Photogrammetry & Robotics Lab

Triangulation and Absolute Orientation

Cyrill Stachniss

5 Minute Preparation for Today

https://www.ipb.uni-bonn.de/5min/

Motivation

Given the relative orientation of two images, compute the points in 3D

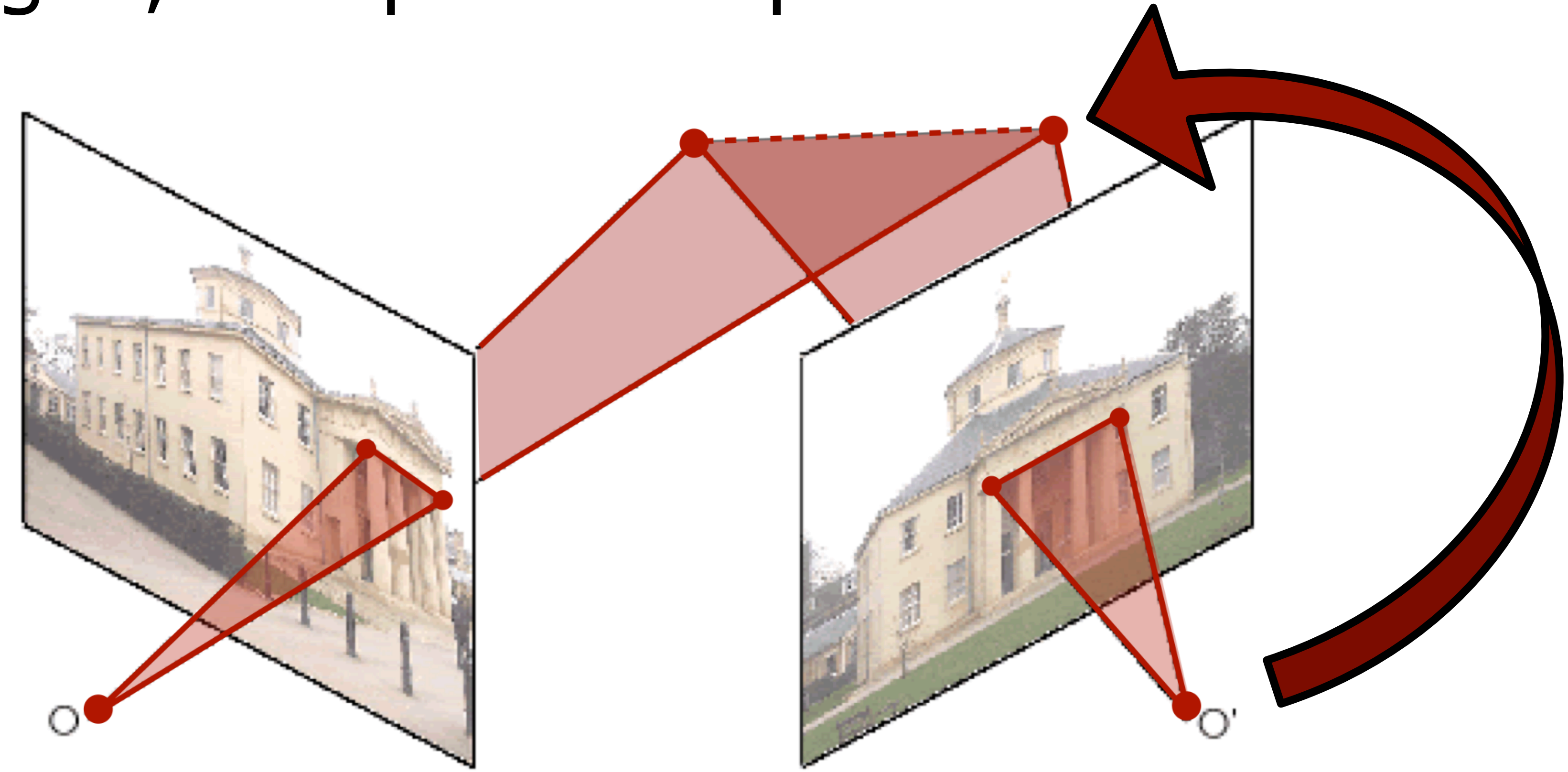
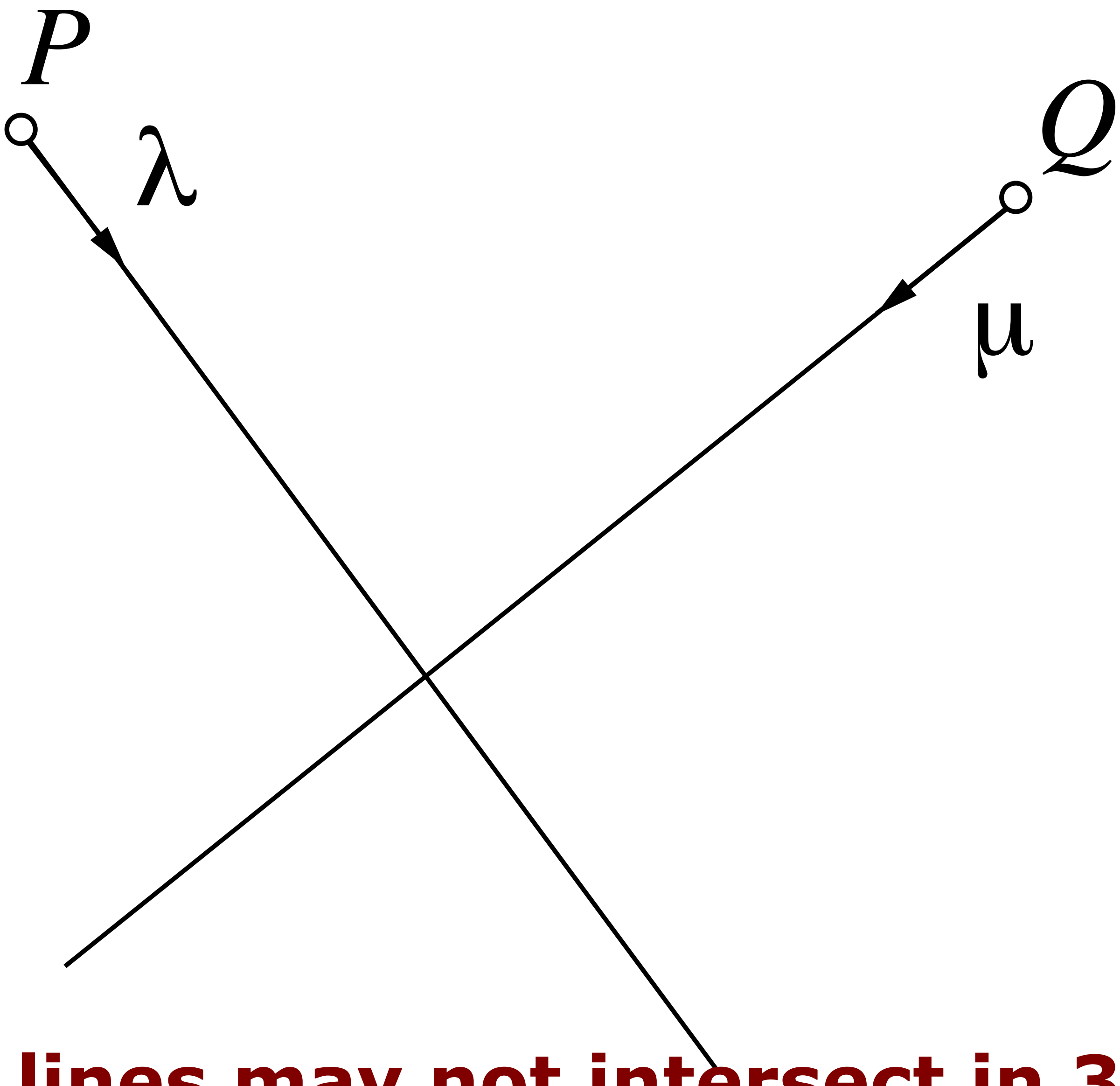


Image courtesy: Schindler

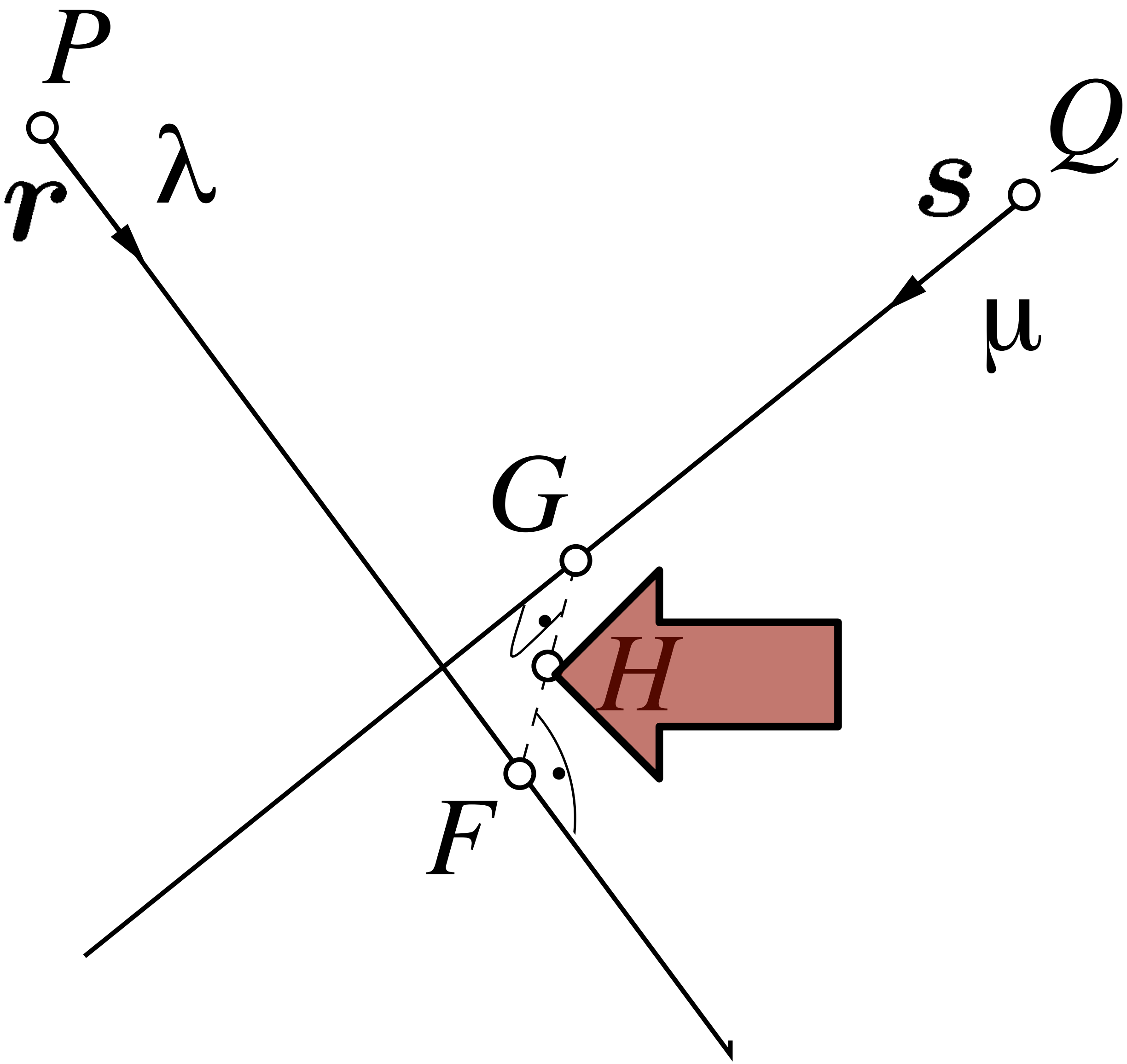
1.
Geometric Approach

The Problem



The lines may not intersect in 3D!

