3D Computer Vision (SoSe2024)

About this course

Prof. Dr. Ulrich Schwanecke

RheinMain University of Applied Sciences

CVMR
Computer Vision & Mixed Reality Group

by Decker

Course Goal and Content

- **Goal**
    - Gain an understanding of the theoretical and practical concepts of **3D Computer Vision**, e.g.
        - Camera calibration
        - Epipolar geometry
        - Structure-from-Motion
        - Image rectification
        - Block matching
        - Volumetric fusion
        - ...
    - Be able to
        - develop and train computer vision models,
        - repoduce results and conduct original research

- **(Planned) Content**
    1. Image Formation
    2. 3D Projective Space and 3D Motion
    3. Conic Sections and Quadrics
    4. Camera Models and Calibration
    5. Shape from Shading and Photometric Stereo
    6. Structure from Motion
    7. Multi View Reconstruction, Optical Flow
    8. Siamese Networks, End-to-End Learning
    9. Data Driven 3D Reconstruction
    10. Neural Scene Representations
    11. Diverse Topics in 3D Computer Vision

Organization

- **SWS 2V + 2Ü, 6 ECTS, Total Workload: 180h**
- **Lecture** (13)
  - Monday, 14:15-15:45, 04 422
    - Apr. 15/22/29, May. 06/13/27, Jun. 03/10/17/24, Jul. 01/08/15
  - All lecture related information at http://cvmr.info/lectures/3DCVSS24/ (user: 3DCV passwd: sose2024)
- **Exercise Sessions**
  - Exercises are mandatory [Day/time to be determined]
- **Exam**
  - Content: lectures and exercises [Very likely written (Day/time will be announced)]
  - To qualify for the exam you have to
    - have $\geq 50\%$ of all achievable points ($\geq 25\%$ for each problem set) and **present at least one assignment**

Course Materials

- **Books**
  - Y. Ma, et. al, *An Invitation to 3-D Vision - From Images to Geometric Models*, 2004, https://www.eecis.udel.edu/~cer/arv/readings/old_mkss.pdf
  - R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, 2003, https://www.robots.ox.ac.uk/~vgg/hzbook/
  - R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2011, https://szeliski.org/Book
  - I. Gooldfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016, https://www.deeplearningbook.org
  - J. E. Solem, *Programming Computer Vision with Python*, O'Reilly, 2012
  - V. K. Ayyadevara, Y. Reddy, *Modern Computer Vision with PyTorch*, Packt, 2020
  - M. P. Deisenroth, et al, *Mathematics for Machine Learning*, https://mml-book.github.io
  - K. B. Petersen, M. S. Pedersen, *The Matrix Cookbook*, http://www.cs.toronto.edu/~bonner/courses/2012s/csc338/matrix_cookbook.pdf

Course Materials

- **Tutorials**
  - The Python Tutorial: https://docs.python.org/3/tutorial
  - Numpy Quickstart: https://numpy.org/devdocs/user/quickstart.html
  - PyTorch Tutorial: https://pytorch.org/tutorials
- **Frameworks, IDEs**
  - Visual Studio Code: https://code.visualstudio.com/
  - Google Colab: https://colab.research.google.com
- **Courses**
  - Slide deck covering *Szeliski's book* https://szeliski.org/Book
  - I. Gkioulekas, *Computer Vision* https://www.cs.cmu.edu/~16385/
  - A. Owens, *Foundations of Computer Vision* https://web.eecs.umich.edu/~ahowens/eecs504/w20/

Prerequisites

- Basic math skills
  - Linear Algebra, Calculus, Probability
- Basis computer science skills
  - Variables, functions, loops, classes, algoritms
- Basic Python coding skills
  - https://docs.python.org/3/turorial/
- Basic PyTorch coding skills
  - https://pytorch.org/turorials

Prerequisites

- **Linear Algebra**
  - Vectors: $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$
  - Matrices: $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{m \times n}$
  - Operations:
    - $\mathbf{x}^\top \mathbf{y}, \mathbf{A}\mathbf{x}, \mathbf{x} \times \mathbf{y}$
    - $\mathbf{A}^\top, \mathbf{A}^{-1}$, trace$(\mathbf{A})$, det$(\mathbf{A})$, $\mathbf{A} + \mathbf{B}, \mathbf{AB}$
  - Norms: $||\mathbf{x}||_1, ||\mathbf{x}||_2, ||\mathbf{x}||_\infty, ||\mathbf{A}||_F$
  - Eigenvalues, Eigenvectors, SVD: $\mathbf{A} = \mathbf{UDV}^\top$
- **Calculus**
  - Multivariate functions: $f : \mathbb{R}^n \to \mathbb{R}$
  - Partial derivatives: $\frac{\partial f}{\partial x_i}, i = 1, \ldots, n$, Gradient
  - Integrals: $\int f(x)dx$
- **Probability**
  - Probability distributions: $P(X = x)$
  - Expectation: $\mathbb{E}_{x \sim p}[f(x)] = \int_x p(x)f(x)dx$
  - Variance: Var$(f(x)) = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])^2]$
  - Marginal: $p(x) = \int p(x, y)dy$
  - Conditional: $p(x, y) = p(x|y)p(y)$
  - Bayes rule: $p(x|y) = p(y|x)/p(y)$
  - Distributions: Uniform, Gaussian

Time Management

Activity	Times	Total
Attending (watching) the lecture	2h / week	24h
Self-study of lecture materials	2h / week	24h
Participation in exercise	2h / week	24h
Solving the assignments	6h / week	72h
Preparation for the final exam	36h	36h
Total workload		**180h**

See you on Monday, April 25, 2024, in 04 422

ulrich.schwanecke(at)hs-rm.de