

Übung Computergrafik 2

Abgabetermin:

Die Lösung zu diesem Übungsblatt ist bis zum **20.06.2012** abzugeben.

Form der Abgabe:

- Die Übungen können in Gruppen von 2–3 Studenten bearbeitet werden.
- Die Abgaben bestehen aus den Python-Quelltexten samt aller Bilder, die für die Bearbeitung der Übung verwendet wurden. Idealerweise wird das gesamte PyDev-Projektverzeichnis für die jeweilige Übung über Eclipse als .zip Archiv exportiert. Wichtig: Zur Vermeidung von Namenskonflikten in Eclipse, euren Nachnamen bitte als Präfix vor die Namen der die PyDev-Projekte setzen, z.B.: „müller-übung-x“.
- Teilaufgaben, deren Python-Dateien wegen Syntaxfehlern nicht ausführbar sind, werden nicht weiter korrigiert.
- Textaufgaben sollten in Form einer .doc, .odt oder (bevorzugt) als PDF-Datei abgegeben werden, und sollten (falls erforderlich) die benötigten Ausgabebilder der Aufgaben enthalten.
- Alle Übungsabgaben erfolgen über UniWorX¹.

Inhalt:

In diesem Übungsblatt behandeln wir die Korrelation im Frequenzraum, sowie das Anwenden von Filtern im Frequenzraum und den Umgang mit Rauschen bei der Dekonvolution (Anwendung des Wiener-Filters).

¹<https://uniworx.ifi.lmu.de>

Aufgabe 1 Korrelation im Ortsraum und im Frequenzraum (**)

Sie entwickeln eine mobile Applikation, deren Ziel es ist, Menschen mit Sehbehinderung beim Einkauf zu unterstützen. Konkret soll die Applikation in der Lage sein, mit Hilfe der Kamera des mobilen Geräts bestimmte Produkte in Supermarktregalen zu identifizieren.

Zum Prototyping des letztlich in der Applikation eingesetzten Computer Vision Algorithmus verwenden Sie Python und Numpy.

- a) Implementieren Sie das Ortsraum-Korrelationsverfahren aus der Vorlesung (Folien 6, 33ff., Korrelation im Ortsraum), und versuchen Sie Instanzen der in **shelf-model{1-3}.pgm** abgebildeten Produkte im Supermarktregal **shelf.pgm**² wiederzufinden. Geben Sie dazu die Werte von $cc_{f,m}$ als Bild aus

```
1 plt.imshow(cc, cmap=plt.get_cmap('jet'))
```

Geben Sie auch eine Variante dieses Bildes aus, bei der alle Werte, die kleiner als 75% des Maximalwerts sind, auf 0 gesetzt sind:

```
cc2 = cc.copy()
cc2[cc2 < 0.75 * np.max(cc2)] = 0.0
plt.subplot(248)
4 plt.imshow(cc2, cmap=plt.get_cmap('jet'))
```

Bestandteil der Abgabe sind der Quellcode und die Ausgabebilder.

- b) Implementieren Sie nun das Frequenzraum-Korrelationsverfahren aus der Vorlesung (Folien 6, 38ff, Korrelation im Frequenzraum). Gehen Sie analog zum obigen Aufgabenteil vor und vergleichen Sie die Ergebnisse. Bestandteil der Abgabe sind der Quellcode, die Ausgabebilder und ein kurzer Vergleich der Ergebnisse.

Hinweise zur Bearbeitung dieser Aufgabe:

- PGM ist ein einfaches ASCII-Bildformat ohne Komprimierung. Laden Sie die PGM-Bilder mit **pgm.py**. Dazu muss dieses Modul importiert werden:

```
1 import pgm
...
f = pgm.imread('shelf.pgm')
```

- Zero-Padding kann bei Ausführung der Numpy `fft2`-Funktion durch Setzen eines entsprechenden Arguments automatisch hinzugefügt werden. Zero-Padding ist jedoch nicht zwingend erforderlich.

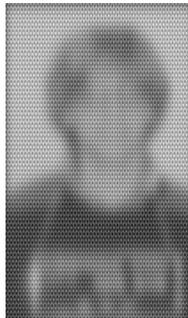
```
h,w = f.shape
2 F = np.fft.fft2(f, (2*h,2*w))
```

- Verwenden Sie, wann immer möglich, Numpy-Array-Operationen, um eine akzeptable Performanz zu erreichen. Erinnerung: Zwei Numpy-Arrays lassen sich durch die Operatoren `*` und `/` *elementweise* multiplizieren bzw. dividieren. D.h. bei der Anwendung von Filtern im Frequenzraum ist es nicht notwendig explizit Schleifen für diese Operationen zu implementieren.

²Abbildung Supermarktregal: (c) Ralf Roletschek (Wikipedia User:Marcela), CC-BY-SA.

