Photogrammetry & Robotics Lab

Projective 3-Point (P3P) Algorithm or Spatial Resection

Cyrill Stachniss

Camera Localization

Task: estimate the pose of the camera

Camera Localization

Given:

- 3D coordinates of object points $\mathbf{X}_i$

Observed:

- 2D image coordinates $\mathbf{x}_i$ of the object points

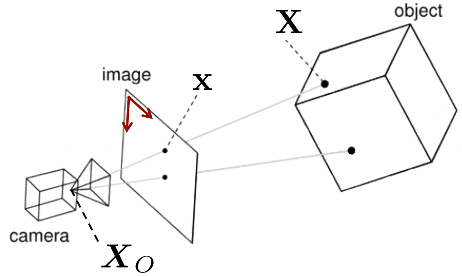
Wanted:

- Extrinsic parameters $R, \boldsymbol{X}_O$ of the calibrated camera

Reminder: Mapping Model

Direct linear transform (DLT) maps any object point $\mathbf{X}$ to the image point $\mathbf{x}$

$$\begin{aligned} \mathbf{x} &= \mathsf{K}R[I_3| - \boldsymbol{X}_O]\mathbf{X} \\ &= \mathsf{P}\ \mathbf{X} \end{aligned}$$



Reminder: Camera Orientation

$$\mathbf{x} = \mathsf{K}R[I_3|-\boldsymbol{X}_O]\mathbf{X} = \mathsf{P}\ \mathbf{X}$$

- **Intrinsics (interior orientation)**
  - Intrinsic parameters of the camera
  - Given through matrix K
- **Extrinsics (exterior orientation)**
  - Extrinsic parameters of the camera
  - Given through $\boldsymbol{X}_O$ and R

Direct Linear Transform (DLT)

Relation to DLT : Compute the **11 intrinsic and extrinsic parameters**

$$\mathbf{x} = \mathsf{K}R[I_3|-\boldsymbol{X}_O]\mathbf{X}$$

observed image point

c, s, m, x_H, y_H

3 rotations

3 translations

control point coordinates (given)

Projective 3-Point Algorithm (or Spatial Resection)

Given the intrinsic parameters, compute the **6 extrinisic parameters**

$$\mathbf{x} = \mathsf{K}R[I_3|-\boldsymbol{X}_O]\mathbf{X}$$

observed image points

c, s, m, x_H, y_H (given)

3 rotations

3 translations

control point coordinates (given)

P3P/SR vs. DLT

- **P3P/SR: Calibrated camera**
  - 6 unknowns
  - We need at least **3 points**
- **DLT: Uncalibrated camera**
  - 11 unknowns
  - We need at least **6 points**
  - Assuming an **affine camera** (straight-line preserving projection)

Orienting a calibrated camera by using ≥ 3 points

P3P/Spatial Resection (direct solution)

Problem Formulation

Given:

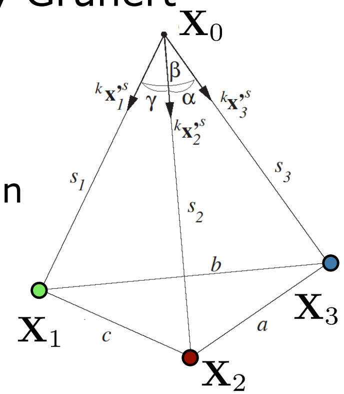
- 3D coordinates $\mathbf{X}_i$ of $I \geq 3$ object points
- Corresponding image coordinates $\mathbf{x}_i$ recorded using a calibrated camera

Task:

- Estimate the 6 parameters $\boldsymbol{X}_O, R$
- Direct solution (no initial guess)

Different Approaches

- Different approaches: Grunert 1841, Killian 1955, Rohrberg 2009, ...
- Here: direct solution by Grunert
- **2-step process**
  1. Estimate length of projection rays
  2. Estimate the orientation

