Photogrammetry & Robotics Lab

Some Math Basics

Cyrill Stachniss

The slides have been created by Cyrill Stachniss.

Motivation

- We use several concepts from math
- **Goal:** Provide a short reminder for few things that we will use on our way

Brief, informal, incomplete, and unordered set of explanations

Motivation

- We use several concepts from math
- **Goal:** Provide a short reminder
- **Topics**
  - Solving Ax=b
  - Solving Ax=0 using SVD
  - Least squares with Gauss Newton
  - Skew-symmetric matrix
  - Derivative of rotation matrices
  - Homogenous coordinates (own lecture)

System of Linear Equations

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

Linear Equation System: Ax=b

Three cases:

- A is squared and has full rank
- A is **over**determined
- A is **under**determined

Solving Ax=b, w/ Exact Solution

- **A is a square matrix with full rank**
- Best-case situation, unique solution
- Can be solved in many ways...

Solving Ax=b, w/ Exact Solution

- **A is a square matrix with full rank**
- Best-case situation, unique solution
- Can be solved through
  - Gauss elimination
  - Inversion of A: $\boldsymbol{x} = A^{-1}\boldsymbol{b}$
  - Cholesky decomposition $\operatorname{chol}(A) = LL^{\mathsf{T}}$
    with lower triangular matrix L
    and then solving $L\boldsymbol{y} = \boldsymbol{b}$ and $L^{\mathsf{T}}\boldsymbol{x} = \boldsymbol{y}$
  - QR decomposition
  - Conjugate gradients

Solving Ax=b, A overdetermined

- **Common real-world situation**
- No exact solution exists
- We aim at finding minimizing $\|A\boldsymbol{x} - \boldsymbol{b}\|$ instead of solving $A\boldsymbol{x} = \boldsymbol{b}$:

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \|A\boldsymbol{x} - \boldsymbol{b}\|$$

- Ordinary least squares approach
- Solution can be obtained through

$$\boldsymbol{x} = (A^\top A)^{-1} A^\top \boldsymbol{b}$$

Solving Ax=b, A underdetermined

- **Infinitively many solutions exist** (or no solution if inconsistent)
- Not enough information available
- Approach: Find $\boldsymbol{x}$ which solves $A\boldsymbol{x} = \boldsymbol{b}$ and minimizes $\|\boldsymbol{x}\|$
- Solution

$$\boldsymbol{x} = A^{\mathsf{T}}(AA^{\mathsf{T}})^{-1}\boldsymbol{b}$$

Homogenous System

$$A\mathbf{x} = \mathbf{0}$$

Homogenous System: Ax=0

- Find a solution $x \neq \mathbf{0}$ fulfilling $Ax = \mathbf{0}$
- Means system is underdetermined
- There exists a null space of *A* called $\mathrm{null}(A)$ and all x fulfilling $Ax = \mathbf{0}$ are elements of it
- *A*'s rank deficiency defines the dimensionality of the null space

Eigenvalues

- For a squared matrix, we have

$$\dim(A) = \dim(\mathrm{null}(A)) + \mathrm{rank}(A)$$

- Which impact does this have on the Eigenvalues of A ?

Eigenvalues

- For a squared matrix, we have

$$\dim(A) = \dim(\mathrm{null}(A)) + \mathrm{rank}(A)$$

- Which impact does this have on the Eigenvalues of A ?
- There are $\mathrm{rank}(A)$ non-zero Eigenvalues
- There are $\dim(\mathrm{null}(A))$ Eigenvalues that are zero

Eigenvector

- For each Eigenvector $\boldsymbol{\nu}$ holds $A\boldsymbol{\nu} = \lambda\boldsymbol{\nu}$
- Thus, for those with Eigenvalue 0 we have $A\boldsymbol{\nu} = 0\boldsymbol{\nu} = 0$

Eigenvector

- For each Eigenvector $\boldsymbol{\nu}$ holds $A\boldsymbol{\nu} = \lambda\boldsymbol{\nu}$
- Thus, for those with Eigenvalue 0 we have $A\boldsymbol{\nu} = 0\boldsymbol{\nu} = 0$
- **Result:** all Eigenvectors corresponding to an Eigenvalue of 0 solve $A\boldsymbol{x} = \mathbf{0}$
- The same holds for all linear combinations of these Eigenvectors
- These Eigenvectors form $\mathrm{null}(A)$

