Summer term 2024 – Cyrill Stachniss

5 Minute Preparation for Today

https://www.youtube.com/watch?v=26nV4oDLiqc

Photogrammetry & Robotics Lab

Camera Calibration: Zhang's Method

Cyrill Stachniss

The slides have been created by Cyrill Stachniss.

3D Point to Pixel: Estimating the Parameters of P

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

pixel coordinate

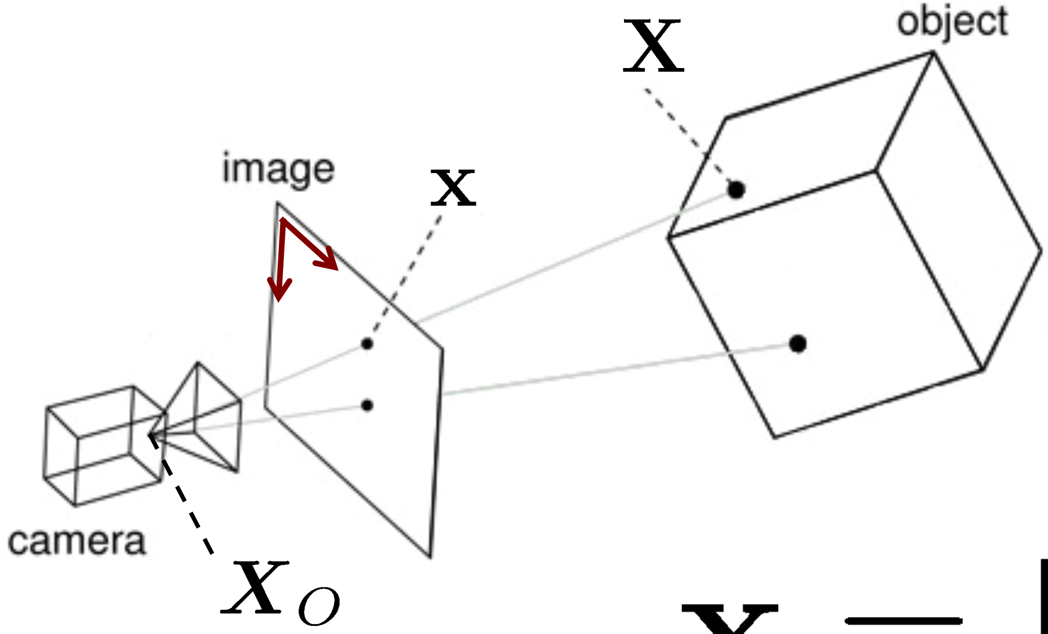
trans-formation

world coordinate

This time we only want the intrinsics!

Mapping (Recap)

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



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

Direct Linear Transform (Recap)

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)

Zhang's Method

Compute the **5 intrinsic parameters**

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

observed image point

c, s, m, x_H, y_H

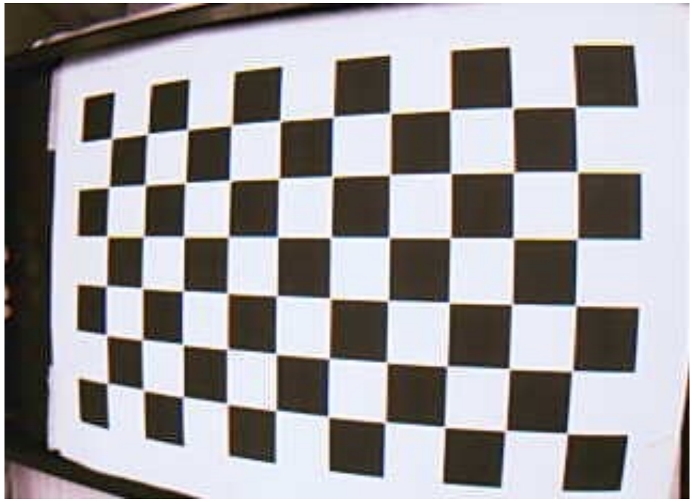
3 rotations

3 translations

control point

Assumption:
You know how DLT works!

