DTU

Daniel Olesen, DTU Space

30540 – Photogrammetry (2)

Warm up exercise (~15 min)

- Part 1: Euclidean transformation (Rotation and Translation)
  - Use MATLAB to generate the simple square

```
X=zeros(2,44);
X(:,1:11)=[-1:0.2:1; repmat(1,1,11)]; X(:,12:22)=[repmat(1,1,11); -(-1:0.2:1)];
X(:,23:33)=[-(-1:0.2:1); repmat(-1,1,11)]; X(:,34:44) =[repmat(-1,1,11); -1:0.2:1;];
figure, plot(X(1,:),X(2,:),'*'); axis([-1.2 1.2 -1.2 1.2]);
```

  - Now apply a rotation (+30 degrees) and a translation of t = [3;7] on your generated square and visualize the result
  - Do the rotation and translation using both inhomogenous and homogenous coordinates

- Part 2: Homogeneous Coordinates and Perspective Projection (Pairs)
  - Discuss what is the advantages of using homogenous coordinates compared to inhomogeneous coordinates
  - What properties of the camera and scene influences the perspective effects?

Todays Lecture

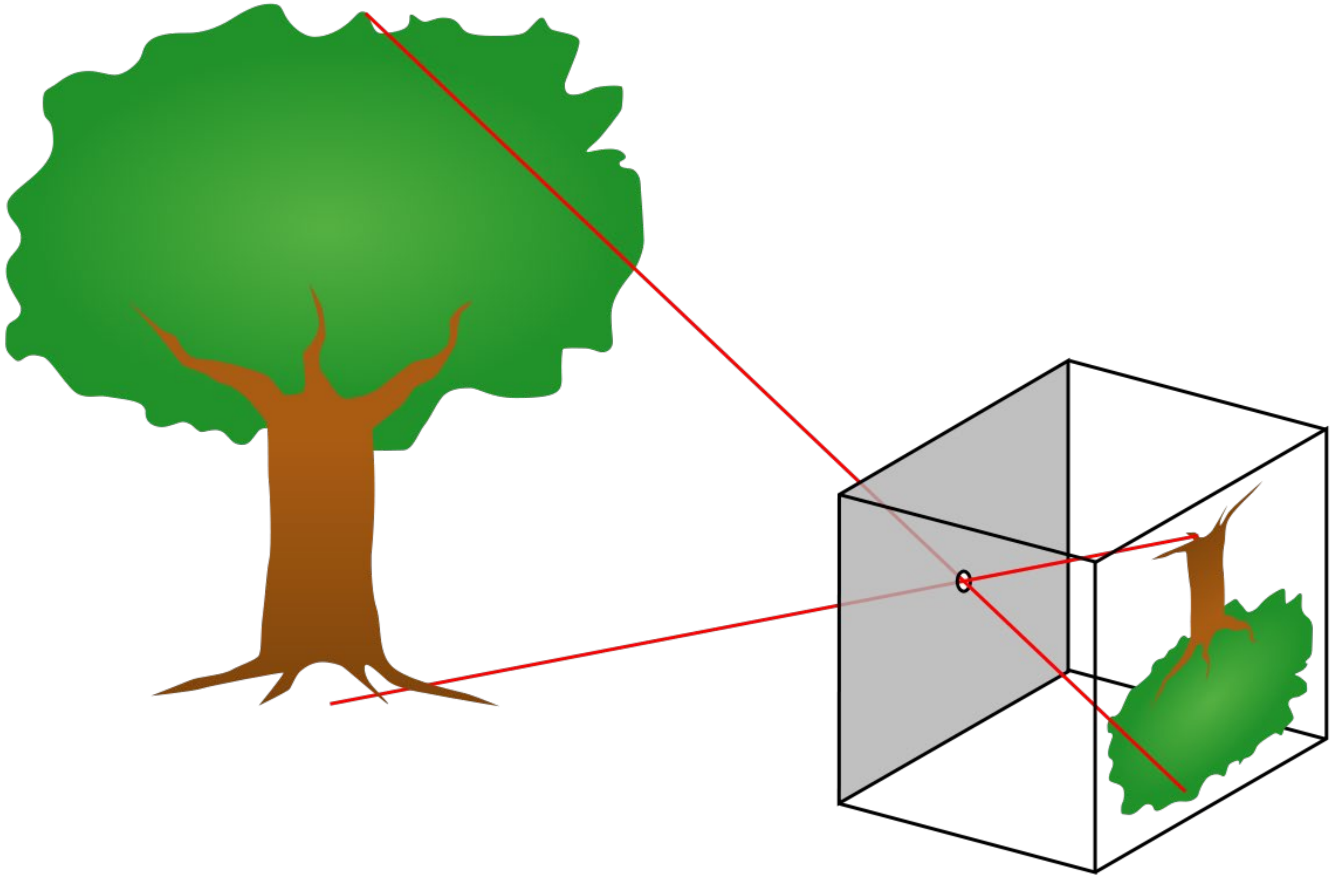
- Pinhole Camera Model
  - Extrinsics parameters (Rigid 3D to 3D transformation)
    - World and Camera coordinate systems
    - Transformation betwen systems
  - Intrinsic parameters (Projective 3D to 2D)
    - Focal length, principal point, scaling and skew

- Spatial Resection (Perspective-N-Point)
  - Finding the side lengths of a tetrahedron
  - Estimation of camera centre location

- Radial and tangential lens distortion

- Camera-calibration

The Pinhole Camera



https://simple.wikipedia.org/wiki/Pinhole_camera

Pinhole Camera Model

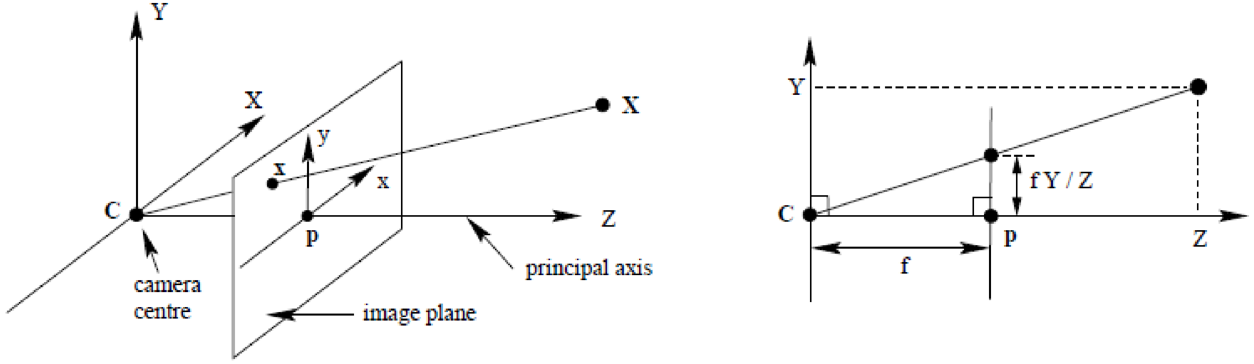


Fig. 6.1. **Pinhole camera geometry.** C *is the camera centre and* p *the principal point. The camera centre is here placed at the coordinate origin. Note the image plane is placed in front of the camera centre.*

$$\boxed{(X, Y, Z)^T \mapsto (fX/Z, fY/Z)^T}$$ → Cartesian coordinates

$$\boxed{\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}}$$ → Homogeneous coordinates

