Institut für Informationsverarbeitung (TNT)
Appelstraße 9a, 30167 Hannover
www.tnt.uni-hannover.de

Prof. Dr.-Ing. Bodo Rosenhahn
Tom Wehrbein, M.Sc.

WiSe 2025/26 **28.10.2025**

Praktische Übung: Computer Vision für medizinische und industrielle Anwendungen

Versuch 1: Lokale Operatoren

Allgemeine Hinweise

- Aufgaben, die mit einem Stern (*) gekennzeichnet sind, sind optional und vertiefen vorhergehende Aufgaben oder dienen zur weiteren Einarbeitung in MATLAB.
- Sobald Sie mit allen Aufgaben fertig sind, melden Sie sich bei Ihrem Betreuer, um die Aufgaben abnehmen zu lassen.
- Zur Abgabe der Aufgaben schicken Sie alle m-Files (keine mlx-Files!), sowie eine stichpunktartige Beantwortung der Kontrollfragen (z.B. als txt-Datei) in einem zip-Archiv `GruppeX_VersuchY.zip` an

labormatlab@tnt.uni-hannover.de

Aufgabe 1 - eindimensionale Faltung

a) Das Skript `faltung1D_test.m` bietet bereits die Möglichkeit über eine Faltung die Ableitung einer Funktion zu approximieren und darzustellen. Die Faltung wird mit dem MATLAB-eigenen Befehl `conv` durchgeführt. Welche Auswirkung hat der Parameter `'same'`? Welche Varianten sind ebenfalls implementiert?

b) Testen Sie weitere Filterkerne, die ebenfalls die Ableitung approximieren:
1. Ableitung: $\frac{1}{\Delta x}$`[1,-1,0]`, $\frac{1}{\Delta x}$`[0,1,-1]`, $\frac{1}{\Delta x}$`[0.5, 0, -0.5]`
2. Ableitung: $\frac{1}{\Delta x}$`[1,-2,1]`

Welche Unterschiede bestehen zwischen den Filtern? Wofür könnten die verschiedenen Filter genutzt werden?

c) Welche Randbehandlung verwendet die Funktion `conv`? Implementieren Sie eine eigene Randbehandlung, indem Sie den Vektor $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ für einen Filter der Länge $2m + 1$ anpassen, z.B.

- Zero padding:
$$\mathbf{x}^{padded} = (\underbrace{0, \ldots, 0}_{m-fach}, x_1, \ldots, x_n, \underbrace{0, \ldots, 0}_{m-fach})$$

- Spiegelung:
$$\mathbf{x}^{padded} = (x_m, \ldots, x_1, x_1, \ldots, x_n, x_n, \ldots, x_{n-m+1})$$

Hier werden manuell Werte am Rand hinzugefügt, auf die bei der Faltung zugegriffen werden kann. Achten Sie darauf beim Ergebnis überzählige Werte wieder zu entfernen oder die Option `'same'` geeignet anzupassen. Was sind Vor- und Nachteile der verschiedenen Varianten?

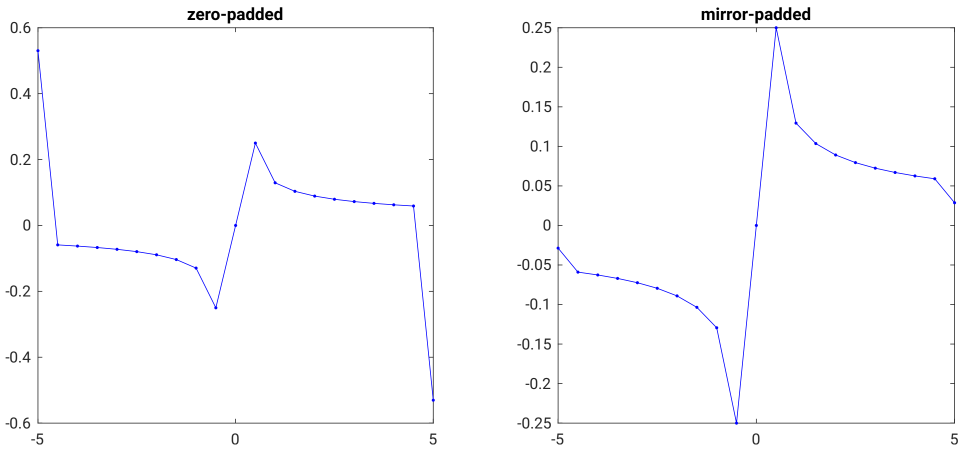

Abbildung 1: Beispielhafte Visualisierung der unterschiedlichen Randbehandlungen

Aufgabe 2 - zweidimensionale Faltung

a) Implementieren Sie einen Gauss-Filter zur Glättung von Bildern (`gauss_filter.m`), das als Eingabe ein Bild und die Standardabweichung σ erhält.

