

# Computer Vision

## Introduction to Jupyter Notebooks and Colab for Exercise 3

Autonomous Vision Group  
University of Tübingen / MPI-IS

EBERHARD KARLS  
UNIVERSITÄT  
TÜBINGEN



e l l i s  
European Laboratory for Learning and Intelligent Systems

# Environment Setup

You have two choices to complete our exercises:

1. Setup jupyter notebook locally on your machine
2. Use google colab in your browser

# 1. Local Environment Setup

# 1. Local Environment Setup

- ▶ Follow the instructions for your OS to install the Python package manager `conda`:  
`https://docs.conda.io/projects/conda/en/latest/user-guide/install/`
- ▶ Download the archive for exercise 1 and open a terminal in the `code` directory
- ▶ Create the new environment `lecturecv` with required packages (numpy, etc.):  
`conda env create -f environment.yml`
- ▶ Before launching your notebook you need to activate the environment:  
`conda activate lecturecv-ex02` (same as for the last exercise)
- ▶ Run this command from the directory where the jupyter notebooks are located:  
`jupyter-notebook`

# 1. Local Environment Setup

You can then navigate to the respective notebook and edit it in the browser

**Computer Vision Lecture - Exercise 1 - Image Formation**

In this exercise, you will gain hands-on experience regarding the image formation process and geometric transformations. More specifically, we will develop our own simple renderer and play around with focal lengths, BRDF-based shading, rotations, translations, and much more! In the second part of the notebook, we will have a look at homographies and how they can be used to stitch together images to form panoramas.

This notebook guides you through the relevant steps. When you see helper functions, you don't need to do anything - they are already implemented. The functions you need to implement are indicated as Exercise Function. Sometimes, you can find Hints - these are written upside-down so you can first try to find the solution without reading them.

Good luck and let's have fun!

### Preliminaries

Let's first import relevant libraries and define hyperparameters. For the latter, we set the image height  $H$  and width  $W$  to 128 pixels.

```
In [ ]: import numpy as np
import itertools
import matplotlib
import matplotlib.pyplot as plt
from scipy.spatial.transform import Rotation as R
from IPython.display import HTML
from matplotlib import animation
from matplotlib.patches import Polygon
import cv2

# Let's first define hyperparameters. In our case, we set the image height  $H$  and width  $W$  to 128 pixels.
H, W = 128, 128
```

We need an object in our scene in order to render more than an empty image! For this, we define a helper function which returns the faces (and optionally the face normals) of a cube.

```
In [ ]: #####
##### Helper Function #####
#####
def get_cube(center=(0, 0, 2), rotation_angles=[0., 0., 0.], with_normals=False, scale=1.):
    """Returns an array containing the faces of a cube.

    Args:
        center (tuple): center of the cube
        rotation_angles (tuple): Euler angles describing the rotation of the cube
        with_normals (bool): whether to return the normal vectors of the faces
        scale (float): scale of cube

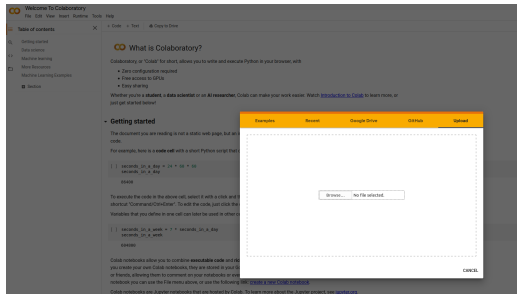
    """
    # A cube consists of 6 faces and 8 corners:
    #
    #   +-----+
    #   |         |
    #   |         |
    #   |         |
    #   |         |
    #   |         |
    #   |         |
    #   +-----+
    #
    # Let's first consider the unit cube. The corners are:
```

## 2. Online Environment Setup: Google Colab

## 2. Online Environment Setup: Google Colab

Alternatively, you can use Google Colab online

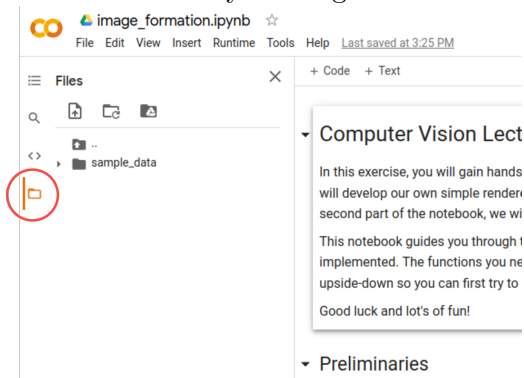
- ▶ Create a free Google account at: <https://google.com>
- ▶ Navigate to <https://colab.research.google.com/> in your browser
- ▶ Click on File → Upload notebook and upload the respective notebook



## 2. Online Environment Setup: Google Colab

Alternatively, you can use Google Colab online

- ▶ Create a free Google account at: <https://google.com>
- ▶ Navigate to <https://colab.research.google.com/> in your browser
- ▶ Click on File → Upload notebook and upload the respective notebook
- ▶ You can also upload additional files by clicking on the folder symbol on the left:

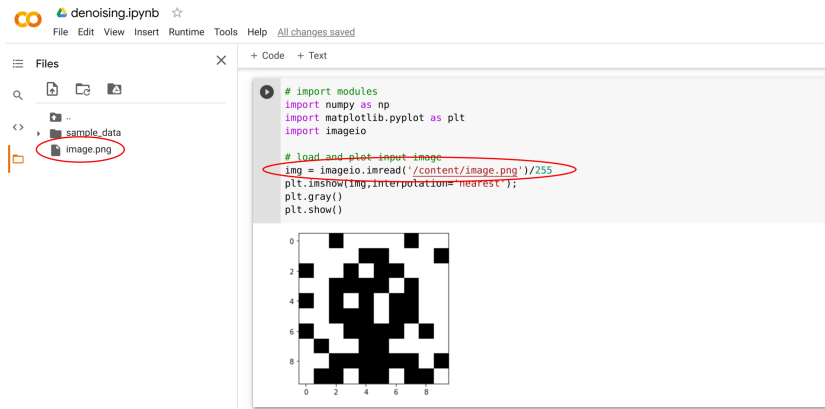




## 2. Google Colab: Upload denoising image

denoising.ipynb: Upload the image for denoising

- ▶ Upload the image `gfx/image.png` to e.g. `/content/`
- ▶ In the code, read the image from `/content/image.png`



Questions?