Find the Point H



Geometric Solution

- Equation for two lines in 3D

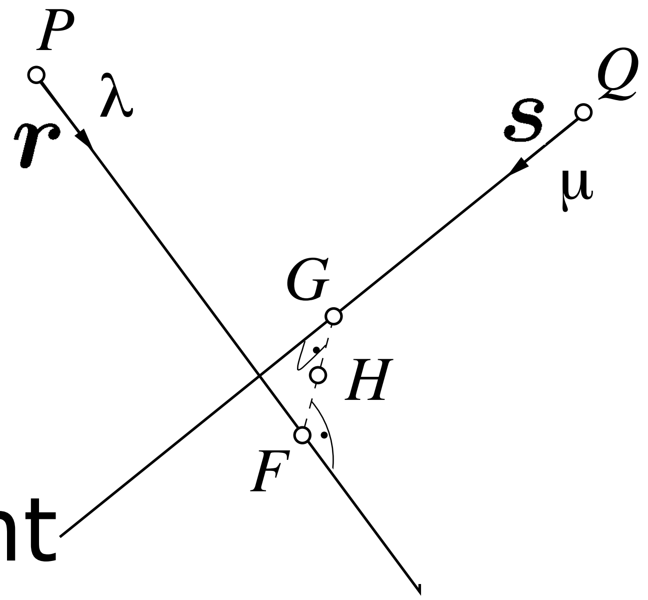
$$\boldsymbol{f} = \boldsymbol{p} + \lambda \boldsymbol{r} \qquad \boldsymbol{g} = \boldsymbol{q} + \mu \boldsymbol{s}$$

- with the points $\boldsymbol{p} = \boldsymbol{X}_{O'} \quad \boldsymbol{q} = \boldsymbol{X}_{O''}$
- and the directions (calibrated camera)

$$\boldsymbol{r} = R'^{\mathsf{T}}\ {}^{k}\mathbf{x}' \qquad \boldsymbol{s} = R''^{\mathsf{T}}\ {}^{k}\mathbf{x}''$$

- with ${}^{k}\mathbf{x}' = (x', y', c)^{\mathsf{T}} \quad {}^{k}\mathbf{x}'' = (x'', y'', c)^{\mathsf{T}}$

Geometric Solution



- The shortest connection requires that FG is orthogonal to both lines
- This leads to the constraint

$$(\boldsymbol{f} - \boldsymbol{g}) \cdot \boldsymbol{r} = 0 \qquad (\boldsymbol{f} - \boldsymbol{g}) \cdot \boldsymbol{s} = 0$$

which directly leads to

$$\begin{aligned} (\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{s} &= 0 \\ (\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{r} &= 0 \end{aligned}$$

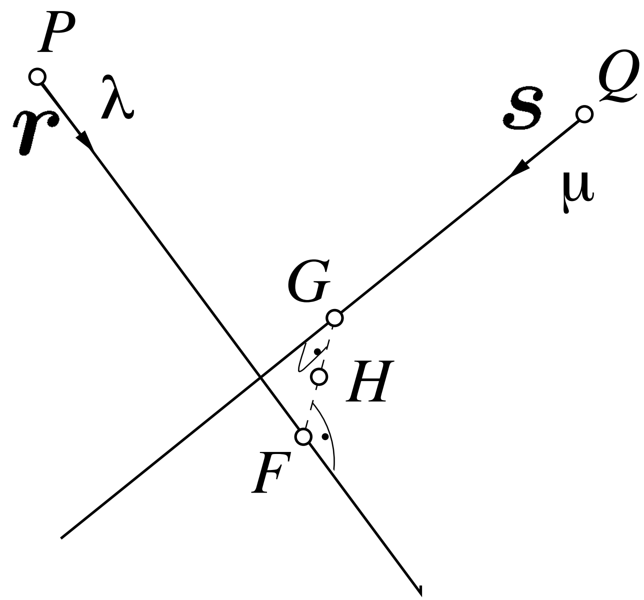
- Two equations, two unknowns

Geometric Solution

- By solving the equations

$$\begin{aligned}(\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{s} &= 0 \\ (\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{r} &= 0\end{aligned}$$

  we obtain λ, μ
- λ, μ directly yield F and G
- We compute H as the middle of the line connecting F and G



More Concretely...

- For the stereo setup

$$
\begin{aligned}
(\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{s} &= 0 \\
(\boldsymbol{q} + \lambda \boldsymbol{s} - \boldsymbol{p} - \mu \boldsymbol{r}) \cdot \boldsymbol{r} &= 0
\end{aligned}
$$

- Projection centers $\boldsymbol{p} = \boldsymbol{X}_{O'} \quad \boldsymbol{q} = \boldsymbol{X}_{O''}$
- Ray direction vectors $\boldsymbol{r} = R'^{\mathsf{T}}\, {}^{k}\mathbf{x}' \quad \boldsymbol{s} = R''^{\mathsf{T}}\, {}^{k}\mathbf{x}''$

$$
\begin{aligned}
(\boldsymbol{X}_{O'} + \lambda \boldsymbol{r} - \boldsymbol{X}_{O''} - \mu \boldsymbol{s})^{\top} \boldsymbol{r} = 0 \\
(\boldsymbol{X}_{O'} + \lambda \boldsymbol{r} - \boldsymbol{X}_{O''} - \mu \boldsymbol{s})^{\top} \boldsymbol{s} = 0
\end{aligned}
$$

More Concretely...

- Rearranging

$$(\boldsymbol{X}_{O'} + \lambda \boldsymbol{r} - \boldsymbol{X}_{O''} - \mu \boldsymbol{s})^\top \boldsymbol{r} = 0$$
$$(\boldsymbol{X}_{O'} + \lambda \boldsymbol{r} - \boldsymbol{X}_{O''} - \mu \boldsymbol{s})^\top \boldsymbol{s} = 0$$

- leads to

$$(\boldsymbol{X}_{O'} - \boldsymbol{X}_{O''})^\top \boldsymbol{r} + \lambda \boldsymbol{r}^\top \boldsymbol{r} - \mu \boldsymbol{s}^\top \boldsymbol{r} = 0$$
$$(\boldsymbol{X}_{O'} - \boldsymbol{X}_{O''})^\top \boldsymbol{s} + \lambda \boldsymbol{r}^\top \boldsymbol{s} - \mu \boldsymbol{s}^\top \boldsymbol{s} = 0$$

More Concretely...

- We can transforms

$$
\begin{aligned}
(\boldsymbol{X}_{O'} - \boldsymbol{X}_{O''})^\top \boldsymbol{r} + \lambda \boldsymbol{r}^\top \boldsymbol{r} - \mu \boldsymbol{s}^\top \boldsymbol{r} = 0 \\
(\boldsymbol{X}_{O'} - \boldsymbol{X}_{O''})^\top \boldsymbol{s} + \lambda \boldsymbol{r}^\top \boldsymbol{s} - \mu \boldsymbol{s}^\top \boldsymbol{s} = 0
\end{aligned}
$$

- into matrix form

$$
\begin{bmatrix} \boldsymbol{r}^\top \boldsymbol{r} & -\boldsymbol{s}^\top \boldsymbol{r} \\ \boldsymbol{r}^\top \boldsymbol{s} & -\boldsymbol{s}^\top \boldsymbol{s} \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \end{bmatrix} = \begin{bmatrix} (\boldsymbol{X}_{O''} - \boldsymbol{X}_{O'})^\top \\ (\boldsymbol{X}_{O''} - \boldsymbol{X}_{O'})^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{s} \end{bmatrix}
$$

More Concretely...

- So that we can solve

$$\begin{bmatrix} \boldsymbol{r}^\top \boldsymbol{r} & -\boldsymbol{s}^\top \boldsymbol{r} \\ \boldsymbol{r}^\top \boldsymbol{s} & -\boldsymbol{s}^\top \boldsymbol{s} \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \end{bmatrix} = \begin{bmatrix} (\boldsymbol{X}_{O''} - \boldsymbol{X}_{O'})^\top \\ (\boldsymbol{X}_{O''} - \boldsymbol{X}_{O'})^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{s} \end{bmatrix}$$

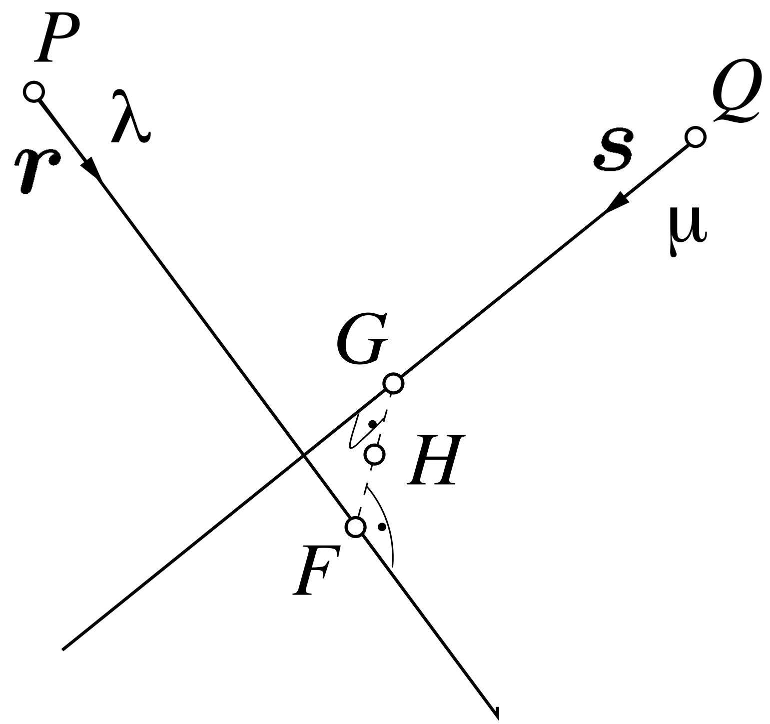
$$A \qquad \boldsymbol{x} \qquad \boldsymbol{b}$$

- with our standard $A\boldsymbol{x} = \boldsymbol{b}$ formulation
- Knowing λ, μ allows us to compute the intersecting point

Solution

- λ, μ directly yield F and G
- The 3D point H is the middle of the line connecting F and G
- The solution is:

$$\boldsymbol{H} = \frac{\boldsymbol{F} + \boldsymbol{G}}{2}$$

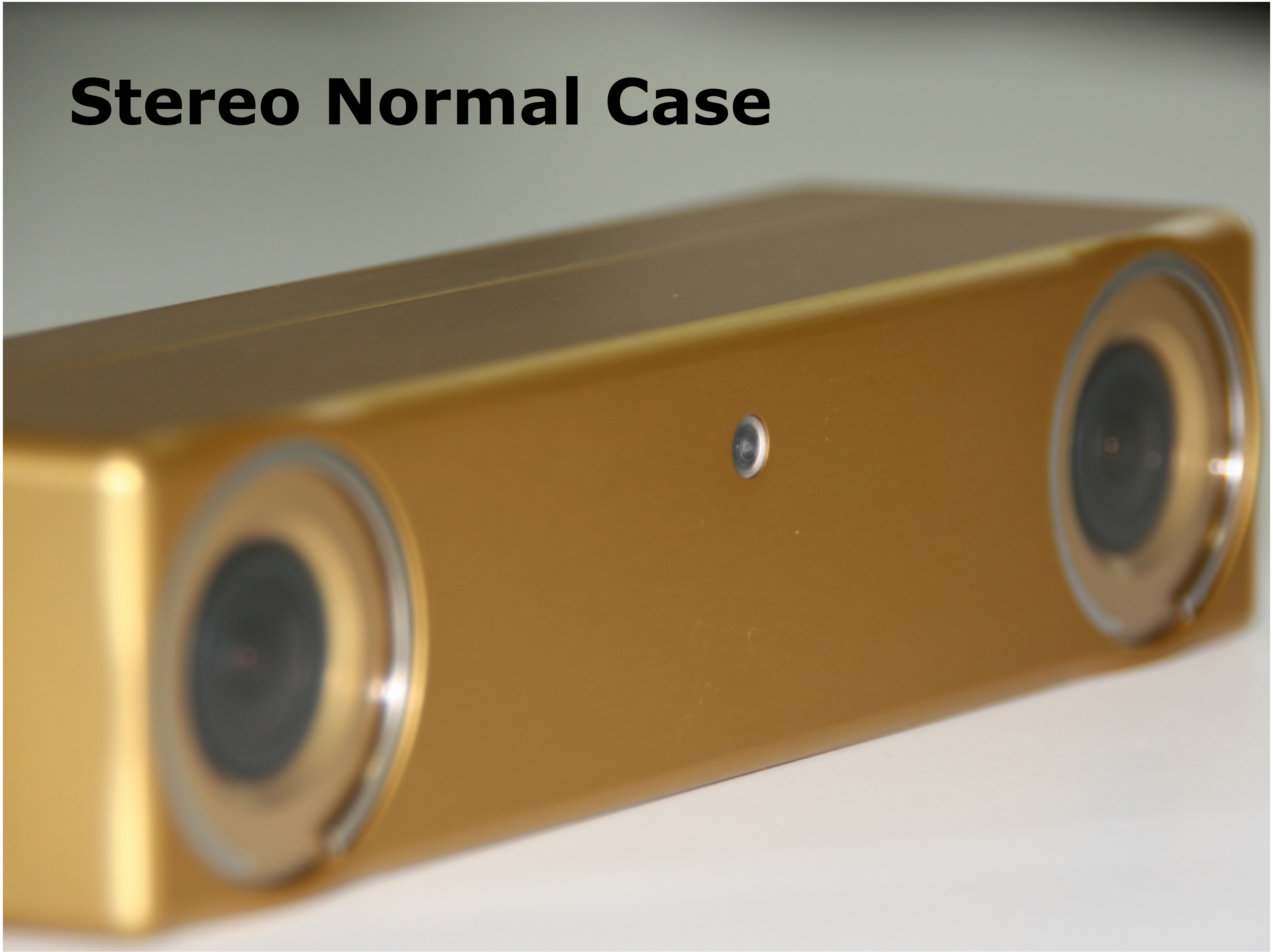


Geometric Solution

- Simple 3D geometry allows us to compute a solution
- Boils down to solving a system of two linear equations with two unknowns
- Does not take into account uncertainties, not statistically optimal

