

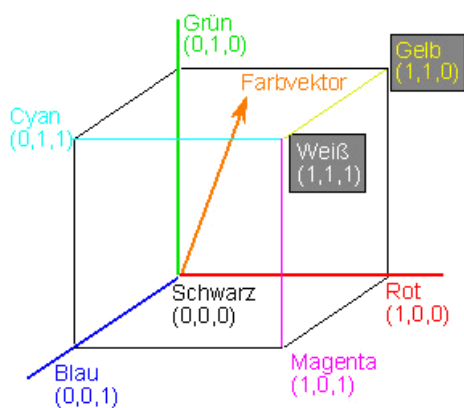
Allgemein

- Dient zur Reduzierung der Datenmenge
- Hat ihre Ursprünge in der Standbildkompression (JPEG)
 - Einzelne Bilder wurden unabhängig voneinander komprimiert
 - Ton wurde nicht komprimiert
- Heutzutage: Ähnlichkeiten zwischen Bildern wird genutzt um höhere Kompressionsrate zu erreichen (bis zu 1:200)
 - Ton wird auch komprimiert

Farbräume

RGB-Farbraum

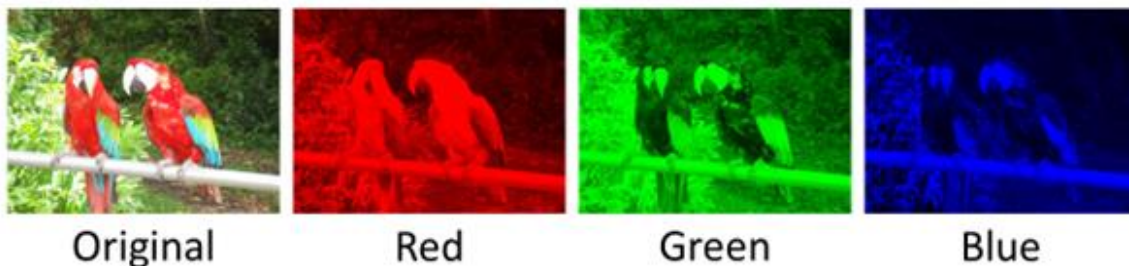
- RGB steht für Rot, Grün und Blau
- Ist für effiziente Codierung **NICHT** geeignet



Kann dreidimensional dargestellt werden.

X-Achse: Rot Y-Achse: Grün Z-Achse: Blau

Daraus entsteht ein Würfel, wobei an jeder Ecke eine andere Farbe zu finden ist. Um darzustellen welche Farbe verwendet wird, wird ein Farbevektor eingezeichnet.



Wie an diesem Beispiel zu sehen ist, wird jedes Bild in drei Einzelteile zerlegt. Das ganz linke Bild zeigt das ganze Bild. Das zweite zeigt die Rot Anteile, das dritte die Grün Anteile und das letzte die Blau Anteile.

YCbCr-Farbraum

- Ist für effiziente Codierung geeignet
- Konvertierung von RGB zu YCbCr ist verlustfrei (bis auf Rundungsfehler)
- Mensch ist auf Helligkeitsänderung empfindlicher als Farbänderung → Kanäle Cb & Cr werden in halber Auflösung ausgeführt (erste Komprimierung)

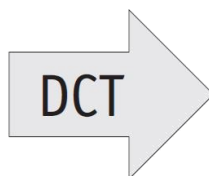


Auch in diesem Farbraum wird jedes Bild in drei Einzelteile zerlegt. Ganz links befindet sich das originale Bild. Das zweite Bild zeigt die Helligkeitsverteilung. Man kann hier gut erkennen, dass der die Blätter eher heller und die linke obere Ecke eher dunkler dargestellt wird. Das dritte Bild zeigt den Cb-Kanal welcher die Kombination aus Blau und Gelb darstellt. Das letzte Bild zeigt den Cr-Kanal welcher die Rot-Grün Kombination zeigt.

Diskrete Cosinustransformation (DCT)

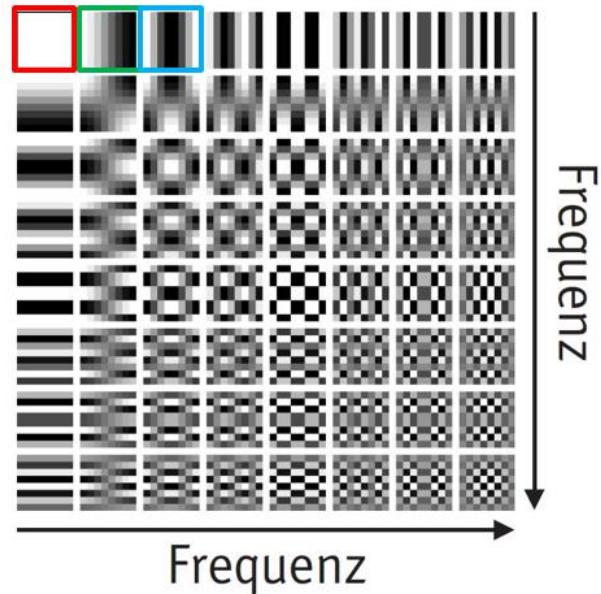
- Jedes einzelne Bild wird in 8x8 Blöcke eingeteilt welche einzeln codiert werden