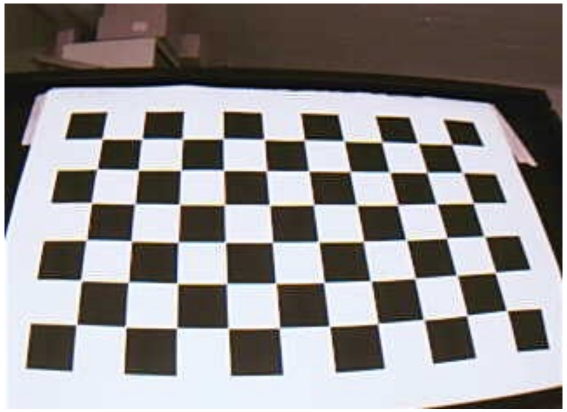
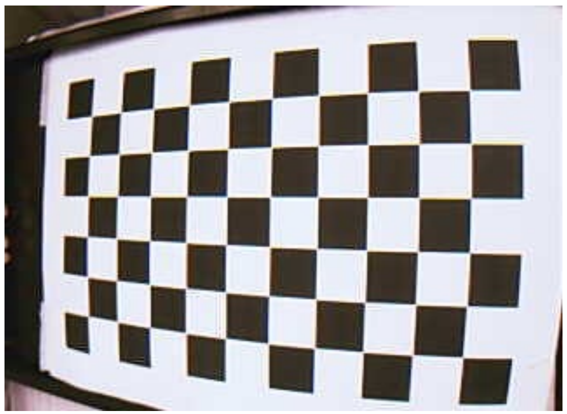
Zhang's Method for Camera Calibration Using a Checkerboard



Zhang, A Flexible New Technique for Camera Calibration, MSR-TR-98-71

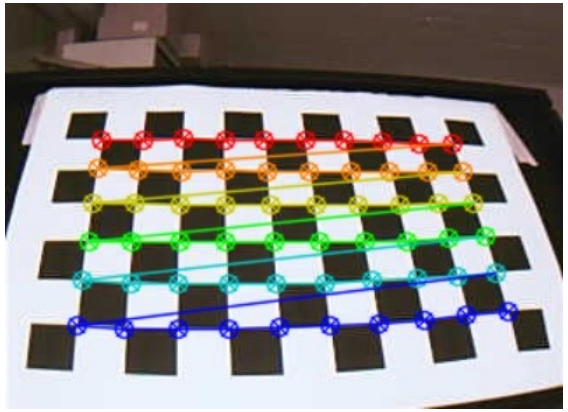
Camera Calibration Using a 2D Checkerboard

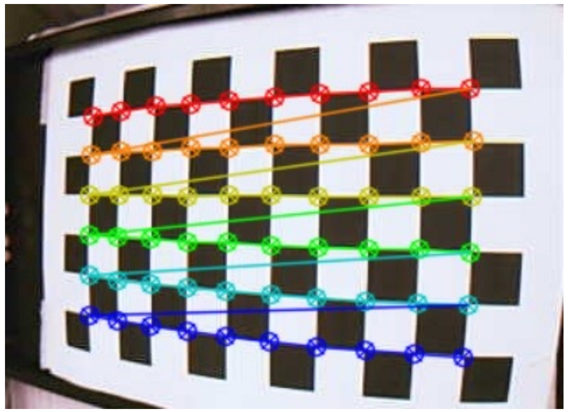
- Observed 2D pattern (checkerboard)
- **Known size and structure**