2.
For the Stereo Normal Case

Stereo Normal Case



Stereo Normal Case

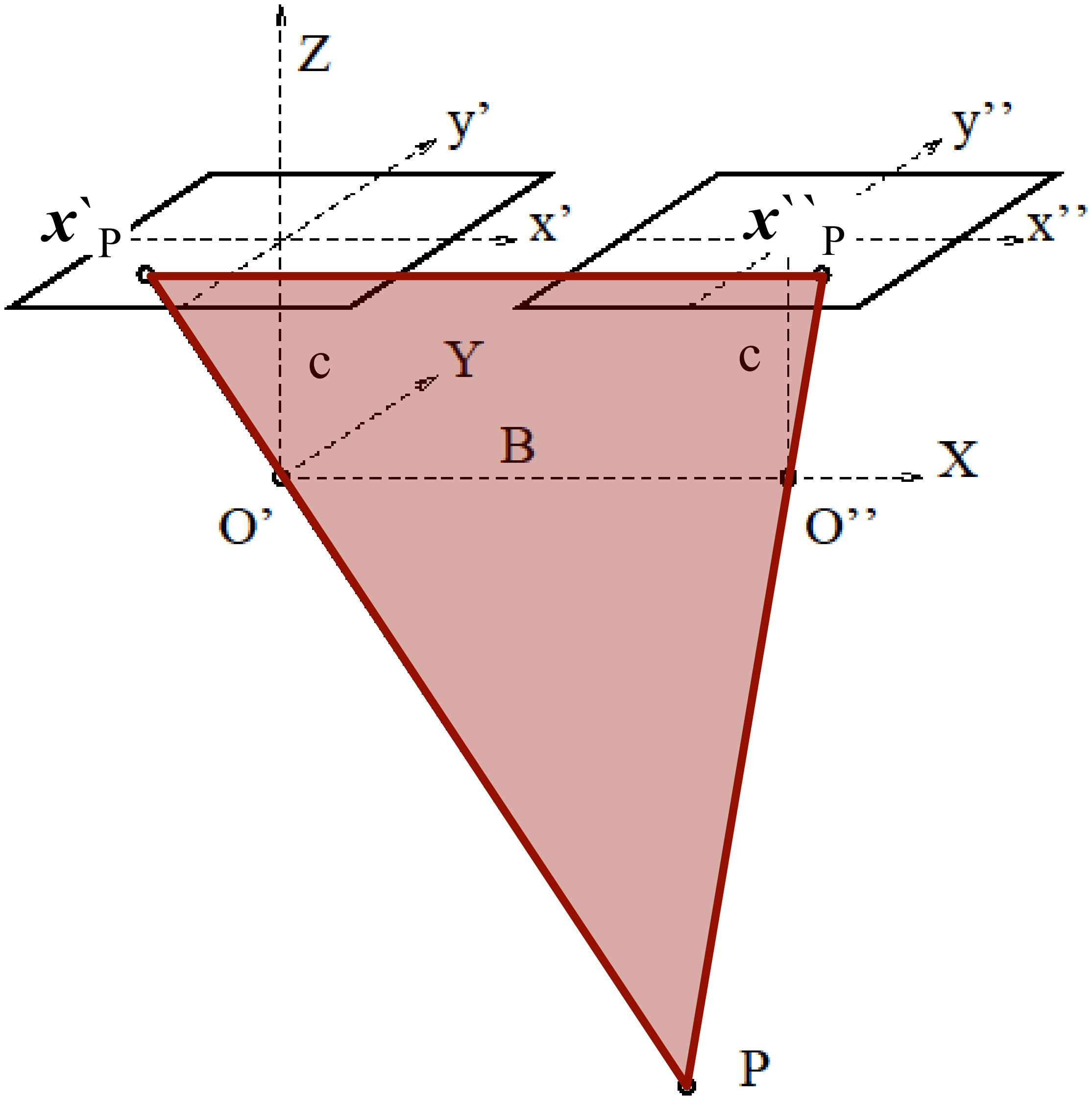


Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

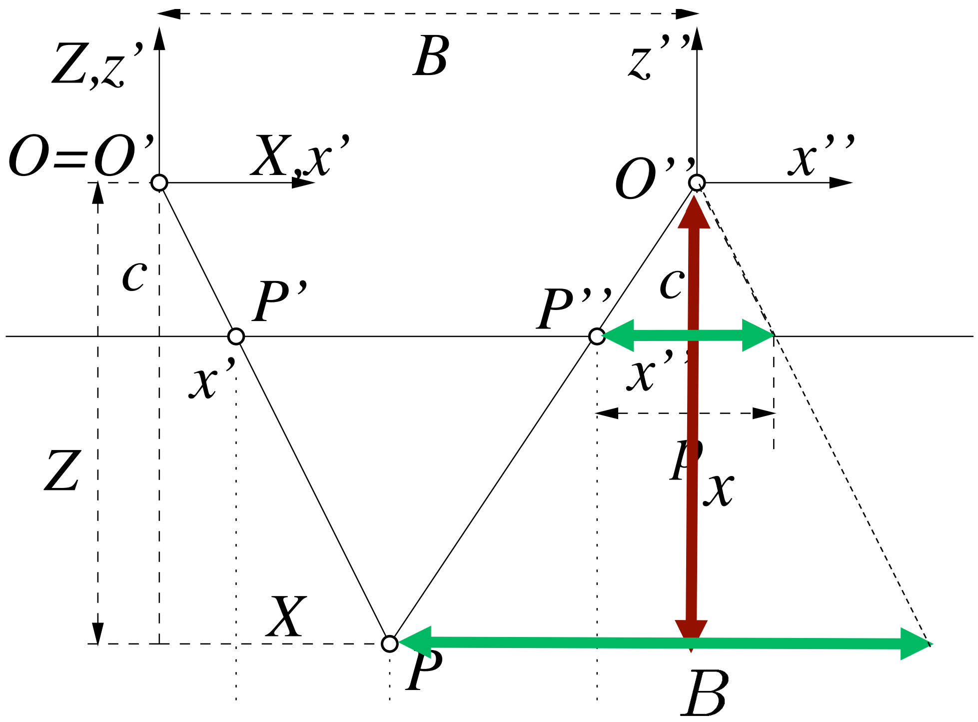
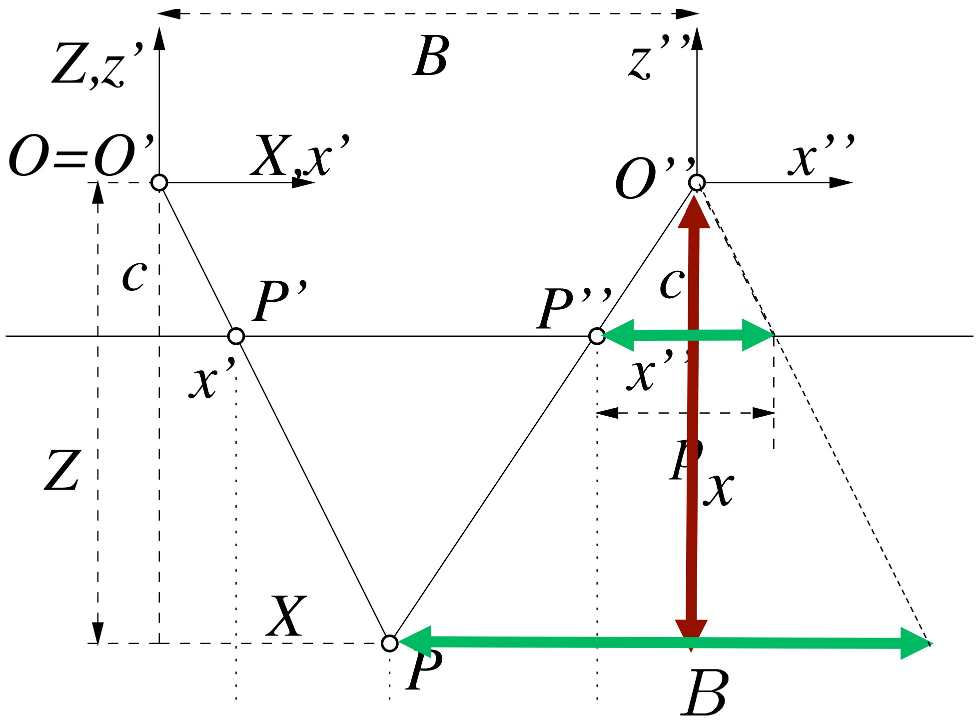


Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem



$$\frac{Z}{c} = \frac{B}{x' - x''}$$

Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

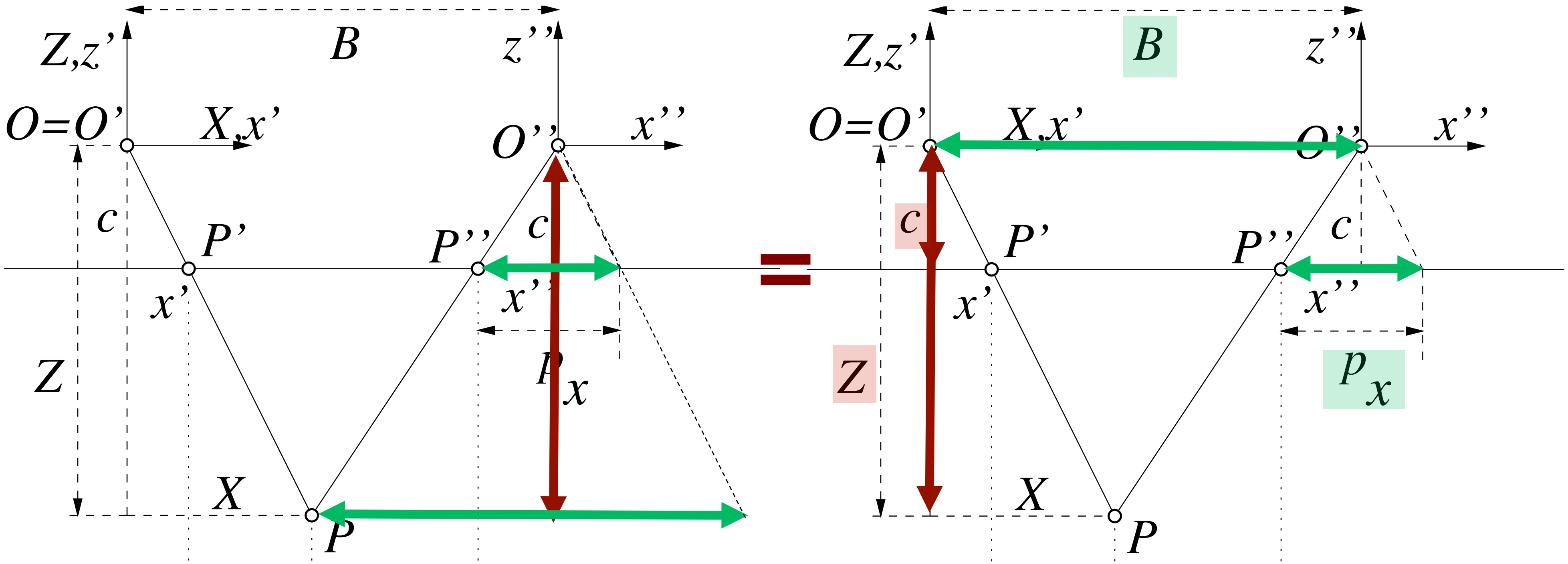
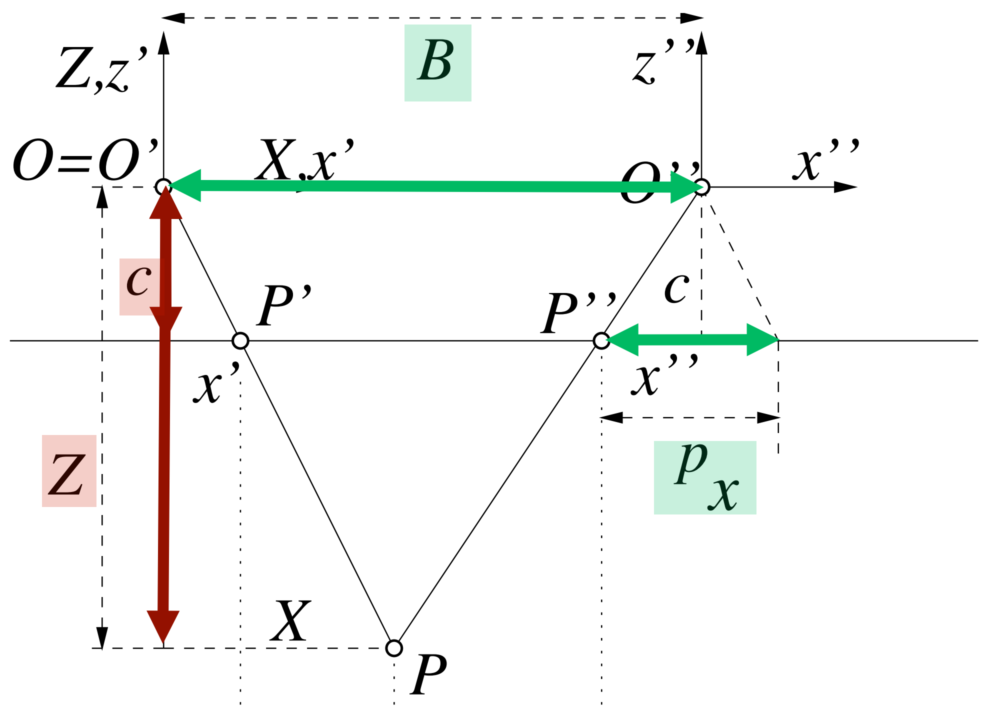


Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$\frac{Z}{c} = \frac{B}{-\underbrace{(x'' - x')}_{p_x}}$$



Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$\frac{Z}{c} = \frac{B}{\underbrace{-(x'' - x')}_{p_x}}$$

$$\Rightarrow \quad Z = c\frac{B}{-(x'' - x')}$$

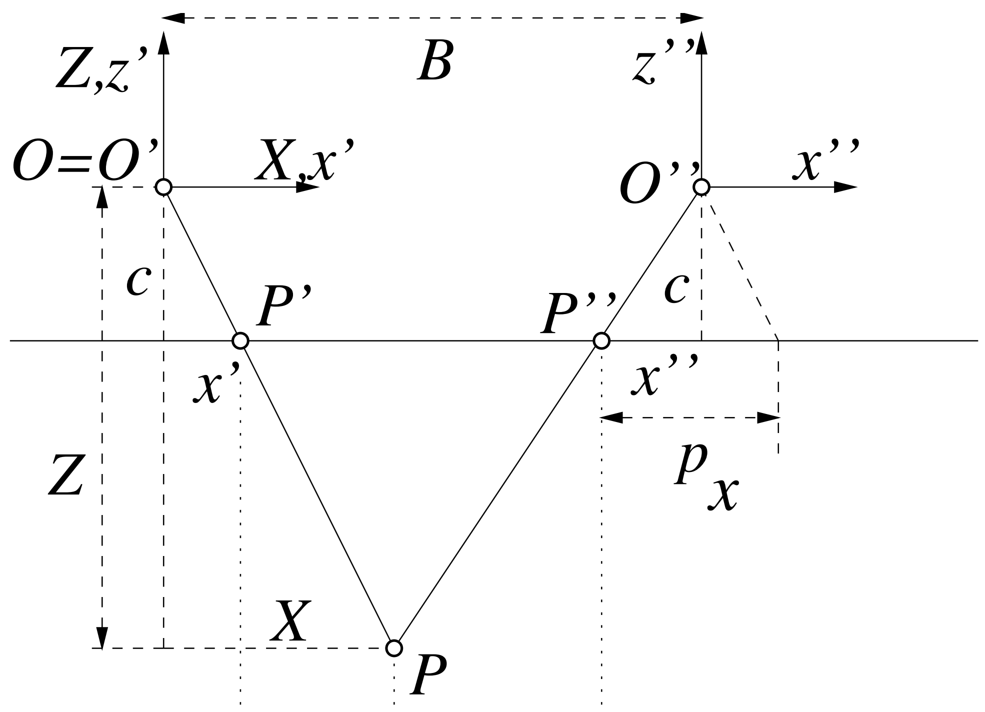


Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$Z = c\frac{B}{-(x'' - x')}$$

2. X-coordinate

$$\frac{X}{x'} = \frac{Z}{c}$$

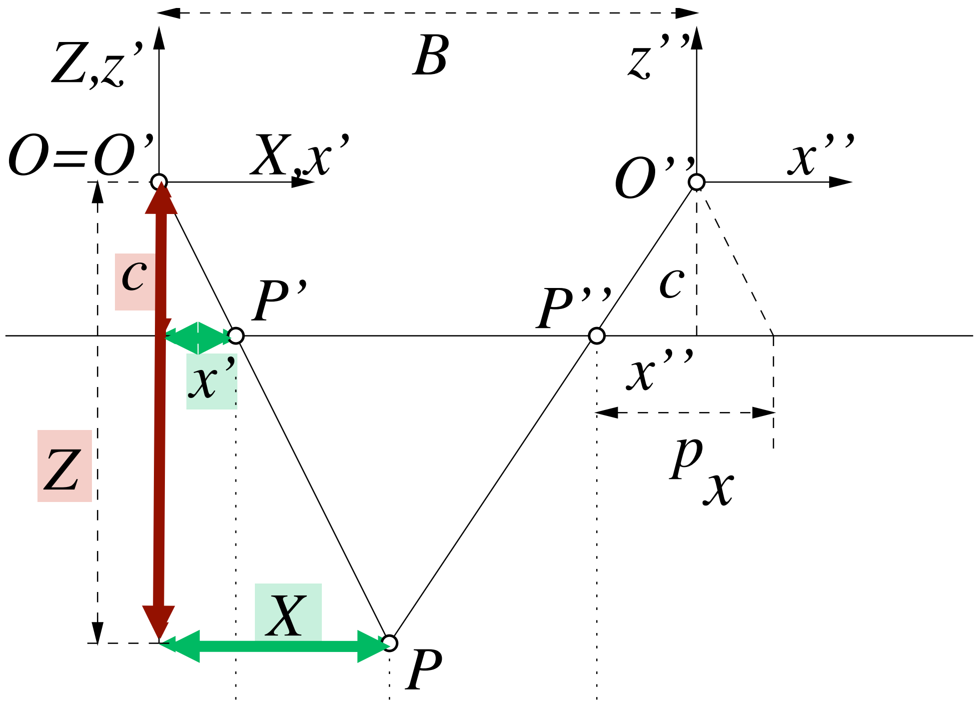


Image courtesy: Förstner

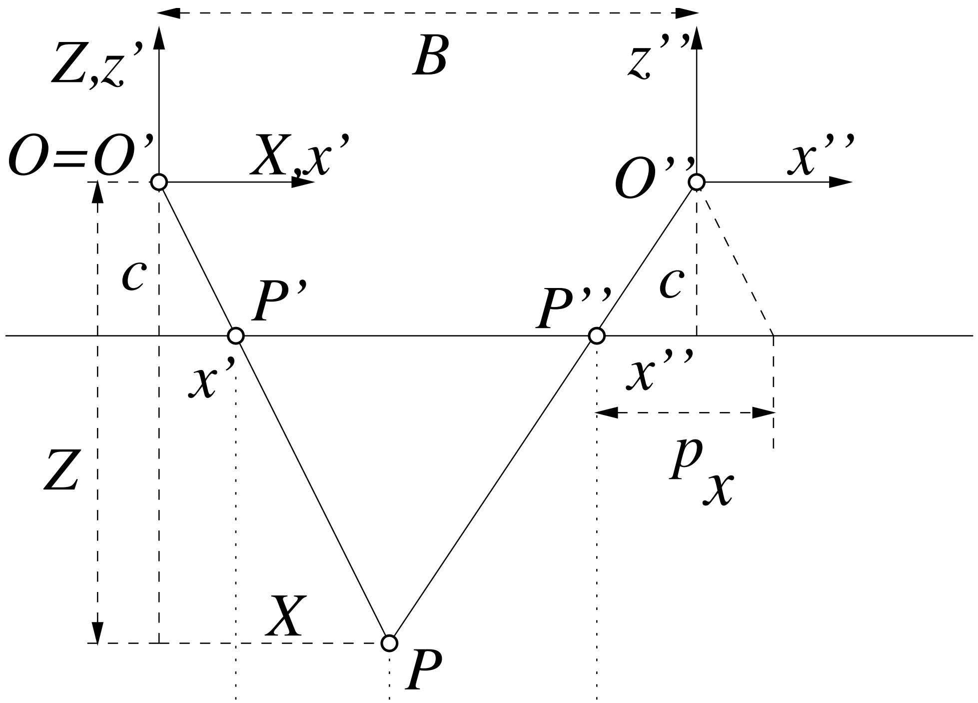
Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$Z = c\frac{B}{-(x'' - x')}$$

2. X-coordinate

$$\frac{X}{x'} = \frac{Z}{c}$$



$$\Rightarrow \quad X = x'\frac{B}{-(x'' - x')}$$

Image courtesy: Förstner

Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$Z = c\frac{B}{-(x'' - x')}$$