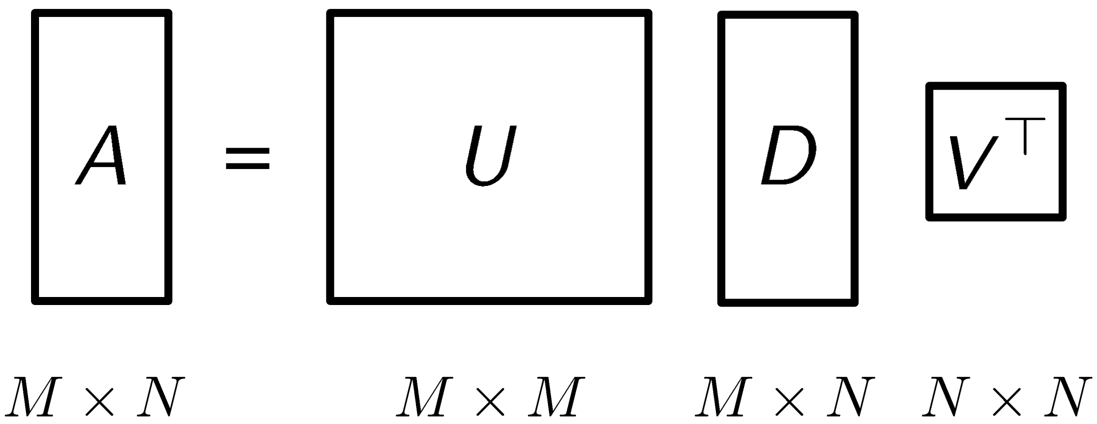
Eigenvector & Singular Vectors

- If A is square, real, symmetric and has non-negative Eigenvalues, then Eigenvalues equal to singular values
- Singular vectors and values also defined for non-square matrices
- We can use SVD to compute the singular values and vectors

