



Design, Entwicklung und Evaluation eines Low Motion Videocodecs

Benjamin Langmann

Betreuer: Zefir Kurtisi

Seminarvortrag zur Studienarbeit

Institut für Betriebssysteme und Rechnerverbund
Technische Universität Braunschweig



Motivation

Einleitung

Funktionsweise

Evaluation

Zusammenfassung

- ✘ Wichtiger Bestandteil gängiger Videocodecs ist die Bewegungskompensation:
 - ▶ Aufwändigster Schritt der Enkodierung
 - ▶ Vergrößert die Abhängigkeiten der Videodaten untereinander, Auswirkungen von Datenverlusten erhöht

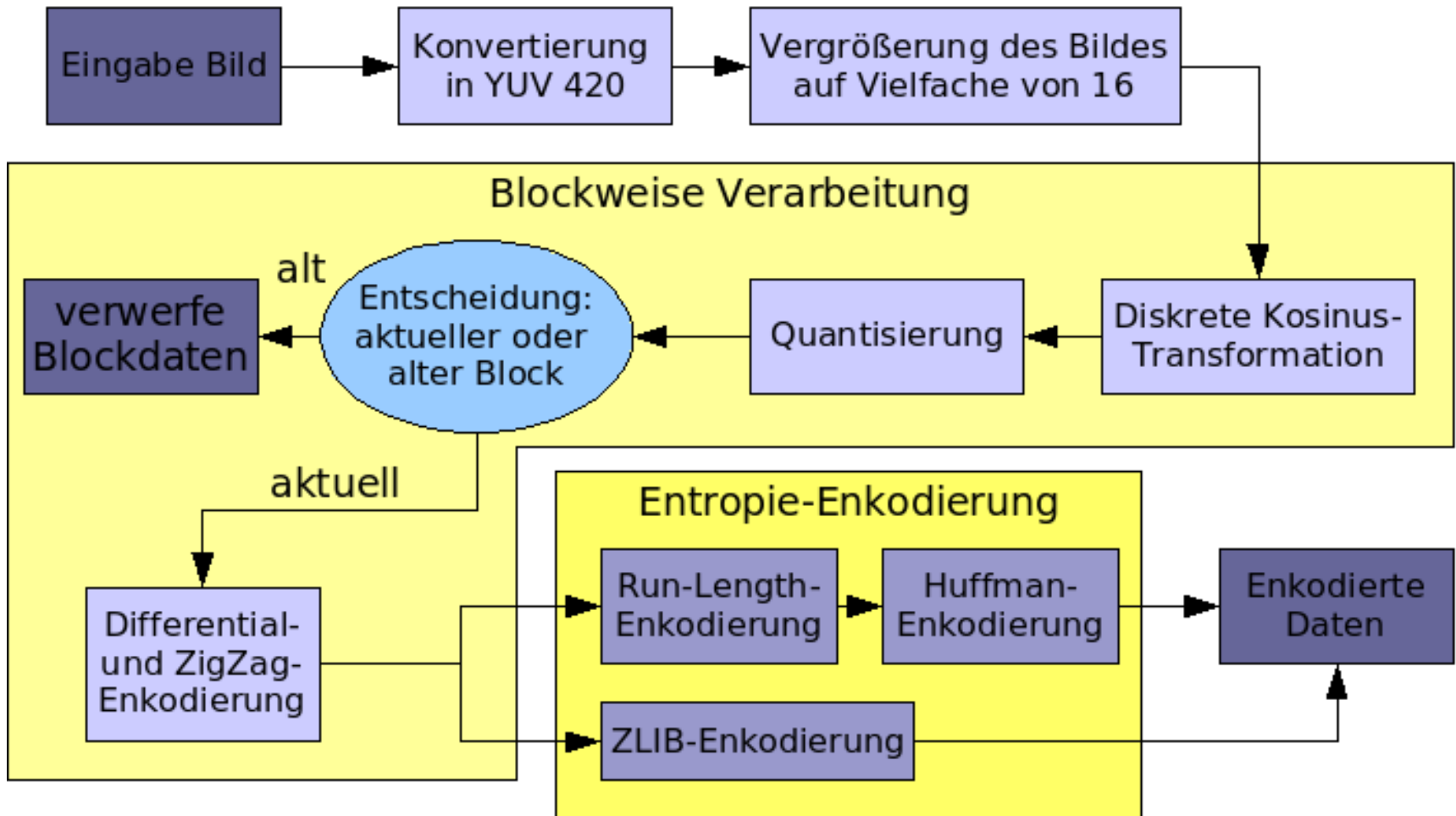
- ✘ Motivation: Untersuchung bei welchen Videotypen die Bewegungskompensation effizient und bei welchen unnötig ist.

