

# Industrielle Bildverarbeitung (SS 2022)

Sebastian Edelmann (Hochschule RheinMain)

Praktikum 3 am 23. Mai 2022 / Abgabe: 05. Juni 2022

## Aufgabe 1: Farbbildverarbeitung (8 Punkte)

Schreiben Sie ein Skript 'Park.m', in dem Sie zunächst das Bild 'ParkGraySmall.png' laden und es anzeigen. Sie sehen einen schönen Park mit grauen Himmel, den Sie in dieser Aufgabe durch einen schönen blauen Himmel ersetzen sollen.

1. Wählen Sie dazu mit der Maus einzelne Pixel des grauen Himmels aus (ca. 5-10 reichen hier schon) und speichern die Positionen ab. Hierfür steht Ihnen die Matlab-Funktion 'getpts' zur Verfügung. Durch Drücken der linken Maustaste wählen Sie die Punkte aus und durch Drücken der rechten Maustaste beenden Sie die Eingabe (Hinweis: die Position beim Beenden der Eingabe wird mit übernommen). Verwenden Sie für die interaktive Eingabe folgende Zeilen: `'h=figure; [x,y]=getpts(h);'`. Da `x` und `y` allerdings double-Werte sind, wandeln Sie diese in uint32 um.
2. Bestimmen Sie für jeden Farbkanal jeweils das Minimum und das Maximum Ihrer ausgewählten Punkte.
3. Ersetzen Sie nun den grauen durch einen blauen Himmel (z.B. [100,200,250] oder einer anderen blauen Färbung). Da Ihre Stichprobe (von ca. 5-10 Pixel) nicht alle Punkte des grauen Himmels enthält, werden die ermittelten Farbbereiche des R-, G- und B-Kanals von  $[Min, \dots, Max]$  auf z.B.  $[0, 9 \cdot Min, \dots, 1, 1 \cdot Max]$  des jeweiligen Farbkanals erweitert.
4. Beachten Sie die grauen Haare der älteren Dame :-)



Abbildung 1: Park mit grauem Himmel und zwei Varianten mit blauem Himmel.

## Aufgabe 2: Local Thresholding (6 Punkte)

Gegeben ist ein Textauszug, der aufgrund eines Kopiererefehlers durch sinusförmigen Verdunklungen gestört ist. Ihre Aufgabe ist es, diese Störungen zu beseitigen und den Text gut lesbar darzustellen, (s. Abb. 1).

1. Schreiben Sie ein Skript 'TextLocalThresh.m', in dem Sie zunächst das Bild 'sine\_shaded\_text\_small.png' laden. Nun soll für jedes Pixel entschieden werden, ob es Text oder Hintergrund ist. Dazu betrachten Sie für jedes zu klassifizierende Pixel eine Umgebung von insgesamt 21x21 Pixel (10 Pixel nach jeder Seite). Bestimmen Sie für jede Umgebung den Mittelwert  $m$  und die Standardabweichung  $s$  und bestimmen daraus den lokalen Schwellwert  $T(u, v)$  gemäß  $T(u, v) = m + k \cdot s$ ,  $k \in [-1, 0]$ . Pixelwerte, die kleiner als der lokale Schwellwert sind, sollen als Schrift klassifiziert werden und Pixelwerte, die größer sind, als Hintergrund. Variieren Sie den Parameter  $k$  und finden den optimalen Wert  $k_{opt}$

Randeffekte können vernachlässigt werden, d.h., dass erste Pixel, das klassifiziert werden soll, liegt in Zeile 11 und Spalte 11. Entsprechend können Sie „früher“ aufhören.

2. Das Ergebnis aus Aufgabenteil 1 soll mittels einer morphologischen Operationen verbessert werden. Überlegen Sie sich, welche hierfür geeignet ist. Kleine Hilfe: verwenden Sie als Strukturelement  $SE = \begin{bmatrix} 0 & 1 & 0; & 1 & 1 & 1, & 0 & 1 & 0 \end{bmatrix}$ .



Abbildung 2: Gestörter Text (li), mit globaler und lokaler Schwelle (Mitte, re).

### Aufgabe 3: Überblenden (Mischen) zweier Farbbilder (6 Punkte)

Schreiben Sie ein Skript 'AlphaBlendingFarbe.m', in dem Sie zunächst die beiden Bilder 'apple1.jpg' und 'orange1.jpg' laden und anzeigen. Es soll nun die rechte Hälfte des Apfels durch die rechte Hälfte der Orange ersetzt werden. Um einen fließenden Übergang zu erzeugen, verwenden Sie für jede Bildzeile des Apfelbildes bzw. Orangenbildes jeweils den folgenden  $\alpha$ -Verlauf:

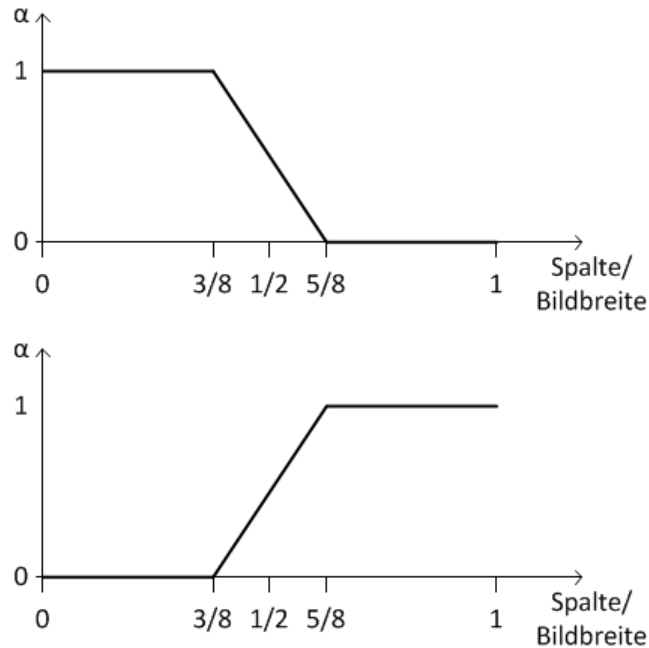


Abbildung 3:  $\alpha$ -Verläufe für Apfel (oben) und Orange (unten).

Vorschlag: Wandeln Sie zunächst die beiden Farbbilder in Grauwertbilder und überblenden diese. Falls das geklappt hat, gehen Sie zu den Farbbildern über und wenden die Überblendung auf jeden Farbkanal an.