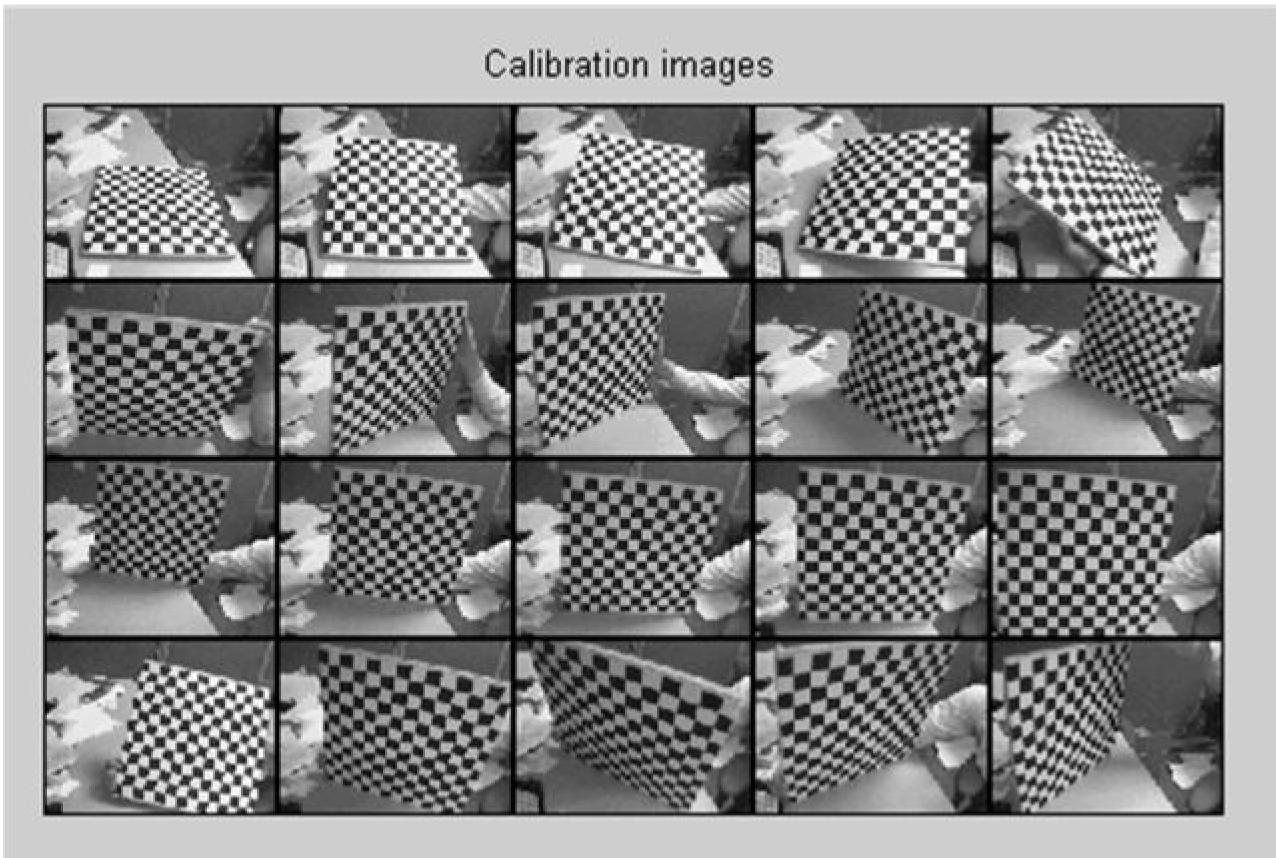
Calibration Demo

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

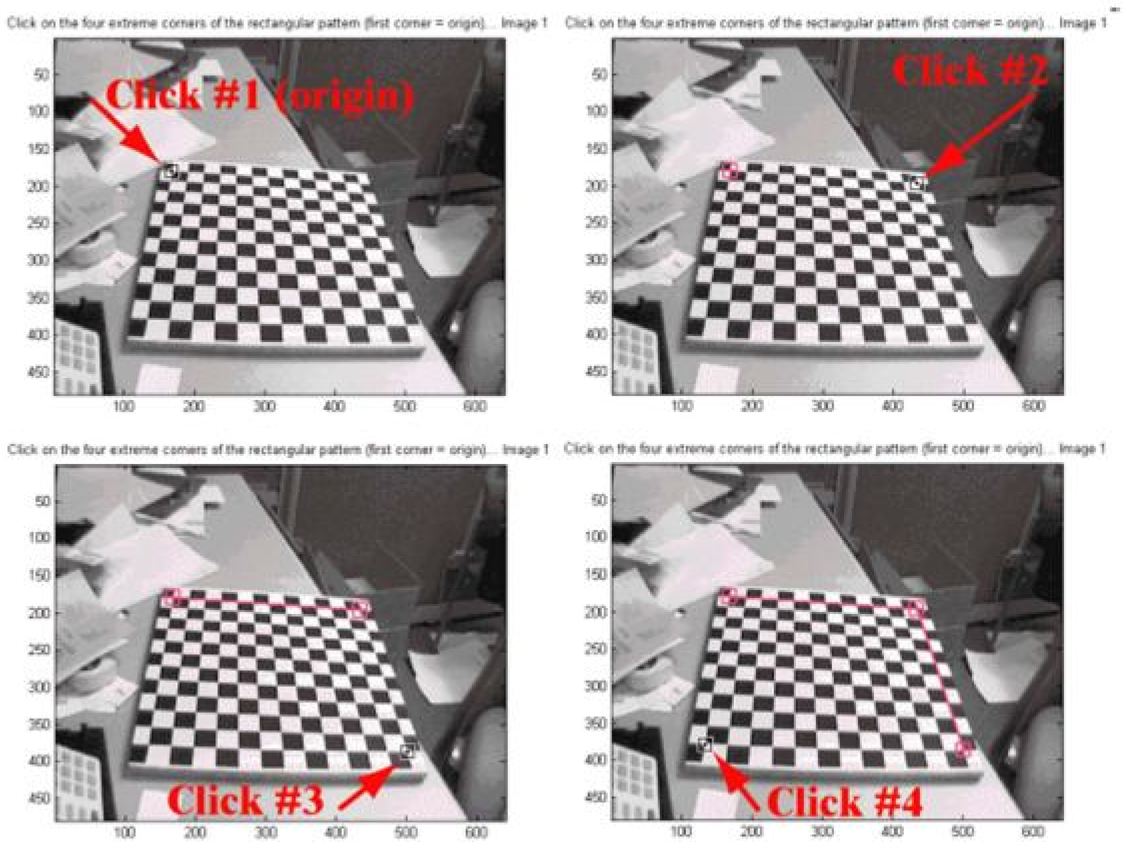


http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#examples

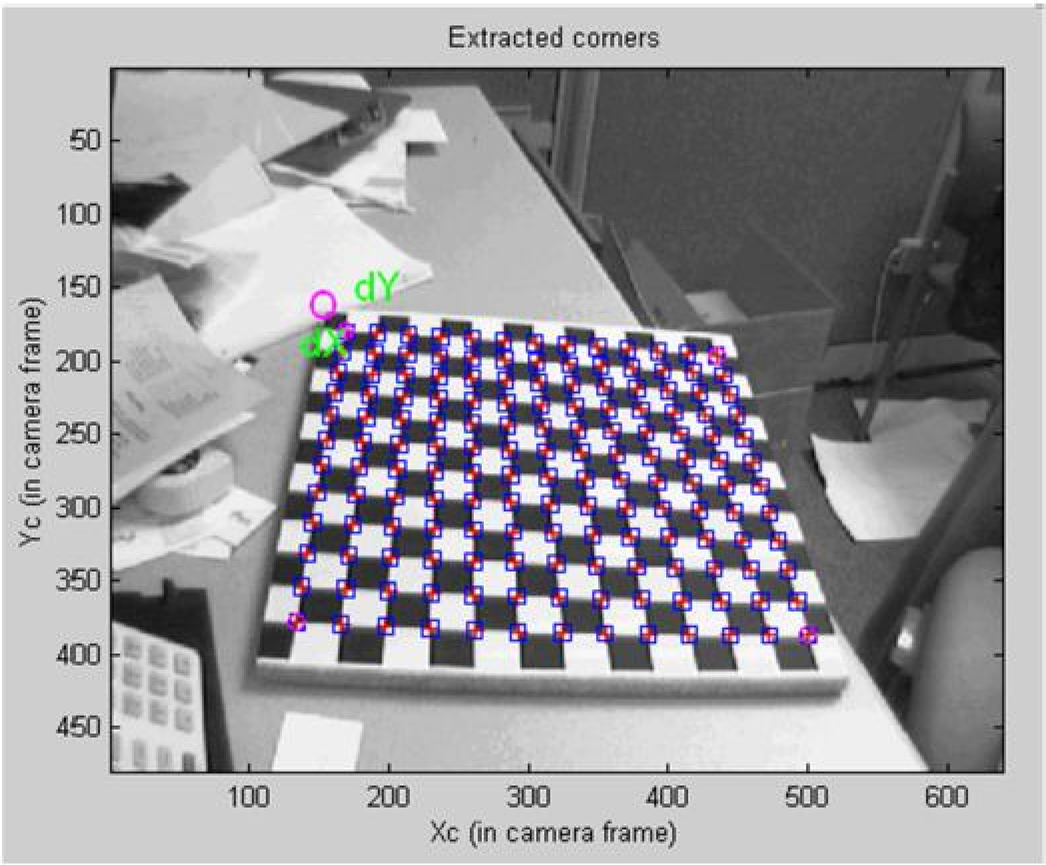