Trick for Checkerboard Calibration

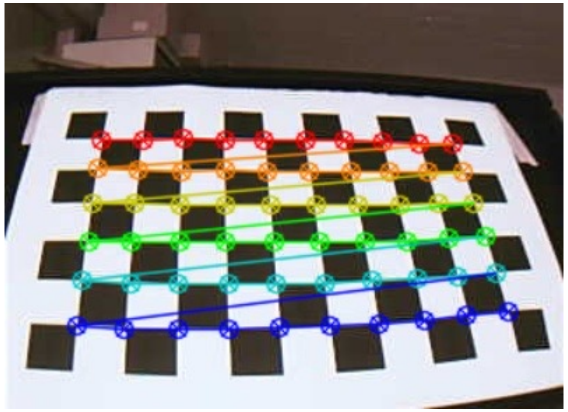
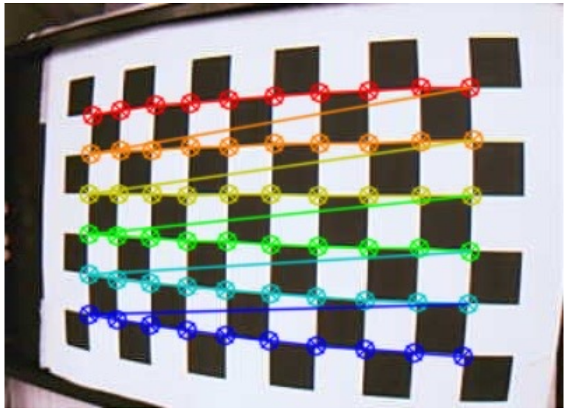
- **Set the world coordinate system to the corner of the checkerboard**





Trick for Checkerboard Calibration

- **Set the world coordinate system to the corner of the checkerboard**
- All points on the checkerboard lie in the X/Y plane, i.e., Z=0

DLT Mapping

- We start from the affine camera model $\mathbf{x} = \mathsf{K}R[I_3 | -\boldsymbol{X}_O]\mathbf{X}$
- Expanded

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Coordinate System Definition

- By definition of the coordinate system, the Z coordinate of each point on the checkerboard is equal to zero

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ \boxed{Z} \\ 1 \end{bmatrix}$$

Coordinate System Definition

- By definition of the coordinate system, the Z coordinate of each point on the checkerboard is equal to zero

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & \boxed{r_{13}} & t_1 \\ r_{21} & r_{22} & \boxed{r_{23}} & t_2 \\ r_{31} & r_{32} & \boxed{r_{33}} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ \boxed{Z} \\ 1 \end{bmatrix}$$

Simplification

- By definition of the coordinate system, the Z coordinate of each point on the checkerboard is equal to zero

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & \cancel{r_{13}} & t_1 \\ r_{21} & r_{22} & \cancel{r_{23}} & t_2 \\ r_{31} & r_{32} & \cancel{r_{33}} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ \cancel{Z} \\ 1 \end{bmatrix}$$

- We can delete the 3rd column of the extrinsic parameter matrix

Simplification

- Z coordinate of each point on the checkerboard is equal to zero
- Deleting the 3rd column of the extrinsic parameter matrix leads to

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

One point observed on the checkerboard generates such an equation

Setting Up the Equations for Determining the Parameter

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

Setting Up the Equations for Determining the Parameter

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

One point generates the equation:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathsf{K}[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

Setting Up the Equations for Determining the Parameter

- For multiple observed points on the checkerboard (in the same image), we obtain

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \underset{3\times3}{\mathsf{H}} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix} \qquad i = 1, \ldots, I$$

How to proceed?

Setting Up the Equations for Determining the Parameter

- For multiple observed points on the checkerboard (in the same image), we obtain

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \underset{3\times 3}{\mathsf{H}} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix} \qquad i = 1, \ldots, I$$

- Analogous to first **steps of the DLT**

DLT-Like Estimation

- We estimate a 3x3 homography instead of a 3x4 projection matrix
- Rest is identical
- We use $\boldsymbol{a}_{x_i}^{\mathsf{T}} \boldsymbol{h} = 0$

  $\boldsymbol{a}_{y_i}^{\mathsf{T}} \boldsymbol{h} = 0$
- with

$$
\begin{aligned}
\boldsymbol{h} &= (h_k) = \mathrm{vec}(\mathsf{H}^{\mathsf{T}}) \\
\boldsymbol{a}_{x_i}^{\mathsf{T}} &= (-X_i,\ -Y_i,\ -\cancel{Z_i},\ -1, 0,\ 0,\ \cancel{0},\ 0, x_iX_i,\ x_iY_i,\ \cancel{x_iZ_i},\ x_i) \\
\boldsymbol{a}_{y_i}^{\mathsf{T}} &= (0,\ 0,\ \cancel{0},\ 0, -X_i,\ -Y_i,\ -\cancel{Z_i},\ -1, y_iX_i,\ y_iY_i,\ \cancel{y_iZ_i},\ y_i)
\end{aligned}
$$