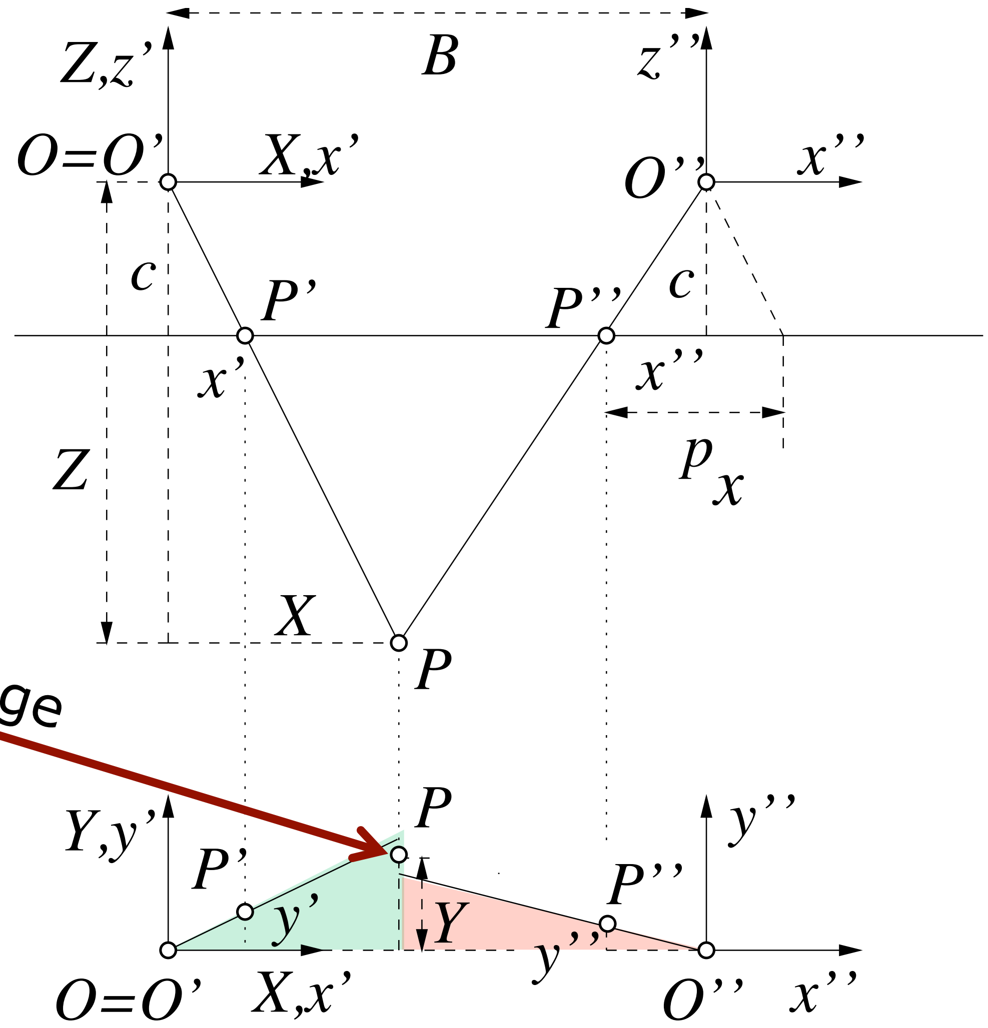
2. X-coordinate

$$X = x'\frac{B}{-(x'' - x')}$$

3. Y-coordinate by mean

$$\frac{Y}{X} = \frac{\boxed{\frac{y'+y''}{2}}}{x'}$$

average



Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

$$Z = c\frac{B}{-(x'' - x')}$$

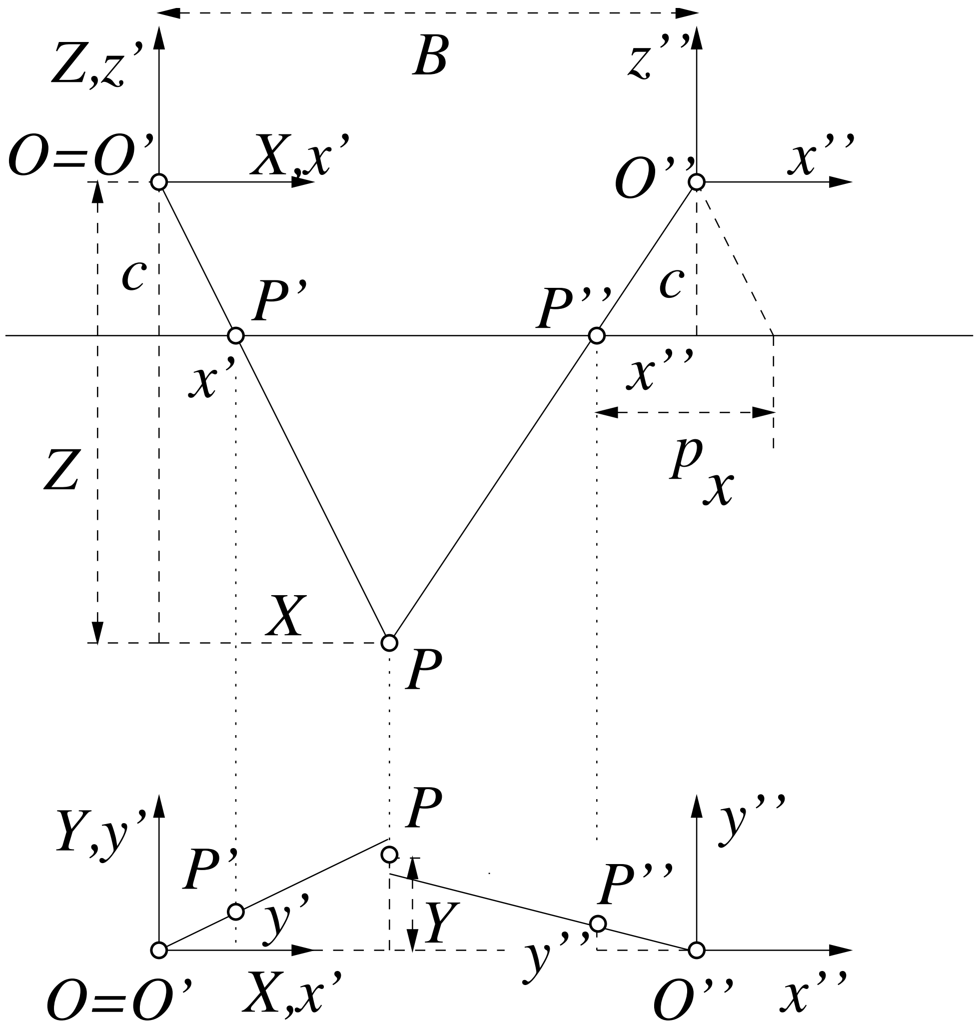
2. X-coordinate

$$X = x'\frac{B}{-(x'' - x')}$$

3. Y-coordinate by mean

$$\frac{Y}{X} = \frac{\frac{y'+y''}{2}}{x'}$$

$$Y = \frac{y' + y''}{2}\frac{B}{-(x'' - x')}$$



Stereo Normal: Intersection

1. Z-coordinate from intercept theorem

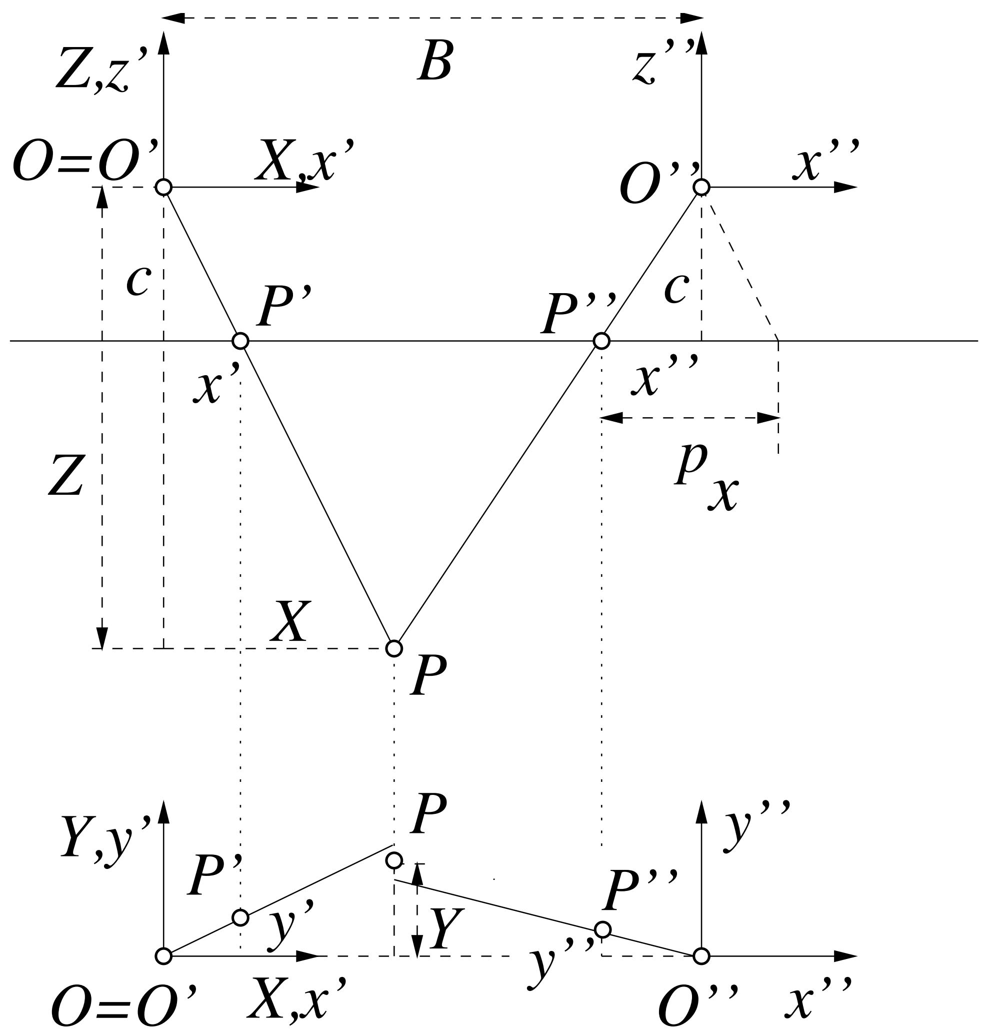
$$Z = c\frac{B}{-(x'' - x')}$$

2. X-coordinate

$$X = x'\frac{B}{-(x'' - x')}$$

3. Y-coordinate by mean

$$Y = \frac{y' + y''}{2}\frac{B}{-(x'' - x')}$$



Intersection of Two Rays for the Stereo Normal Case

$$X = x' \frac{B}{-(x'' - x')} \qquad Y = \frac{y' + y''}{2} \frac{B}{-(x'' - x')} \qquad Z = c \frac{B}{-(x'' - x')}$$

- x-parallax $p_x = x'' - x'$ corresponds to depth Z
- y-parallax $p_y = y'' - y'$ corresponds to the consistency of image points and should be small (due to stereo normal case)
- The parallax is also called disparity

X-Parallax (X-Disparity)

- The x-parallax is a key element

$$
\begin{aligned}
X &= x' \frac{B}{-(x'' - x')} \\
Y &= \frac{y' + y''}{2} \frac{B}{-(x'' - x')} \\
Z &= c \frac{B}{-(x'' - x')}
\end{aligned}
$$

X-Parallax and Scale Number

- The x-parallax is a key element

$$
\begin{aligned}
X &= x' \boxed{\frac{B}{-(x''-x')}} \\
Y &= \frac{y'+y''}{2} \boxed{\frac{B}{-(x''-x')}} \\
Z &= c \boxed{\frac{B}{-(x''-x')}}
\end{aligned}
$$

image scale number

X-Parallax and Scale Number

- The x-parallax is a key element

$$
\begin{aligned}
X &= x' \boxed{\frac{B}{-(x''-x')}} \\
Y &= \frac{y'+y''}{2} \boxed{\frac{B}{-(x''-x')}} \\
Z &= c \boxed{\frac{B}{-(x''-x')}}
\end{aligned}
$$

image scale number

- Image scale number: $M = \frac{-B}{x''-x'} = \frac{Z}{c}$

$$\boxed{X = Mx' \qquad Y = M\frac{y'+y''}{2} \qquad Z = Mc}$$

Intersection of Two Rays for the Stereo Normal Case

- If the y-parallax is zero, we obtain

$$X = x' \frac{B}{-p_x} \qquad Y = y' \frac{B}{-p_x} \qquad Z = c \frac{B}{-p_x}$$

Intersection of Two Rays for the Stereo Normal Case

- If the y-parallax is zero, we obtain

$$X = x' \frac{B}{-p_x} \qquad Y = y' \frac{B}{-p_x} \qquad Z = c \frac{B}{-p_x}$$

- We can write this as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} -\frac{B}{p_x} & 0 & 0 \\ 0 & -\frac{B}{p_x} & 0 \\ 0 & 0 & -\frac{B}{p_x} \end{bmatrix} \begin{bmatrix} x' \\ y' \\ c \end{bmatrix}$$

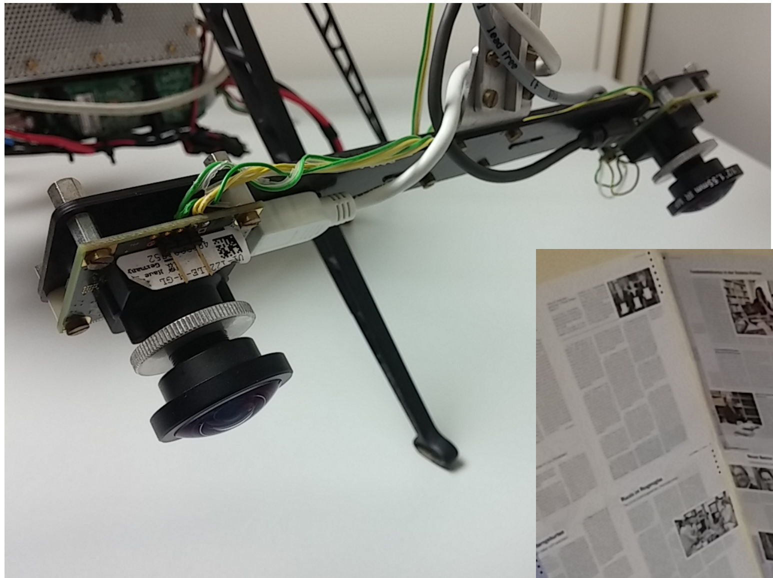
Parallax Map

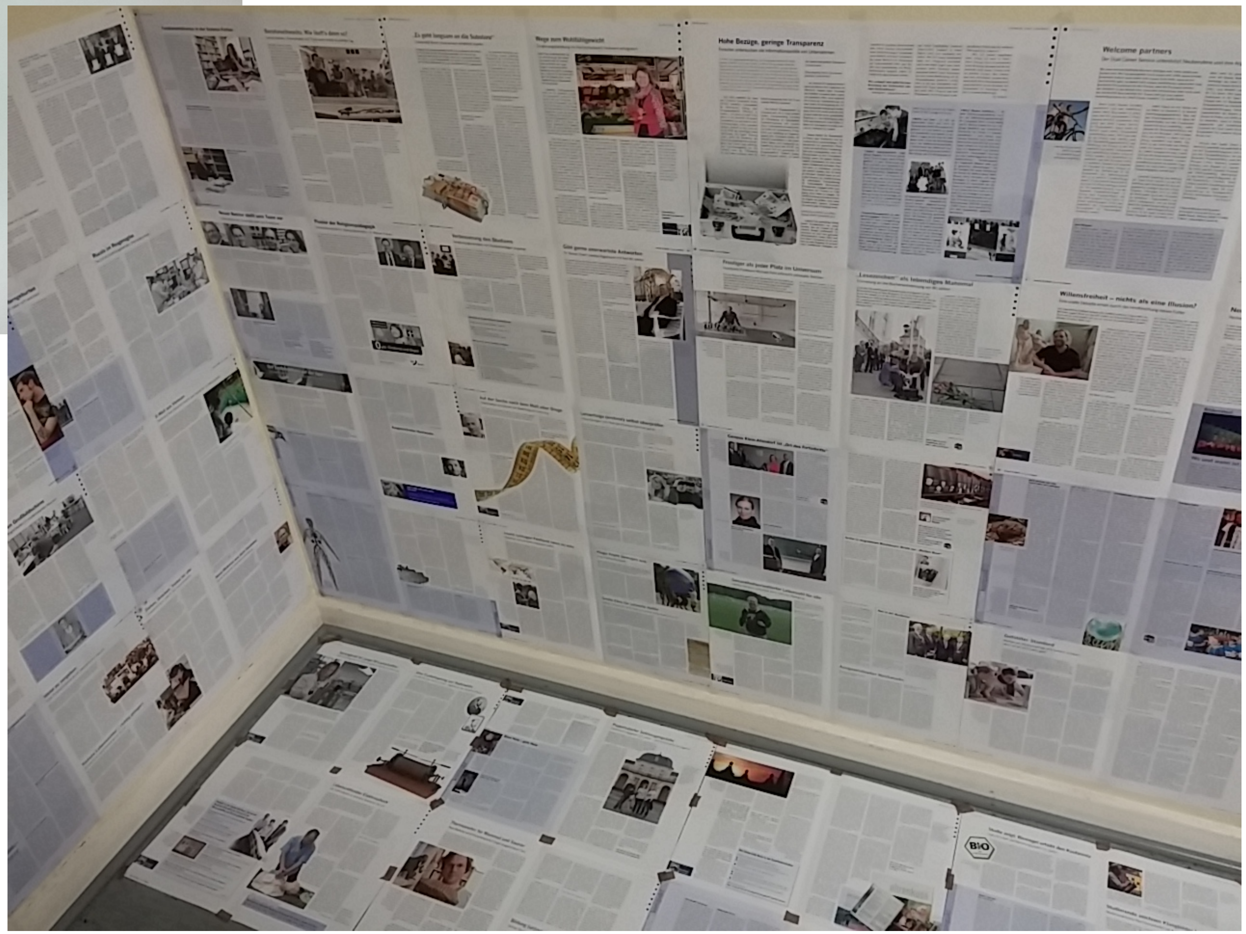
- Using H.C. and the parallax as input

$$\begin{bmatrix} U \\ V \\ W \\ T \end{bmatrix} = \begin{bmatrix} B & 0 & 0 & 0 \\ 0 & B & 0 & 0 \\ 0 & 0 & Bc & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \\ p_x \end{bmatrix}$$

- For a set of points $\{x', y'\}$ in the first image, $\{x', y', p_x\}$ is called **parallax map**
- **The parallax map directly leads to the 3D coordinates of the point**