Calibration Demo



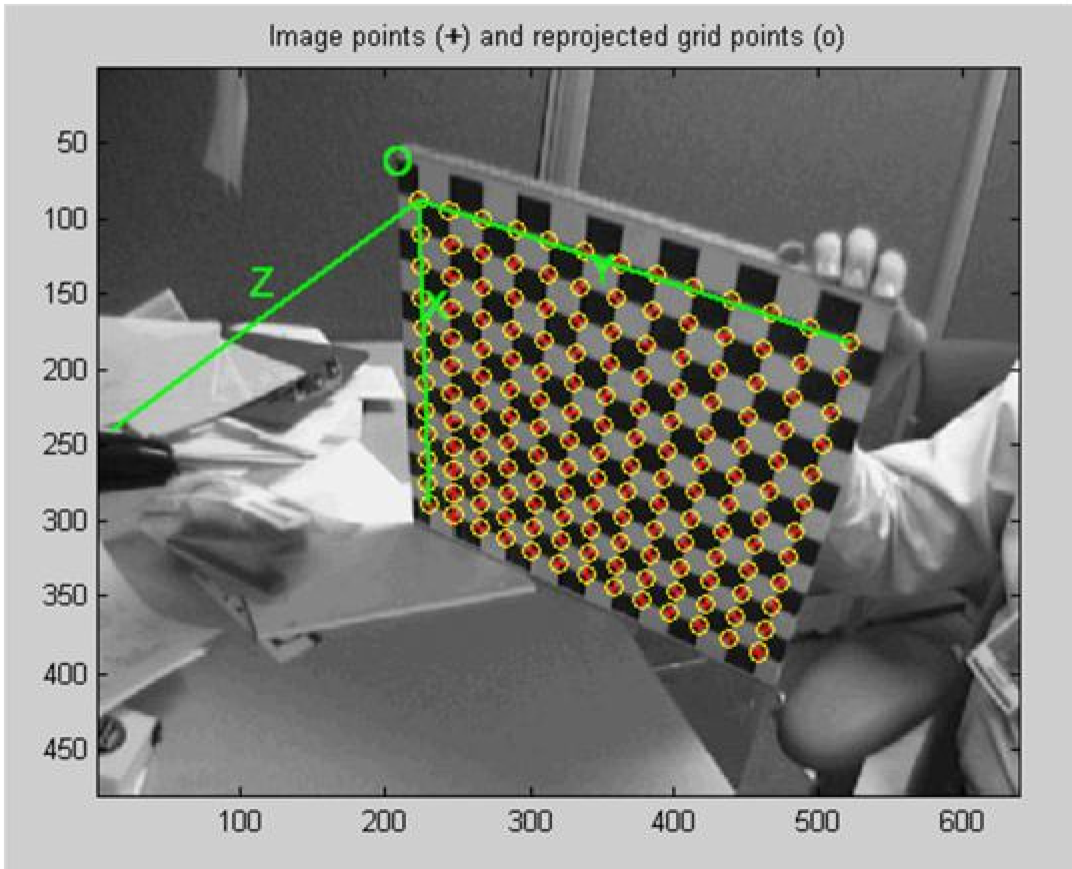
Calibration Demo



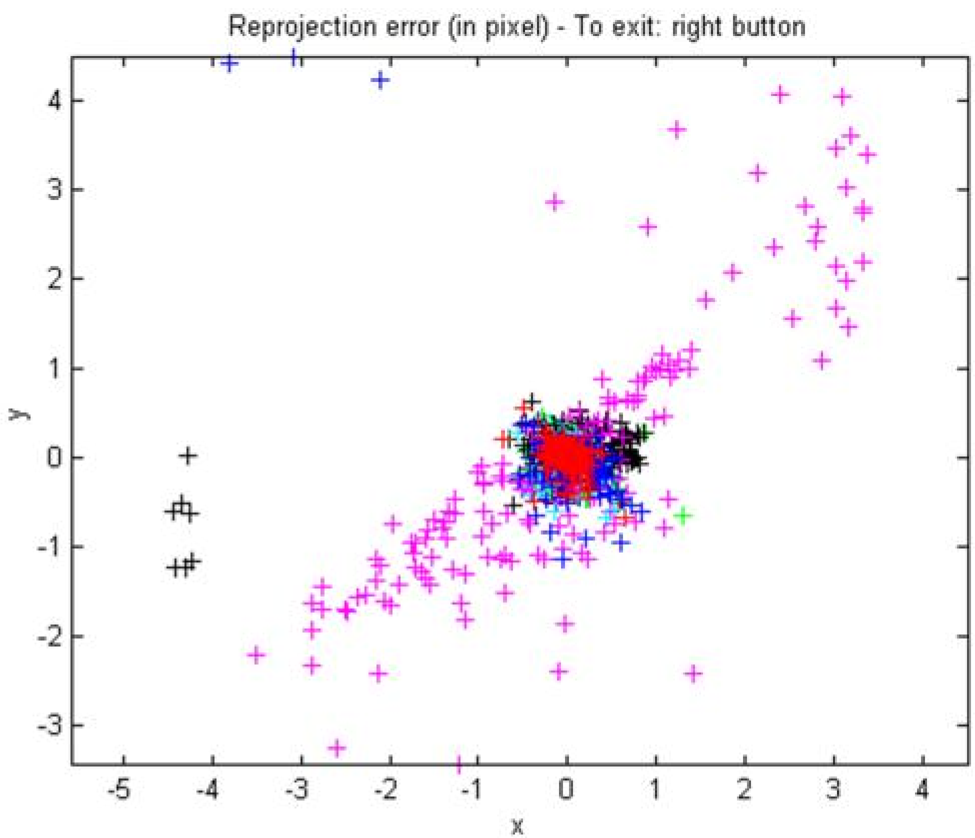
Calibration Demo



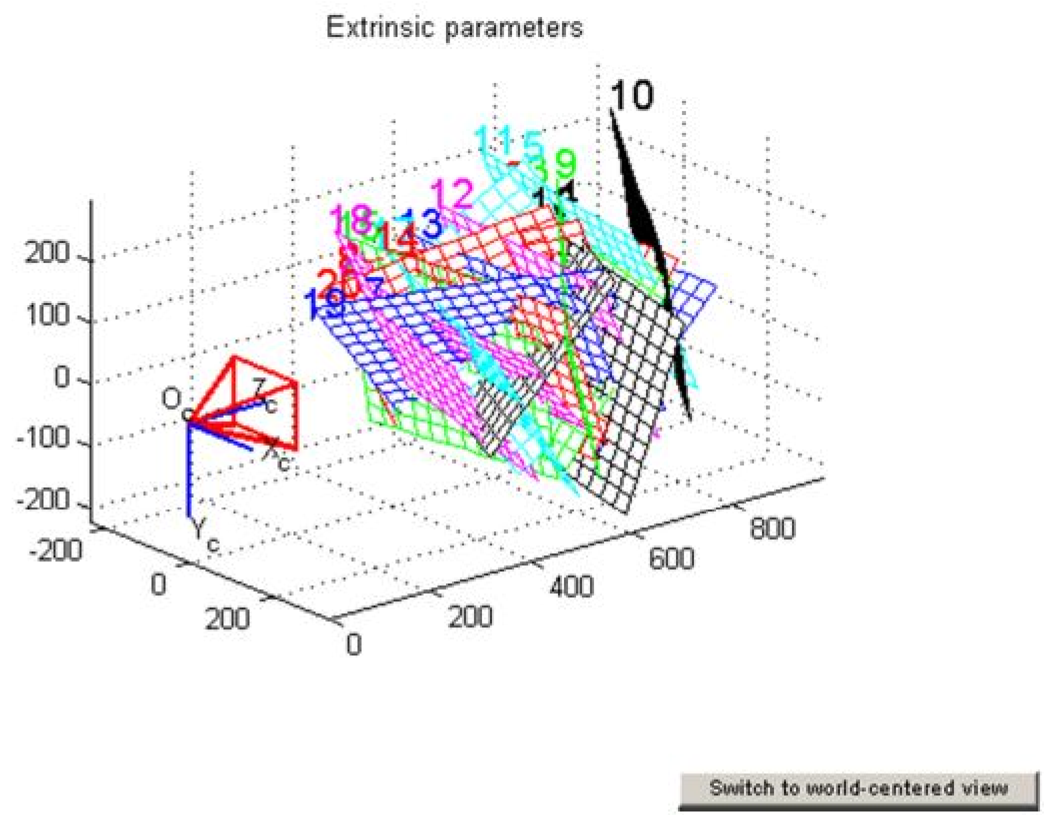
Calibration Demo



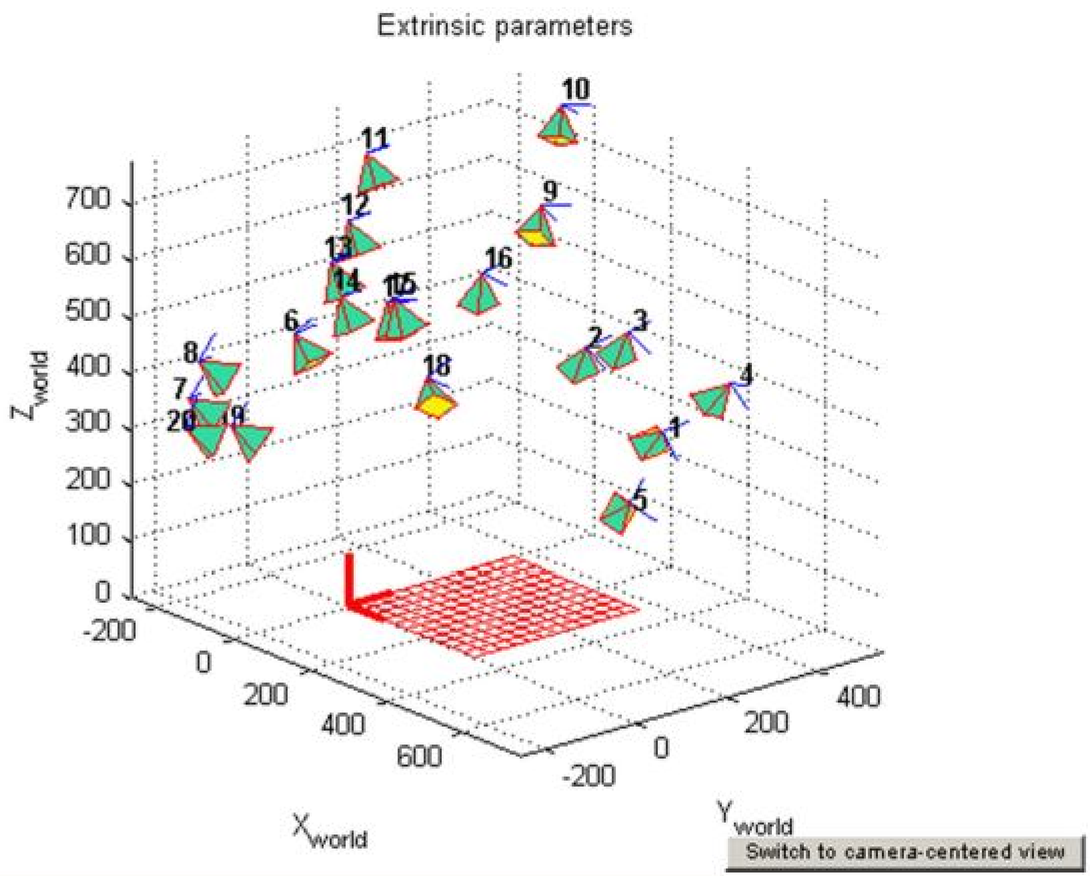
