

# Einführung in die Bildverarbeitung

## Übung

Christian Wilms

Computer Vision Group  
Universität Hamburg

Sommersemester 2020

20. April 2020

# Wer wir sind

## Computer Vision Group (CV)

### Übung



Christian Wilms

- [wilms@informatik.uni-hamburg.de](mailto:wilms@informatik.uni-hamburg.de)
- Büro R-108

### Vorlesung



Prof. Simone Frintrop

- [frintrop@informatik.uni-hamburg.de](mailto:frintrop@informatik.uni-hamburg.de)
- Büro R-105

Wendet euch bei Fragen jederzeit an mich!  
**[wilms@informatik.uni-hamburg.de](mailto:wilms@informatik.uni-hamburg.de)**

# Übungen in der Übersicht

Im Rahmen der Übung werden die in der Vorlesung erlernten Konzepte vertieft.

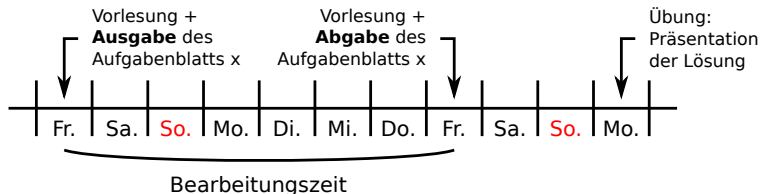
- jede Woche gibt es ein Aufgabenblatt im Moodle
- es gibt theoretische Aufgaben (selber rechnen etc.) und praktische Aufgaben (Programmieraufgaben)
- komplett gelöste Aufgaben werden per Mail abgeben
- zum Bestehen der Übung werden 50% aller Punkte benötigt (Details s. Scheinkriterien)
- anstelle einer gemeinsamen Besprechung der Aufgaben werden Videos mit Lösungen und Erklärungen im Moodle bereitgestellt

Details können wir im Laufe des Semesters je nach Entwicklung anpassen. Wir lernen gerade alle gemeinsam!

# Ablauf der Übungen

Jede Übungswoche beginnt und endet mit der Vorlesung!

- Ausgabe der Aufgabenblätter via Moodle i.A. freitags zur VL
- zusätzlich wird es auch jeweils ein kurzes Video mit Erklärungen und ggf. Hinweisen geben (ebenfalls im Moodle)
- Abgabe der Lösungen i.A. bis zum jeweils nächsten Freitag, 10:00 (Abgabedatum steht immer auf dem Aufgabenblatt)
- am darauf folgenden Montag werden im Moodle Videos bereitgestellt, in denen ich die Lösungen erkläre



# Aufgabenblätter

Die Aufgabenblätter enthalten zwei Typen von Aufgaben:

Programmieraufgaben

Theoriaufgaben

# Aufgabenblätter

Die Aufgabenblätter enthalten zwei Typen von Aufgaben:

## Programmieraufgaben

Schreibt ein kleines Programm zur Lösung eines gegebenen Problems!

- das Programm muss in Python 3.x geschrieben werden
- weitere Einschränkungen zur Nutzung von Bibliotheken etc. werden in den Aufgaben genannt
- der Code muss wohl strukturiert, nachvollziehbar und kommentiert sein

## Theoriaufgaben

# Aufgabenblätter

Die Aufgabenblätter enthalten zwei Typen von Aufgaben:

## Programmieraufgaben

## Theoriaufgaben

- typischerweise Rechnungen, Beweise, etc.
- zur Lösung müsst ihr stets nicht nur das Ergebnis sondern auch den Rechenweg nachvollziehbar und sauber darlegen

# Abgaben

Die Abgabe erfolgt in Gruppen von bis zu 3 Studis.

- Termin für die Abgabe ist i.A. jeweils freitags um 10:00
- später eingesendete Abgaben werden nicht berücksichtigt
- es werden nur Lösungen zu komplett gelösten Aufgaben abgegeben

Programmieraufgaben

Theoriaufgaben



# Abgaben

Die Abgabe erfolgt in Gruppen von bis zu 3 Studis.

- Termin für die Abgabe ist i.A. jeweils freitags um 10:00
- später eingesendete Abgaben werden nicht berücksichtigt
- es werden nur Lösungen zu komplett gelösten Aufgaben abgegeben

## Programmieraufgaben

- eine .py-Datei je Aufgabe
- achtet darauf, dass man den Code ausführen kann

## Theoriaufgaben

# Abgaben

Die Abgabe erfolgt in Gruppen von bis zu 3 Studis.

- Termin für die Abgabe ist i.A. jeweils freitags um 10:00
- später eingesendete Abgaben werden nicht berücksichtigt
- es werden nur Lösungen zu komplett gelösten Aufgaben abgegeben

## Programmieraufgaben

## Theoriaufgaben

- Abgaben bitte nur als pdf! Vor allem keine Word-Dateien!
- ihr könnt eure Lösungen handschriftlich verfassen und scannen/fotografieren oder digital erstellen (z.B. mit  $\text{\LaTeX}$ )

# Fragen/Probleme

Möglichkeit 1

Möglichkeit 2

# Fragen/Probleme

## Möglichkeit 1

Schreibt in den E-Mailtext eurer Abgabe (nicht ins pdf oder die py-Dateien):

- Fragen, die ihr zu den Aufgaben habt
- Probleme, wegen derer ihr die Aufgaben nicht lösen konntet
- Sachen, auf die bei der Präsentation der Lösung nochmal gesondert eingegangen werden soll

Ich werde in den Lösungsvideos auf die Anliegen eingehen.

## Möglichkeit 2

## Möglichkeit 1

## Möglichkeit 2

- wir machen optionale Zoom-Meetings in den Übungszeiten
- das erste wird am 27. April ab 12:00 stattfinden
- es wird vermutlich nicht jede Woche ein solches Meeting stattfinden

# Punkt sammeln

Zum Bestehen der Übung braucht ihr am Ende:

- 50% aller Punkte aus den Programmieraufgaben
- 50% aller Punkte aus den Theorieaufgaben

## Punkte sammeln

- ihr bekommt alle Punkte für alle Aufgaben, deren Lösung ihr abgibt
- daher nur vollständig gelöste Aufgaben abgeben
- ich kontrolliere bei allen Abgaben Vollständigkeit und Korrektheit grob und ...
- ... bei jeweils einer Aufgabe pro Gruppe im Detail

# Scheinkriterien

- regelmäßige und aktive Teilnahme an den Übungen
- ihr braucht 50% aller Punkte aus den Programmieraufgaben
- ihr braucht 50% aller Punkte aus den Theorieaufgaben
- es muss auf jedem Aufgabenblatt mindestens eine Aufgabe gelöst werden
- das Abgeben von offensichtlichen Teillösungen, offensichtlich falschen Lösungen oder Plagiaten kann zum Nichtbestehen führen
- die Abgaben müssen rechtzeitig erfolgen (siehe jeweiliges Aufgabenblatt)

# Gruppenbildung

Sucht euch Gruppenmitglieder unter den Teilnehmern! → Nutzt dafür Moodle, dort könnt ihr die Teilnehmer sehen.

Falls ihr niemanden kennt oder anschreiben wollt:

- schickt mir bis Freitag (24.04.2020) 7:00 eine Mail
- ich nehme dann die Zuordnung in Gruppen vor
- ihr bekommt am Freitagvormittag eine entsprechende Nachricht von mir
- dies gilt auch für 2er-Gruppen, die gerne noch eine dritte Person aufnehmen wollen

Nutzt zur Kommunikation und zum kollaborativen Arbeiten Tools wie Zoom, MS Teams, Git, ...