



# Maschinelles Sehen

## Getting ready

SoSe 2019

Die Bearbeitung der Übungsaufgaben im Kurs erfolgt in der Programmiersprache Python mittels des Computer Vision Frameworks OpenCV <http://opencv.org/>. Diese Aufgabe dient zur Installation und Einarbeitung in Programmiersprache und Bibliothek.

Diese und alle weiteren Aufgaben sollen ausschließlich allein bearbeitet werden. Implementieren Sie die Aufgaben eigenständig. Die Abgabe der Aufgaben erfolgt einzeln über Moodle bzw. in der Übung. Halten Sie sich bei der Bearbeitung der Aufgaben und des Projektes an folgenden Honor Code (Inspiriert vom Honor Code der Stanford University):

*You may consult any papers, books, online references, or publicly available implementations for ideas and code that you may want to incorporate into your strategy or algorithm, so long as you clearly cite your sources in your code and your writeup. However, under no circumstances may you look at another group's code or incorporate their code into your project or assignment.*

**Diese Aufgabe muss nicht abgegeben werden und wird nicht bewertet.**

### Aufgabe 1: Installation Anaconda + OpenCV

Installieren Sie zunächst das Framework Anaconda (<https://www.anaconda.com/distribution/#download-section>) in der Python 3.7.1 Version. Das installiert sich normalerweise in das Userverzeichnis z.B. `/home/you/anaconda3` unter Linux. Anaconda ist ein umfangreiches Framework für Wissenschaftliches Rechnen, Computer Vision und Data Science in Python.

Als nächstes benötigen wir OpenCV in einer älteren Version. Erfahrene Python User sollten vielleicht eine virtual environment benutzen, wenn Python sonst nicht verwendet wird reicht auch einfach folgendes Kommando:

```
1 $ pip install opencv-contrib-python==3.4.2.17
```

Starten Sie Python und probieren Sie bitte folgenden Pythoncode aus:

```
1 $ python
2 Python 3.7.1 (default, Dec 14 2018, 19:28:38)
3 [GCC 7.3.0] :: Anaconda, Inc. on linux
4 Type "help", "copyright", "credits" or "license" for more information.
5 >>> import cv2
6 >>> import numpy
7 >>> exit()
```

Wenn alles fehlerfrei ausgeführt wird sind Sie startbereit.

### Aufgabe 2: Arbeit mit Python, Numpy und OpenCV

Stellen Sie nach der Installation der Pakete sicher dass der mitgelieferte Pythoncode `[test.py]` ohne Fehler läuft und Ihnen das Bild Ihrer Webcam oder einen Platzhalter zeigt. Beenden können Sie das Programm mit 'q':

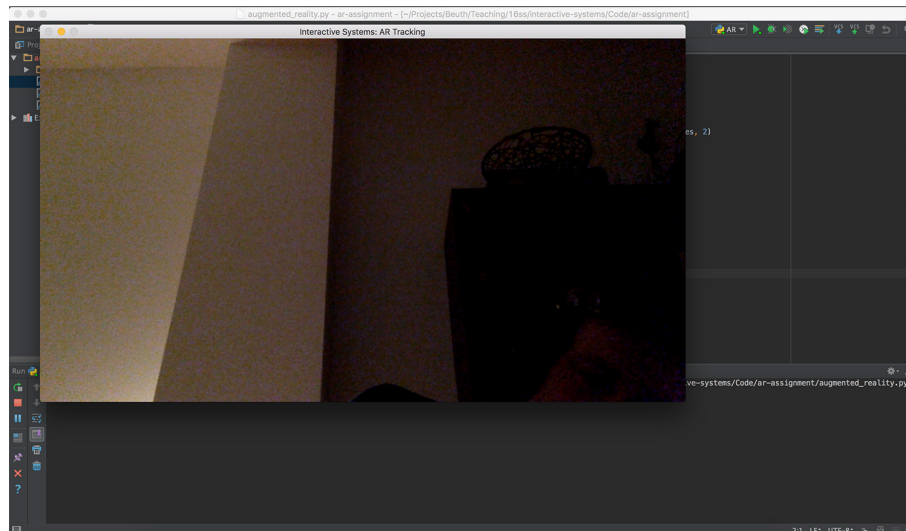


Abbildung 1: Bild einer Webcam

```
1 import numpy as np
2 import cv2
3 import requests
4
5 sift = cv2.xfeatures2d.SIFT_create()
6 cap = cv2.VideoCapture(0)
7
8 req = requests.get('https://cdn-images-1.medium.com/max/1000/1*n-
9 rhQ7gilqLbYaiON4bPUQ.jpeg?q=20')
10 arr = np.asarray(bytearray(req.content), dtype=np.uint8)
11 img = cv2.imdecode(arr, -1) # 'Load it as it is'
12
13 while(True):
14     # Capture frame-by-frame
15
16     ret, frame = cap.read()
17
18     # Our operations on the frame come here
19     try:
20         gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
21     except:
22         gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
23
24     if cv2.waitKey(100) & 0xFF == ord('q'):
25         break
26
27     # Display the resulting frame
28     cv2.imshow('frame', gray)
29
30
31
32 # When everything done, release the capture
33 cap.release()
34 cv2.destroyAllWindows()
```

Arbeiten Sie sich in die grundlegenden Sprachkonzepte von Python ein (<https://www.python.org/about/gettingstarted/>). Zusätzlich empfehle ich Ihnen folgendes Tutorial (<http://cs231n.github.io/python-numpy-tutorial/>). Dieses gibt Ihnen auch eine Einführung in die Bibliothek Numpy (<http://www.numpy.org/>).

Arbeiten Sie sich in die grundlegende Funktionsweise von OpenCV ein. Einige hilfreiche Tutorials und Code-Beispiele finden Sie hier: [http://opencv-python-tutroals.readthedocs.org/en/latest/py\\_tutorials/py\\_tutorials.html](http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_tutorials.html)

### **Aufgabe 3: (Optional) Gitlab**

Machen sie sich mit dem Gitlab Server der Hochschule vertraut. Sie können sich dort mit den Zugangsdaten ihres HRZ-Accounts anmelden. Gitlab stellt eine komfortable Benutzerschnittstelle für die Erstellung und Verwaltung von Git-Repositories bereit.

Für den einfachen und sicheren Datenaustausch mit dem Git-Server empfiehlt sich die Erstellung eines SSH-Schlüssels. Hinweise dazu finden sich im Help-Bereich.