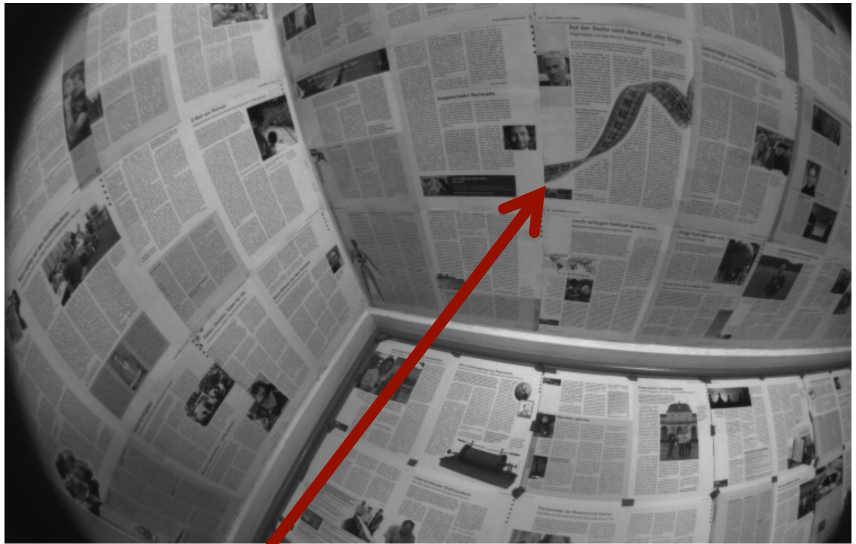
Example – Setup





Example – Image Pair

Example – Parallax Map

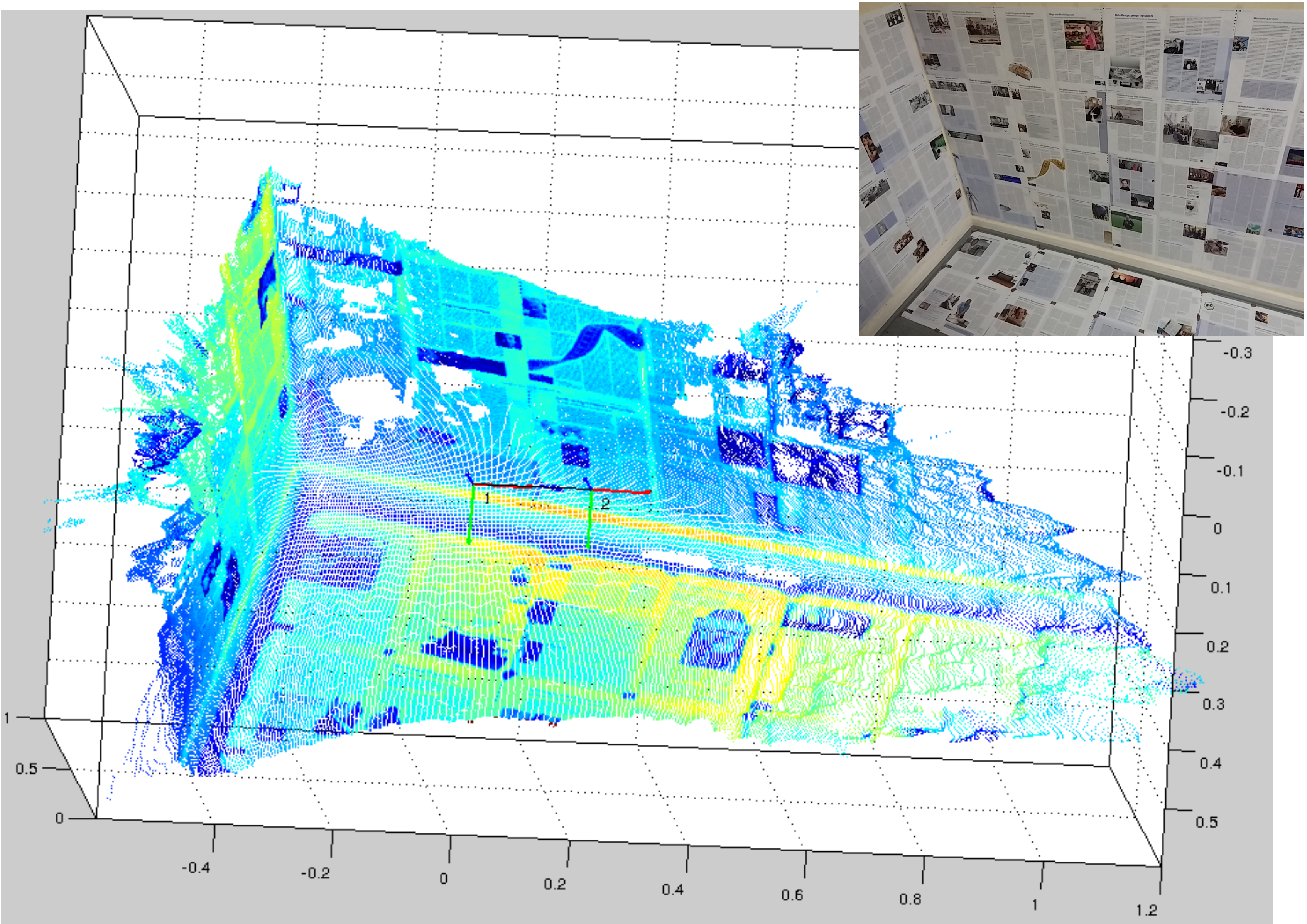
 

parallax map $\{x', y', p_x\}_1 \ldots \{x', y', p_x\}_N$

Example – Parallax Map

parallax map $\{x', y', p_x\}_1 \ldots \{x', y', p_x\}_N$

Example – 3D Point Cloud



3.
Quality of the 3D Points

Quality of the 3D Points

What influences the quality of the 3D points obtained in the normal case?

Quality of the 3D Points

What influences the quality of the 3D points obtained in the normal case?

A. Quality of the orientation parameters

B. Quality of the measured image coordinates

Quality of the 3D Points

What influences the quality of the 3D points obtained in the normal case?

A. Quality of the orientation parameters

B. Quality of the measured image coordinates

Quality of the 3D Points

- Assuming that we measure the image coordinates in x/y with $\sigma_{x'} = \sigma_{y'}$
- Starting from

$$X = Mx' \qquad Y = M\frac{y' + y''}{2}$$

- Directly yields the uncertainty in x/y

$$\boxed{\sigma_X = M\sigma_{x'} = \frac{Z}{c}\sigma_{x'}} \qquad \boxed{\sigma_Y = \frac{\sqrt{2}}{2}M\sigma_{y'} = \frac{\sqrt{2}}{2}\frac{Z}{c}\sigma_{y'}}$$

Quality of the 3D Points

- For the Z coordinate, we obtain for the relative precision

$$\boxed{\frac{\sigma_Z}{Z} = \frac{\sigma_{p_x}}{p_x}}$$

The relative precision of the height is the relative precision of the x-parallax

Height/Depth Precision

- Starting from $\frac{\sigma_Z}{Z} = \frac{\sigma_{p_x}}{p_x}$ we obtain:

$$\sigma_Z = \frac{Z}{p_x}\sigma_{p_x} = \frac{cB}{p_x^2}\sigma_{p_x} = \frac{Z^2}{cB}\sigma_{p_x} = \frac{Z}{c\;B/Z}\sigma_{p_x}$$

$$Z = \frac{cB}{p_x} \qquad \frac{1}{p_x} = \frac{Z}{cB} \qquad Z = \frac{1}{1/Z}$$

Height/Depth Precision

- Starting from $\frac{\sigma_Z}{Z} = \frac{\sigma_{p_x}}{p_x}$ we obtain:

$$\sigma_Z = \frac{Z}{p_x}\sigma_{p_x} = \frac{cB}{p_x^2}\sigma_{p_x} = \frac{Z^2}{cB}\sigma_{p_x} = \frac{Z}{c\ B/Z}\sigma_{p_x}$$

Standard deviation of Z depends

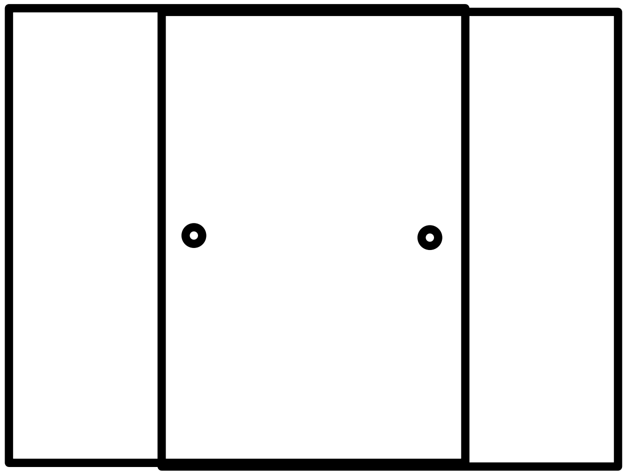
- on the **x-parallax standard deviation**
- inverse quadratically on the **x-parallax**
- quadratically on the **depth**
- inversely on the **base/depth ratio**

Example: Aerial Image Analysis

- Typical values

1. $\sigma_{x'} = 7\ \mu\text{m}, \rightarrow \sigma_{p_x} \approx 10\ \mu\text{m}$
2. $p_x \approx b = 0.4 \times 23\ \text{cm} = 92\ \text{mm}$

$\Rightarrow$ $\sigma_Z = ?$



60% overlap results in an avg. parallax of ~0.4x23cm

Example: Aerial Image Analysis

- Typical values

1. $\sigma_{x'} = 7\ \mu\text{m}, \rightarrow \sigma_{p_x} \approx 10\ \mu\text{m}$

2. $p_x \approx b = 0.4 \times 23\ \text{cm} = 92\ \text{mm}$

$$\Rightarrow \quad \sigma_Z = Z\frac{\sigma_{p_x}}{p_x} = Z\frac{10\ \mu\text{m}}{92\ \text{mm}} \approx \frac{1}{10.000}\,Z$$

Example: Aerial Image Analysis

- Typical values

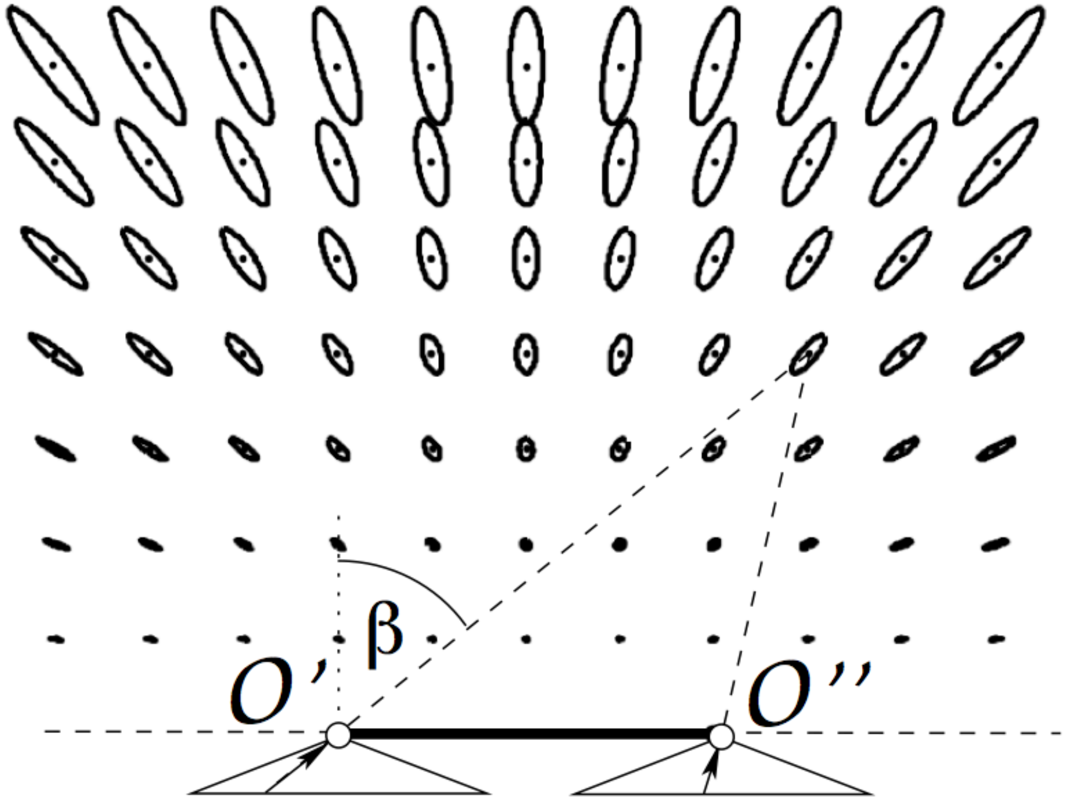
1. $\sigma_{x'} = 7\ \mu\text{m}, \rightarrow \sigma_{p_x} \approx 10\ \mu\text{m}$

2. $p_x \approx b = 0.4 \times 23\ \text{cm} = 92\ \text{mm}$

$$\Rightarrow \quad \sigma_Z = Z \frac{\sigma_{p_x}}{p_x} = Z \frac{10\ \mu\text{m}}{92\ \text{mm}} \approx \frac{1}{10.000} Z$$

The precision of the elevation from an aerial stereo image is approx. 1:10000 of the flight altitude

Stereo Uncertainty Field



x', x''

$\sigma^2_{x'} = \sigma^2_{x''}$

Image courtesy: Förstner and Wrobel

Summary - Triangulation

- We can estimate 3D point locations (in the camera frame) given corresponding points and orientation parameters through triangulation
- Geometric approach
- Triangulation for the stereo normal case
- Quality of the 3D Points for the stereo normal case

Part II
Absolute Orientation

“Where are the points in the world?”

