Image Processing - Organization

Institute for Biomedical Imaging, Hamburg University of Technology

- 👨‍🏫 Lecture: Prof. Dr.-Ing. Tobias Knopp
- 🧑‍🏫 Exercise: Konrad Scheffler, M.Sc.

Table of Contents

Lecture

- Classical in person lectures
- Notebooks uploaded on studIP

Exercise

- Programming based exercises, which will be autograded and give you bonus points for the final exam.
- Will be handed out each Tuesday after the lecture and need to be submitted until the following Tuesday 13 pm.
- Exercises are solved in the Julia programming language, usually using Pluto notebooks.
- The solution **must** be a single Julia file: **LastName_FirstName_ExerciseN.jl** where N is the number of the exercise sheet. No other naming scheme or file extension (like zip) are allowed.
- Sometimes there will be non-programming tasks, which have to be carried out on an extra sheet of paper. These tasks will not be corrected, however they are important to understand the mathematical objects we use and as a preparation for the exam.
- You can seek help from each other using the Stud.IP forum. Please also form learning groups.
- Each week on thursday (15:00-16:00) an exercise meeting will take place, where we discuss your solutions, have a look at the current tasks and questions are answered.
- Sample solutions will also be provided after the submission deadline.

Course Material

The course is designed to be self-contained and the provided material should be used as the primary ressource. This course is based on and guided by the following book:

- Digital Image Processing, Rafael Gonzalez, Richard Woods, Ben Lauwens and Allen Downey

You can find a pdf of the book online but of course need to buy it.

Programming Environment

This course uses the programming language Julia for programming examples. Julia is a young language for scientific programming that is as fast as C/C++ but as convenient to use as high-level languages (Matlab/Python). Julia has some high-quality image processing libraries (see Images.jl), which we will use in parts of the course.

The course material is created using the notebook system Pluto.jl, which allows to mix text, images, and code. You can execute the notebooks yourself. This can be very handy if you want to understand the algorithms we develop. Many of the graphics / illustrations are generated using actual code and you can click on the eye in the upper left corner to access the code

The following section explains how to get started with Julia and Pluto notebooks.

Installation

- Download Julia in the newest version (1.11). Binaries for Mac, Windows, and Linux are available. Please note that you should never install Julia from a package manager but we encourage you to use `juliaup`.
- If you are using Windows, we strongly encourage that you start Julia from the command line (e.g. PowerShell).
- Once you have started Julia you can add Pluto by entering

```
julia> using Pkg
julia> Pkg.add("Pluto")
```

- After that you can run the notebook server:

```
julia> using Pluto
julia> Pluto.run()
```

- This will open a Pluto window in your local web browser. Now you can enter the path to a Pluto notebook (Organization.jl) and load it. In this way you can run all course material (lectures, exercises, solutions). You can also export the *.jl* file into *html* or *pdf* (symbol shown on the upper right). This can be handy for offline usage.

Note

The export to pdf function just works well in Chrome/Chromium browsers. You can use

```
using Pluto; Pluto.run(launch_browser=false)
```

to select another browser than the standard browser.