DLT-Like Estimation

- We estimate a 3x3 homography instead of a 3x4 projection matrix
- Rest is identical
- We use $$\boldsymbol{a}_{x_i}^{\mathsf{T}} \boldsymbol{h} = 0$$ $$\boldsymbol{a}_{y_i}^{\mathsf{T}} \boldsymbol{h} = 0$$
- with

$$\begin{aligned}
\boldsymbol{h} &= (h_k) = \mathrm{vec}(\mathsf{H}^{\mathsf{T}}) \\
\boldsymbol{a}_{x_i}^{\mathsf{T}} &= (-X_i,\ -Y_i,\ -1, 0,\ 0,\ 0, x_i X_i,\ x_i Y_i,\ x_i) \\
\boldsymbol{a}_{y_i}^{\mathsf{T}} &= (0,\ 0,\ 0, -X_i,\ -Y_i,\ -1, y_i X_i,\ y_i Y_i,\ y_i)
\end{aligned}$$

DLT-Like Estimation

- Solving the system of linear equations leads to an estimate of H
- **How many points are needed to estimate** H**?**

DLT-Like Estimation

- Solving the system of linear equations leads to an estimate of H
- We need to identify at least 4 points as H has 8 DoF and each point consists of 2 observations (x and y)

This provides an estimate of H

DLT-Like Estimation

- Solving the system of linear equations leads to an estimate of H
- We need to identify at least 4 points as H has 8 DoF and each point consists of 2 observations (x and y)

After we have estimated H**,**
we need to compute K **from** H

Computing K Given H

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

no rotation matrix, thus QR decomposition is not applicable as for the DLT

How to obtain the matrix decomposition?

Computing K Given H

- We need to extract K from the matrix $\mathsf{H} = \mathsf{K}[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]$ we computed via SVD

Computing K Given H

- We need to extract K from the matrix $\mathsf{H} = \mathsf{K}[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]$ we computed via SVD

Four step procedure:

1. Exploit constrains about $\mathsf{K}, \boldsymbol{r}_1, \boldsymbol{r}_2$
2. Define a matrix $\mathsf{B} = \mathsf{K}^{-\top}\mathsf{K}^{-1}$
3. This B can be computed by solving another homogeneous linear system
4. Decompose matrix B

Computing K Given H is Different from the DLT Solution

- Homography H has only 8 DoF
- No direct DLT-like decomposition
- Exploit constraints on the parameters

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

$$[\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \mathsf{K}[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]$$

Exploiting Constraints for Determining the Parameter

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

$$[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}] = \mathsf{K}^{-1}[\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3]$$

$$\Rightarrow \quad \boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

Exploiting Constraints for Determining the Parameter

$$\mathsf{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3] = \underbrace{\begin{bmatrix} c & cs & x_H \\ 0 & c(1+m) & y_H \\ 0 & 0 & 1 \end{bmatrix}}_{\mathsf{K}} \underbrace{\begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}}_{[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}]}$$

$$[\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{t}] = \mathsf{K}^{-1}[\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3]$$

$$\Rightarrow \quad \boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

As $\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3$ form an orthonormal basis

$$\Rightarrow \quad \boldsymbol{r}_1^\top \boldsymbol{r}_2 = 0 \qquad \|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\|$$

Exploiting Constraints

$$\boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

$$\xrightarrow{\boldsymbol{r}_1^\top \boldsymbol{r}_2 = 0} \qquad \boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0$$

Note: $\mathsf{K}^{-\top} = \left(\mathsf{K}^{-1}\right)^\top$

Exploiting Constraints

$$\boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

$$\xrightarrow{\boldsymbol{r}_1^\top \boldsymbol{r}_2 = 0} \quad \boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0$$

$$\xrightarrow{\|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\|} \quad \boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_1 = \boldsymbol{h}_2^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2$$

Note: $\mathsf{K}^{-\top} = \left(\mathsf{K}^{-1}\right)^\top$

Exploiting Constraints

$$\boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

$\boldsymbol{r}_1^\top \boldsymbol{r}_2 = 0$

$$\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0$$

$\|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\|$

$$\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_1 = \boldsymbol{h}_2^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2$$

$$\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_1 - \boldsymbol{h}_2^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0$$