- ✘ Dazu Entwicklung eines vergleichbaren Videocodecs, der auf Bewegungskompensation verzichtet.

- ✘ Verfahren basiert auf MPEG4 (grob vereinfacht):
 - ▶ Bildraumkonvertierung (Yuv 420)
 - ▶ Diskrete Kosinus Transformation (DCT) von 8x8-Pixel Bildblöcken
 - ▶ Quantisierung
 - ▶ *bei P-Frames: Prüfung, welche Blöcke abgespeichert und welche vernachlässigt werden sollen*
 - ▶ Entropie-Enkodierung (RLE, Huffman-Enkodierung), zusätzlich ist auch ZLIB-Komprimierung möglich

- ✘ Parameter der Enkodierung:
 - ▶ Quantisierungsparameter (Quantizer von 1 - 31)
 - ▶ Schwellenwert der Blockprüfung (1 – 200)
 - ▶ Maximales Keyframe-Intervall (1 - 10000)
 - ▶ Entropie-Enkodierungs Variante (RLE + Huffman oder ZLIB)

Enkodierung



✘ Ablauf der P-Frame Blockprüfung:

- ▶ Ein einfaches Maß bestimmt, ob ein Block abgespeichert wird: Summieren der Differenzen der Einträge von Block B und dem entsprechenden Block C des letzten Bildes:

$$d = (1 \dots 1) \cdot |C - B| \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

- ▶ Liegt diese Summe d unter einem gegebenen Schwellenwert, so wird der Block abgespeichert, andernfalls ausgelassen.

(Das Ergebnis ist also ähnlich zu dem eines MPEG4-Codecs mit Radius 0 bei der Suche in der Bewegungskompensation.)

- ✘ Implementierung in C mit Video for Windows Anbindung und mit einem Kommandozeileninterface mit Anbindung an die libavcodec/avformat Bibliotheken aus FFMpeg.