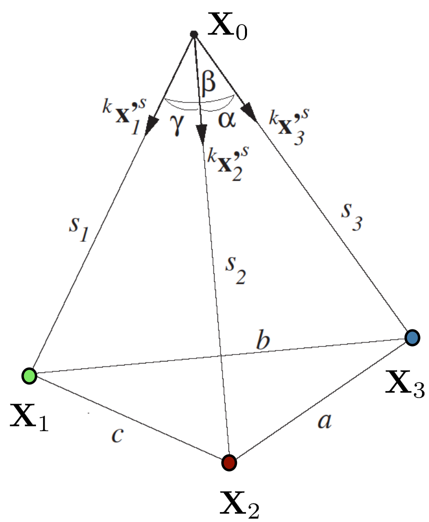


Direct Solution by Grunert

2-Step process

Estimate

1. length of projection rays
2. orientation

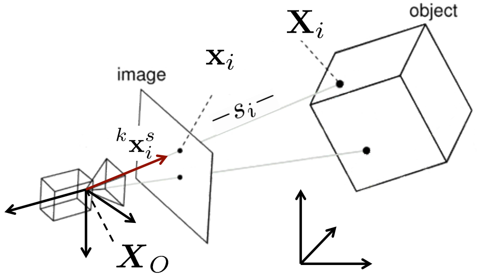


P3P/SR Model

- Coordinates of object points **within the camera system** are given by

$$s_i \, {}^k\mathbf{x}_i^s = \mathsf{R}(\boldsymbol{X}_i - \boldsymbol{X}_O) \qquad\qquad i = 1, 2, 3$$

ray directions pointing to the object points



P3P/SR Model

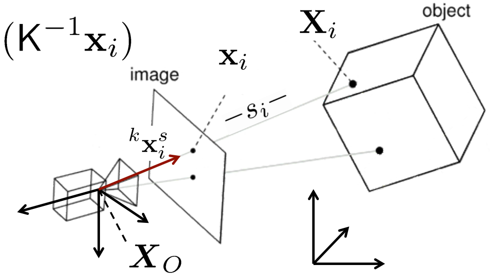
- Coordinates of object points within the camera system are given by

$$s_i \, {}^k\mathbf{x}_i^s = \mathsf{R}(\boldsymbol{X}_i - \boldsymbol{X}_O) \qquad\qquad i = 1, 2, 3$$

- From image coordinates, we obtain the directional vector of projection ray

$${}^k\mathbf{x}_i^s = -\mathrm{sign}(c)\mathsf{N}(\mathsf{K}^{-1}\mathbf{x}_i)$$

ensure ray directions are pointing to the object points



P3P/SR Model

- Coordinates of object points within the camera system are given by

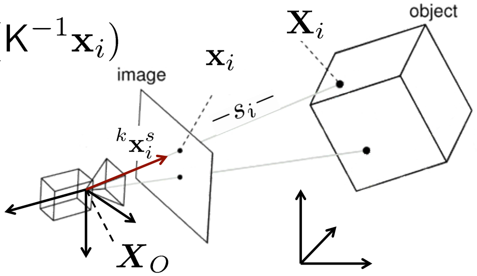
$$s_i \, {}^k\mathbf{x}_i^s = \mathsf{R}(\boldsymbol{X}_i - \boldsymbol{X}_O) \qquad\qquad i = 1, 2, 3$$