Singular Value Decomposition

- SVD decomposes a matrix A into

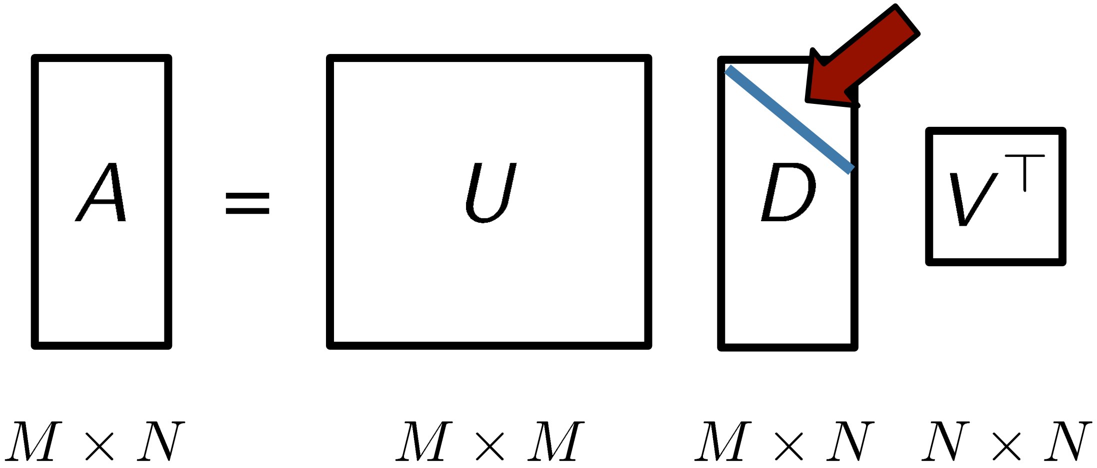
$$A = UDV^{\mathsf{T}}$$



Singular Values

- SVD decomposes a matrix A into

$$A = UDV^{\top}$$

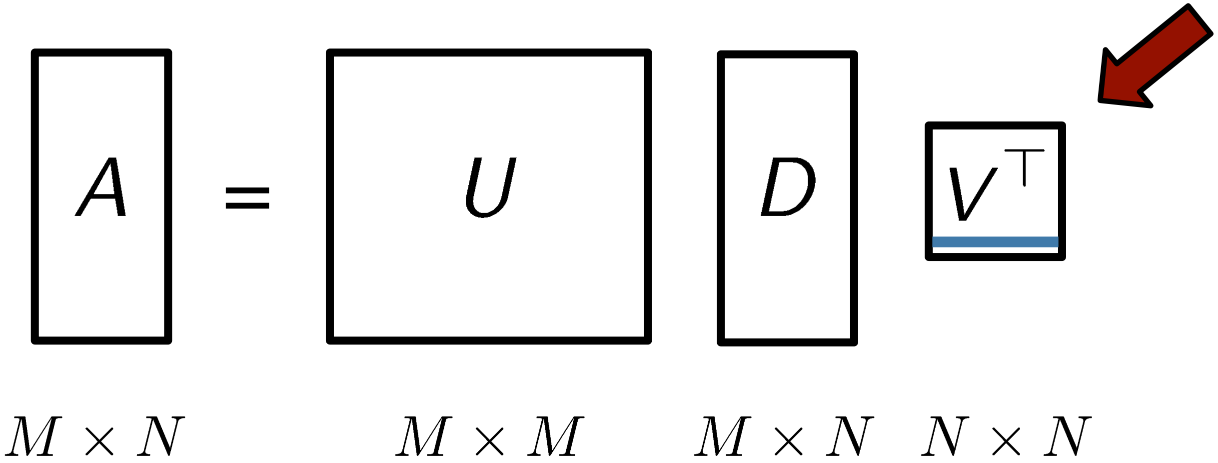


- D is a diagonal matrix of singular values sorted from large to small
- U, V are orthogonal matrices

Singular Vectors

- SVD decomposes a matrix A into

$$A = UDV^{\top}$$

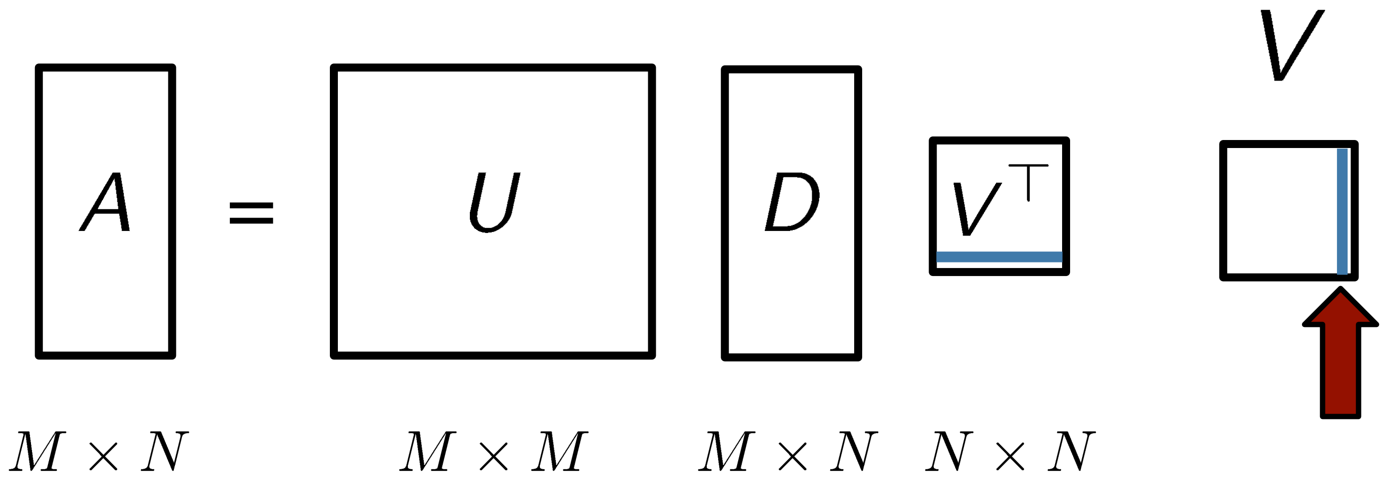


- $V^{\top}$ stores the corresponding singular vectors to the values

Singular Vectors

- SVD decomposes a matrix A into

$$A = UDV^{\top}$$



- Math libraries often returns V not $V^{\top}$
- The last column of V stores the vector corresponding to the smallest value

Solution to Ax=0 via SVD

- Decompose A using SVD: $A = UDV^\mathsf{T}$
- Check of the smallest singular value in D is zero: $D_{NN} \stackrel{?}{=} 0$
- **If so, the last column of V is a non-trivial solution $\boldsymbol{x}$ to $A\boldsymbol{x} = \mathbf{0}$**

Solution to Ax=0 via SVD

- Decompose A using SVD: $A = UDV^{\mathsf{T}}$
- Check of the smallest singular value in D is zero: $D_{NN} \stackrel{?}{=} 0$
- If so, the last column of V is a non-trivial solution $\boldsymbol{x}$ to $A\boldsymbol{x} = \mathbf{0}$
- **If not**, there is **no non-trivial solution** (i.e., only the trivial exists)
- **However,** the last column of V represents the vector that minimizes $\|A\boldsymbol{x}\|$ under the constraint $\|\boldsymbol{x}\| = 1$

Least Squares
(an non-Geodetic view)