Ablauf der Evaluation

Einleitung

Funktionsweise

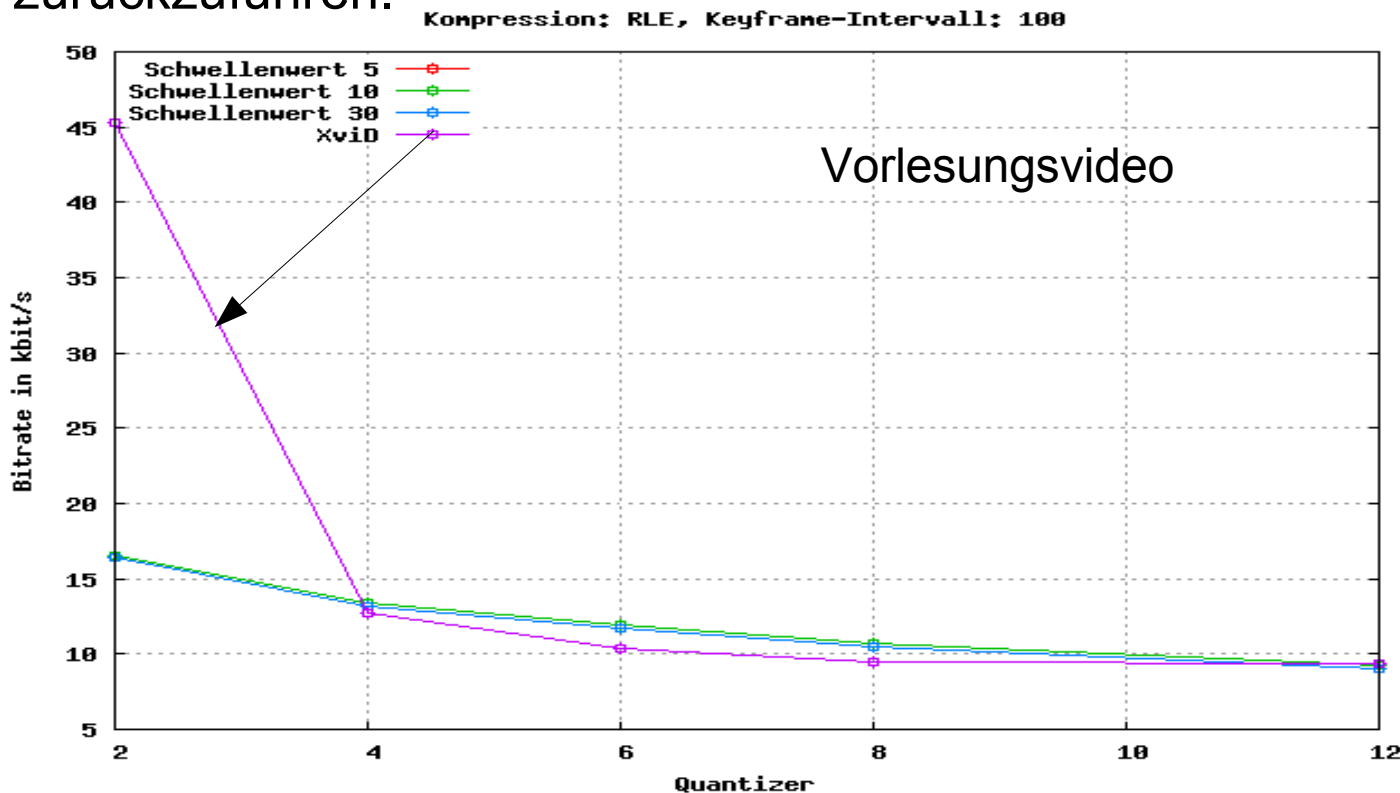
Evaluation

Zusammenfassung

- ✘ Vergleich dieses Codecs mit XviD
- ✘ Auswahl der Testvideos:
 - ▶ Es wurden vier sehr unterschiedliche Testvideos ausgewählt: Vorlesungs-, Vortrags-, Nachrichtenvideo und Kinotrailer.
(nahezu keine Bewegungen / Bewegungen nur im Vordergrund / viele Kameranews, Zooms, Szenenwechsel)
- ✘ Testdurchläufe:
 - ▶ Quantisierung mit festem Quantizern: 2, 4, 6, 8, 12, 18, 24, 31
 - ▶ Schwellenwerte bei diesem Codec: 1, 5, 10, 15, 30, 50
 - ▶ Bei Xvid wurden “einfache” Einstellungen ohne Zusatzoptionen, wie B-Frames, Qpel, GMC und adaptive Quantisierung, gewählt.
- ✘ Auswertung:
 - ▶ Vergleich der Dateigrößen
 - ▶ Vergleich der Bildqualität anhand von PSNR-Werten

✘ Ergebnisse der Evaluation:

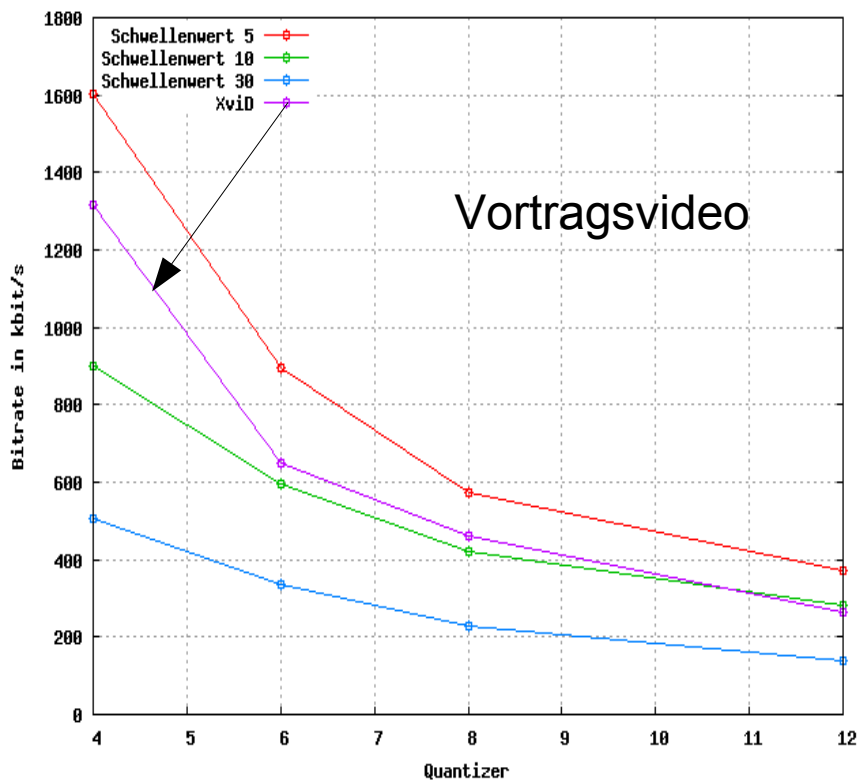
- ▶ Die Unterschiede bei bildänderungsarmen Videos sind klein und wahrscheinlich nur auf verschiedene Entropie-Enkodierungen zurückzuführen.