Calibration Demo



Calibration Demo



Calibration Demo



Properties of Projection

- Points project to points
- Lines project to lines

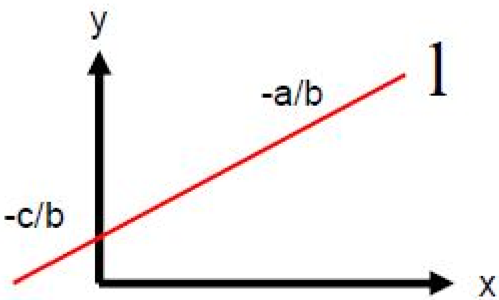
Properties of Projection

- Angles are not preserved
- Parallel lines meet

Vanishing point

Lines in a 2D plane

$$ax + by + c = 0$$



$$l = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

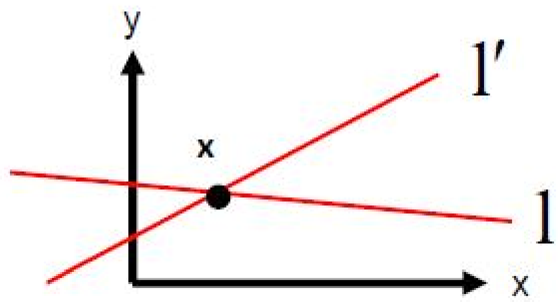
If $x = [x_1, x_2]^T \in l$

$$\begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix}^T \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0$$

Lines in a 2D plane

Intersecting lines

$$\mathrm{x} = \mathrm{l} \times \mathrm{l}'$$



Proof

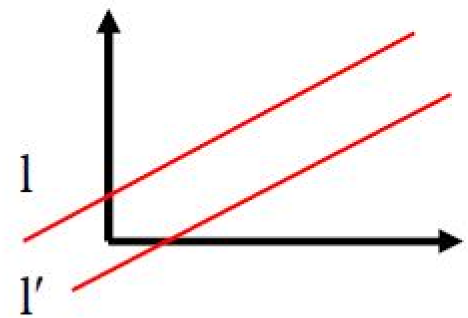
$$\mathrm{l} \times \mathrm{l}' \perp \mathrm{l} \quad \rightarrow (\mathrm{l} \times \mathrm{l}') \cdot \mathrm{l} = 0 \quad \rightarrow x \in l$$

$$\mathrm{l} \times \mathrm{l}' \perp \mathrm{l}' \quad \rightarrow \underbrace{(\mathrm{l} \times \mathrm{l}')}_{\mathrm{x}} \cdot \mathrm{l}' = 0 \quad \rightarrow x \in l'$$