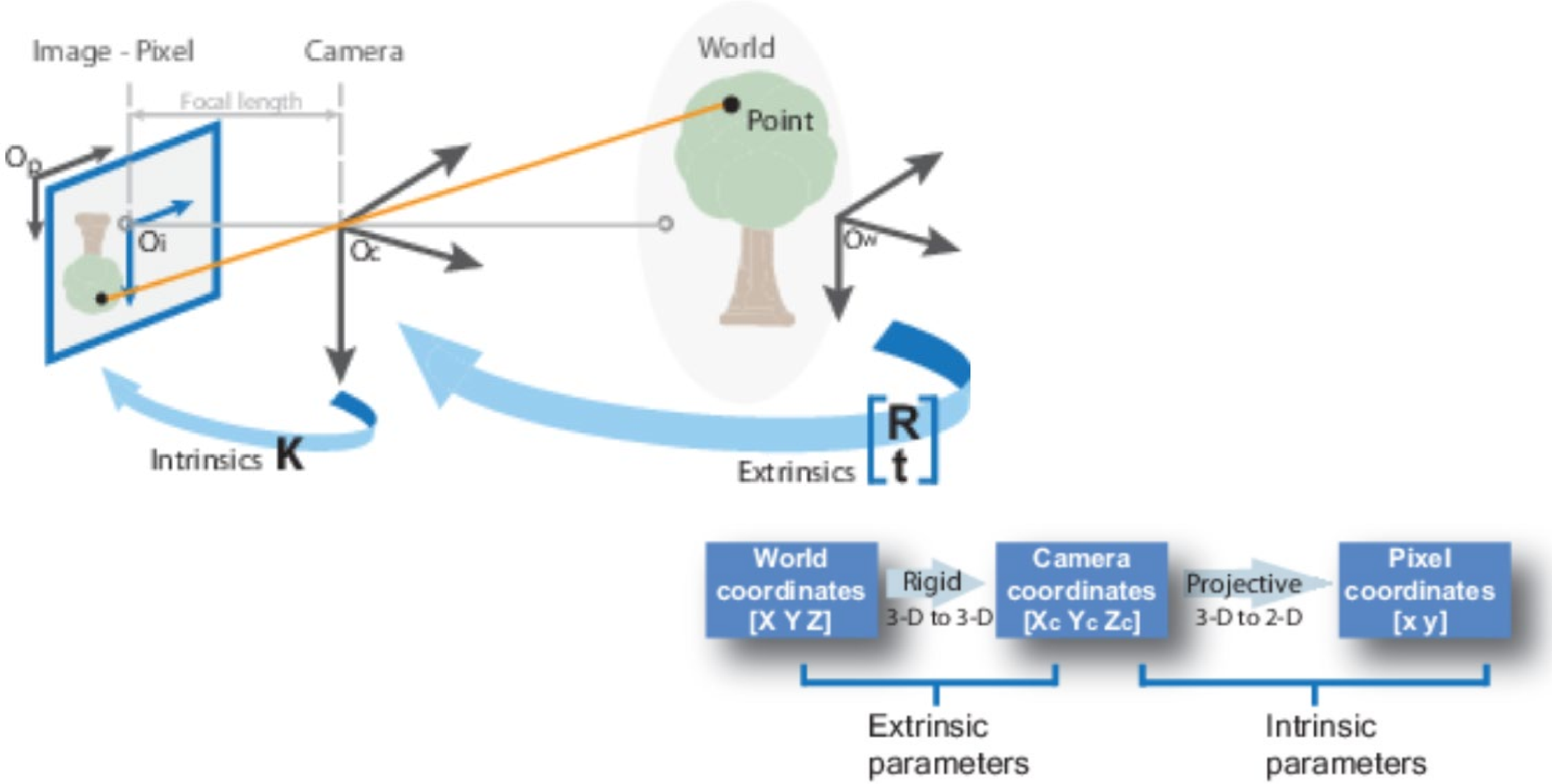
$$\mathbf{x} = \mathbf{PX}$$

Source: Hartley & Zisserman: Multiple View Geometry

Pinhole Camera Model

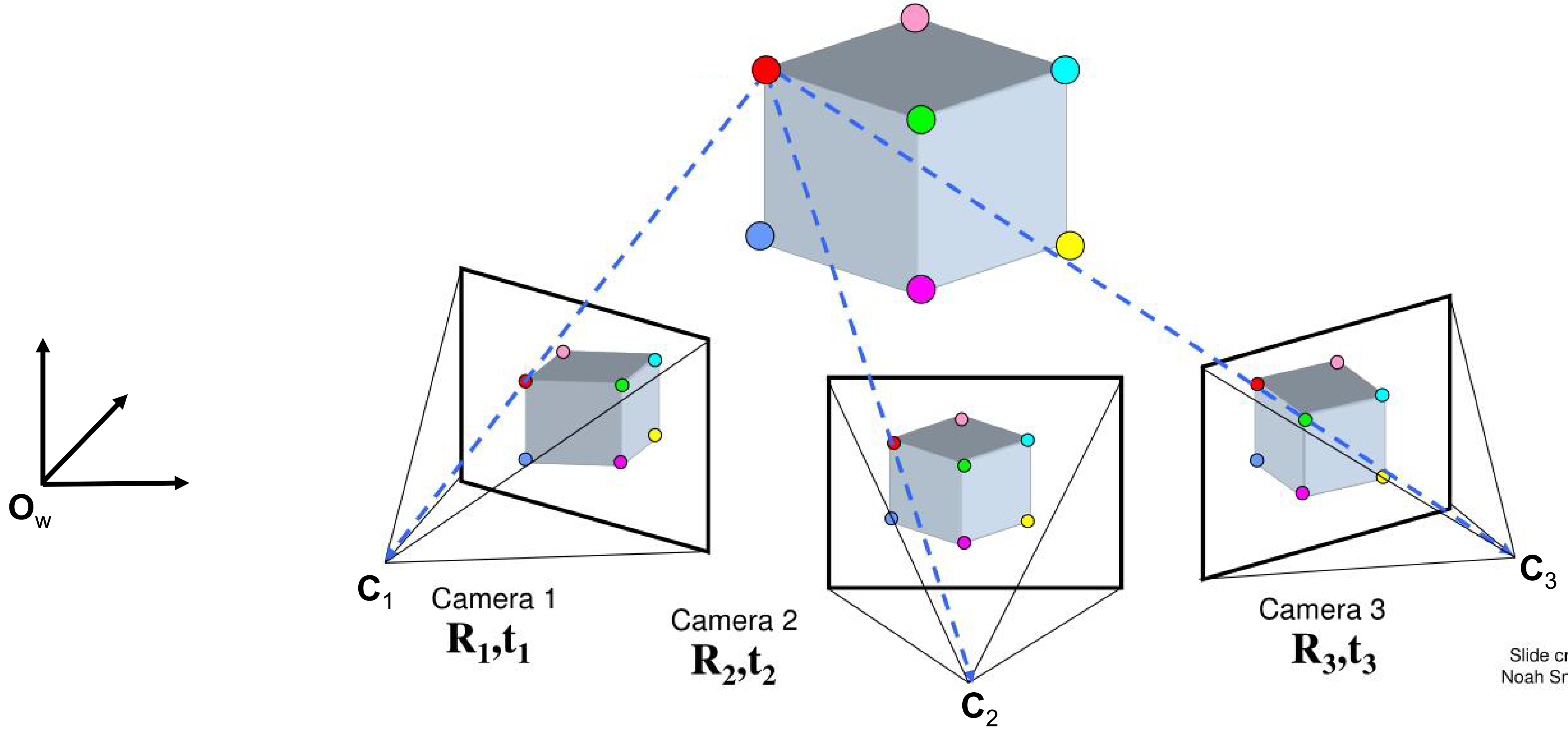
- In the camera model, we assume that the camera resides in the **origin** of a coordinate system and that the z-axis are aligned with the optical axis (depth) of the camera
- This convention is practical when we observe object points from only **one** camera
- It is however not very practical, if we are dealing with multiple views of the same scene, as we then need to express the relative position and orientation between the cameras. In this case we express scene-points and cameras in a world-coordinate system.
- In the following we will distinguish between intrinsic camera parameters, which describes a 3D to 2D transformation and extrinsic parameters which describes a rigid 3D to 3D transformation from the origin of a world coordinate system and the perspective center of the camera.

Extrinsic and intrinsic parameters



Source: https://se.mathworks.com/help/vision/ug/camera-calibration.html

World and Camera coordinate systems (Extrinsic parameters)



Slide credit: Noah Snavely

Rotation-matrix and Euler angles

$$\mathbf{R} = \mathbf{R_z}(\boldsymbol{\alpha})\mathbf{R_y}(\boldsymbol{\beta})\mathbf{R_x}(\boldsymbol{\gamma})$$

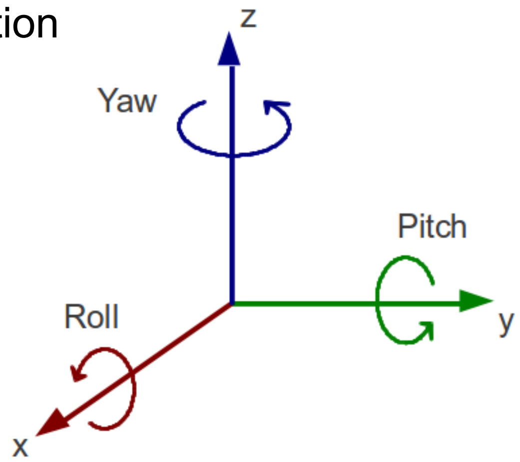
NOT commutative!, i.e. $\mathbf{R_z}(\boldsymbol{\alpha})\mathbf{R_y}(\boldsymbol{\beta})\mathbf{R_x}(\boldsymbol{\gamma}) \neq \mathbf{R_x}(\boldsymbol{\gamma})\mathbf{R_y}(\boldsymbol{\beta})\,\mathbf{R_z}(\boldsymbol{\alpha})$

$$\mathbf{R_x}(\boldsymbol{\gamma}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & \sin(\gamma) \\ 0 & -\sin(\gamma) & \cos(\gamma) \end{bmatrix}$$

$$\mathbf{R_y}(\boldsymbol{\beta}) = \begin{bmatrix} \cos(\beta) & 0 & -\sin(\beta) \\ 0 & 1 & 0 \\ \sin(\beta) & 0 & \cos(\beta) \end{bmatrix}$$