- From image coordinates, we obtain the directional vector of projection ray

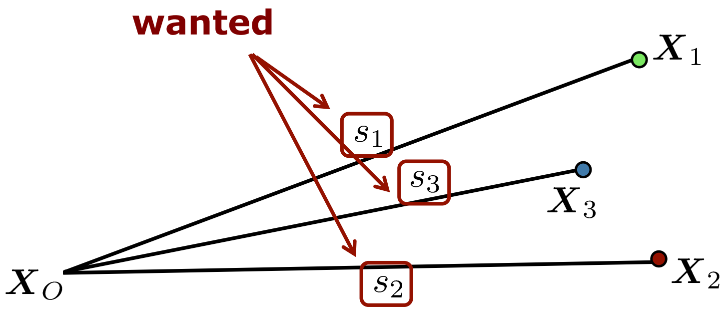
$${}^k\mathbf{x}_i^s = -\mathrm{sign}(c)\mathsf{N}(\mathsf{K}^{-1}\mathbf{x}_i)$$

spherical normalization

$$\mathsf{N}(\mathbf{x}) = \frac{\mathbf{x}}{|\mathbf{x}|}$$



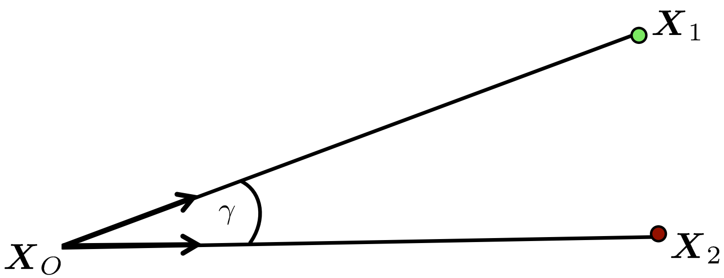
1. Get Length of Projection Rays



1. Get Length of Projection Rays

- Start with computing the angle between rays:

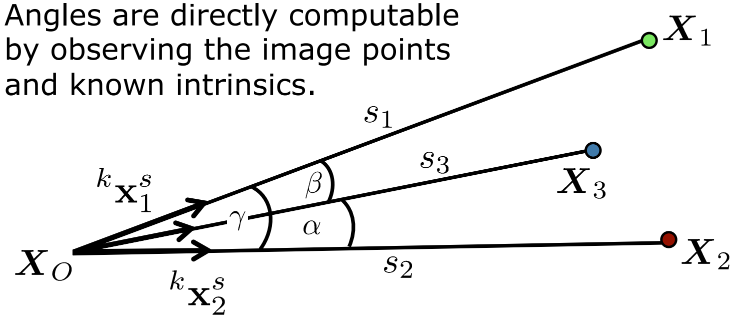
$$\cos\gamma = \frac{(\boldsymbol{X}_1 - \boldsymbol{X}_0)\cdot(\boldsymbol{X}_2 - \boldsymbol{X}_0)}{||\boldsymbol{X}_1 - \boldsymbol{X}_0||\,||\boldsymbol{X}_2 - \boldsymbol{X}_0||}$$



1. Get Length of Projection Rays

$$\begin{aligned}\alpha &= \arccos\left({}^k\mathbf{x}_2^s,\ {}^k\mathbf{x}_3^s\right)\\ \beta &= \arccos\left({}^k\mathbf{x}_3^s,\ {}^k\mathbf{x}_1^s\right)\\ \gamma &= \arccos\left({}^k\mathbf{x}_1^s,\ {}^k\mathbf{x}_2^s\right)\end{aligned}$$

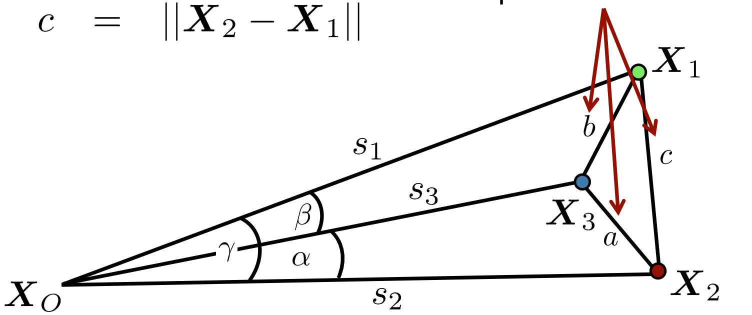
Angles are directly computable by observing the image points and known intrinsics.