Erstellen Sie zunächst ein zweidimensionales Filter mit der Impulsantwort

$$h(x,y) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\},$$

indem Sie die Funktion h an geeigneten Stützstellen (x,y) um den Nullpunkt $(0,0)$ auswerten (Matlab: `meshgrid`). Beachten Sie, dass Pixel üblicherweise den Abstand 1 haben und dies auch für die Stützstellen gelten muss. Die Impulsantwort ist nach 3σ ausreichend klein, um vernachlässigt zu werden. Korrekt umgesetzt ist das Filter eine symmetrische Matrix, deren Einträge sich in etwa zu 1 addieren.

Nutzen Sie anschließend `conv2` oder `imfilter`. Welche Unterschiede bestehen zwischen den Funktionen?

Erstellen Sie ein Testscript `faltung2D_test`, das ein Bild einliest, glättet und das Ergebnis anzeigt. Beachten Sie die Variablentypen, die in Matlab nicht explizit angegeben werden müssen. Ein Bild hat typischerweise den Typ `uint8`, zur Faltung müssen die Werte aber als Gleitkommazahlen gegeben sein (z.B. `double`). Matlab geht davon aus, dass ein Bild mit `double`-Werten den Wertebereich $[0,1]$ hat. Zur Darstellung mit `imshow` muss das Bild entsprechend angepasst werden, zurück zu `uint8` gecastet werden oder die Parameter in der Funktion entsprechend gesetzt werden. Alternativ kann zur Darstellung `image` oder `imagesc` verwendet werden.

Abbildung 2: Beispielhaftes Ergebnis einer Glättung mit Gauss-Filter

b) Die zweidimensionale Faltung ist relativ rechenintensiv. Nutzen Sie die Separierbarkeit des Gauss-Filters

$$\begin{aligned} h(x,y) &= \frac{1}{2\pi\sigma^2}\exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\} \\ &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{x^2}{2\sigma^2}\right\}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{y^2}{2\sigma^2}\right\} \\ &= h(x)\cdot h(y), \end{aligned}$$

um die Glättung durch die Anwendung zweier eindimensionaler Gauss-Filter $h(x) = h(y)$ durchzuführen. Implementieren Sie eine Funktion `gaussFilterSep.m` in der Sie zunächst das Bild in x-Richtung und anschließend mit dem transponierten Filter in y-Richtung falten.

Vergleichen Sie die Laufzeiten der beiden Methoden, wenn Sie "größere" Bilder oder Filter verwenden (MATLAB: `tic, toc`). Wählen Sie hierzu ein größeres σ oder erstellen Sie sich eine zufällige Matrix als Bildersatz (MATLAB: `rand`).

Aufgabe 3 - Harris-Corner-Detektor

a) Vervollständigen Sie die Funktion `harrisCorner.m`, die Sie Schritt für Schritt durch die Implementierung eines Harris-Corner-Detektors führt.

b) Testen Sie Ihre Funktion an den gegebenen Bildern (`harris_test.m`). Probieren Sie auch verschiedene Parameter aus. Lassen sich in dem Beispiel `rectangles.png` alle Ecken finden?

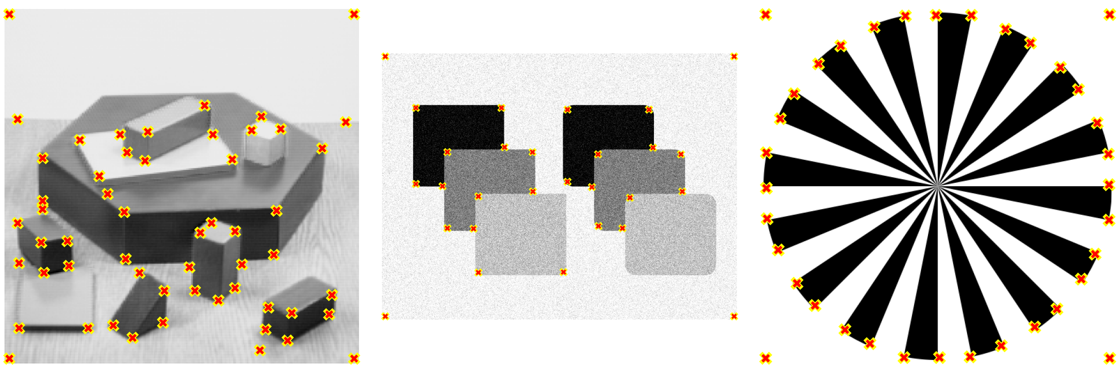

Abbildung 3: Beispielhafte Detektion von Ecken auf verschiedenen Bildern

Kontrollfragen

a) Welche Unterschiede bestehen zwischen den Parametern `'same'`,`'full'`,`'valid'` in den Funktionen `conv` und `conv2`? Was bedeutet das für die Faltung?

b) Nennen Sie mindestens eine weitere Möglichkeit zur Randbehandlung, die nicht in Aufgabe 1 genannt ist und einen Vorteil gegenüber den genannten Varianten.

c) Was ist der wesentliche Unterschied zwischen Glättungs- und Ableitungsfiltern?

d) Welche Möglichkeiten gibt es, bei der Verwendung des Harris-Corner-Detektors Einfluss auf die Wahl der detektierten Ecken zu nehmen?