Note: $\mathsf{K}^{-\top} = \left(\mathsf{K}^{-1}\right)^\top$

Exploiting Constraints

$$\boldsymbol{r}_1 = \mathsf{K}^{-1}\boldsymbol{h}_1 \qquad \boldsymbol{r}_2 = \mathsf{K}^{-1}\boldsymbol{h}_2$$

$$\boxed{\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0}$$

$$\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_1 = \boldsymbol{h}_2^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2$$

$$\boxed{\boldsymbol{h}_1^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_1 - \boldsymbol{h}_2^\top \mathsf{K}^{-\top}\mathsf{K}^{-1}\boldsymbol{h}_2 = 0}$$

Note: $\mathsf{K}^{-\top} = \left(\mathsf{K}^{-1}\right)^\top$

Exploiting Constraints

$$\boldsymbol{h}_1^\top \underline{\mathsf{K}^{-\top}\mathsf{K}^{-1}} \boldsymbol{h}_2 = 0$$

$$\boldsymbol{h}_1^\top \underline{\mathsf{K}^{-\top}\mathsf{K}^{-1}} \boldsymbol{h}_1 - \boldsymbol{h}_2^\top \underline{\mathsf{K}^{-\top}\mathsf{K}^{-1}} \boldsymbol{h}_2 = 0$$

- Define matrix $\mathsf{B} := \mathsf{K}^{-\top}\mathsf{K}^{-1}$

Note: $\mathsf{K}^{-\top} = \left(\mathsf{K}^{-1}\right)^\top$

Exploiting Constraints

$$\boldsymbol{h}_1^\top \underline{\mathsf{B}} \boldsymbol{h}_2 = 0$$

$$\boldsymbol{h}_1^\top \underline{\mathsf{B}} \boldsymbol{h}_1 - \boldsymbol{h}_2^\top \underline{\mathsf{B}} \boldsymbol{h}_2 = 0$$

- with $\mathsf{B} := \mathsf{K}^{-\top}\mathsf{K}^{-1}$

Exploiting Constraints

$$\boldsymbol{h}_1^\top \underline{\mathrm{B}} \boldsymbol{h}_2 = 0$$

$$\boldsymbol{h}_1^\top \underline{\mathrm{B}} \boldsymbol{h}_1 - \boldsymbol{h}_2^\top \underline{\mathrm{B}} \boldsymbol{h}_2 = 0$$

- with $\mathrm{B} := \mathrm{K}^{-\top}\mathrm{K}^{-1}$
- From B, the calibration matrix can be recovered through Cholesky decomp.

$$\mathrm{chol}(\mathrm{B}) \to \mathrm{A}$$

$$\mathrm{B} = \mathrm{A}\mathrm{A}^\top \quad \Rightarrow \quad \mathrm{A} = \mathrm{K}^{-\top}$$

⇒ If we know B, then we can compute K

Exploiting Constraints

$$\boldsymbol{h}_1^\top \underline{\mathsf{B}} \boldsymbol{h}_2 = 0$$

$$\boldsymbol{h}_1^\top \underline{\mathsf{B}} \boldsymbol{h}_1 - \boldsymbol{h}_2^\top \underline{\mathsf{B}} \boldsymbol{h}_2 = 0$$

- with $\mathsf{B} := \mathsf{K}^{-\top}\mathsf{K}^{-1}$
- If we know B, we can compute K
- Inspect the equations above:
  - B consists of the unknowns
  - h are known
  - Two equations that relate B and h

Next Step: Compute B

- $\mathsf{B} := \mathsf{K}^{-\top}\mathsf{K}^{-1}$
- B is symmetric and positive definite
- This means

$$\mathsf{B} = \begin{pmatrix} * & 0 & 0 \\ * & * & 0 \\ * & * & * \end{pmatrix} \begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{pmatrix} = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{12} & b_{22} & b_{23} \\ b_{13} & b_{23} & b_{33} \end{pmatrix}$$

