

DAVT

verlustbehaftete Kompression

von Samu, Lukas, Toni

Bilder

Verlustfreie Kompression: Reduziert die Dateigröße ohne Qualitätsverlust (Bsp. PNG)

-> nutzt man eher für Grafiken

Verlustbehaftete Kompression: Spart mehr Speicher aber Bildinformationen gehen unwiderruflich verloren (Bsp. JPEG)

- nutzt man eher für Fotos

Wie funktioniert die verlustbehaftete Kompression

Step 1:

RGB umwandeln in:

Y = Helligkeit und

Cb , Cr = Farbkanäle

Eigener Kanal für
Helligkeit, weil der
Mensch ja auf Helligkeit
stärker reagiert.



Luminance

Chrominance

Step 2:
Chroma Subsampling
Farbkanäle werden
Kleiner scaled

Chroma Subsampling

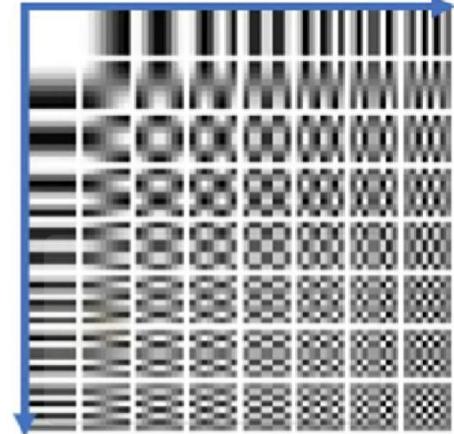
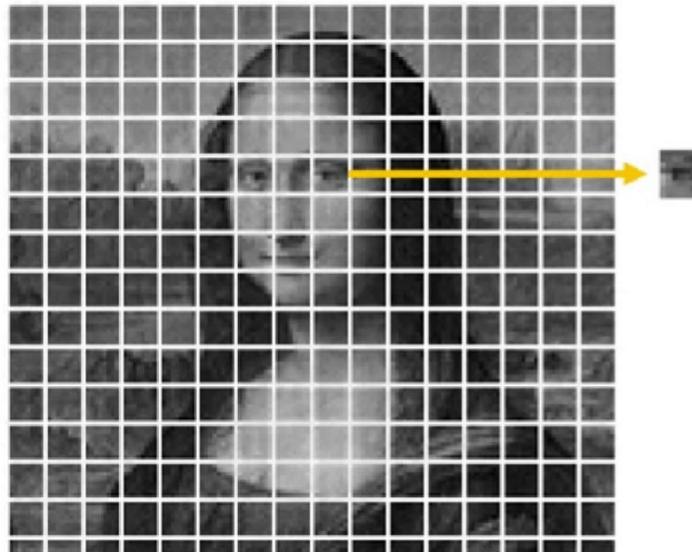


Chrominance

Luminance

Step 3:
In 8x8Pixel Blöcke aufteilen

Step 4:
 $DCT = (\text{Discrete Cosine Transform})$
Über die Verschiedenen Frequenzen die vorkommen
Werden Zahlen vergeben



5.29 Decoding

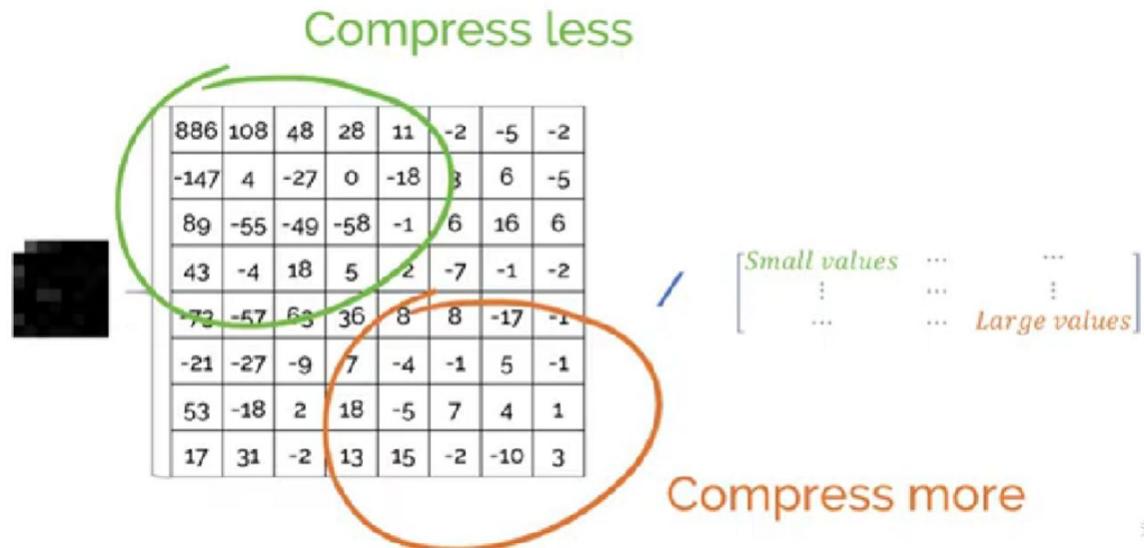
Große Muster (niedrige Frequenz) bekommen hohe Werte → werden behalten.

