

Bildverarbeitung Praktikum

Übung 2, Abgabe 07.05.2020

Dr. Christina Gillmann

April 23, 2020

1 Implementierung des Gaußfilters

In Python haben sie die Möglichkeit wie bei allen anderen Programmiersprachen Funktionen zu erstellen. Diese können direkt in ihren Programmcode eingefügt werden. Eine kurze Zusammenfassung finden sie hier:

<https://pythonbuch.com/funktion.html>

Sie sollen eine Funktion schreiben, die ihnen einen Gaußfilter mit beliebiger Größe (Input Parameter der Funktion) erstellt.

Wie schon im vorherigen Übungsblatt können sie die Funktion `filter2D` von `openCV` nutzen um verschiedene Filter auf ein Bild anzuwenden.

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

- a) Schreiben sie eine Funktion, die ihnen beliebig große Gaußfilter erzeugt.
- b) Nutzen sie ihre Funktion um Gaußfilter der Größe 3, 7, und 15 auf ein beliebiges Bild anzuwenden. Was fällt ihnen auf?

2 Kantendetektion

In der Vorlesung haben wir verschiedene Ansätze zur Kantendetektion ausprobiert. Diese sollen sie nun an einem beliebigen Bild ausprobieren (am besten alle auf dem gleichen Bild. Erstellen sie dazu einen Plot, der mehrere Subplots erhält. Somit können sie die Ergebnisse direkt vergleichen.

- a) Implementieren sie den Laplace Kantendetektor.
- b) Implementieren sie den Sober Kantendetektor. Dazu müssen sie erst den Kantendetektor in x und dann in y berechnen und dann den Betrag aus beiden Ergebnissen.
- c) Implementieren sie den Canny-Edge-Detektor. Hierzu finden sie eine Funktion in `openCV` https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html
- d) Implementieren sie den LoG Filter.
- e) Implementieren sie den DoG Filter.
- f) Überlegen sie sich welche Vor und Nachteile die jeweiligen Kantendetektoren haben.

3 Jacobi Matrix

Schreiben sie einen Kantendetektionsalgorithmus auf Grundlage der Jacobi-Matrix. Nutzen sie dazu ein Graubild. Berechnen sie eine JacobiMatrix für jeden Punkt ihres Bildes und daraus die Eigenwerte der Jacobimatrix. Dies erreichen sie mit

```
1 ew, ev = linalg.eig(A)
```

welches sie durch

```
1 import numpy.linalg as linalg
```

importieren können.

a) Detektieren sie Grauwerte in ihrem Bild, die sich nur in eine Richtung ändern.

b) Detektieren sie Grauwerte in ihrem Bild, die sich in zwei Richtungen ändern.

Zeigen sie alle Ergebnisse in einem Plot an.