Aufgabe 2 Filtern im Frequenzraum (☆☆)

Ein prominenter Programmierer hat versucht, ein eher unangenehmes Bild von sich durch zwei Operationen im Frequenzraum zu „verschlüsseln“ (**wholsThis.pgm**):



- a) Probieren Sie, die Identität der abgebildeten Person festzustellen, indem Sie die auf das Bild angewandten Operationen im Frequenzraum rückgängig machen. Konvertieren Sie dazu das „verschlüsselte“ Bild zunächst in den Frequenzraum und inspizieren Sie das Spektrum. Überlegen Sie sich anhand Ihrer Analyse, welche Filter und Operationen im Frequenzraum auf das Bild möglicherweise angewendet wurden, und versuchen Sie diese im Frequenzraum mit anschließender Konvertierung in den Ortsraum rückgängig zu machen, um ein möglichst scharfes und störungsfreies Bild zu erhalten. Bestandteil der Abgabe sind der Quellcode zum Aufheben der Bildstörungen, sowie Bilder, die die Schritte beim Entfernen der Artefakte zeigen.

Hinweise zur Bearbeitung dieser Aufgabe:

- Laden Sie das Bild wieder mit **pgm.py**.
- Verwenden Sie diesmal kein Zero-Padding bei der FFT, also

```
F = np.fft.fft2(f)
```
- Der „Hebewert“ a des in dem Bild verwendeten Tiefpass-Gauß-Filters $H_a(u, v) = a + (1 - a)H(u, v)$ beträgt $a = 0.001$, wobei $H(u, v) = \exp(-\frac{D^2(u, v)}{2D_0^2})$. Den Parameter D_0 müssen Sie durch Ausprobieren ermitteln.

Aufgabe 3 Bildverbesserung bei Rauschen – das Wiener Filter (☆☆)

Ein Journalist im Auslandseinsatz hat einen wichtigen Schriftzug fotografiert und wollte ihn so schnell wie möglich an seine Redaktion zurückschicken. Leider hat er dabei „versehentlich“ einen Gaußschen Tiefpass-Filter auf das Bild angewendet und überdies gab es Probleme mit der Übertragungstechnik, so dass das Bild mit einem leichten gleichmäßig verteilten Rauschen überlagert worden ist (**Schriftzug.pgm**). Es ist bekannt, dass das Filter einen „Hebewert“ $a = 0,001$ hat (siehe Hinweise zur vorigen Aufgabe). Zum Glück hat der Journalist noch den Durchmesser des ausgeführten Gaußfilters in seiner Bildverarbeitungssoftware wiedergefunden: $D_0 = 20$ Pixel. Helfen Sie der Redaktion den wichtigen Schriftzug zu entziffern, indem Sie verschiedene aus der Vorlesung bekannte Verfahren anwenden.

- a) Probieren Sie zunächst ein einfaches inverses Filter (Folien 6, 69) $H_a^I(u, v) = 1/H_a(u, v)$. Variieren Sie D_0 leicht und betrachten Sie das Ergebnis. Bestandteil der Abgabe ist das Resultatbild des inversen Filters.

- b) Wie in der Vorlesung beschrieben, lassen sich bessere Ergebnisse erzielen, wenn der Radius des inversen Filters auf einen bestimmten Maximalwert beschränkt wird (Folien 6, 73), da dies die Auswirkungen des Rauschens bei hohen Frequenzen reduziert. Um dies zu erreichen, multiplizieren Sie das inverse Filter $H_a^I(u, v)$ zunächst mit einem Butterworth-Tiefpass-Filter (BLPF, n, D_0) der Ordnung $n = 10$ (Folien 6, 14), bevor Sie das Resultat auf das Ausgangsbild anwenden. Experimentieren Sie mit verschiedenen Werten D_0 für das BLPF. Beobachten Sie, wie sich das Resultatbild mit D_0 verändert. Geben Sie in der Abgabe an, bei welchem D_0 das beste Resultat erzielt wird, sowie das dazugehörige Resultatbild.
- c) Wenden Sie nun das aus der Vorlesung bekannte heuristische Wiener Filter (Folien 6, 75) auf das vom Journalisten erhaltene Bild an. Probieren Sie dazu mehrere Größenordnungen von K aus ($K = 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, \dots$), bis Sie ein entzifferbares Ergebnis erhalten. Geben Sie das K für das beste Ergebnis, das dazugehörige Ergebnisbild, sowie den entzifferten Text an.

Hinweise zur Bearbeitung dieser Aufgabe:

- Verwenden Sie kein Zero-Padding bei der FFT, um die Berechnungen zu beschleunigen.
- Empfehlung: Erhöhen Sie D_0 in Fünfer-Schritten – Werte zwischen $50 < D_0 < 100$ sollten hilfreiche Resultate liefern.
- Erinnerung: $|H(u, v)|^2 = H(u, v)H^*(u, v) = \text{re}(H(u, v))^2 + \text{im}(H(u, v))^2$.