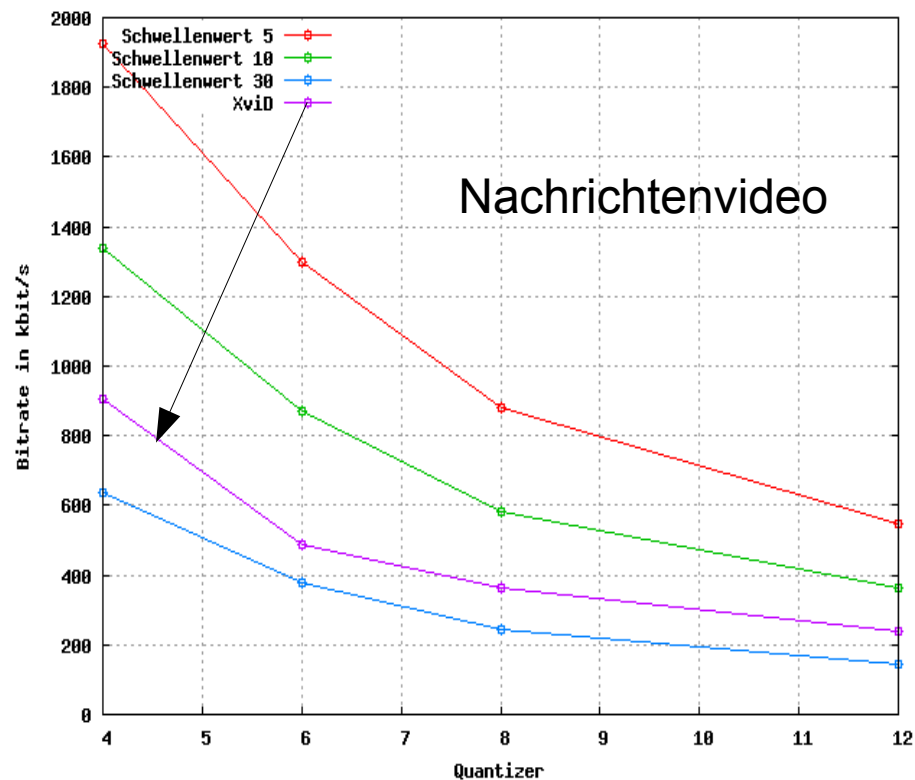
✘ Ergebnisse der Evaluation:

- ▶ Bei Videos mit leichten Bildänderungen lassen sich vergleichbare, wenn auch eher unterlegende Ergebnisse erzielen.