Feine Muster (hohe Frequenz) bekommen kleine Werte → werden später durch Quantisierung fast auf 0 bzw. auf 0 gesetzt.

Step 5:

Quantisierung

Werte werden abgerundet,
dabei gehen dann nochmal
weiter Infos verloren



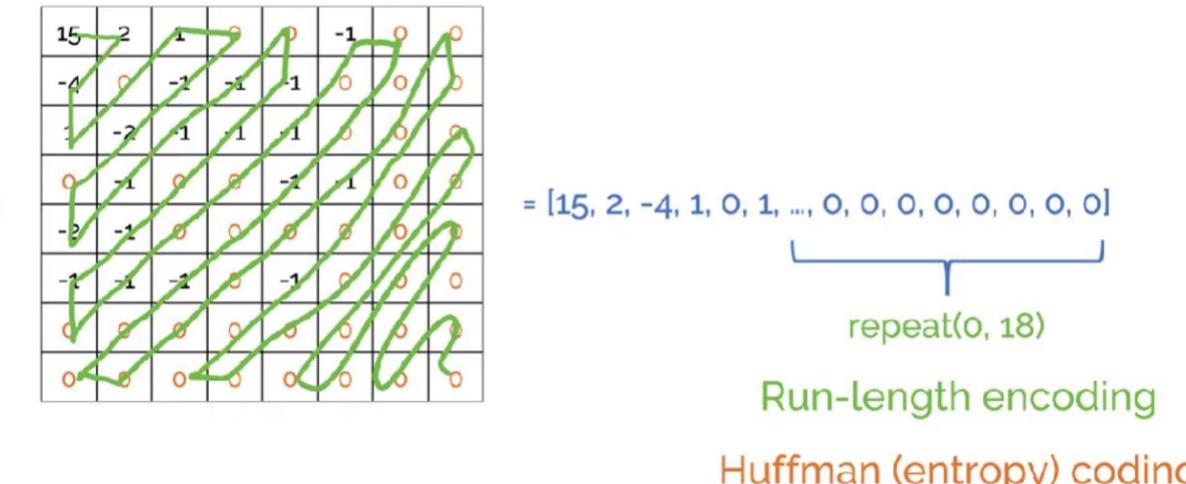
Step 6:

Entropie-Kodierung
Zahlenwerte werden
Effektiv gespeichert

Von oben links
Nach unten rechts

Z.b. mit RLE und
Huffman.

Lossless Compression



Video Compression

Intra-Frame-Compression

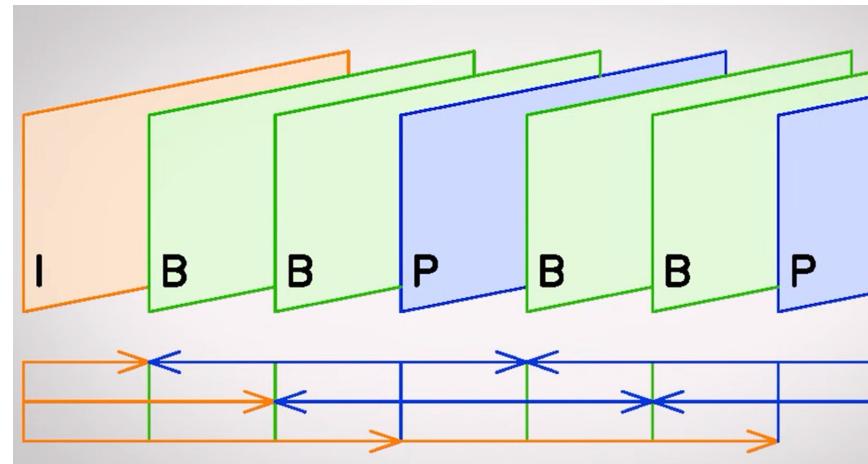
- Daten per Frame rauschmeißen, die für das Menschliche Auge unsichtbar sind (JPEG)

