

Bildverarbeitung Praktikum

Übung 1

Dr. Christina Gillmann

April 15, 2020

1 Extraktion und Darstellung von Farbkanälen

In Python haben sie die Möglichkeit mehrere Bilder nebeneinander oder untereinander in einem Subplot darzustellen. Dazu gibt es die Funktion

```
1 fig , axs = plt.subplots(x, y, figsize=(v,w))
```

Dabei gibt x die Anzahl der Bilder in einer Reihe und y die Anzahl der Bilder in einer Spalte an. v und w entscheiden über die Größe der eingefügten Bilder.

a) Laden sie ein beliebiges Bild und erstellen 3 Bilder, die jeweils nur den rot, grün und blau Kanal des Bildes enthalten. Zeigen sie das Originalbild und die Bilder der drei Farbkanäle in dem Subplot an. Sie müssen dazu in den Subplots eine entsprechende Colormap verwenden. Als beispiel sehen sie folgendes Beispiel, dass eine blaue Colormap verwendet. Blues, kann dabei durch Reds und Greens ersetzt werden.

```
1 axs[0].imshow(blueImage, origin='upper', cmap='Blues')
```

b) Erstellen sie drei neue Bilder, die den Cyan, Magenta und Yellow Kanal ihres gewählten Bildes enthalten. Das erstellen neuer Bilder in Python entspricht dabei der Erstellung eines neuen Numpy Arrays. Dies erreichen sie wie folgt:

```
1 result = np.full((x, y), value)
```

Dabei wird ein Array mit x mal y Werten erstellt indem jeweils der Wert value gespeichert wird.

Um die richtige Berechnung durchzuführen müssen sie die Formeln aus den Folien und eine Funktion von numpy namens *subtract* benutzen. Beachten sie dabei, dass ihr Werte Bereich nicht von 0 bis 1, sondern von 0 bis 255 angeben ist.

2 Box Filter

Für diese Aufgabe benötigen sie eine weitere Bibliothek in Python, die sie durch die Anaconda Shell wie folgt laden können.

```
1 conda install -c conda-forge opencv
```

OpenCV ist eine Bibliothek, die ihnen sehr viele Möglichkeiten der Bildverarbeitung liefert. Unter anderem die Funktion

```
1 result = cv2.filter2D(img, -1, kernel)
```

Hiermit können sie einen beliebigen Kernel auf ein Bild anwenden. Hierzu müssen sie sich einen Kernel erstellen. Dies können sie durch ein neues Numpy Array erreichen, wie auch in Aufgabe 1.

a) Erstellen sie einen Box Kernel (Größe 5x5) und wenden sie ihn auf ein beliebiges Bild an. Vergessen sie die Normalisierung nicht. b) Manipulieren sie den Wert des Zentralen Eintrags in ihrem Kernel. Nutzen sie dabei immer größere Werte. (Achten sie darauf, dass sie auch hier entsprechend normalisieren müssen. Wenn sie den Wert um 5 erhöhen, muss ihre Normalisierung entsprechend auch um 5 Erhöht werden). Was fällt bei beliebig großen Werten auf?