

Übung 6: Bildfilter/DCT

Einleitung: Für die Übung benötigt ihr wieder eure eigenen Bildaufnahmen (Detail und Fläche).

Aufgabe 1: Mittelwertfilter (Tiefpass)

a. Lies die Bilder ein und konvertiere sie zunächst mit der Funktion `rgb2gray` in Graustufenbilder. Erweitere das Skript nun so, dass ein Filter mit einem Kernel von 3x3 Pixeln angewandt werden kann:

```
RGB = imread('my_pic.bmp');
I = rgb2gray(RGB);
%H ist der 3x3 Filter-Kernel, hier ein Unity-Filter, das das Bild
%unverändert lässt
H = [0 0 0; 0 1 0; 0 0 0];
Y = filter2(H,I,'full');
```

Beim Mittelwertfilter wird ein Zielpixel aus dem Mittelwert aller Pixel im Kernel im Originalbild berechnet. Beschreibe die Wirkung des Filters auf deine Bilder!

b. Wie verfährt man bei der Berechnung des Filters mit den Randpixeln?

c. Erweitere das Skript so, dass das Differenzbild zwischen dem gefilterten und dem Originalbild berechnet werden kann! Dieses muss offensichtlich die Information zeigen, die das Filter entfernt hat. Beachte dabei, dass du das Differenzbild in den Bereich 0-255 verschieben und möglicherweise im Kontrast verstärken musst (siehe Übung 6).

Aufgabe 2: Gradientenfilter

a. Wende nun den folgenden 3*3 Kernel auf deine Bilder an! Beachte dabei, dass bei der Anwendung der Vorschrift Werte < 0 oder > 255 entstehen können und entsprechend behandelt werden müssen!

0	-2	0
-2	12	-2
0	-2	0

3*3 Filterkernel

b. Berechne wiederum das Differenzbild zwischen dem gefilterten und dem Originalbild! Beachte dabei ebenfalls, dass du das Differenzbild möglicherweise im Kontrast verstärken musst.

Aufgabe 3: DCT

Die DCT transformiert ein Pixelbild in seine Ortsfrequenzen. Berechne die DCT für deine beiden Bilder, indem du das komplette Bild nach Anwendung von `rgb2gray` mit der Funktion `dct2` in den Ortsfrequenzbereich transformierst und stelle die DCT-Koeffizienten wieder als Graustufenbild dar. Dazu müssen die Werte der DCT-Koeffizienten ggf. skaliert und wieder in den Bereich von 0 bis 255 verschoben werden.

Erkläre das Aussehen des Bildes der Transformation an Hand der Strukturen in deinen Bildern. Nun setze im Ergebnis der DCT, d.h. der Matrix der DCT-Koeffizienten, Werte mit einem Index (x,y)

a) > (128,128)

b) > (64,64)

c) > (16,16) auf 0 und transformiere die modifizierte Matrix der DCT Koeffizienten wieder in den Pixelbereich, indem du `idct2` verwendest. Wie verändert sich dein ursprüngliches Bild? Erkläre,

was die genannte Operation prinzipiell für einen Effekt hat, indem du die Bedeutung der DCT Koeffizienten interpretierst.

Ins Protokoll: Beschreibung zu den Unterpunkten, entstehende Bilder und Codeausschnitte.