Least Squares in 5 Minutes

https://www.youtube.com/watch?v=87S82fh4rI4

Graphical Explanation

$$\mathbf{x}$$

$$f_1(\mathbf{x}) = \hat{\mathbf{z}}_1 \qquad \mathbf{z}_1$$

$$f_2(\mathbf{x}) = \hat{\mathbf{z}}_2 \qquad \mathbf{z}_2$$

$$\ldots$$

$$f_n(\mathbf{x}) = \hat{\mathbf{z}}_n \qquad \mathbf{z}_n$$

state (unknown)	predicted measurements	real measurements

Error Function

- Error $\mathbf{e}_i$ is typically the **difference** between the **predicted and actual** measurement

$$\mathbf{e}_i(\mathbf{x}) \quad = \quad \mathbf{z}_i - f_i(\mathbf{x})$$

- We assume that the error has **zero mean** and is **normally distributed**
- Gaussian error with information matrix $\mathbf{\Lambda}_i$
- The squared error of a measurement depends only on the state and is a scalar

$$e_i(\mathbf{x}) \quad = \quad \mathbf{e}_i(\mathbf{x})^T \mathbf{\Lambda}_i \mathbf{e}_i(\mathbf{x})$$

Linearizing the Error Function

- Approximate the error functions around an initial guess $\mathbf{x}$ via Taylor expansion

$$\mathbf{e}_i(\mathbf{x} + \mathbf{\Delta x}) \simeq \underbrace{\mathbf{e}_i(\mathbf{x})}_{\mathbf{e}_i} + \mathbf{J}_i(\mathbf{x})\mathbf{\Delta x}$$

- $\mathbf{J}$ is the Jacobian

$$\mathbf{J}_f(x) = \begin{pmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \cdots & \frac{\partial f_2(x)}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m(x)}{\partial x_1} & \frac{\partial f_m(x)}{\partial x_2} & \cdots & \frac{\partial f_m(x)}{\partial x_n} \end{pmatrix}$$

Gauss-Newton

Iterate the following steps:

- Linearize around $\mathbf{x}$ and compute for each measurement

$$\mathbf{e}_i(\mathbf{x}+\boldsymbol{\Delta}\mathbf{x}) \simeq \mathbf{e}_i(\mathbf{x}) + \mathbf{J}_i \boldsymbol{\Delta}\mathbf{x}$$

- Compute the terms for the linear system $\mathbf{b}^\top = \sum_i \mathbf{e}_i^\top \boldsymbol{\Lambda}_i \mathbf{J}_i \quad \mathbf{H} = \sum_i \mathbf{J}_i^\top \boldsymbol{\Lambda}_i \mathbf{J}_i$
- Solve the linear system

$$\boldsymbol{\Delta}\mathbf{x}^* \;=\; -\mathbf{H}^{-1}\mathbf{b}$$

- Updating state $\mathbf{x} \leftarrow \mathbf{x} + \boldsymbol{\Delta}\mathbf{x}^*$

Skew-Symmetric Matrices

Skew-Symmetric Matrices

- A skew-symmetric matrix is a matrix S for which holds $S^\top = -S$

Skew-Symmetric Matrices

- A skew-symmetric matrix is a matrix S for which holds $S^\top = -S$
- S has zeros on the main diagonal
- $\forall S \in \mathbb{R}^{3\times 3} : \det(S) = 0$
- $\det(S) = 0 \quad \text{if } \dim(S) \text{ odd.}$

Skew-Symmetric Matrices in 3D

- In $\mathbb{R}^3$ we can express the cross product through a skew-symmetric matrix

$$\boldsymbol{a} \times \boldsymbol{b} = [\boldsymbol{a}]_\times \boldsymbol{b} = S_a \boldsymbol{b}$$

$$[\boldsymbol{a}]_\times = S_a = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

Skew-Symmetric Matrices in 3D

- In $\mathbb{R}^3$ we can express the cross product through a skew-symmetric matrix

$$\boldsymbol{a} \times \boldsymbol{b} = [\boldsymbol{a}]_\times \boldsymbol{b} = S_a \boldsymbol{b}$$

$$[\boldsymbol{a}]_\times = S_a = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}}_{\boldsymbol{a}} \times \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}}_{\boldsymbol{b}} = \begin{bmatrix} -a_3 b_2 + a_2 b_3 \\ a_3 b_1 - a_1 b_3 \\ -a_2 b_1 + a_1 b_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}}_{S_a} \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}}_{\boldsymbol{b}}$$