1. Get Length of Projection Rays

$$\begin{aligned}a &= ||\boldsymbol{X}_3 - \boldsymbol{X}_2||\\ b &= ||\boldsymbol{X}_1 - \boldsymbol{X}_3||\\ c &= ||\boldsymbol{X}_2 - \boldsymbol{X}_1||\end{aligned}$$

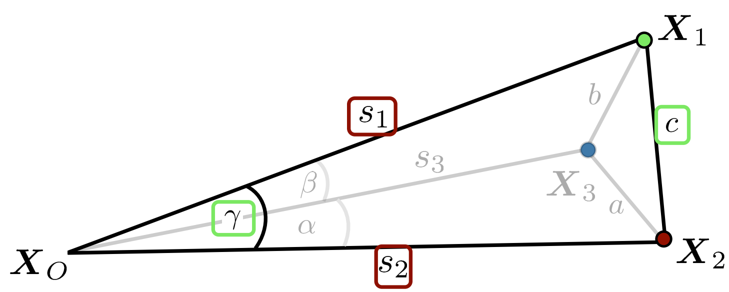
Given through known control point coordinates



Use the Law of Cosines

In triangle $\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2$

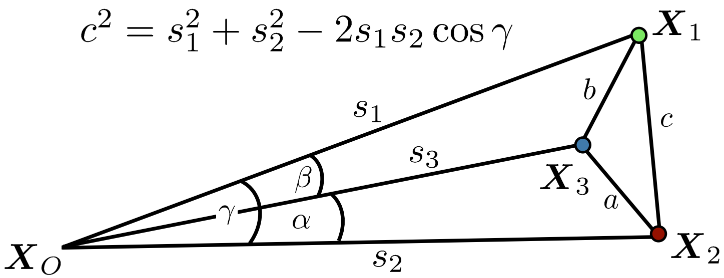
$$s_1^2 + s_2^2 - 2\underbrace{\boxed{s_1}\boxed{s_2}}_{\text{wanted}}\cos\underbrace{\boxed{\gamma} = \boxed{c^2}}_{\text{known}}$$



Use the Law of Cosines

Analogously in all three triangles

$$a^2 = s_2^2 + s_3^2 - 2 s_2 s_3 \cos\alpha$$
$$b^2 = s_1^2 + s_3^2 - 2 s_1 s_3 \cos\beta$$
$$c^2 = s_1^2 + s_2^2 - 2 s_1 s_2 \cos\gamma$$



Compute Distances

- We start from:

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

- Define: $u = \frac{s_2}{s_1} \qquad v = \frac{s_3}{s_1}$
- Substitution leads to:

$$a^2 = s_1^2 (u^2 + v^2 - 2uv\cos\alpha)$$

- Rearrange to: $s_1^2 = \frac{a^2}{u^2+v^2-2uv\cos\alpha}$

Compute Distances

- Use the same definition

$$u = \frac{s_2}{s_1} \qquad v = \frac{s_3}{s_1}$$

- And perform the substitution again for:

$$b^2 = s_1^2 + s_3^2 - 2 s_1 s_3 \cos\beta$$
$$c^2 = s_1^2 + s_2^2 - 2 s_1 s_2 \cos\gamma$$

Compute Distances

Analogously, we obtain

$$\begin{aligned} s_1^2 &= \frac{a^2}{u^2 + v^2 - 2uv\cos\alpha} \\ &= \frac{b^2}{1 + v^2 - 2v\cos\beta} \\ &= \frac{c^2}{1 + u^2 - 2u\cos\gamma} \end{aligned}$$

Rearrange Again

Solve one equation for u put into the other

$$s_1^2 = \frac{a^2}{u^2+v^2-2uv\cos\alpha}$$

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

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

Results in an fourth degree polynomial

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

Forth Degree Polynomial

$$\boxed{A_4}v^4 + \boxed{A_3}v^3 + A_2v^2 + A_1v + A_0 = 0$$

$$\boxed{A_4} = \left(\frac{a^2-c^2}{b^2} - 1\right)^2 - \frac{4c^2}{b^2}\cos^2\alpha$$