Relative Orientation

- The result of the R.O. is the so-called **photogrammetric model**
- It contains the
  - parameters of the relative orientation of both cameras
  - 3D coordinates of N points in a local coordinate frame

$$ {}^{m}\boldsymbol{X}_n = ({}^{m}X_n, {}^{m}Y_n, {}^{m}Z_n)^{\mathsf{T}} \qquad n = 1, \cdots, N $$

- Known up to a similarity transform (for calibrated cameras)

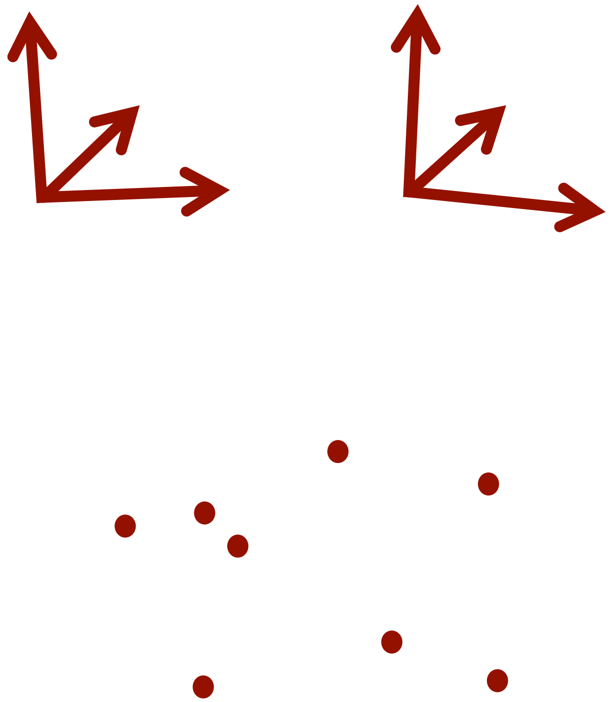
Absolute Orientation

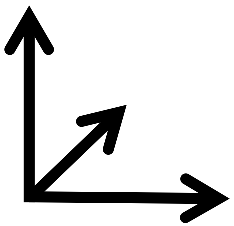
- A similarity transform maps the photogrammetric model into the object reference frame

$$ {}^{o}\boldsymbol{X}_n = \underline{\lambda R}\, {}^{m}\boldsymbol{X}_n + \underline{\boldsymbol{T}} $$

- 7 DoF for the similarity transform (3 rotation, 3 translation, 1 scale)
- **Control points** are required

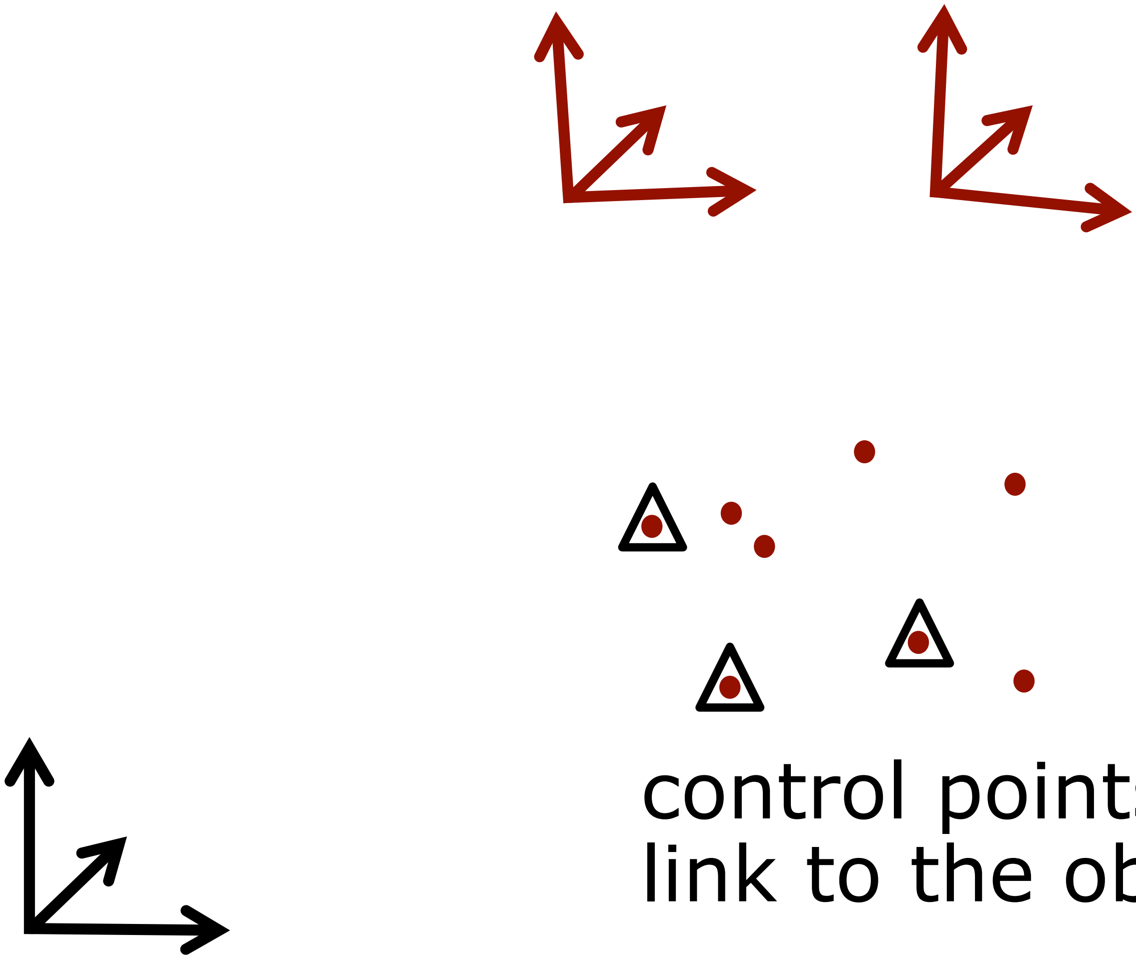
Photogrammetric Model





missing

Object Reference Frame



control points create the
link to the object frame

Least Squares Solution

- Non-linear least squares solution
- At least 3 control points (X,Y,Z known)
- Similar to the ICP algorithm

Sketch of the Solution

- Points in object and local system

$$\boldsymbol{y}_n = \lambda R \boldsymbol{x}_n - \boldsymbol{T} \qquad n = 1, \cdots, N$$

- Can be written as

$$\underbrace{\lambda^{-\frac{1}{2}}(\boldsymbol{y}_n - \boldsymbol{y}_0)}_{\boldsymbol{b}_n} = R\,\underbrace{\lambda^{\frac{1}{2}}(\boldsymbol{x}_n - \boldsymbol{x}_0)}_{\boldsymbol{a}_n}$$

- Function to minimize (w/ weights p_n)

$$\Phi(\boldsymbol{x}_0, \lambda, R) = \sum [\boldsymbol{b}_n - R\boldsymbol{a}_n]^{\mathsf{T}} [\boldsymbol{b}_n - R\boldsymbol{a}_n] p_n$$

Minimize $\Phi(\boldsymbol{x}_0, \lambda, R)$

- Computing the first derivatives yields

$$\frac{\partial \Phi}{\partial \boldsymbol{x}_0} = 0 \quad \rightarrow \quad \boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n}$$

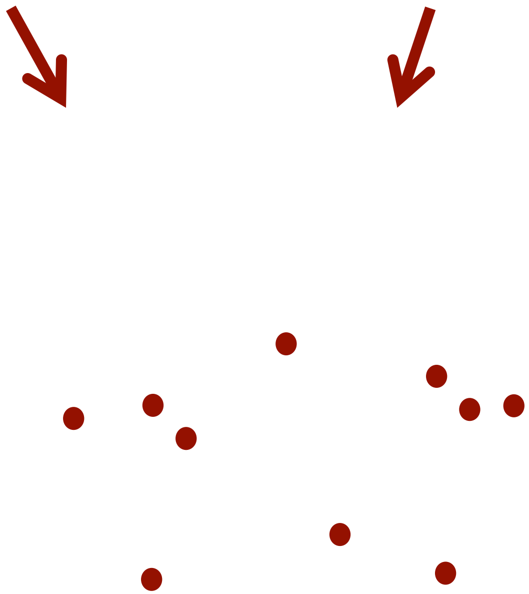
$$\frac{\partial \Phi}{\partial \lambda} = 0 \quad \rightarrow \quad \lambda^2 = \frac{\sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\mathsf{T} (\boldsymbol{y}_n - \boldsymbol{y}_0) p_n}{\sum (\boldsymbol{x}_n - \boldsymbol{x}_0)^\mathsf{T} (\boldsymbol{x}_n - \boldsymbol{x}_0) p_n}$$

- 3D rotation via SVD

$$H = \sum_{i=1}^{k} (\boldsymbol{a}_n \boldsymbol{b}_n^T) p_n \ , \ \ \mathrm{svd}(H) = UDV^T \quad \rightarrow \quad R = VU^T$$

- Details: Lecture “Absolute Orientation”

Overview – Initial Stage

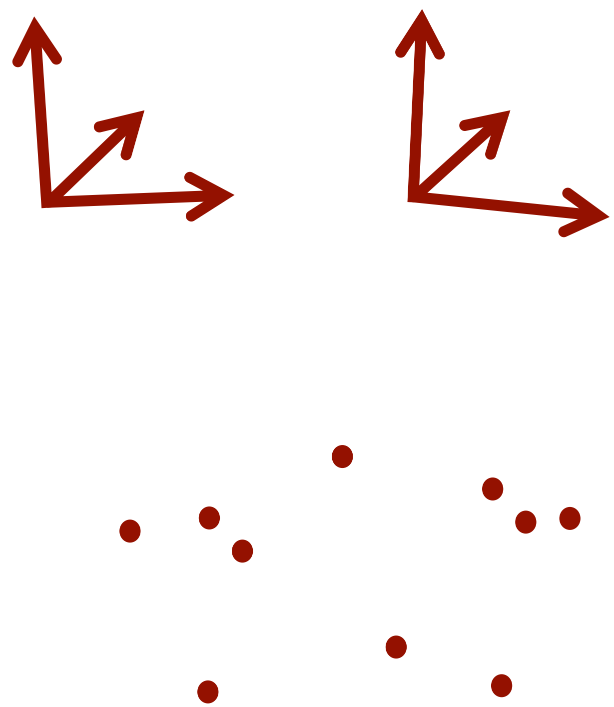


two cameras

+

corresponding points in both images

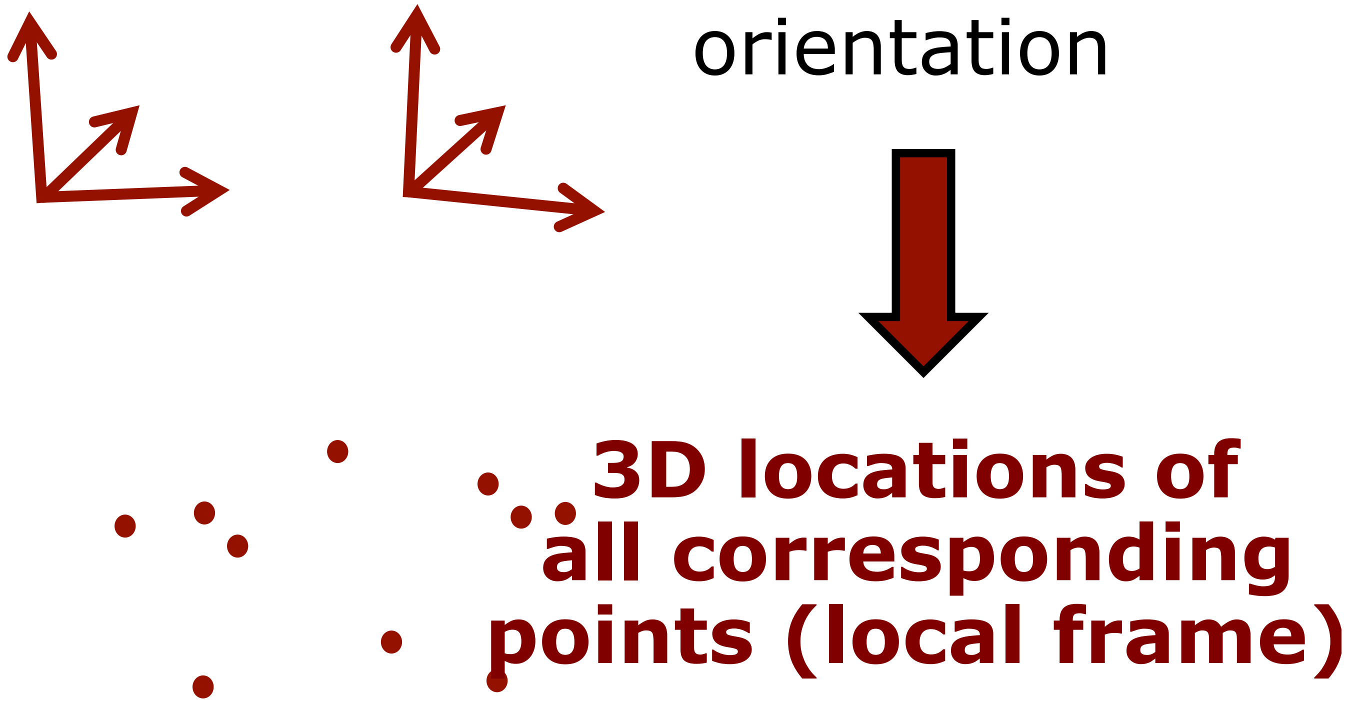
Overview – 1st Step



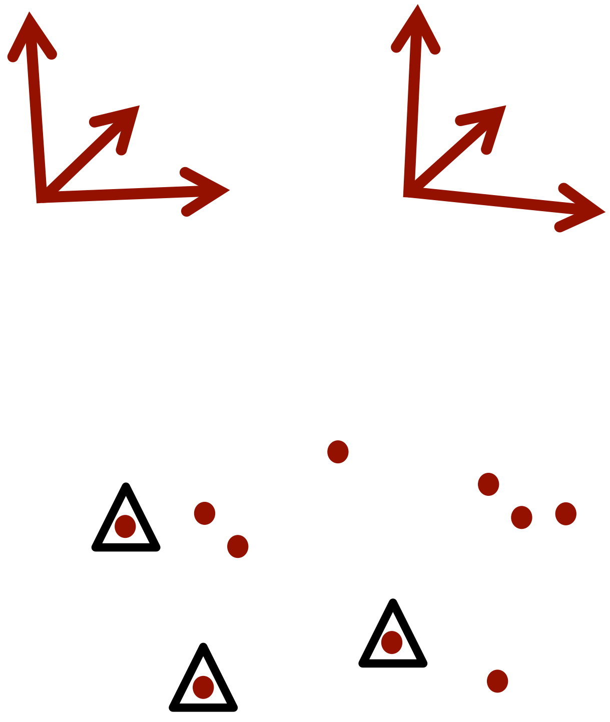
relative orientation

corresponding points in both images

Overview – 1st Step



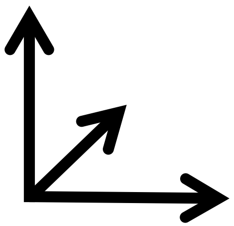
Overview – 2nd Step



relative orientation

+

control points



3D location of all corresponding points in the object frame

2-Step Solution

- **Relative orientation** without control points and 3D location of correspond. points in a local frame
- **Absolute orientation** of cameras and corresponding points through control points