Kompression: RLE, Keyframe-Intervall: 100

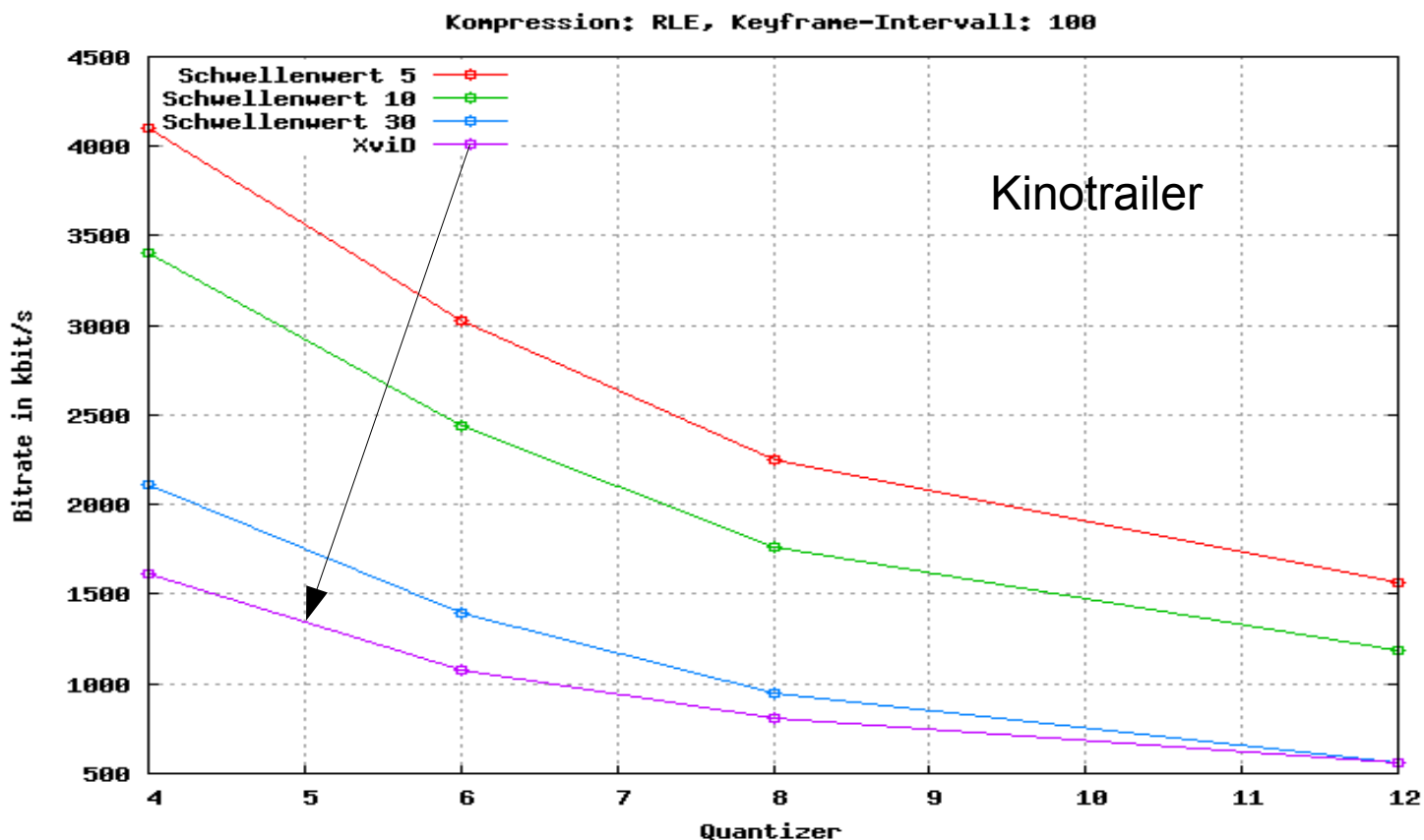


Kompression: RLE, Keyframe-Intervall: 100

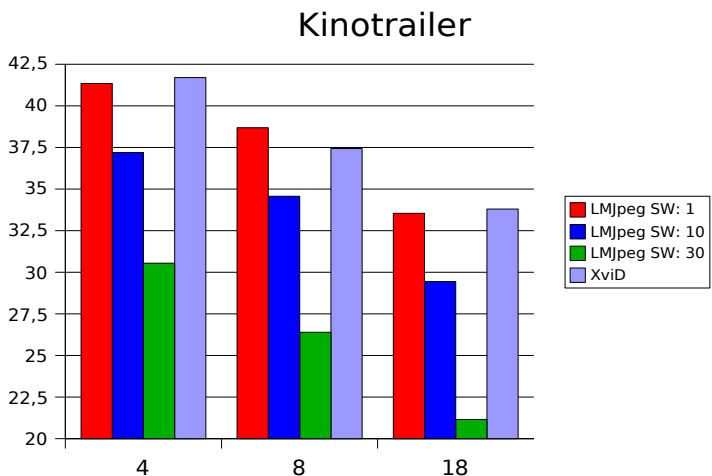
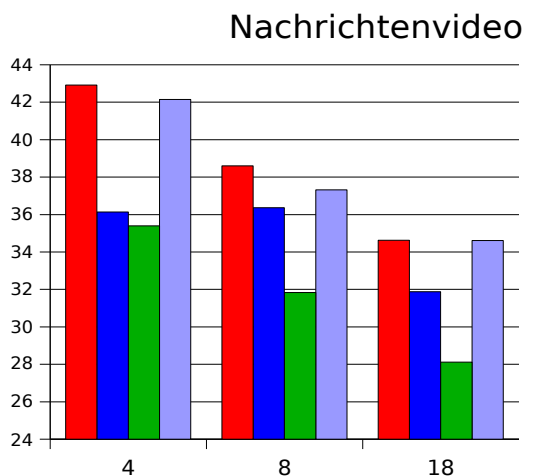
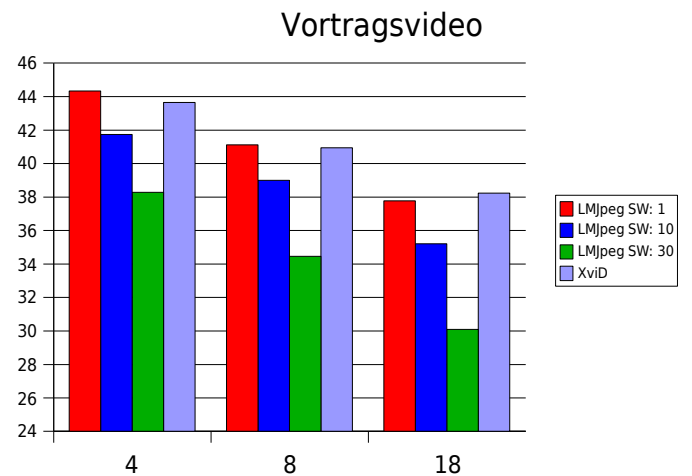
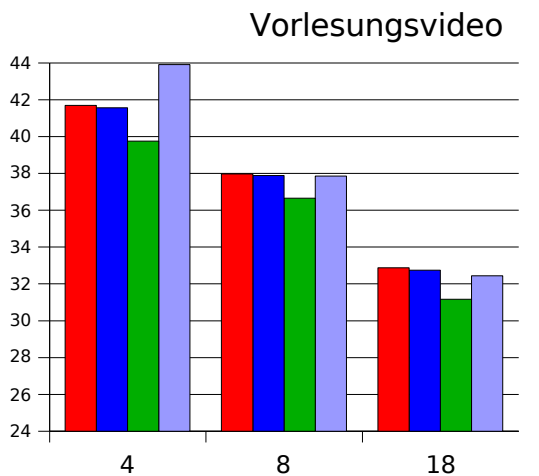


✘ Ergebnisse der Evaluation:

- ▶ Bei "normalen" Videos sind die Ergebnisse deutlich unterlegen.



✗ XviDs Ergebnisse entsprechen im Schnitt Schwellenwert 2-3



- ✘ Vorteile:
 - ▶ geringe Komplexität
 - ▶ geringe Abhängigkeiten der Daten untereinander
 - ▶ vergleichbare Ergebnisse bei bewegungsarmen Videos

- ✘ Nachteile:
 - ▶ Höhere Datenraten notwendig
 - ▶ Eingeschränkte Anwendungsbereiche

- ✘ Geeignete Anwendungsbereiche:
 - ▶ Mobile Anwendungen
 - ▶ Videostreaming



- ✘ Verbesserungsmöglichkeiten, Anmerkungen:
 - ▶ Optimierungen z.B. der Entropie-Enkodierung und der Bildraumkonvertierung
 - ▶ Verbesserung oder Anpassung des Blockprüfungsverfahrens und der Parameter.
 - ▶ Die Netzwerктаuglichkeit wurde in dieser Arbeit nicht betrachtet, d.h. mögliche Latenzen und die Auswirkungen von Paketverlusten. (Da es keine Berücksichtigung und Kodierung von Bewegungen gibt und Bildteile unabhängig vom vorherigen Bild kodiert werden, sollten Datenverluste geringere Auswirkungen haben.)