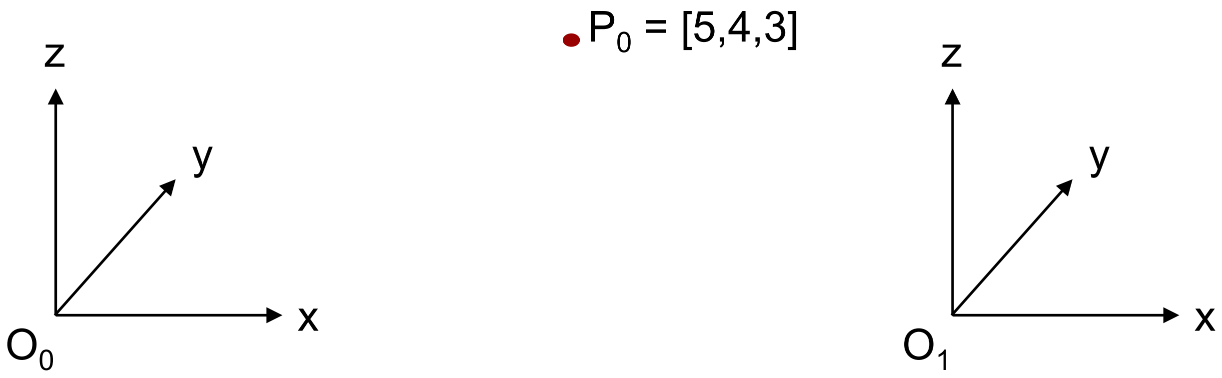
$$\mathbf{R_z}(\boldsymbol{\alpha}) = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Orthogonal matrix
- The inverse of a rotation matrix can be found by its transpose, i.e., $\mathbf{R}^{-1} = \mathbf{R}^{\mathbf{T}}$
- The euler-angles, α, β, γ, is called yaw, pitch, roll
- Be careful with different conventions for rotation matrices and euler-angles. There are many variants.



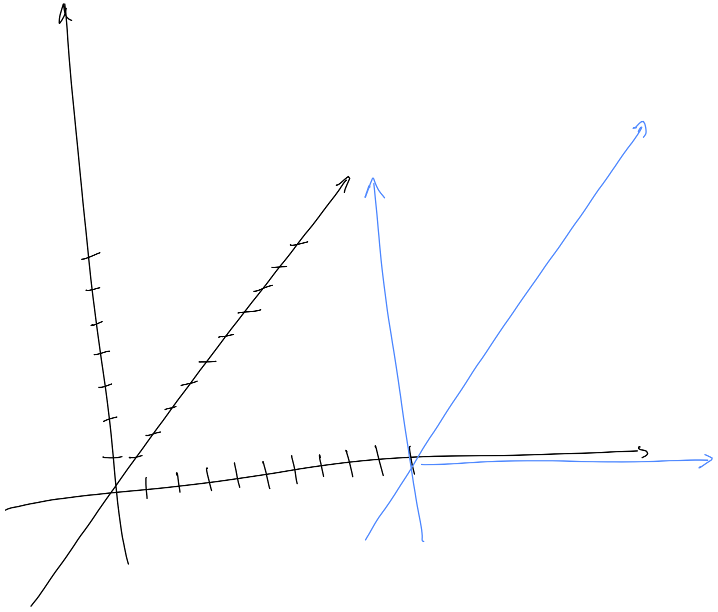
Class exercise (~10 minutes)

- In this exercise we look at two coordinate systems, where one is described from the other. The task will be to convert a point from the first to the second system



- If we assume that O_1, has its origin in [10,0,0] in O_0, what will the coordinates of P_1 be (the point expressed relative to O_1)?
- We now rotate O_1, -90 degrees around the z-axis, such that the y-axis in O_1 becomes parallel with the x-axis in O_0. What is the coordinates of P_1^* now? Which rotation matrix R do we need to use to convert the point?

- Can you find a general relationship, so that we can find P_1^* from O_1, R and P_0



$P_1 = ?(P_0, O1)$

$P_{1,0} = \begin{bmatrix} x_{1,0} \\ y_{1,0} \\ z_{1,0} \end{bmatrix}$ in O_0

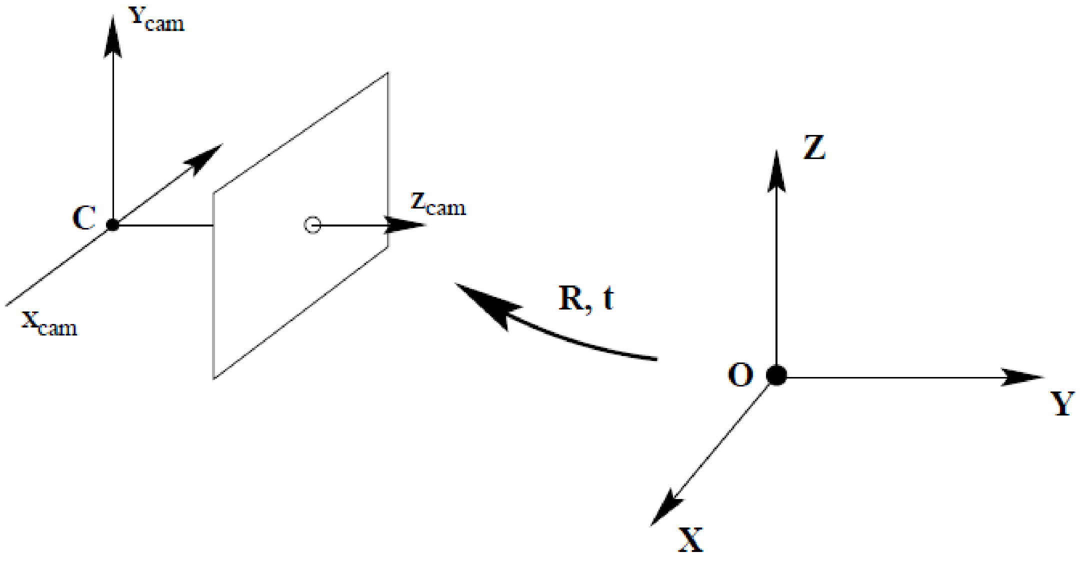
$P_{1,1} = \begin{bmatrix} x_{1,0} - (O_{1x} - O_{0x}) \\ y_{1,0} - (O_{1y} - O_{0y}) \\ z_{1,0} - (O_{1z} - O_{0z}) \end{bmatrix}$

$O_0 \to O_1$

$y_1 \Leftrightarrow x_0 \begin{bmatrix} \sin\varphi \\ \cos\varphi \\ 0 \end{bmatrix}$

$x_1 \Leftrightarrow y_0 \begin{bmatrix} -\cos\varphi \\ +\sin\varphi \\ 0 \end{bmatrix}$

Transformation between points in world- and camera coordinate system



Camera-centre in World coordinates

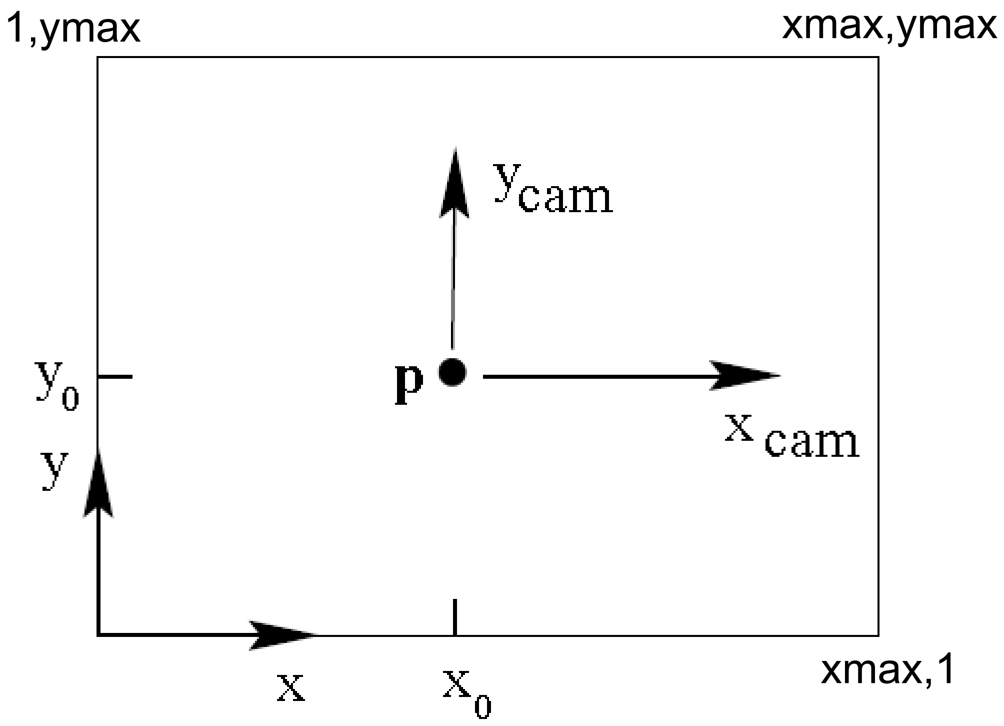
$$X_{cam} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \mathbf{X}$$