- We define a vector of six unknowns

$$\boldsymbol{b} = (b_{11}, b_{12}, b_{13}, b_{22}, b_{23}, b_{33})$$

Exploiting Constraints

- Unknown vector $\boldsymbol{b} = (b_{11}, b_{12}, b_{13}, b_{22}, b_{23}, b_{33})$
- From our two equations

$$\boldsymbol{h}_1^\top \mathrm{B} \boldsymbol{h}_2 = 0 \qquad \boldsymbol{h}_1^\top \mathrm{B} \boldsymbol{h}_1 - \boldsymbol{h}_2^\top \mathrm{B} \boldsymbol{h}_2 = 0$$

- Construct a system of linear equations $V\boldsymbol{b} = 0$ exploiting the prev. constraints:

$$\boldsymbol{v}_{12}^\top \boldsymbol{b} = 0 \qquad \boldsymbol{v}_{11}^\top \boldsymbol{b} - \boldsymbol{v}_{22}^\top \boldsymbol{b} = 0$$

(eqn. from first constraint)
$\boldsymbol{r}_1^\top \boldsymbol{r}_2 = 0$

(eqn. from second constraint)
$\|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\|$

The Matrix V

- The matrix V is given as

$$V = \begin{pmatrix} \boldsymbol{v}_{12}^\top \\ \boldsymbol{v}_{11}^\top - \boldsymbol{v}_{22}^\top \end{pmatrix} \text{ with } \boldsymbol{v}_{kl} = \begin{bmatrix} h_{1k}h_{1l} \\ h_{1k}h_{2l} + h_{2k}h_{1l} \\ h_{3k}h_{1l} + h_{1k}h_{3l} \\ h_{2k}h_{2l} \\ h_{3k}h_{2l} + h_{2k}h_{3l} \\ h_{3k}h_{3l} \end{bmatrix}$$

elements of H

- For one image, we obtain

$$\begin{pmatrix} \boldsymbol{v}_{12}^\top \\ \boldsymbol{v}_{11}^\top - \boldsymbol{v}_{22}^\top \end{pmatrix} \boldsymbol{b} = \boldsymbol{0}$$

The Matrix V

- For multiple images, we stack the matrices to a 2n x 6 matrix

$$\begin{array}{r} \text{image 1} \longrightarrow \\ \\ \\ \text{image n} \longrightarrow \\ \\ \end{array} \begin{pmatrix} \boldsymbol{v}_{12}^{\top} \\ \boldsymbol{v}_{11}^{\top} - \boldsymbol{v}_{22}^{\top} \\ \cdots \\ \boldsymbol{v}_{12}^{\top} \\ \boldsymbol{v}_{11}^{\top} - \boldsymbol{v}_{22}^{\top} \end{pmatrix} \boldsymbol{b} = \mathbf{0}$$

- We need to solve the linear system $V\boldsymbol{b} = 0$ to obtain $\boldsymbol{b}$ and thus K

Solving the Linear System

- The system $V\boldsymbol{b} = 0$ has a trivial solution (invalid matrix B)
- Impose additional constraint $\|\boldsymbol{b}\| = 1$

Solving the Linear System

- The system $Vb = 0$ has a trivial solution (invalid matrix B)
- Impose additional constraint $\|\boldsymbol{b}\| = 1$
- Real measurements are noisy
- Find the solution that minimizes the squared error

$$\boldsymbol{b}^* = \arg\min_{\boldsymbol{b}} \|V\boldsymbol{b}\| \text{ with } \|\boldsymbol{b}\| = 1$$

- Solve via SVD as in DLT computation

What is Needed?

- We need at least **4 points per plane** to compute the matrix H
- Each **plane** gives us **two equations**
- Since B has 5 DoF, we need at least **3 different views of a plane**
- Solve $Vb = 0$ via SVD to compute K

Non-Linear Parameters?