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

Forth Degree Polynomial

$$A_4v^4 + A_3v^3 + \boxed{A_2}v^2 + A_1v + A_0 = 0$$

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

Forth Degree Polynomial

$$A_4v^4 + A_3v^3 + A_2v^2 + \boxed{A_1}v + \boxed{A_0} = 0$$

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

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

Forth Degree Polynomial

$$A_4 v^4 + A_3 v^3 + A_2 v^2 + A_1 v + A_0 = 0$$

Solve for v to get s_1, s_2, s_3 through:

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

$$s_3 = v\ s_1$$

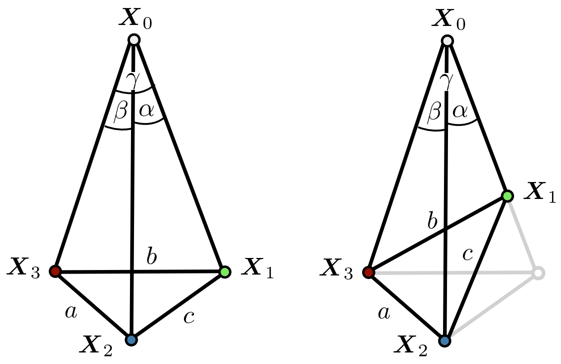
$$a^2 = s_2^2 + s_3^2 - 2 s_2 s_3 \cos\alpha \Rightarrow s_2 = \ \cdots$$

Problem:
up to 4 possible solutions !

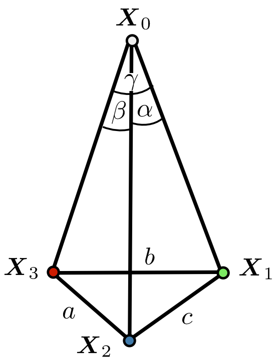
$$\{s_1, s_2, s_3\}_{1\ldots4}$$

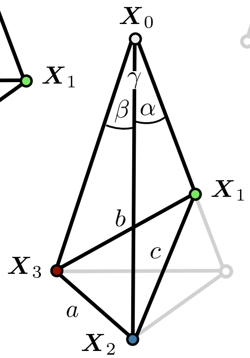
Example for Multiple Solutions

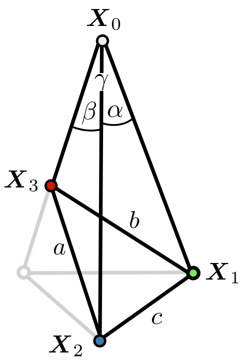
- Assume $a = b = c$ and $\alpha = \beta = \gamma$
- Tilting the triangle $(\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3)$ has no effect on (a, b, c) and (α, β, γ)

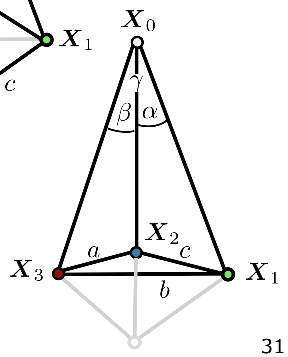


Four Solutions









How to Eliminate This Ambiguity?