Inter-Frame-Compression

- Unterschiede zwischen Frames nur gespeichert

Inter-Frame-Compression

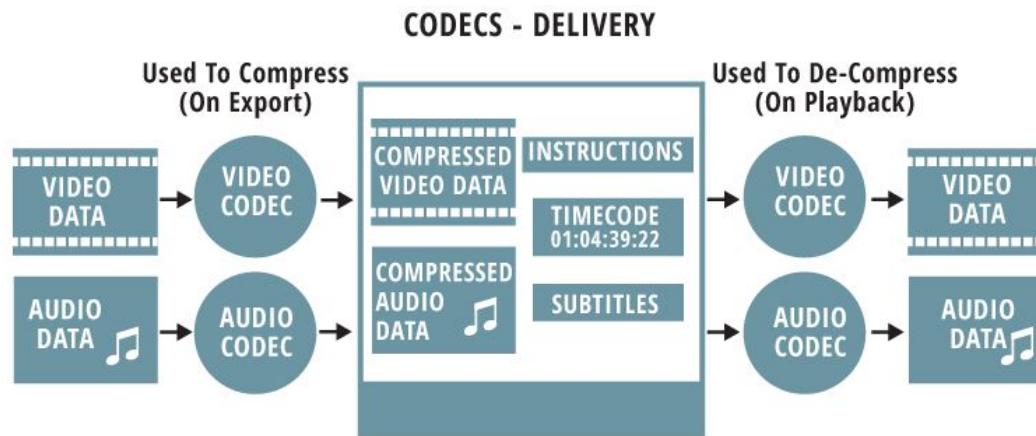
- Wie JPEG jeder Frame in 8x8 Megablocks geteilt (I-Frame) → Reference Frame
- P-Frames: Anleitungen pro Megablock was zu tun ist mit Pixeln des jeweiligen Blocks (rotieren, Farbe ändern, usw.) → Residual Frame
- -> Hälfte der Daten als ein I-Frame
- B-Frames sind Interpolationen zwischen I- und P-Frames → $\frac{1}{4}$ der Daten



Codec vs Container

Container beinhalten mehrere verschiedene Arten von Codecs und Dateiarten

Codecs für Videokompression sind Softwareprogramme für nur einen Datentyp



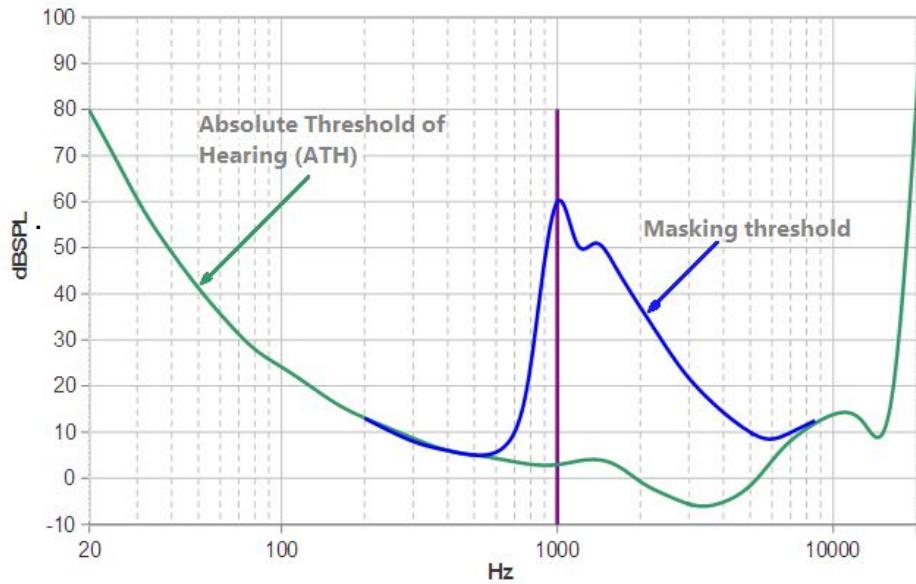
AV1 Codec

- Royalty-Free
- 25% mehr effizient als VP9 Format in einem Netflix Test
- Designed für höhere Resolutions als H.264 Format (MPEG-4)
- Effizienter, ohne großen aufwands Zuwachs, als HEVC

Audio - Auditory Masking

In der Umgebung eines lauten Tons wird der Threshold erhöht, sodass leisere Töne unter dem Threshold nicht zu hören sind

→ leisere Klänge werden durch laute Klänge verdeckt



Audio - Lossy audio compression removes redundant audio data

Es werden nicht unnötige Daten verworfen, sondern quantization noise erhöht

Bei niedrigen Bitraten können low-pass-filter angewandt werden um die Genauigkeit zu erhöhen

Bei hohen Bitraten ist der Verlust hoher Frequenzen nicht wahrnehmbar

Audio - typische Formate und Anwendungen

Encoding format	Typical application
AAC	DVB standard (MPEG-2, MPEG-4) Apple (iTunes), video streaming services
Dolby AC-3	DVD audio, video
DTS	DVD audio, video
MP3	First MPEG standard, widespread support, DVB
Opus	YouTube
Vorbis	Spotify

Audio - Qualität

Bitrate und Encoder-Design:

Encoder haben einen großen Einfluss auf die Qualität, wenn die Bitrate niedrig ist.

Ein Encoder, spezialisiert auf Geschwindigkeit, könnte Files mit schlechterer Qualität produzieren.

Ein Encoder, spezialisiert auf Qualität, produziert Files mit hoher Qualität bei gleicher Bitrate.

Psychoakustische Modelle:

Das Menschliche Ohr kann nicht alle Informationen eines ankommenden Tons analysieren, deshalb kann man eine Klangdatei stark verändern, ohne das es auffällt.

Es können z.B. Teile der Klanganteile in sehr hohen und sehr tiefen Frequenzbereichen mit stärker verminderter Präzision gespeichert oder verworfen werden.