Source: Hartley & Zisserman: Multiple View Geometry

Intrinsic parameters: Principal point offset

offset we have in pixels

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}.$$

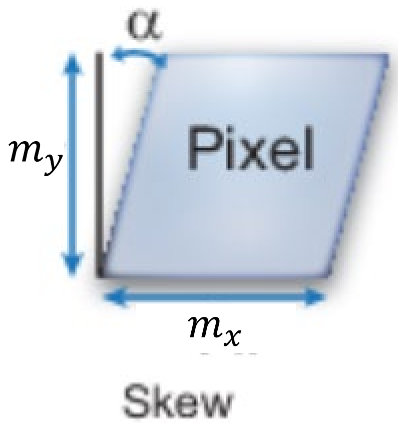


$$K = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix}$$

Intrinsic param. matrix

$$\boxed{\mathbf{x} = K[I \mid 0]\mathbf{X}_{\mathrm{cam}}}$$

Intrinsic parameters: Scaling and skew



Mathworks Inc

$$K = \begin{bmatrix} f \cdot m_x & s & p_x \\ 0 & f \cdot m_y & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

f	Focal-length in pixels
m_x, m_y	Scaling-factor in x- and y-directions
p_x, p_y	Principal-point offset in pixels
s	Skew-parameter $s = f \cdot m_y \cdot \tan(\alpha)$

Complete model

$\mathbf{x} = \mathbf{P}\mathbf{X}$

Intrinisic (or calibration parameters)

$$\begin{bmatrix} \lambda x \\ \lambda y \\ \lambda \end{bmatrix}_{imag} = K[\mathbf{R} \quad \mathbf{t}] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{world}$$

Extrinsics (rotation + translation)

There is a total of 11 parameters in this model (5 intrinsics + 6 extrinsics

Write pixel coordinates (u,v) = $\{\frac{\lambda x}{\lambda}, \frac{\lambda y}{\lambda}\}$

Break!

Todays Lecture

✓ Pinhole Camera Model
  ✓ Extrinsics parameters (Rigid 3D to 3D transformation)
    ✓ World and Camera coordinate systems
    ✓ Transformation betwen systems
  ✓ Intrinsic parameters (Projective 3D to 2D)
    ✓ Focal length, principal point, scaling and skew

- Spatial Resection (Perspective-N-Point)
  - Finding the side lengths of a tetrahedron
  - Estimation of camera centre location

- Radial and tangential lens distortion

- Camera-calibration

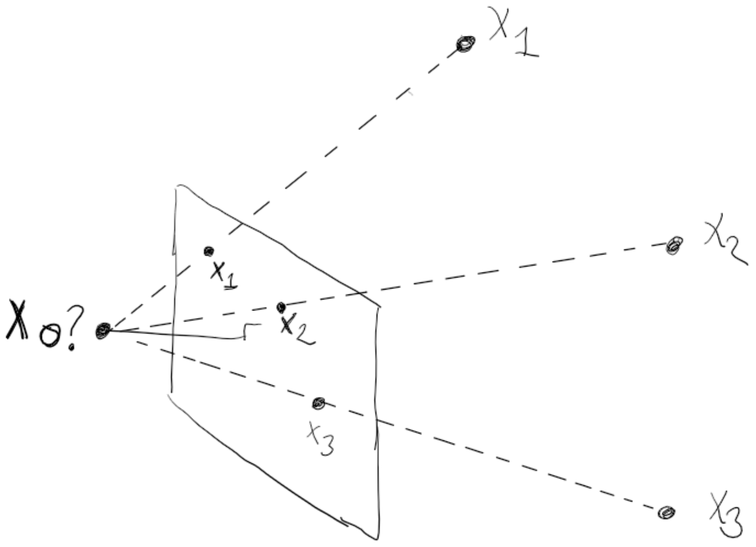
Spatial Resection (P3P, PnP)

- Spatial resection is the problem of finding the coordinates of the camera centre and orientation w.r.t. the world given known coordinates of world-points and image points

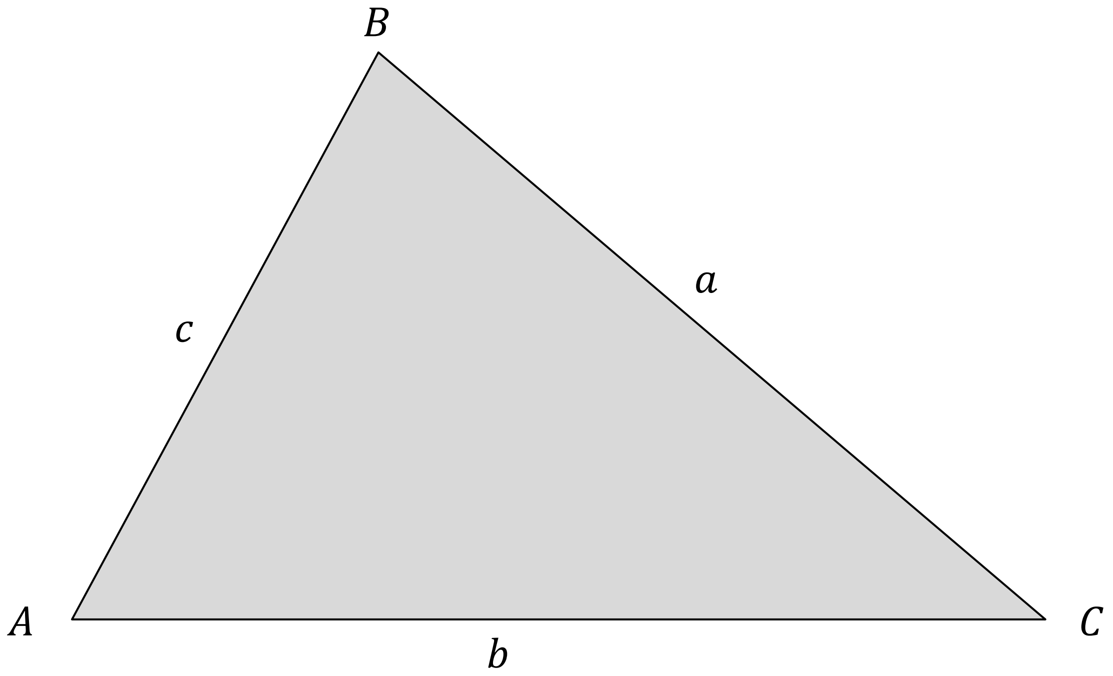
- Fundamental for Camera-calibration and also widely-used in image-based navigation

- The problem was first treated in 1841 by Grünert but has received a lot of attention within Photogrammetry and Computer Vision.

- We assume that the camera is calibrated (intrinsics and distortions are known).



Prerequisite: Cosine rule



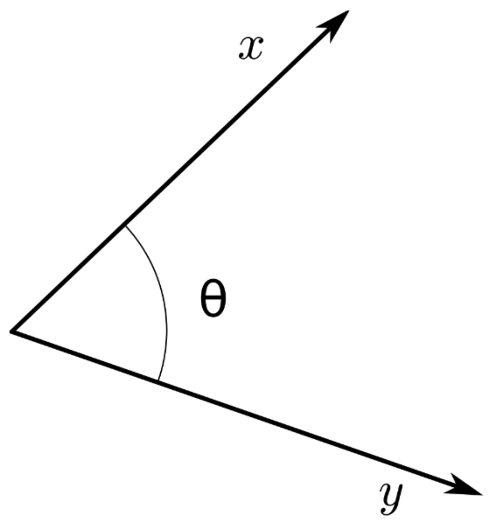
$$a^2 = b^2 + c^2 - 2bc \cdot cosA$$

$$b^2 = a^2 + c^2 - 2ac \cdot cosB$$

$$c^2 = a^2 + b^2 - 2ab \cdot cosC$$

https://www.mathalino.com/reviewer/derivation-of-formulas/derivation-of-cosine-law

Prerequisite: Angles between vectors

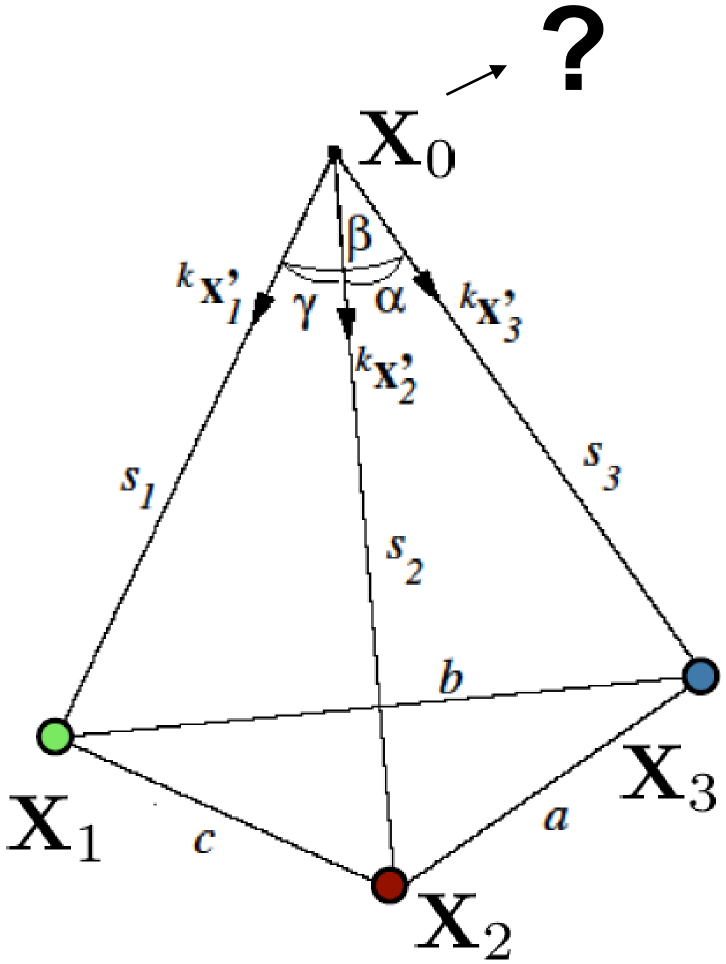


$\vec{x} \cdot \vec{y} = |x| \cdot |y| \cdot cos\theta$

In case, that $\vec{x}$ and $\vec{y}$, is unit vectors, we get:

$\vec{x} \cdot \vec{y} = cos\theta \Leftrightarrow \theta = \cos^{-1}(\vec{x} \cdot \vec{y})$

Spatial Resection



$\mathbf{X_0}$ is the unknown camera-centre

$\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}$ is **known** world points

The angles between the points can be deduced as:

$$\alpha = \cos^{-1}(\vec{x_2} \cdot \vec{x_3}), \beta = \cos^{-1}(\vec{x_1} \cdot \vec{x_3}),$$
$$\gamma = \cos^{-1}(\vec{x_1} \cdot \vec{x_2})$$

The sides, a, b, c can be found as:
$a = |\mathbf{X_3} - \mathbf{X_2}|, b = |\mathbf{X_3} - \mathbf{X_1}|, c = |\mathbf{X_2} - \mathbf{X_1}|$

How do we determine s_1, s_2, s_3?

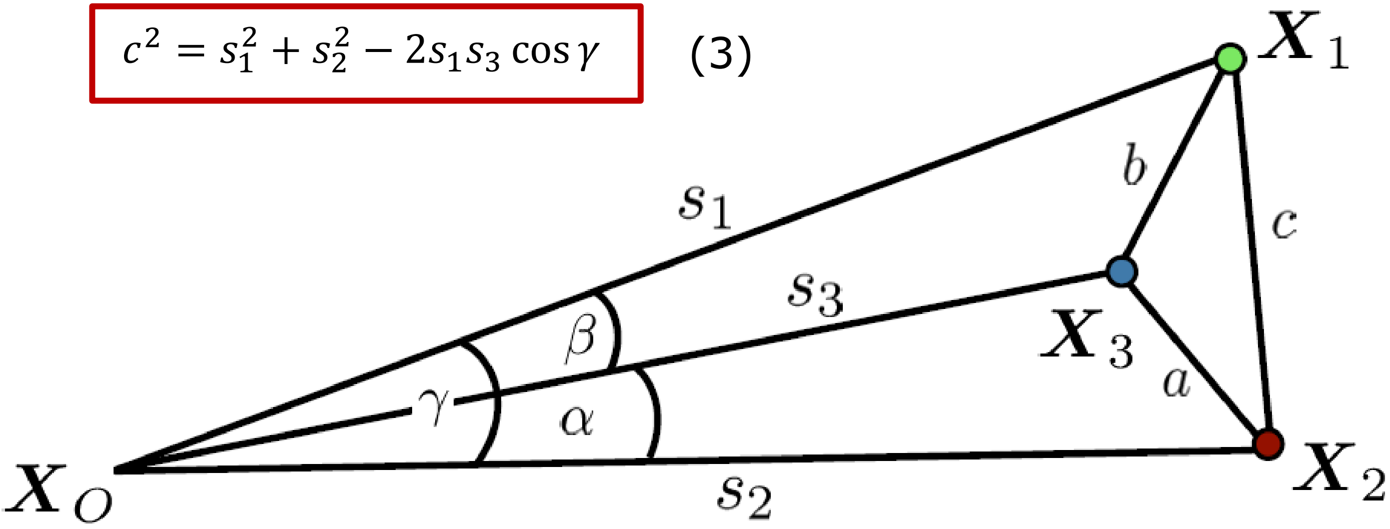
Spatial Resection

The Law of cosines!

$$\boxed{a^2 = s_2^2 + s_3^3 - 2s_2s_3\cos\alpha} \quad (1)$$

$$\boxed{b^2 = s_1^2 + s_3^2 - 2s_1s_3\cos\beta} \quad (2)$$

$$\boxed{c^2 = s_1^2 + s_2^2 - 2s_1s_3\cos\gamma} \quad (3)$$



Spatial Resection

If we define $u = \frac{s_2}{s_1}$ and $v = \frac{s_3}{s_1}$, then by substitution into (1)

$$a^2 = s_2^2 + s_3^3 - 2s_2 s_3 \cos\alpha \Longrightarrow a^2 = s_1^2(u^2 + v^2 - 2uv\cos\alpha) \Leftrightarrow$$
$$s_1^2 = \frac{a^2}{u^2 + v^2 - 2uv\cos\alpha}$$

For (2) and (3), we get the following expressions:

$$s_1^2 = \frac{b^2}{1 + v^2 - 2v\cos\beta}$$
$$s_1^2 = \frac{c^2}{1 + u^2 - 2u\cos\gamma}$$

From the above equations, it is now possible to express u in terms of v. This forms a 4th order polynomial, see Grünert 1841, Haralick 1994 (p.334),

$$A_4v^4 + A_3v^3 + A_2v^2 + A_1v^1 + A_0 = 0$$

Spatial Resection

$$
\begin{aligned}
A_3 = 4\Bigg[&\frac{a^2-c^2}{b^2}\left(1-\frac{a^2-c^2}{b^2}\right)\cos\beta \\
&-\left(1-\frac{a^2+c^2}{b^2}\right)\cos\alpha\cos\gamma \\
&+2\frac{c^2}{b^2}\cos^2\alpha\cos\beta\Bigg]
\end{aligned}
$$

$$
\begin{aligned}
A_2 = 2\Bigg[&\left(\frac{a^2-c^2}{b^2}\right)^2-1+2\left(\frac{a^2-c^2}{b^2}\right)^2\cos^2\beta \\
&+2\left(\frac{b^2-c^2}{b^2}\right)\cos^2\alpha \\
&-4\left(\frac{a^2+c^2}{b^2}\right)\cos\alpha\cos\beta\cos\gamma \\
&+2\left(\frac{b^2-a^2}{b^2}\right)\cos^2\gamma\Bigg]
\end{aligned}
$$

$$
\begin{aligned}
A_1 = 4\Bigg[&-\left(\frac{a^2-c^2}{b^2}\right)\left(1+\frac{a^2-c^2}{b^2}\right)\cos\beta \\
&+\frac{2a^2}{b^2}\cos^2\gamma\cos\beta \\
&-\left(1-\left(\frac{a^2+c^2}{b^2}\right)\right)\cos\alpha\cos\gamma\Bigg]
\end{aligned}
$$

$$
A_0 = \left(1+\frac{a^2-c^2}{b^2}\right)^2-\frac{4a^2}{b^2}\cos^2\gamma.
$$

From Haralick (1994) (p.334)

Solve polynomial with MATLAB function: roots

Spatial Resection

From the solution of v, s_1 can be found as:

$$s_1^2 = \frac{b^2}{1 + v^2 - 2v\cos\beta}$$

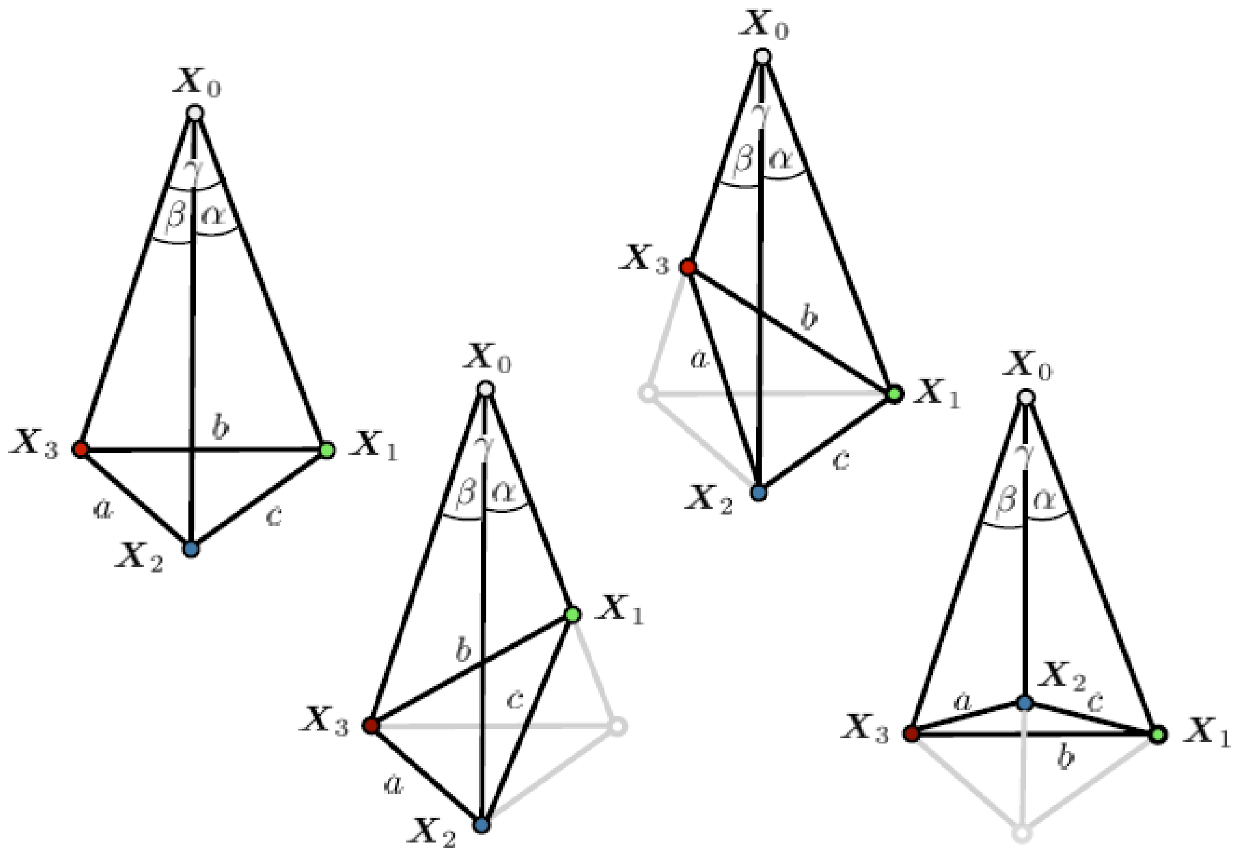
s_3, can be found as:

$$v = \frac{s_3}{s_1} \Leftrightarrow s_3 = vs_1$$

And finally, s_2 from:

$$a^2 = s_2^2 + s_3^2 - 2s_2s_3\cos\alpha$$

Multiple solutions (solution of 4th order polynomial)

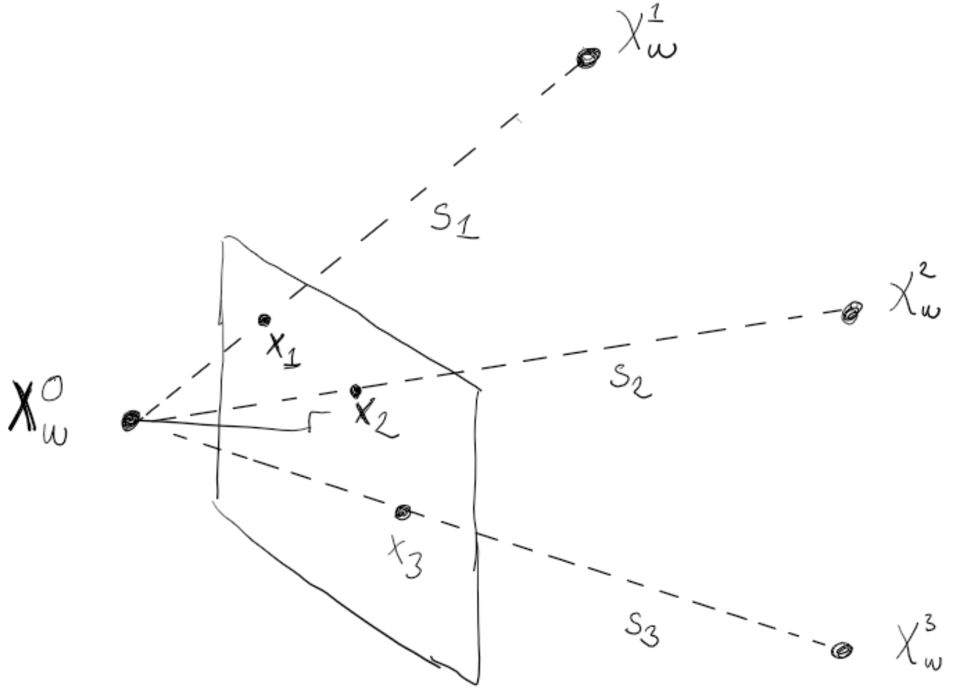


Express the points in camera coordinate system

- Before we can calculate the position of the camera center in the world coordinate system, we need to express $\mathbf{X}_\mathbf{w}^\mathbf{1}$, $\mathbf{X}_\mathbf{w}^\mathbf{2}$ and $\mathbf{X}_\mathbf{w}^\mathbf{3}$ in the camera coordinate system.

- This can be done by finding the unit-vectors from the pixel coordinates and scale with the identified lengths from previous step, i.e.

$$\mathbf{X}_\mathbf{C}^\mathbf{i} = \mathbf{s}_\mathbf{i} \cdot \frac{\mathbf{x}_\mathbf{i}}{|\mathbf{x}_\mathbf{i}|}$$



Finding the Camera centre…

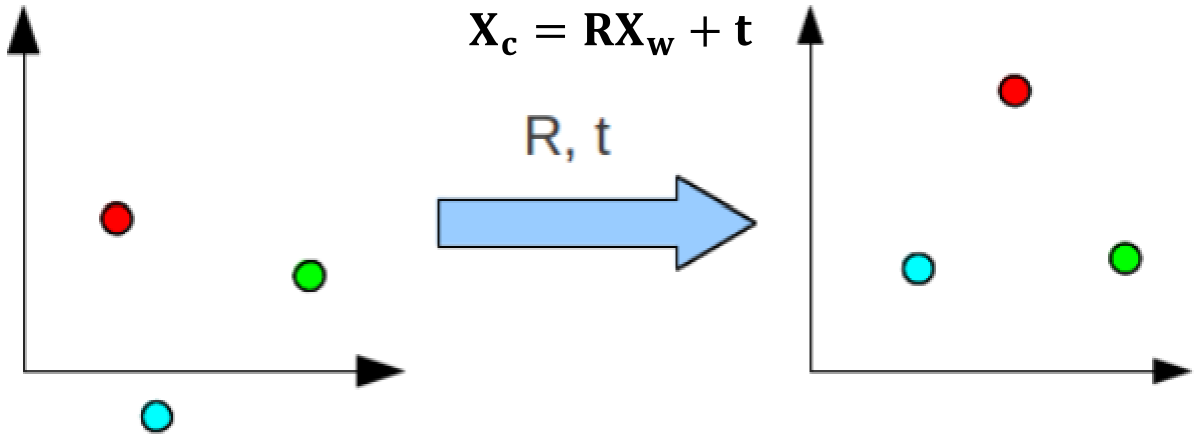
- From the previous slides, we can now express the known World-points in the camera-coordinate frame -> However the task was to identify the world-coordinates of the camera centre…

We first need to solve for $(\mathbf{R}, \mathbf{t})$,

Camera-centre in camera-cordinates

Camera-centre in world-cordinates

$$\mathbf{X_c} = \mathbf{R}\mathbf{X_w} + \mathbf{t}$$



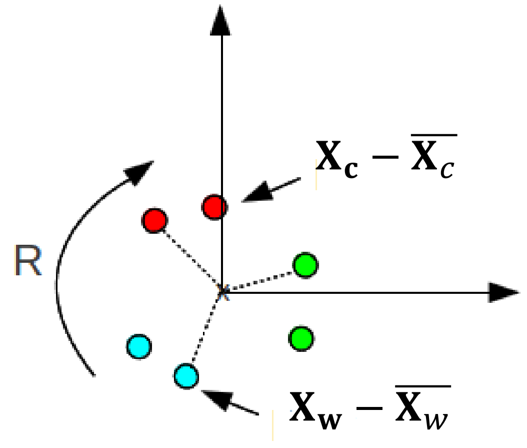
http://nghiaho.com/?page_id=671

Finding the Camera centre…

- First step, is to calculate the centroids (average point) from each representation (world and camera coords.)

$$\overline{\mathbf{X}_C} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_C^i,\ \overline{\mathbf{X}_w} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_w^i,$$

- Subtracting the centroids from the original point-sets, leaves only the Rotation unknown…



$$\mathbf{H} = \sum_{i}^{N}\left(\mathbf{X}_w^i, -\overline{\mathbf{X}_w}\right)\left(\mathbf{X}_c^i, -\overline{\mathbf{X}_c}\right)^T$$

$$[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \boldsymbol{SVD}(\mathbf{H})$$

$$\mathrm{d} = \mathrm{sgn}(\det(\mathbf{VU^T}))$$

$$\mathbf{R} = \mathbf{V}\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & \mathrm{d}\end{bmatrix}\mathbf{U^T}$$

http://nghiaho.com/?page_id=671

Finding the Camera centre…

- Finally, we can find the translation as:

$$\mathbf{t} = -\mathbf{R}\overline{\mathbf{X}_w} + \overline{\mathbf{X}_c}$$

The camera centre can hence be found as…

$$\mathbf{X_c} = \mathbf{R}\mathbf{X_w} + \mathbf{t} \Leftrightarrow \mathbf{X_w} = \mathbf{R}^{-1}(\underbrace{\mathbf{X_c}}_{[0 \quad 0 \quad 0]^T} - \mathbf{t}) = -\mathbf{R}^{-1}\mathbf{t} = -\mathbf{R}^{\mathrm{T}}\mathbf{t}$$