$\rightarrow$ x is the intersecting point

Points at infinity (ideal points)

$$\mathrm{x} = \begin{bmatrix} \mathrm{x}_1 \\ \mathrm{x}_2 \\ \mathrm{x}_3 \end{bmatrix}, \mathrm{x}_3 \neq 0$$



$$\mathrm{l} = \begin{bmatrix} \mathrm{a} \\ \mathrm{b} \\ \mathrm{c} \end{bmatrix} \qquad \mathrm{l}' = \begin{bmatrix} \mathrm{a} \\ \mathrm{b} \\ \mathrm{c}' \end{bmatrix}$$

$$x_\infty = \begin{bmatrix} x_1 \\ x_2 \\ 0 \end{bmatrix}$$

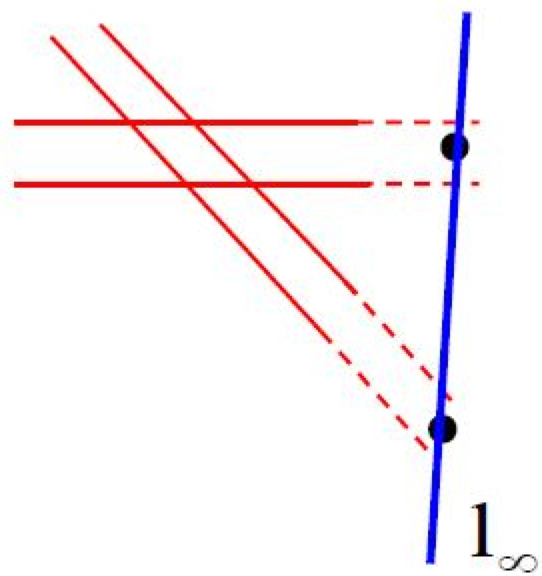
Let’s intersect two parallel lines: $\rightarrow \mathrm{l} \times \mathrm{l}' = (\mathrm{c} - \mathrm{c}') \begin{bmatrix} \mathrm{b} \\ -\mathrm{a} \\ 0 \end{bmatrix}$

Agree with the general idea of two lines intersecting at infinity

Lines at infinity $\mathbf{l}_\infty$

Set of ideal points lies on a line called the line at infinity
How does it look like?

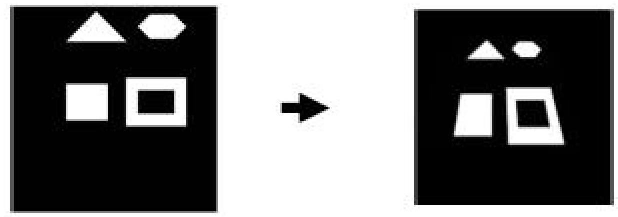
$$\mathbf{l}_\infty = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$



Indeed: $\begin{bmatrix} x_1 \\ x_2 \\ 0 \end{bmatrix}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$

Projective projections of lines at infinity (2D)

$$H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}$$



$$l' = H^{-T}\, l$$

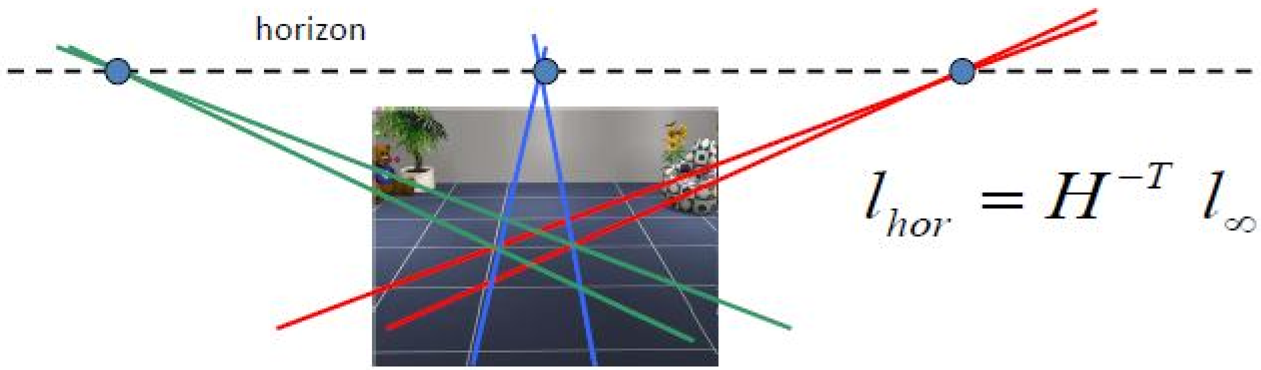
is it a line at infinity?

$$H_A^{-T}\, l_\infty = ? \quad = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} A^{-T} & 0 \\ -t^T A^{-T} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$H^{-T}\, l_\infty = ? = \begin{bmatrix} A & t \\ v & b \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ b \end{bmatrix}$$

…no!

