Teilleistung 1

MI-EMI-B

Christian Zeck
Mikhael Glazunov

Aufgabe 1: Einleitung

(a)

Die Augenbewegungen beim Erkennen größerer Objekte laufen wie folgt ab.
Die Augen wechseln stetig zwischen ruckartigen Bewegungen, sogenannten Sakkaden, und Ruhepausen ab. Die Ruhepausen dauern von 250 ms bis zu 2 s an und sind wichtig, damit das Auge auf einen neuen Fixationspunkt scharf stellen kann.

(b)

Für eine Videothek haben sich durch den technischen Fortschritt folgende Veränderungen an der Wertschöpfungskette von Multimediadiensten ergeben:
Im Bereich „Inhalt" existieren nun qualitativ höherwertigere und neuere Produktionsmethoden, wie beispielsweise Animationsfilme und 3D-Versionen. Außerdem werden mehr Medien produziert, die in Videotheken angeboten werden und andere Formate, wie Serien, gewinnen an Einfluss.
Die Verteilung der Medien wird nun hauptsächlich über digitale Netzwerke geregelt, sodass die traditionelle Verbreitung, wie Verleih, stark zurückgegangen ist. Dies führt dazu, dass die Produkte nicht durch ihre physische Anzahl limitiert sind. Sie sind als Download oder auch als Stream beliebig oft und meistens auch jederzeit verfügbar.
Dank der Digitalisierung hat der Nutzer die Möglichkeit Filme einfacher durch WLAN oder USB-Sticks weiterzugeben und es ist eine Betrachtung der Medien auf einer Vielzahl von Geräten möglich.

(c)

Der Satz „Du lässt deine Wohnung ja völlig vorkommen!" wird als Laute durch Schallwellen transportiert. Die Syntax des Satzes entspricht der deutschen Grammatik. Meine Mutter will damit ausdrücken, dass meine Wohnung ungepflegt ist und ich schleunigst mal wieder aufräumen sollte.
Sie benutzt das nicht alltägliche Wort „verkommen", um die grässliche Situation meiner Wohnung zu verdeutlichen.

Aufgabe 2: Kanäle, Codecs und Medien

(a)

Das Abtasttheorem besagt, dass ein Quellsignal mit Frequenz f mit mehr als der doppelten Frequenz f ($2f$) abgetastet werden muss, um eine verlustfreie Digitalisierung zu garantieren.
Das Abtasttheorem gilt außer bei Audio auch beim Scannen von Bildern. Hierbei werden statt der Frequenz die feinste Linie bzw. die einzelnen Pixel zum Berechnen genommen.

(b)

1 Punkt pro cm = 2,54 dpi
(http://de.wikipedia.org/wiki/Punktdichte)

=> $$1\,Punkt\ pro\ mm\ = 25{,}4\,dpi$$

Feinste Linien: 0,01mm

=> $$1\,px\ /\ 0{,}01\,mm = 100\,px\,/\,mm$$

$$100\,px\,/\,mm \cdot 25{,}4\,dpi\,/\,(px\,/\,cm) = 2540\,dpi$$

Abtasttheorem:
mehr als
$$2540\,dpi \cdot 2 = 5080\,dpi$$

Das Millimeterpapier sollte mit mehr als *5080 dpi* , also mit mindestens *5081 dpi*, eingescannt werden.

Bild:
$$1920px \cdot 1080px\ |\ 24cm \cdot 15\,cm$$
Pixel pro cm:

Länge: $1920px \div 24cm = 80\,px\,/\,cm$

Breite: $1080px \div 15cm = 72\,px\,/\,cm$

=> Die Länge bestimmt die Abtastrate, da dort eine höhere Auflösung, herrscht als in der Breite (***`80 px/cm > 72 px/cm`***)

$$80\,px\,/\,cm \cdot 2{,}54\,dpi\,/\,(px\,/\,cm) = 203{,}2\,dpi$$

Abtasttheorem:
mehr als
$$2 \cdot 203{,}2\,dpi = 406{,}4\,dpi$$

Mit der Annahme, dass das Bild 24cm x 15cm groß ist, muss das Bild mit mehr als *508 dpi*, also mit mindestens *509 dpi*, eingescannt werden.