Derivative of a Rotation Matrix

Derivative of a Rotation Matrix

- Skew-symmetric matrices are useful to formulate the derivative of a rotation matrix
- For any rotation matrix R holds

$$RR^\top = I$$

- Consider a rotation by θ around x-axis

$$R_x(\theta)$$

- Then, we have $R_x(\theta)R_x^\top(\theta) = I$

Derivative of a Rotation Matrix

- Compute derivative (chain rule)

$$R_x(\theta)R_x^\top(\theta) = I$$

$$\frac{d}{d\theta}\left(R_x(\theta)R_x^\top(\theta)\right) = \frac{d}{d\theta}I$$

$$\frac{d}{d\theta}R_x(\theta)R_x^\top(\theta) + R_x(\theta)\frac{d}{d\theta}R_x^\top(\theta) = 0$$

Derivative of a Rotation Matrix

- Compute derivative (chain rule)

$$R_x(\theta) R_x^\top(\theta) = I$$

$$\frac{d}{d\theta}\left(R_x(\theta) R_x^\top(\theta)\right) = \frac{d}{d\theta} I$$

$$\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta) + R_x(\theta) \frac{d}{d\theta} R_x^\top(\theta) = 0$$

- Exploiting $(AB)^\top = B^\top A^\top$ leads us to

$$\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta) + \left(\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta)\right)^\top = 0$$

Derivative of a Rotation Matrix

- Rewrite $\underline{\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta)} + \left(\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta) \right)^\top = 0$
- as $S + S^\top = 0$
- This directly leads to $S^\top = -S$, which is a skew-symmetric matrix
- We can now exploit the fact that $\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta)$ is a skew-symmetric matrix

Derivative of a Rotation Matrix

- We have $S = \frac{d}{d\theta} R_x(\theta) R_x^\top(\theta)$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}$$

- So

$$S = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin\theta & -\cos\theta \\ 0 & \cos\theta & -\sin\theta \end{bmatrix}}_{\frac{d}{d\theta} R_x(\theta)} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}}_{R_x^\top(\theta)}$$

Derivative of a Rotation Matrix

$$
\begin{aligned}
S &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin\theta & -\cos\theta \\ 0 & \cos\theta & -\sin\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \\
&= S_{\boldsymbol{e}_x}
\end{aligned}
$$

with the unit vector $\boldsymbol{e}_x = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

Derivative of a Rotation Matrix

- This means $\frac{d}{d\theta} R_x(\theta) R_x^\top(\theta) = S_{\boldsymbol{e}_x}$
- and thus

$$\frac{d}{d\theta} R_x(\theta) = \frac{d}{d\theta} R_x(\theta) \underbrace{R_x^\top(\theta) R_x(\theta)}_{I} = S_{\boldsymbol{e}_x} R_x(\theta)$$

- The derivative of a rotation matrix $R_x(\theta)$ is the skew-symmetric matrix $S_{\boldsymbol{e}_x}$ times the rotation matrix itself

$$\frac{d}{d\theta} R_x(\theta) = S_{\boldsymbol{e}_x} R_x(\theta)$$

The Same for x,y,z Axes

- We can repeat the same to x, y, z and obtain

$$\frac{d}{d\theta} R_x(\theta) = S_{\boldsymbol{e}_x} R_x(\theta)$$

$$\frac{d}{d\theta} R_y(\theta) = S_{\boldsymbol{e}_y} R_y(\theta)$$

$$\frac{d}{d\theta} R_z(\theta) = S_{\boldsymbol{e}_z} R_z(\theta)$$

- and even for an arbitrary rot. axis $\boldsymbol{r}$

$$\frac{d}{d\theta} R_r(\theta) = S_r R_r(\theta)$$

Infinitesimal Small Rotations

- Similarly, we can also approximate an infinitesimally small rotation by

$$R \approx I + \mathrm{d}R = I + S_{\mathrm{d}r} = I + \begin{bmatrix} 0 & -\mathrm{d}\kappa & \mathrm{d}\phi \\ \mathrm{d}\kappa & 0 & -\mathrm{d}\omega \\ -\mathrm{d}\phi & \mathrm{d}\omega & 0 \end{bmatrix}$$

- Thus,

$$\mathrm{d}R = S_{\mathrm{d}r} = \begin{bmatrix} 0 & -\mathrm{d}\kappa & \mathrm{d}\phi \\ \mathrm{d}\kappa & 0 & -\mathrm{d}\omega \\ -\mathrm{d}\phi & \mathrm{d}\omega & 0 \end{bmatrix}$$

Summary

This lecture was a **brief and informal reminder** of concepts we will need

- Solving Ax=b
- Solving Ax=0 using SVD
- Least squares with Gauss Newton
- Skew-symmetric matrics
- Derivative of a rotation matrix
- Own lecture: Homogenous coordinates

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