Projective projections of lines at infinity (2D)



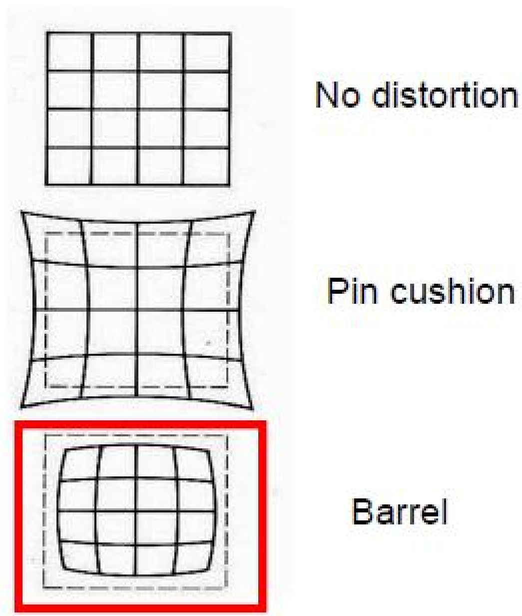
$$l_{hor} = H^{-T}\, l_{\infty}$$

Are these two lines parallel or not?

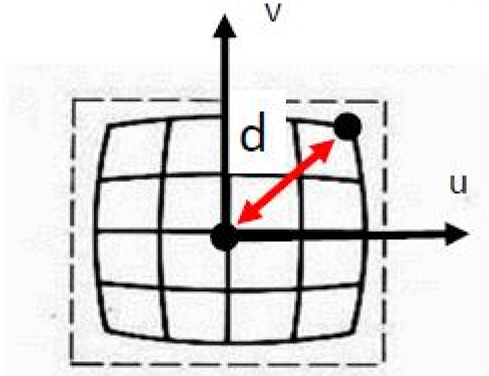
- Recognize the horizon line
- Measure if the 2 lines meet at the horizon
- if yes, these 2 lines are //

Radial Distortion

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



Radial Distortion



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

Distortion coefficient

$$d^2 = a\,u^2 + b\,v^2 + c\,u\,v \qquad \underbrace{\lambda = 1 \pm \sum_{p=1}^{3} \boxed{\kappa_p} d^{2p}}_{\text{Polynomial function}}$$

To model radial behavior

Radial Distortion

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

$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} \dfrac{\mathbf{q}_1 P_i}{\mathbf{q}_3 P_i} \\ \dfrac{\mathbf{q}_2 P_i}{\mathbf{q}_3 P_i} \end{bmatrix} \rightarrow \overbrace{\begin{cases} u_i \mathbf{q}_3 P_i = \mathbf{q}_1 P_i \\ v_i \mathbf{q}_3 P_i = \mathbf{q}_2 P \end{cases}}^{\text{Non-linear system of equations}}$$

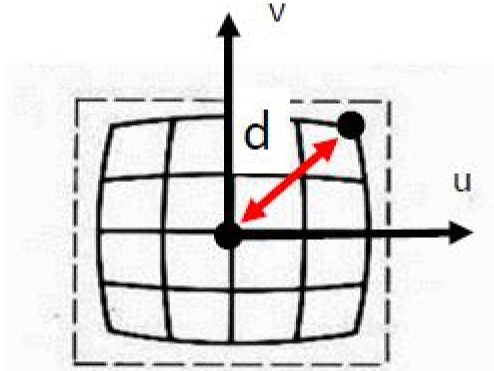
Tsai's calibration technique

1. Estimate $\mathbf{m}_1$ and $\mathbf{m}_2$ first:

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

How to do that?

Hint:



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

Tsai's calibration technique

1. Estimate $\mathbf{m}_1$ and $\mathbf{m}_2$ first:

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

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

Tsai's calibration technique

2. Once that $\mathbf{m}_1$ and $\mathbf{m}_2$ are estimated, estimate $\mathbf{m}_3$:

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

$\mathbf{m}_3$ 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