- Known approximate solution (e.g. from GPS) or
- Use 4th points to confirm the right solution

Unique solution for
s_1, s_2, s_3

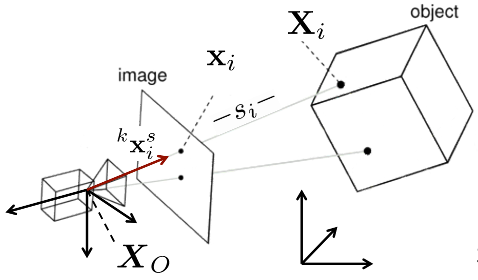
2. Orientation of the Camera

Given:

- Distances and direction vectors to the control points

Task:

- Estimate 6 extrinsic parameters

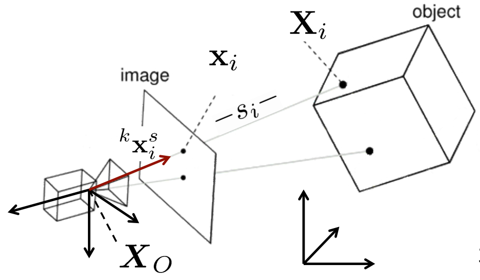


2. Orientation of the Camera

1. Compute 3D coordinates of the control points in the camera system

$$ {}^k\boldsymbol{X}_i = s_i \, {}^k\mathbf{x}_i^s \qquad i = 1, 2, 3 $$

That's what we just discussed!



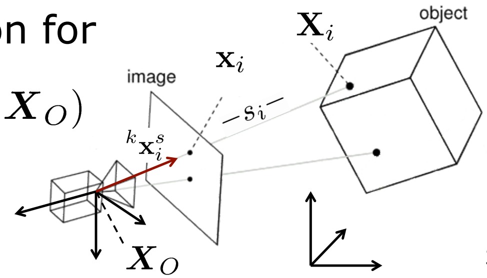
2. Orientation of the Camera

1. Compute 3D coordinates of the control points in the camera system

$$ {}^k\boldsymbol{X}_i = s_i \, {}^k\mathbf{x}_i^s \qquad i = 1, 2, 3 $$

2. Compute coordinate transformation for

$$ {}^k\boldsymbol{X}_i = R(\boldsymbol{X}_i - \boldsymbol{X}_O) $$



P3P/SR in a Nutshell

$$ \mathbf{X}_i, \mathbf{x}_i, \mathsf{K} \qquad i = 1, 2, 3, 4 $$

↓

Compute angles

↓ α, β, γ

Computes distances

↓ $\{s_1, s_2, s_3\}_{1\ldots4}$

Identify correct solution by the 4th point

↓ s_1, s_2, s_3

Compute coordinate transformation

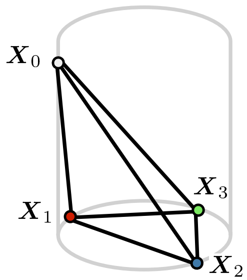
↓

$$ R, \boldsymbol{X}_0 $$

Critical Surfaces

"Critical cylinder"

- If the projection center lies on a cylinder defined by the control points
- Small changes in angles lead to large changes in coordinates
- Instable solution



Outlier Handling with RANSAC

Use **direct solution** to find correct solution among set of corrupted points

- Assume I≥3 points

Repeat

1. Select 3 points randomly
2. Estimate parameters of SR/P3P
3. Count the number of other points that support current hypotheses
4. Select best solution

- Can deal with large numbers of outliers in data

Orienting a calibrated camera by using > 3 points

Spatial Resection Iterative Solution

Overview: Iterative Solution

- Over determined system with $I > 3$
- No direct solution but iterative LS
- Main steps
  - Build the system of observation equations
  - Measure image points $\boldsymbol{x}_i, \ i = 1, \dots I$
  - Estimate initial solution $R, \boldsymbol{X}_o \rightarrow \boldsymbol{x}^{(0)}$
  - Adjustment
    - Linearizing
    - Estimate extrinsic parameter $\widehat{\boldsymbol{x}}$
    - Iterate until convergence

Summary: P3P/SR

- Estimates the **pose** of a **calibrated camera** given control points
- Uses **≥3 points**
- **Direct solution**
  - Fast
  - Suited for outlier detection with RANSAC
- **Statistically optimal solution using iterative least squares**
  - Uses all available points
  - Assumes no outliers
  - Allows for accuracy assessments