



Image Processing & Understanding

Grundlagen der Bildverarbeitung

Übungsblatt 7

Wintersemester 15/16
AG Bildverarbeitung und Bildverstehen
Prof. Klaus Tönnies,
Tim König, Johannes Steffen

Die Lösungen der Aufgaben werden in den Übungen am **08., 09. und 10.12.2015** besprochen. Votieren Sie am Anfang Ihrer Übung für die Aufgaben, die Sie bearbeitet haben und vorstellen können.

Hinweis: Um die Lösungen der Aufgaben zu überprüfen und zu interpretieren, können Sie geeignete Funktionen mit Matlab/Octave programmieren.

1. Was ist Run length Encoding (RLE) und wieso kann bei RLE-basierter Kompression der Speicheraufwand sogar größer als vor der Kompression sein? Wieso verwendet man bei RLE-Kompression auf Bit-Ebenen den Gray-Code?
2. Gegeben sei das Alphabet $\mathcal{A} = \{a, b, c, d, e, f\}$. Den Symbolen können die Wahrscheinlichkeiten $p(a) = 0.45, p(b) = 0.23, p(c) = 0.12, p(d) = 0.11, p(e) = 0.07$ und $p(f) = 0.02$ zugeordnet werden. Erstellen Sie einen Huffman-Code für \mathcal{A} gemäß der Konvention aus der Vorlesung.
Welches Wort wird durch die Zeichenfolge $x = 011100101010011$ kodiert?
3. Die Zeichenfolge

$x = 31 \ 29 \ 27 \ 25 \ 23 \ 16 \ 16 \ 16 \ 16 \ 17 \ 18 \ 19 \ 20$

soll möglichst effektiv (*hinsichtlich der Komprimierungsrate*) verlustfrei komprimiert werden. Wie würden Sie vorgehen?

Hinweis: Bedenken Sie, dass sich betragsmäßig kleine Zahlen durch weniger Bits darstellen lassen. Bei der Datenkompression ist es daher erstrebenswert, die vorhandenen Daten in „kleine Zahlen“ umzuwandeln.

4. Berechnen Sie die Entropie für Bilder mit folgenden Eigenschaften:

a) das Bild hat folgendes Histogramm:

Grauwert	0	1	2	3
Häufigkeit	50	0	50	100

b) das Bild hat folgendes Histogramm:

Grauwert	0	1	2	3
Häufigkeit	100	0	100	200

c) im Bild kommen alle 256 Grauwerte in gleichem Umfang vor.

5. Geben sei M als ein 4×4 Block eines Grauwertbildes, die diskrete Kosinustransformierte D von M und die Quantisierungsmatrix Q , mit der D quantisiert werden soll (siehe JPEG-Kompression):

$$M = \begin{pmatrix} 139 & 149 & 155 & 155 \\ 150 & 160 & 158 & 156 \\ 159 & 161 & 162 & 155 \\ 162 & 161 & 162 & 157 \end{pmatrix}, D = \begin{pmatrix} 625.2 & -5.1 & -8.8 & 0.2 \\ -16.1 & -11.4 & -2.4 & -2.8 \\ -5.2 & -4 & 1.8 & 1.4 \\ -1.7 & -0.3 & 0.2 & 2.4 \end{pmatrix} \text{ und}$$

$$Q = \begin{pmatrix} 16 & 10 & 24 & 51 \\ 14 & 16 & 40 & 69 \\ 18 & 37 & 68 & 103 \\ 49 & 78 & 103 & 120 \end{pmatrix}$$

Führen Sie die Kompression (wie bei JPEG) weiter, d.h.:

- Berechnen Sie die quantisierte DCT-Koeffizienten-Matrix ($D./Q$)
- Sortieren Sie die Koeffizienten mit aufsteigender Frequenz entlang der Diagonalen (Zig-Zag).
- Geben Sie den finalen Huffman-Kodierten Bit-Vektor C an.