

Aufgaben zu Videokodierung

Lösen Sie die nachfolgenden Aufgaben und bereiten Sie diese bis zum nächsten Lehrveranstaltungstermin vor. Unterstrichene Aufgaben sind nach Möglichkeit während der Lehrveranstaltung zu lösen.

LB-VK 01.

- a) Kodieren Sie die zur Verfügung gestellte Videosequenz unter Verwendung von *x264* mit i) 100, ii) 200, iii) 500 kbit/s. Diskutieren Sie den Einfluss der Datenrate auf die (subjektive) Bildqualität, indem Sie sich eine Repräsentation des dekodierten Videos mit dem Parameter `--dump-yuv` erstellen lassen und diese mit *PYUV* (verfügbar im E-Learning-System, Installation über `sudo dpkg -i pyuv_0.7.0-1_amd64.deb`) betrachten.
- b) Kodieren Sie die Sequenz aus a) mit verschiedenen konstanten Quantisierungsparametern derart, dass die Quantisierschrittweiten 1,25, 2,5, 5 etc. (bis zum durch den H.264-Standard beschränkten Maximalwert) abgedeckt sind. Tragen Sie anschließend die von *x264* (mit dem Parameter `--psnr`) ausgegebenen Y-PSNR-Werte über der Dateigröße in einer so genannten Rate-Distortion-Kurve auf und diskutieren Sie deren Verlauf.

LB-VK 02.

- a) Kodieren Sie die Sequenz aus LB-VK 01. a) mit einer Datenrate von 200 kbit/s und allen verfügbaren Presets. Tragen Sie anschließend die (mit dem Befehl `time` gemessenen) benötigten Kodierungsdauern in einem Diagramm auf und diskutieren Sie die zeitlichen Unterschiede anhand der tatsächlich verwendeten Kodierparameter (vgl. Hilfe zu Kommandozeilenparametern, man-Page u.a.) im niedrigsten bzw. höchsten Preset. Beschränken Sie sich dabei auf jene Kodierparameter, deren dazugehörige Konzepte in der Vorlesung behandelt wurden.
- b) Argumentieren Sie anhand der kodierten Sequenzen aus a) einen Preset, der einem aus Ihrer Sicht vernünftigen Kompromiss zwischen Kodierungsdauer und Bildqualität entspricht.

LB-VK 03.

- a) Kodieren Sie die Sequenz aus LB-VK 01. a) mit einer Datenrate von 100 kbit/s und keinem ME-Referenzbild. Erhöhen Sie anschließend die Anzahl der ME-Referenzbilder schrittweise bis zum erlaubten Maximum und tragen Sie die benötigte Kodierungsdauer über der Anzahl der Referenzbilder in einem Diagramm auf.
- b) Diskutieren Sie anhand der kodierten Sequenzen aus a), welchen Einfluss die Anzahl der ME-Referenzbilder auf die (subjektive) Bildqualität hat.

LB-VK 04.

- a) Schreiben Sie ein C-Programm, das eine IYUV-Sequenz (siehe unten) unkomprimierter 8-Bit-Pixel im YCbCr-Farbraum mit 4:2:0-Farbdifferenzkanalunterabtastung (ohne die Verwendung fremder Bibliotheken) einliest. Der Pfad zur Sequenz sowie deren Bildbreite und -höhe sollen dabei in ebendieser Reihenfolge als Kommandozeilenparameter angegeben werden können. Verifizieren Sie Ihr Programm mit der Sequenz aus LB-VK 01. a) und die von Ihrem Programm eingelesenen Werte stichprobenartig mit einem Hex-Editor (z.B. `ghex2` aus dem Paket `ghex`) und/oder anhand einzelner Y-Pixelwerte mit der Software aus LB-VK 01. a) und einem Grafikprogramm wie *GIMP*.
- b) Erweitern Sie Ihr Programm aus a) um einen primitiven ME-Algorithmus, der den $16 \cdot 16$ -Block in der Mitte des zehnten Frames als Ausgangspunkt für seine Suche verwendet. In einem Suchbereich von $32 \cdot 32$ Pixel soll im vorangehenden Frame mit Pixelgenauigkeit unter Verwendung der SAD-Metrik gesucht werden. Geben Sie den zum Treffer gehörenden Bewegungsvektor sowie das MC-Ergebnis aus.
- c) Wie ändert sich das Ergebnis aus b), wenn die MSE- oder die PSNR-Metrik (wählen Sie **eine** der beiden) verwendet wird?

Ergänzung zu LB-VK 04.: IYUV-Dateiformat

Das IYUV-Dateiformat speichert Pixel im YCbCr-Farbraum mit 4:2:0-Farbdifferenzkanalunterabtastung unkomprimiert und planar. Das bedeutet, dass jeder Frame unabhängig von den jeweils anderen gespeichert ist und dessen Farbkanäle getrennt voneinander abgelegt sind.

Die Pixel jedes Farbkanals folgen direkt aufeinander. Das erste gespeicherte Pixel ist jenes, das sich links oben befindet; das letzte ist jenes, das sich rechts unten im dazugehörigen Frame befindet. Nachfolgend ist das beispielhafte (Bitstrom-)Format eines Einzelbildes mit k Bit Farbtiefe, M Zeilen und N Spalten mit 4:2:0-Farbdifferenzkanalunterabtastung dargestellt:

$$\underbrace{Y_{0,0}, Y_{0,1}, \dots, Y_{N-1, M-1}}_{k \text{ Bit}}, Cb_{0,0}, \dots, Cb_{\frac{N}{2}-1, \frac{M}{2}-1}, Cr_{0,0}, \dots, Cr_{\frac{N}{2}-1, \frac{M}{2}-1}$$

$$\underbrace{Y_{0,0}, Y_{1,0}, \dots, Y_{N-1,0}, Y_{0,1}, \dots, Y_{N-1, M-1}}_{k \text{ Bit}}$$

Weitere Illustrationen sind unter <http://www.fourcc.org/yuv.php#IYUV> und <http://en.wikipedia.org/wiki/File:Yuv420.svg> zu finden.