Note, that algorithm for finding the camera centre is known as the Kabsch algorithm. See also the paper from Arun et. al (PAMI-3DLS-1987.pdf)

Todays Lecture

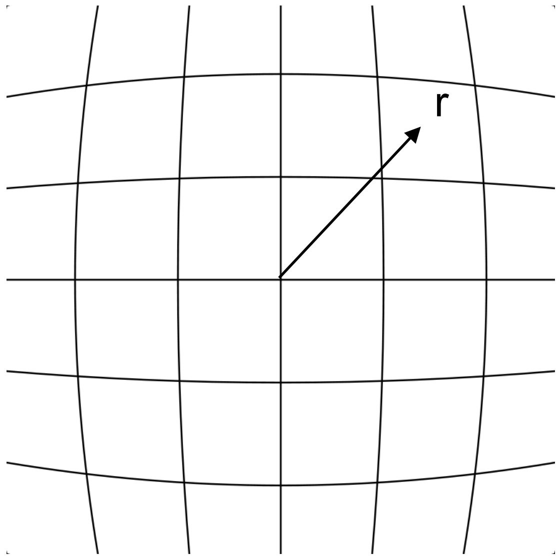
- ✓ Pinhole Camera Model
  - ✓ Extrinsics parameters (Rigid 3D to 3D transformation)
    - ✓ World and Camera coordinate systems
    - ✓ Transformation betwen systems
  - ✓ Intrinsic parameters (Projective 3D to 2D)
    - ✓ Focal length, principal point, scaling and skew

- ✓ Spatial Resection (Perspective-N-Point)
  - ✓ Finding the side lengths of a tetrahedron
  - ✓ Estimation of camera centre location

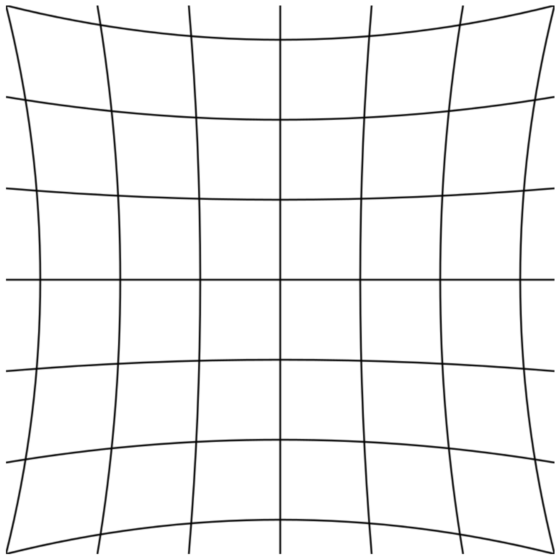
- Radial and tangential lens distortion

- Camera-calibration

Radial Distortion



Barrel Distortion



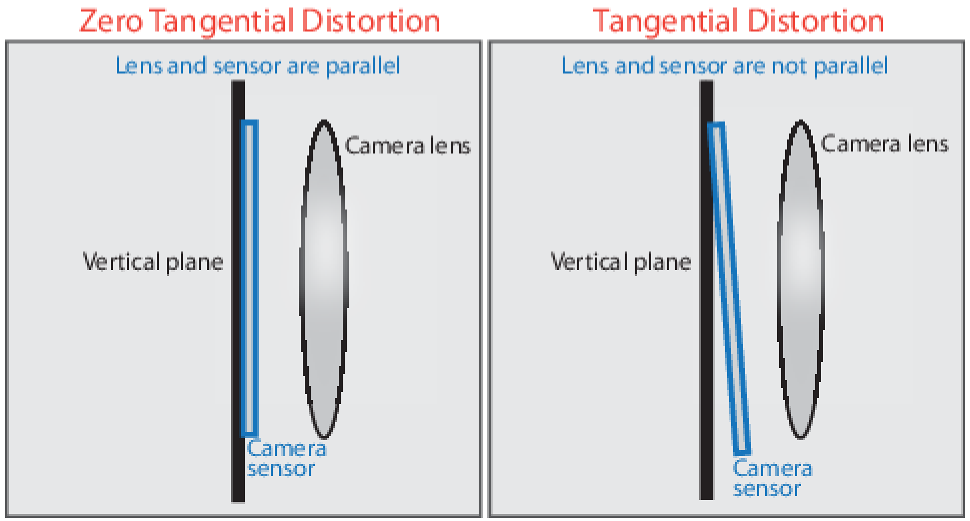
Pincushion Distortion

Typical Distortion Model:

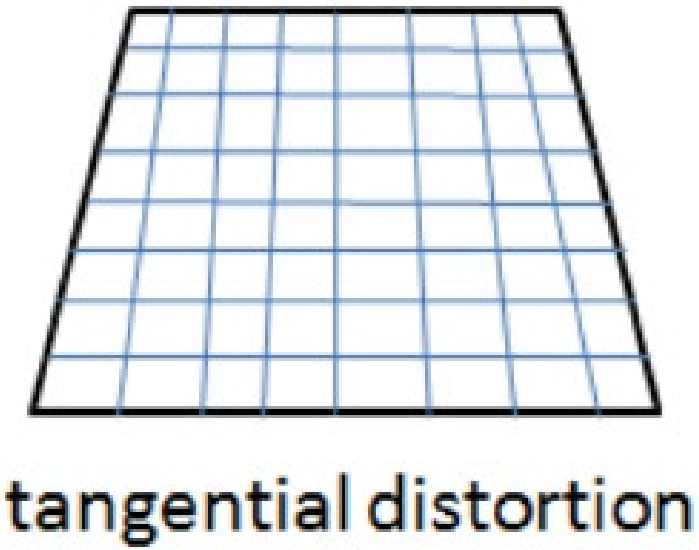
$$r^2 = x^2 + y^2$$
$$x_d = x(1 + a_1 r^2 + a_2 r^4 + a_3 r^6)$$
$$y_d = y(1 + a_1 r^2 + a_2 r^4 + a_3 r^6)$$

Radial Distortion

Tangential Distortion



Mathworks Inc



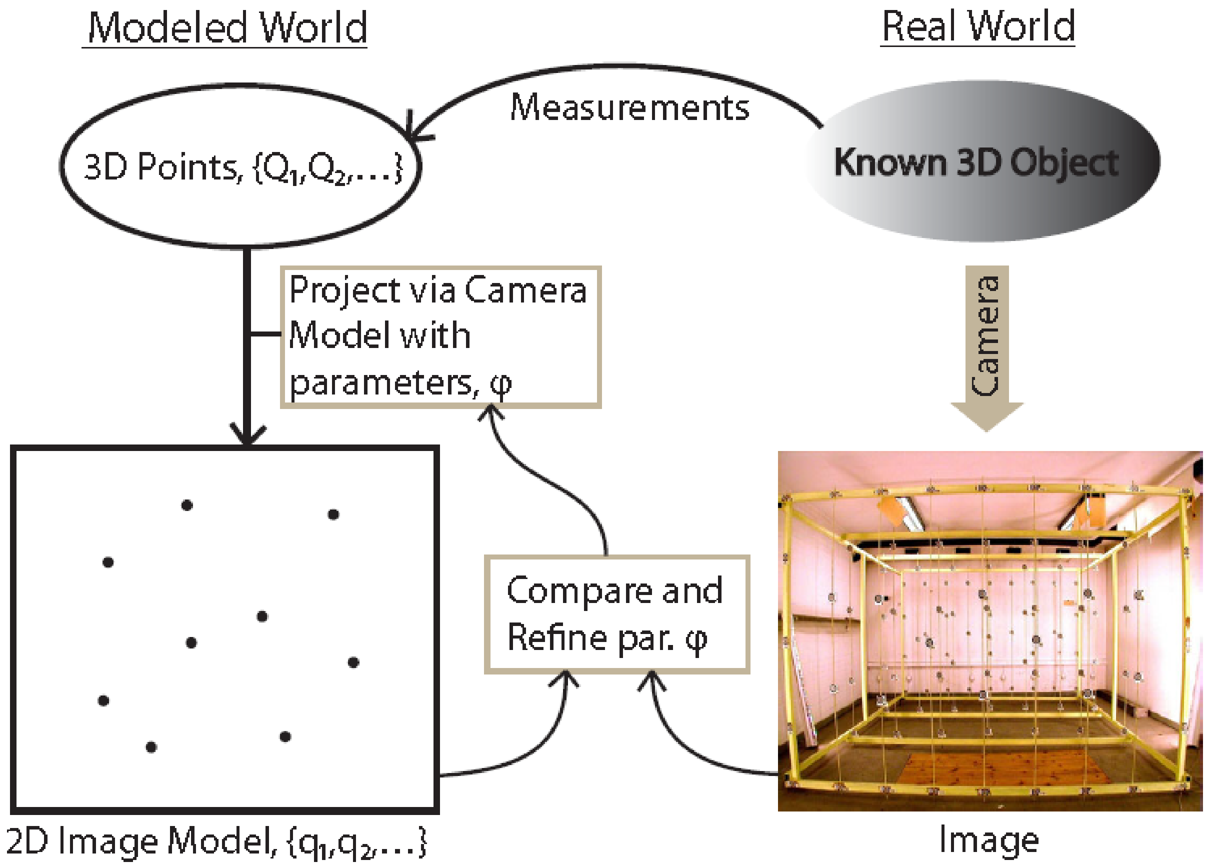
Typical Distortion Model:

$$r^2 = x^2 + y^2$$

$$x_d = x + \left(2p_1xy + p_2(r^2 + 2x^2)\right)$$

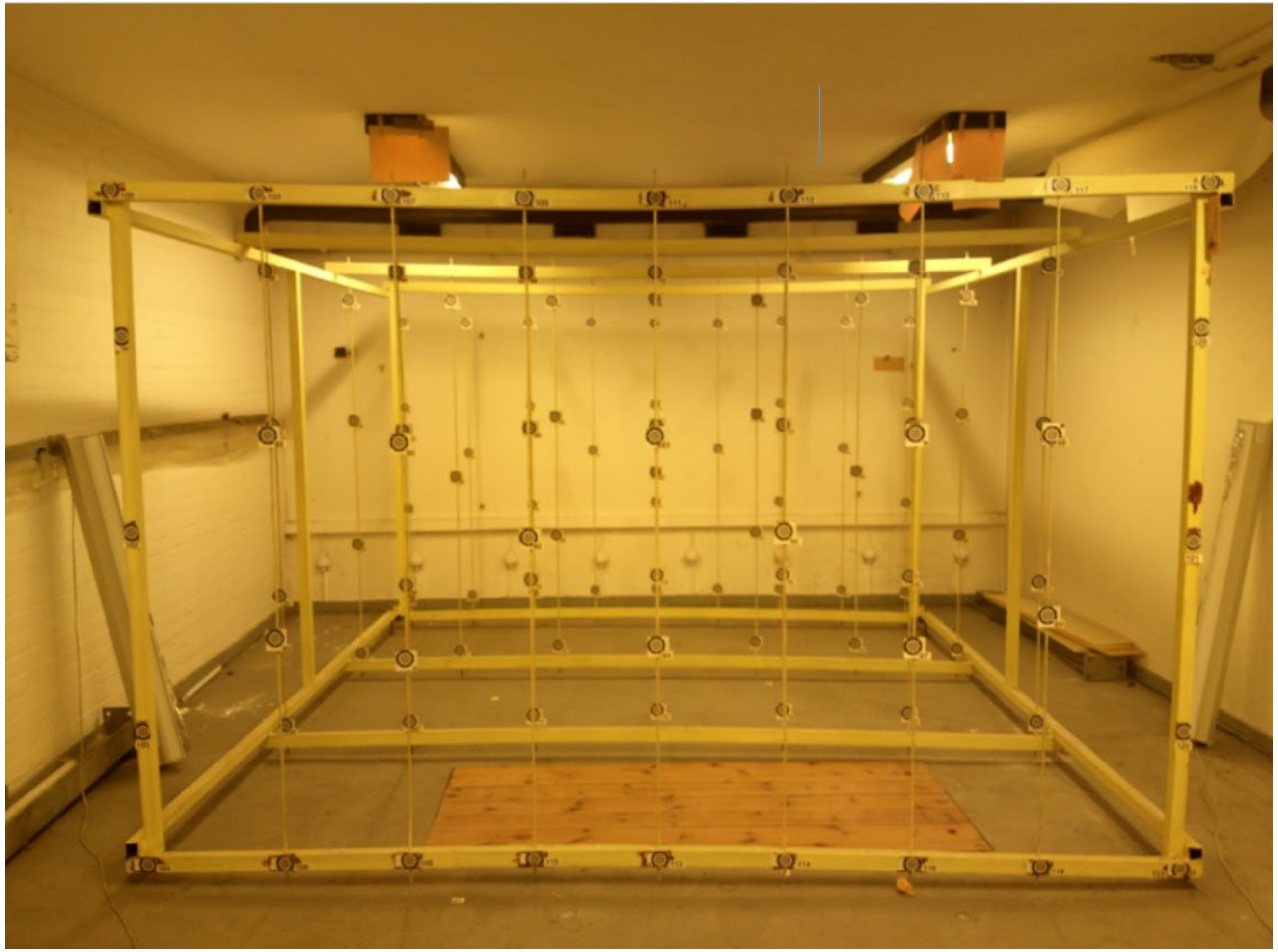
$$y_d = y(p_1(r^2 + 2y^2) + 2p_2xy)$$

Camera Calibration



Source: Henrik Aanæs – Lecture Notes in C.V.

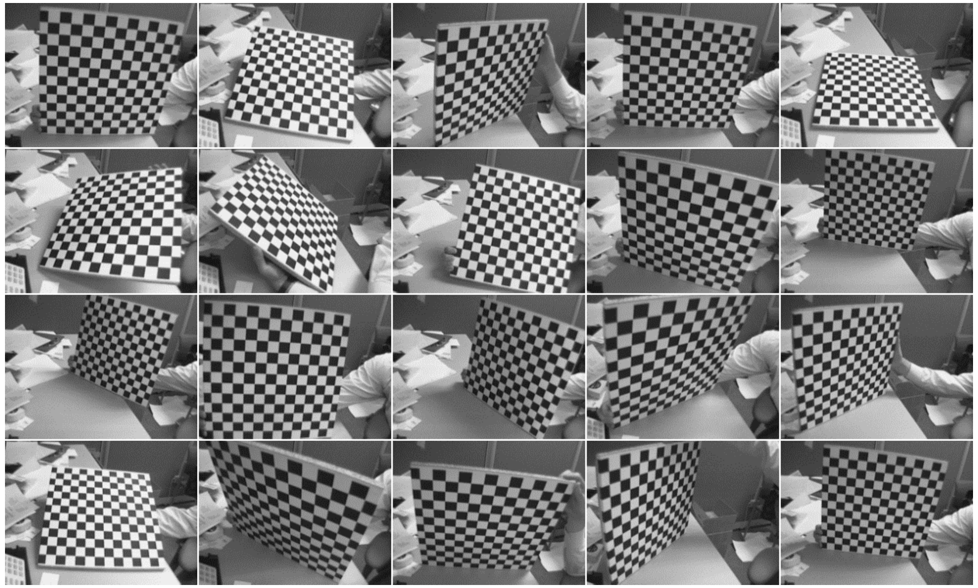
Calibration Rig (Landmålervej)



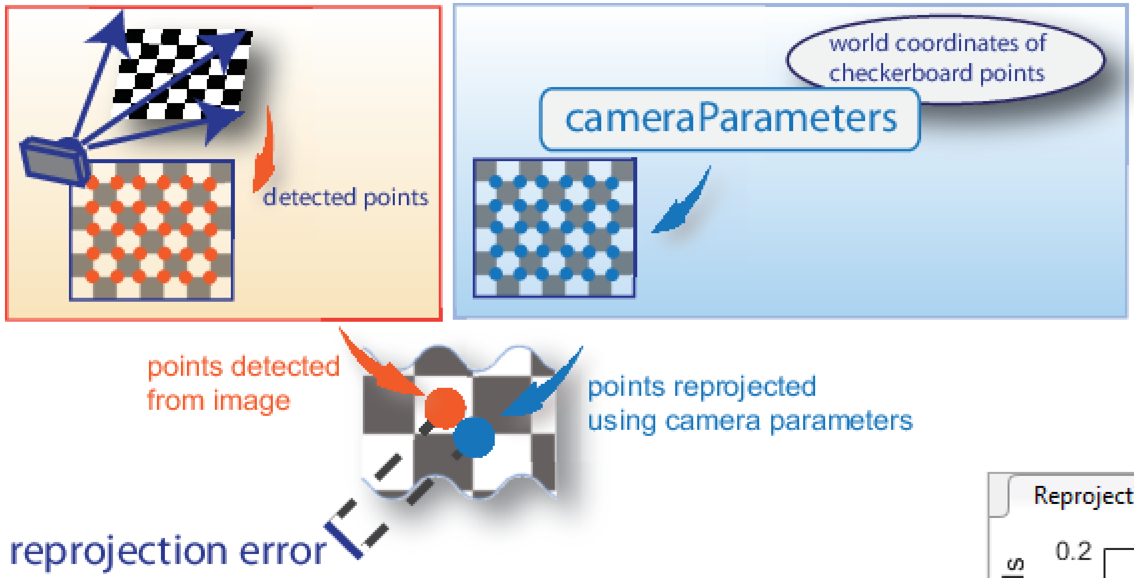
- Coordinates of camera-centre and control points known to high accuracy
- From 3D to image correspondences, intrinsic parameters of the camera can be estimated

Checker-board Calibration

- Using the calibration-rig is cumbersome and unflexible
- A more recent method, involving images of a checkerboard from different views has become the de facto procedure in C.V. and Photogrammetry (Zhang 1998)



Evaluation of Calibration accuracy



Source: Mathworks Inc