



Maschinelles Sehen

Getting ready

WiSe 2019/20

Der 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 + OpenCV3

Installieren Sie zunächst das Framework Anaconda (<https://www.continuum.io/downloads> in der Python 3.7 Version. Das installiert sich normalerweise in das Userverzeichnis z.B. `/Users/kristian/anaconda3` unter MacOS. Anaconda ist ein umfangreiches Framework für Wissenschaftliches Rechnen, Computer Vision und Data Science in Python. Es kommt mit einer eigenen Paketverwaltung, was uns die Installation von Zusatzbibliotheken wie OpenCV und Scikit-learn plattformübergreifend erleichtert. Ich empfehle die Commandline Installation. Installieren Sie die komplette Python Umgebung folgendermaßen unter der Nutzung der mitgelieferten Umgebungskonfigurationsdatei `cv_environment.yml` Datei und folgendem Aufruf: (**Wichtiger Hinweis:** Laden Sie die Datei bitte vom Moodle herunter und gehen in das Verzeichnis, wo sich die heruntergeladene Datei befindet und führen dort das Kommando aus. Wir brauchen für den Kurs genau die Versionen von OpenCV, die in der `.yml` Datei stehen.):

```
1 $ conda env create -f cv_environment.yml
```

Die Bibliothek wurde erfolgreich installiert, wenn sie Python auf der Kommandozeile starten und die folgende Eingabe keine Fehler liefert. Dafür müssen Sie vorab auf der Konsole in die 'cv' Anacondaumgebung wechseln:

```
1 $ source activate cv 'oder' conda activate cv
```

Zum deaktivieren können Sie diese wie folgt wieder verlassen

```
1 $ source deactivate 'oder' conda deactivate
```

Hinweis: Unter Windows müssen Sie dafür ggf. die AnacondaPrompt nutzen.

Probieren Sie bitte folgenden Pythoncode aus:

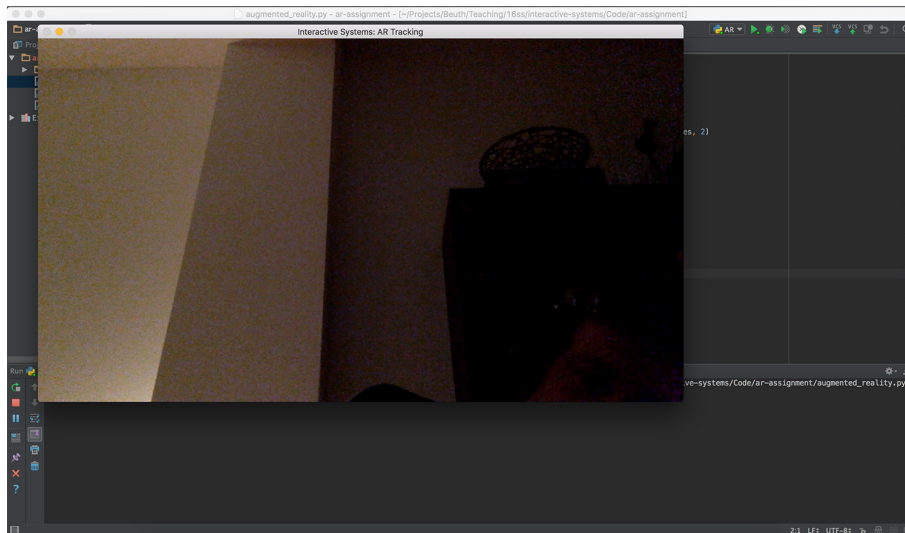


Abbildung 1: Bild einer Webcam

```
1 $ python
2 Python 3.7.4 (default, Aug 13 2019, 15:17:50)
3 [Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
4 Type "help", "copyright", "credits" or "license" for more information.
5 >>> import cv2
6 >>> import numpy
```

Aufgabe 2: Arbeit mit Python, Numpy und OpenCV

Stellen Sie nach der Installation der Pakete sicher das folgender Pythoncode ohne Fehler läuft und Ihnen das Bild Ihrer Webcam zeigt - (**Hinweis: Der Sourcecode liegt in der beigefügten test.py Datei**):

```
1 import numpy as np
2 import cv2
3
4 sift = cv2.xfeatures2d.SIFT_create()
5 cap = cv2.VideoCapture(0)
6
7 while(True):
8     # Capture frame-by-frame
9     ret, frame = cap.read()
10
11     # Our operations on the frame come here
12     gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
13
14     # Display the resulting frame
15     cv2.imshow('frame', gray)
16     if cv2.waitKey(1) & 0xFF == ord('q'):
17         break
18
19 # When everything done, release the capture
20 cap.release()
21 cv2.destroyAllWindows()
```

Arbeiten Sie sich in die grundlegenden Sprachkonzepte von Python ein (<https://www.python.org/about/gettingstarted/>). Ich nutze im Moment die IDE PyCharm, die für Studenten kostenlos und durchaus empfehlenswert ist (<https://www.jetbrains.com/pycharm/>) oder VS Code. 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/>). Ich habe

zusätzlich ein Jupyter Notebook angehängt. Bitte setzen Sie sich selbstständig mit dem Jupyter Notebook Konzept auseinander und

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

Aufgabe 3: 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.