

Übungsaufgaben zur Vorlesung "Bildverarbeitung"

Bernd Neumann

WS 07

Blatt 5

1. Bei der Archivierung von Schriftstücken werden Pixel als "Schrift" oder "Hintergrund" klassifiziert. Bei "Schrift" werden die Farben schwarz, rot, gelb, grün, blau unterschieden. Aufgrund langjähriger Statistiken weiß man,

- dass ein Schriftstück im Mittel zu 10% mit Schrift bedeckt ist,
- die Farben mit den folgenden Häufigkeiten auftreten:

schwarz	80%
rot	5%
gelb	1%
grün	2%
blau	12%

- Welche Entropie haben die Schriftstücke?
- Entwerfen Sie einen Huffman-Code für die Pixel.
- Was ist die mittlere Codewortlänge?
- Welche Redundanz hat der folgende 4-Bit-Code:

Hintergrund	0000
schwarz	1000
rot	1001
gelb	1010
grün	1011
blau	1100

(4 Punkte)

2. Wenden Sie die Karhunen-Loève-Transformation an, um Bilder mit 4 Pixeln x_1 x_2 x_3 x_4 mit 3 Werten y_1 y_2 y_3 zu kodieren und daraus mit möglichst geringem Fehler zu rekonstruieren. Von den Bildern sind die folgenden statistischen Daten bekannt:

Mittelwerte: $m_1 = m_2 = m_3 = m_4 = 0$

Kovarianzmatrix: $\frac{1}{4} * \begin{bmatrix} 15 & 5 & 9 & 3 \\ 5 & 15 & 3 & 9 \\ 9 & 3 & 15 & 5 \\ 3 & 9 & 5 & 15 \end{bmatrix}$

a) Berechnen Sie die 3 x 4 Transformationsmatrix A_3 .

b) Wie groß ist der mittlere quadratische Fehler (MSE) für x_1 bei optimaler Rekonstruktion mit A_3^T ?

(6 Punkte)

b.w.

Nützliche Python-Befehle:

```
from Numeric import *
from LinearAlgebra import *

a = array([[a11, a12, ...],[a21, a22,...],...])
ew, ev = eigenvectors(a)

# ev[i] is the eigenvector belonging to ew[i]
# dot(ev, transpose(ev)) is the matrix multiplication (should here give the identity matrix)

# more documentation for python Numeric can be found in the internet, and by calling
print transpose.__doc__ # likewise for other functions in the Numeric and LinearAlgebra
modules
```