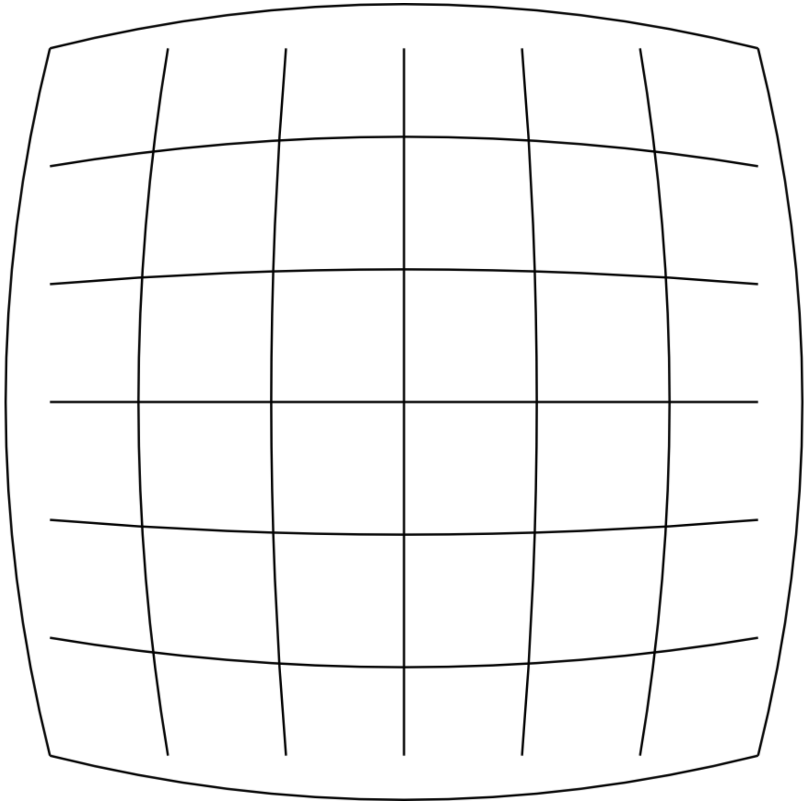
General Calibration Matrix

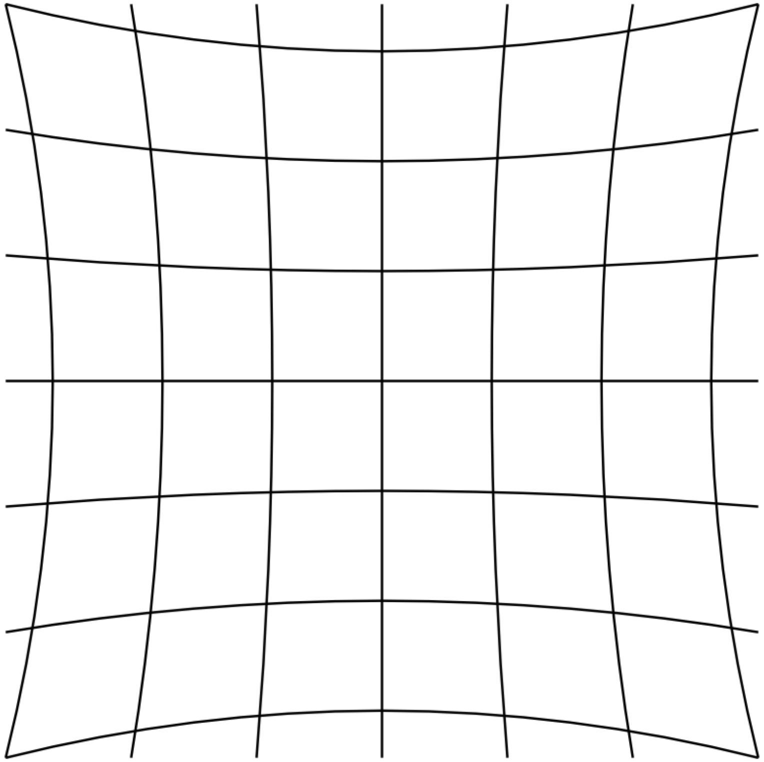
General calibration matrix is obtained by combining the one of the affine camera with the general mapping