2-Step Solution

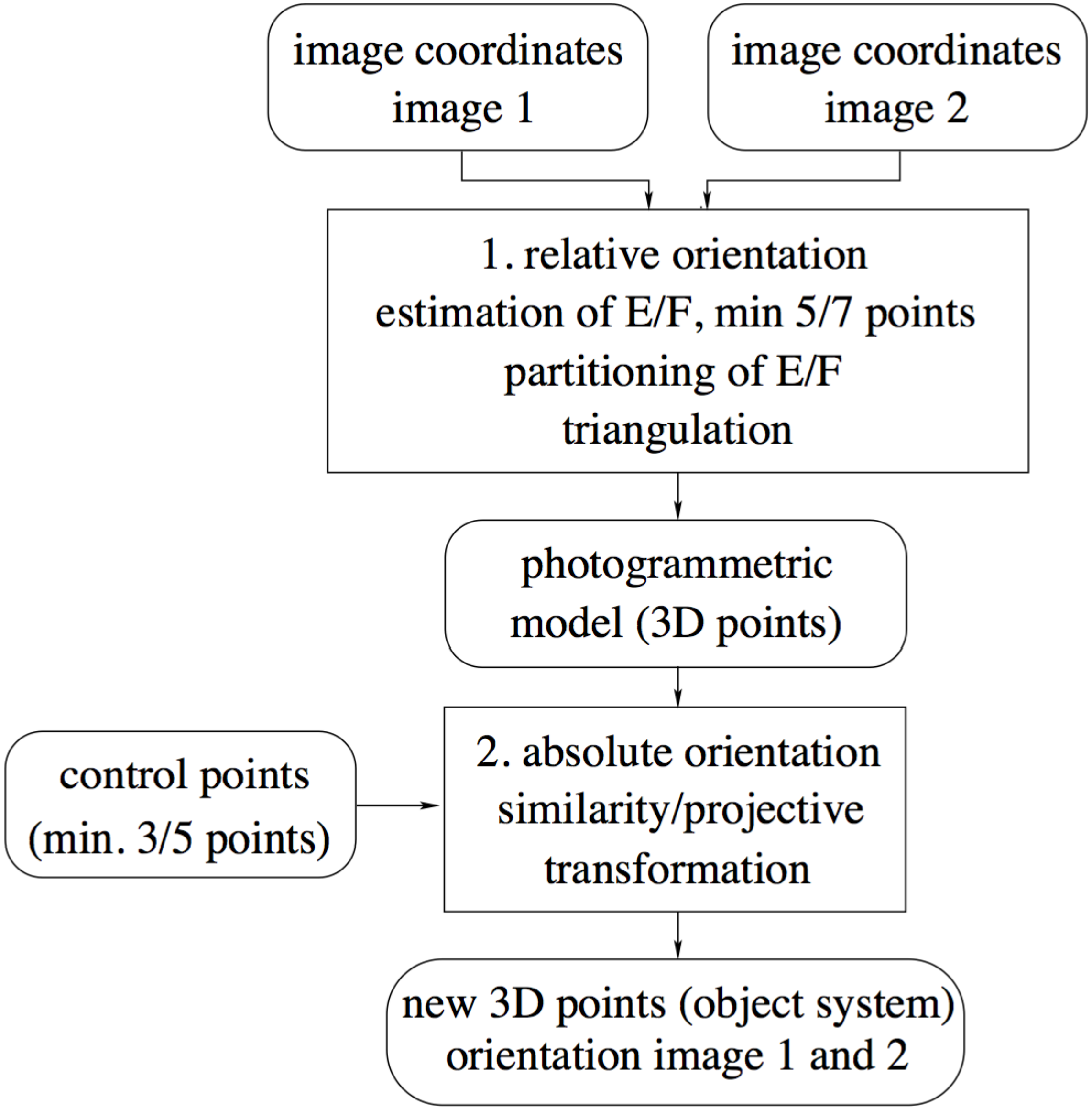


Image courtesy: Förstner and Wrobel

Discussion:

Which Other Orientation Approaches Do We Know?

Approaches to Compute Different Forms of Orientations

- Direct linear transform (DLT)
- Spatial Resectioning (P3P, RRS)
- Relative orientation
- Triangulation
- Absolute orientation

Approaches to Compute Different Forms of Orientations

- Direct linear transform (DLT)
- Spatial Resectioning (P3P, RRS)
- Relative orientation
- Triangulation
- Absolute orientation

How could we achieve the same using combinations of the techniques listed above?

Other Possibilities

Option 1

- DLT for each camera using control pts
- Triangulation for all corresponding pts

Option 2

- P3P for each camera using control pts
- Triangulation for all corresponding pts

Option 3

- One big least squares approach (bundle adjustment)

Option 1 & 2 – DLT / P3P

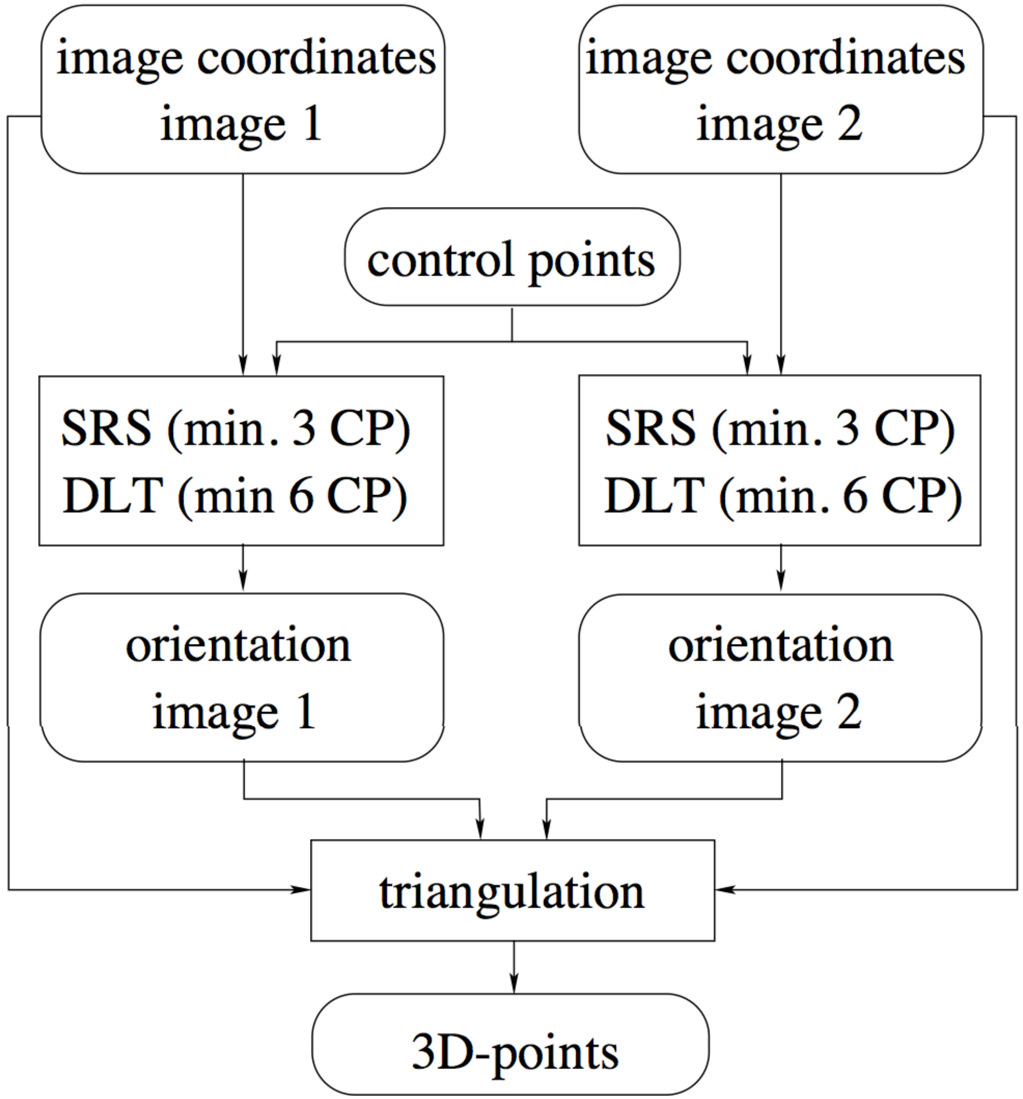


Image courtesy: Förstner and Wrobel

Option 3 – Bundle Adjustment

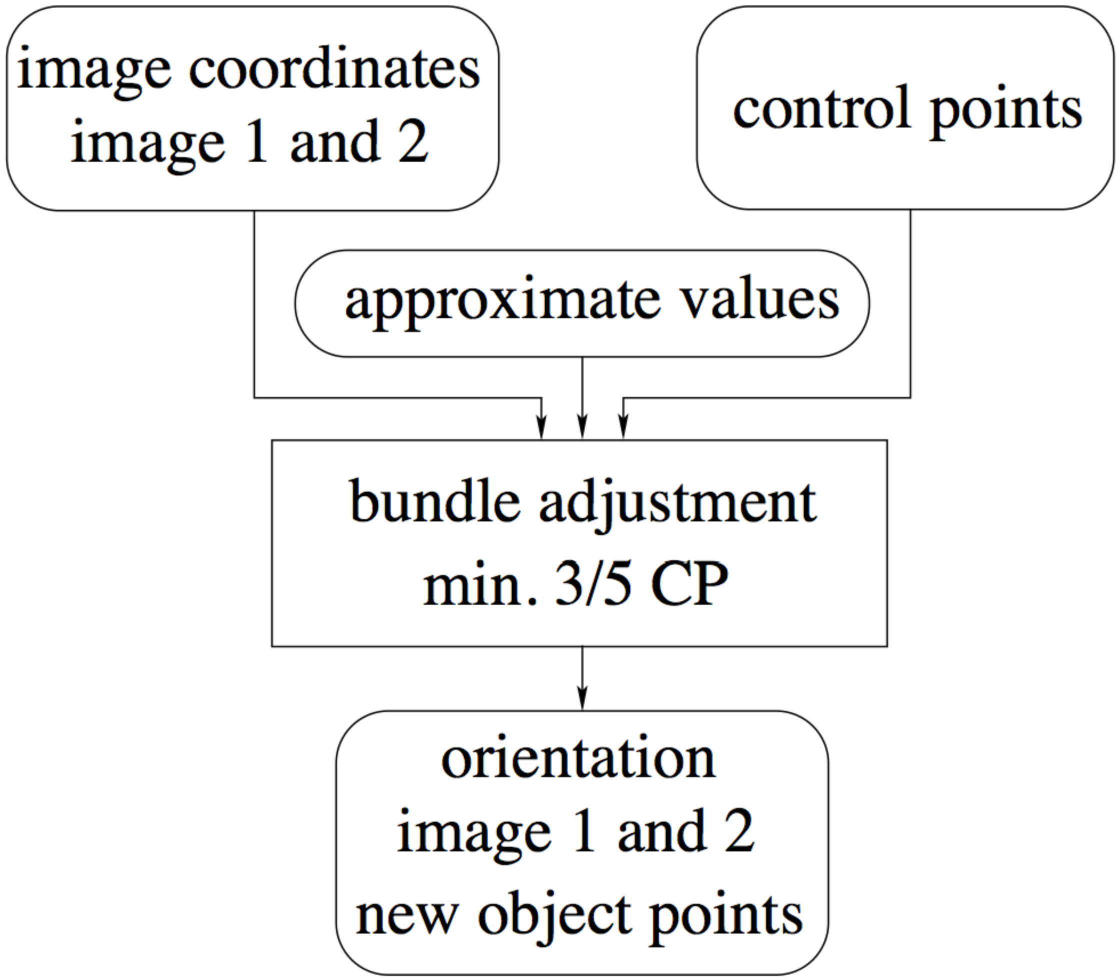


Image courtesy: Förstner and Wrobel

Which Solution is the Best One?

Discussion – Bundle Adjustment

- Two-image bundle adjustment (BA)
- BA leads to the statistically optimal solution (given convergence)
- BA can deal with a moderate amount of gross errors using robust estimators
- BA requires an approximate solution as an initial guess

Discussion – 2-Step Solution

- Two step solution allows for checking the photogrammetric and geodetic measurements separately (corresponding points vs. control points)
- Can handle gross errors for Redundancy/Num-observations > 0.5
- Serves as an initial guess for BA

Discussion – 2 x P3P

- Only applicable if both images observe at least 3 (4) full control pts
- Direct approach can be used to find gross errors
- Less accurate than the 2-step solution in case of large sets of new points

Discussion – 2 x DLT

- Only applicable of both images observe at least 6 full control pts
- Points cannot lie on one plane
- Initial guess for BA in case the calibration parameters are unknown

Summary

- Absolute Orientation transforms the photog. model to the object frame
- Different ways for orienting points absolutely (DLT, P3P, 2-Step, BA)
- Bundle adjustment is the optimal solution in a statistical sense but requires a (good) initial guess

Literature

- Förstner, Wrobel: Photogrammetric Computer Vision, Ch. 12.4 - 12.6

Slide Information

- The slides have been created by Cyrill Stachniss as part of the photogrammetry and robotics courses.
- I tried to acknowledge all people from whom I used images or videos. In case I made a mistake or missed someone, please let me know.
- The photogrammetry material heavily relies on the very well written lecture notes by Wolfgang Förstner and the Photogrammetric Computer Vision book by Förstner & Wrobel.
- Parts of the robotics material stems from the great Probabilistic Robotics book by Thrun, Burgard and Fox.
- If you are a university lecturer, feel free to use the course material. If you adapt the course material, please make sure that you keep the acknowledgements to others and please acknowledge me as well. To satisfy my own curiosity, please send me email notice if you use my slides.

Cyrill Stachniss, cyrill.stachniss@igg.uni-bonn.de, 2014