(c)

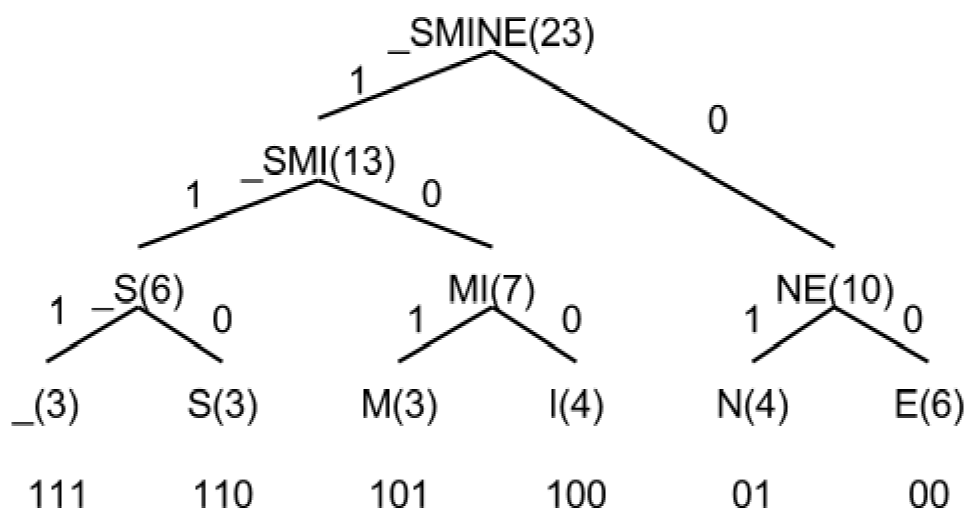
- Bestimmen der Anzahl n der einzelnen Zeichen (mit Leerzeichen “_”) durch nachzählen.
- Berechnung der Auftrittswahrscheinlichkeit $\boldsymbol{p_a}$ der einzelnen Zeichen:

  Anzahl verschiedener Zeichen = 23

  $p_a = n \,/\, 23$
- Berechnung des Entscheidungsgehalts $\boldsymbol{x_a}$ der einzelnen Zeichen:

  $x_a = log_2(1 \div p_a)$
- Berechnung der Entropie:

  $H = \sum_{a \epsilon A} p_a \cdot x_a$
- Herleitung der zugehörigen Huffman-Codierung:



- Berechnung der Durchschnittlichen Wortlänge der Codierung:

  Wortlänge der Codierung eines einzelnen Zeichens:

  $|c(a)|$

  Durchschnittliche Wortlänge:

  $L = \sum_{a \epsilon A} p_a \cdot |c(a)|$
- Berechnung der Redundanz der einzelnen Zeichen:

  R = L – H

	n	p_a	x_a	H	Codierung	|c(a)|	L	R
M	3	0,1304	2,9386	0,3833	**101**	3	0,3913	0,0080
E	6	0,2609	1,9386	0,5057	**00**	2	0,5217	0,0160
I	4	0,1739	2,5236	0,4389	**100**	3	0,5217	0,0829
N	4	0,1739	2,5236	0,4389	**01**	2	0,3478	-0,0911
S	3	0,1304	2,9386	0,3833	**100**	3	0,3913	0,0080
_	3	0,1304	2,9386	0,3833	**111**	3	0,3913	0,0080
	23			**2,5334**			**2,5652**	**0,0318**

Die Codierung ist nicht optimal die Redundanz nicht 0 ist.
Die Huffman-Codierung liefert nur in einem Spezialfall (Werte von ½ x als Wahrscheinlichkeiten) eine optimale Codierung.

Aufgabe 3: Bilder

(a)

Während im RGB-Farbmodell die Farbwerte für die drei Kanäle Rot, Grün und Blau zur Farbdarstellung verwendet werden, spaltet man im YCbCr-Farbmodell die Farben in Luminanz (Helligkeit) und Chrominanz (Farbe) auf.
Da die menschliche Wahrnehmung bei Farben schwächen aufweist, ist in den beiden Chrominanz Kanälen ein großes Komprimierungspotenzial ohne wahrnehmbaren Qualitätsverlust vorhanden. Der Luminanz Kanal wird dabei hochqualitativ abgespeichert, da eine Komprimierung dessen eindeutig vom Menschen wahrgenommen werden.
Im RGB-Farbmodell ist diese Differenzierung nicht möglich, da in allen Kanälen sowohl Helligkeit, als auch Farbe zusammen abgespeichert wird.

Zum Berechnen der YCbCr-Werte wurde folgende Formel benutzt:

$$f_y = 0 + \left(f_R \cdot 0{,}299 + f_G \cdot 0{,}587 + f_B \cdot 0{,}114\right)$$
$$f_{Cb} = 128 + (f_R \cdot (-0{,}168736) + f_G \cdot (-0{,}331264) + f_B \cdot 0{,}5)$$
$$f_{Cr} = 128 + \left(f_R \cdot 0{,}5 + f_G \cdot (-0{,}418688) + f_B \cdot (-0{,}081312)\right)$$

Zum Ermitteln der RGB-Werte haben wir das Bild mit einem Bildberabeitunsprogramm geöffnet und die Werte mithilfe des Pipetten-Werkzeugs abgelesen:

Cyan (0,255,255):

$$f_y = 0 + (0 \cdot 0{,}299 + 255 \cdot 0{,}587 + 255 \cdot 0{,}114) = 178{,}755 \approx 179$$
$$f_{Cb} = 128 + (0 \cdot (-0{,}168736) + 255 \cdot (-0{,}331264) + 255 \cdot 0{,}5) = 171{,}02768 \approx 171$$
$$f_{Cr} = 128 + (0 \cdot 0{,}5 + 255 \cdot (-0{,}418688) + 255 \cdot (-0{,}081312)) = 0{,}5 \approx 1$$

Yellow (255,255,0):

$$f_y = 0 + (255 \cdot 0{,}299 + 255 \cdot 0{,}587 + 0 \cdot 0{,}114) = 225{,}93 \approx 226$$
$$f_{Cb} = 128 + (255 \cdot (-0{,}168736) + 255 \cdot (-0{,}331264) + 0 \cdot 0{,}5) = 0{,}5 \approx 1$$
$$f_{Cr} = 128 + (255 \cdot 0{,}5 + 255 \cdot (-0{,}418688) + 0 \cdot (-0{,}081312)) = 148{,}73456 \approx 149$$

Magenta (255,0,255):

$$f_y = 0 + (255 \cdot 0{,}299 + 0 \cdot 0{,}587 + 255 \cdot 0{,}114) = 105{,}315 \approx 105$$
$$f_{Cb} = 128 + (255 \cdot (-0{,}168736) + 0 \cdot (-0{,}331264) + 255 \cdot 0{,}5) = 212{,}47232 \approx 212$$
$$f_{Cr} = 128 + (255 \cdot 0{,}5 + 0 \cdot (-0{,}418688) + 255 \cdot (-0{,}081312)) = 234{,}76544 \approx 235$$

White (255,255,255):

$$f_y = 0 + (255 \cdot 0{,}299 + 255 \cdot 0{,}587 + 255 \cdot 0{,}114) = 255$$
$$f_{Cb} = 128 + (255 \cdot (-0{,}168736) + 255 \cdot (-0{,}331264) + 255 \cdot 0{,}5) = 128$$
$$f_{Cr} = 128 + (255 \cdot 0{,}5 + 255 \cdot (-0{,}418688) + 255 \cdot (-0{,}081312)) = 128$$

Beim Chroma-Subsampling werden mehrere Pixel durch einen größeren Pixel mit deren Mittelwert dargestellt.
4:2:0 bedeutet hierbei, dass jeweils 2 x 2 Pixelblöcke zu einem Pixel zusammengefasst werden. Da es sich im Beispiel nur um vier verschiedenfarbige Pixel handelt, ist hier einfach der Mittelwert der vier YCbCr-Werte zu berechnen:

$$f_y = (179 + 226 + 105 + 255) \div 4 = 191{,}25 \approx 191$$
$$f_{Cb} = (171 + 1 + 212 + 128) \div 4 = 128$$
$$f_{Cr} = (1 + 149 + 235 + 128) \div 4 = 128{,}25 \approx 128$$

Der erste große Qualitätsverlust entsteht bei der Bildaufbereitung, beim Subsampling, da hierbei mehrere Pixel zusammengefasst werden und so die einzelnen Farbwerte verloren gehen.
Bei der Quantisierung der AC-Koeffizienten wird gerundet, weshalb der Originalwert nicht wieder hergestellt werden kann und Genauigkeitsverluste auftreten können.

(b)

Nearest Neighbour:

„Nearest Neighbour“ bedeutet, die neu skalierten Pixel nehmen den Wert des alten Pixels an, auf welchem ihr neuer Mittelpunkt nun liegt.
Im Beispiel liegt der Mittelpunkt im Bereich des Magentafarbenem Pixel, weshalb auch genau diese Werte übernommen werden (255,0,255)

Bilineare Interpolation:

Bei der „Bilinearen Interpolation“ wird ein Mittelwert aus den Anteilen der Pixel gebildet, die das neue, skalierte Pixel überlagert. Dabei wird auch gewichtet, sodass größere Überlagerungsflächen auch einen größeren Einfluss auf den neuen Farbwert haben.

9/25 des neu skalierten Pixels liegen auf dem Magenta-farbigen Pixel.
6/25 des neu skalierten Pixels liegen auf dem Cyan-farbigen Pixel.
6/25 des neu skalierten Pixels liegen auf dem weißen Pixel.
4/25 des neu skalierten Pixels liegen auf dem gelben Pixel.

$$f_R = 9/25 \cdot 255 + 6/25 \cdot 0 + 6/25 \cdot 255 + 4/25 \cdot 255 = 193{,}8 \approx 194$$
$$f_G = 9/25 \cdot 0 + 6/25 \cdot 255 + 6/25 \cdot 255 + 4/25 \cdot 255 = 163{,}2 \approx 163$$
$$f_B = 9/25 \cdot 255 + 6/25 \cdot 255 + 6/25 \cdot 255 + 4/25 \cdot 0 = 214{,}2 \approx 214$$