$$
\begin{aligned}
{}^a\mathsf{K}(\boldsymbol{x},\boldsymbol{q}) &= {}^a\mathsf{H}_s(\boldsymbol{x},\boldsymbol{q})\,\mathsf{K} \\
&= \begin{bmatrix} 1 & 0 & \Delta x(\boldsymbol{x},\boldsymbol{q}) \\ 0 & 1 & \Delta y(\boldsymbol{x},\boldsymbol{q}) \\ 0 & 0 & 1 \end{bmatrix} \mathsf{K}
\end{aligned}
$$

see lecture on camera parameters

Lens Distortion Example





see lecture on camera parameters

Example: Barrel Distortion

- A standard approach for wide angle lenses is to model the barrel distortion

$$\begin{aligned} {}^a x &= x(1 + q_1\, r^2 + q_2\, r^4) \\ {}^a y &= y(1 + q_1\, r^2 + q_2\, r^4) \end{aligned}$$

- with $[x, y]^\mathsf{T}$ being point as projected by an ideal pin-hole camera
- with r being the distance **of the pixel** in the image to the principal point
- Additional non-linear parameters q_1, q_2

see lecture on camera parameters

Error Minimization

Lens distortion can be calculated by minimizing a non-linear error function

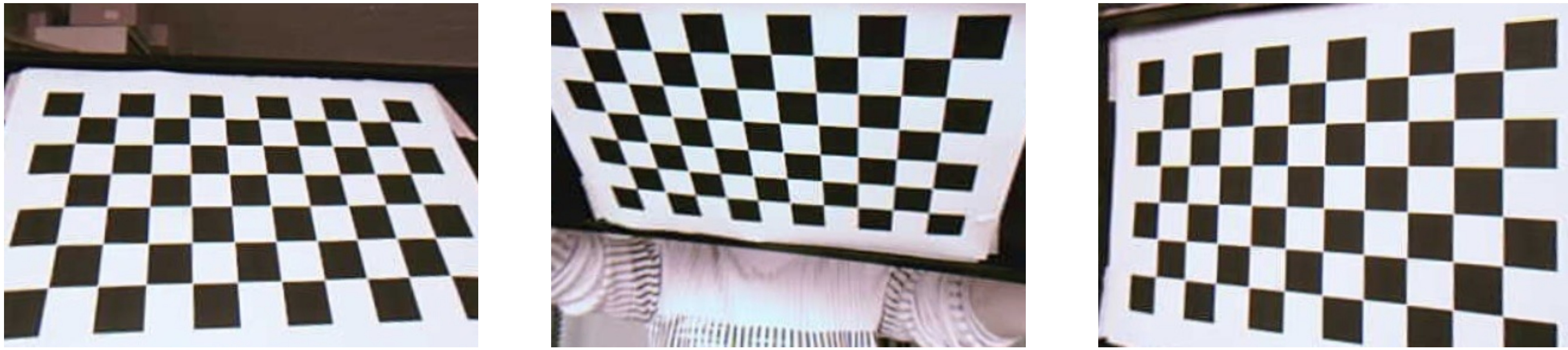
$$\min_{(\mathsf{K},\boldsymbol{q},R_n,\mathbf{t}_n)} \sum_n \sum_i \|\mathbf{x}_{ni} - \hat{\mathbf{x}}(\mathsf{K}, \mathbf{q}, R_n, \mathbf{t}_n, \mathbf{X}_{ni})\|^2$$

...linearize to obtain a quadratic function, compute derivative, set to 0, solve linear system, iterate...
(solved using Levenberg-Marquardt, K by Zhang's method as initial value)

Example Results

- Before calibration:

- After calibration:



Summary on Camera Calibration Using a Checkerboard

- 1st step: Assume affine camera model by using Zhang's method
- 2nd step: Non-linear model for lens distortion starting with Zhang's K
- Approach to camera calibration that
  - accurately determines the camera parameters
  - is relatively easy to realize in practice

Camera Calibration Summary

- Calibration means estimating the (intrinsic) parameters of a camera
- Linear and non-linear errors
- Linear errors: 5 parameters of K
- Zhang's method estimates the 5 linear parameters using a checkerboard
- Estimate non-linear parameters in a second step

Literature

- Zhang, A Flexible New Technique for Camera Calibration, MSR-TR-98-71 (uses a slightly different notation)
- Förstner & Wrobel, Photogrammetric Computer Vision, Chapter 11.2

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