139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	155
150	155	160	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	161	160	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	161	163	158	158	158



235.6	-1.0	-12.1	-5.2	2.1	-1.7	-2.7	1.3
-22.6	-17.5	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-10.9	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0.0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1.0	-1.0
-1.3	-0.4	-0.3	-1.5	-0.5	1.7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

235.6	-1.0	-12.1	-5.2	2.1	-1.7	-2.7	1.3
-22.6	-17.5	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-10.9	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0.0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1.0	-1.0
-1.3	-0.4	-0.3	-1.5	-0.5	1.7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4



$$\text{ROT} * \text{ROT} + \text{GRÜN} * \text{GRÜN} + \text{BLAU} * \text{BLAU} + \dots$$

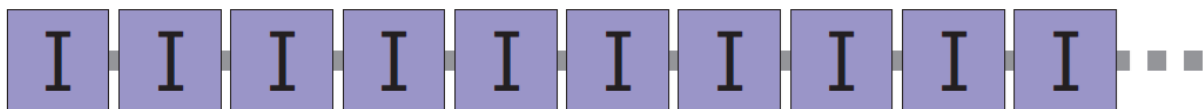
Anschließend werden die transformierten Blöcke mit den 64 charakteristischen Cosinusgrundschwingungen multipliziert und alle Werte miteinander addiert. Dies wird dann übertragen und auf der Empfängerseite rückgewandelt.

Der Wert links oben heißt DC-Koeffizient und alle anderen Werte heißen AC-Koeffizienten. Der DC-Koeffizient ist das Maß für die mittlere Helligkeit.

Frames

I-Frames

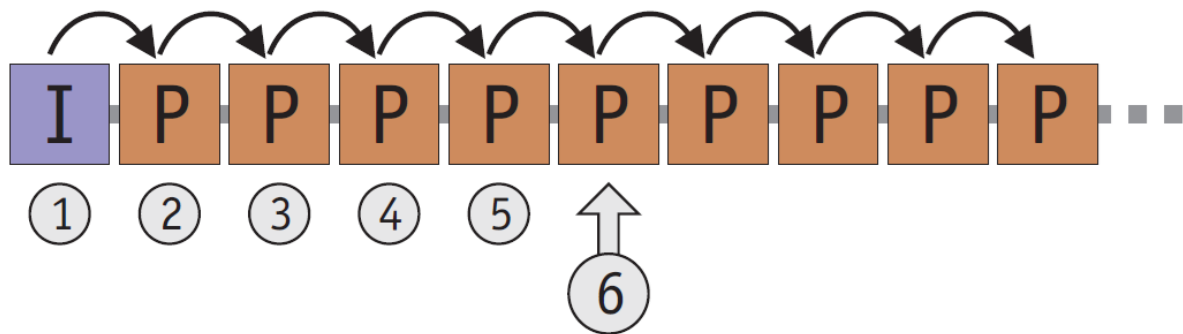
- Aneinanderreihung von Einzelbildern
- Vorteile:
 - Direkter Zugriff auf jedes Bild
 - Falls Decodierung nicht möglich → einfach auslassen
- Nachteil:
 - Niedrige Codiereffizienz



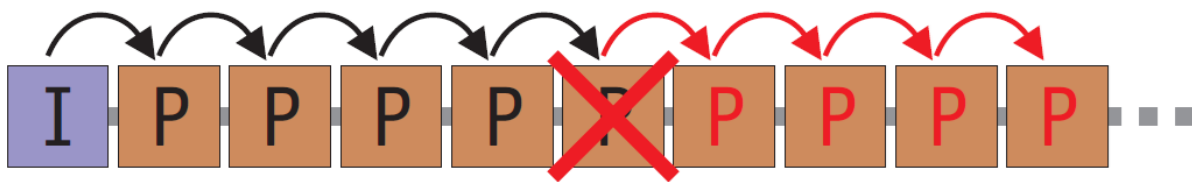
P-Frames

- Bilden Differenz zum letzten Frame
- Zuerst wird ein I-Frame übertragen und anschließend nur noch P-Frames
- Vorteil:
 - Höhere Codiereffizienz

- Nachteil:
 - Um auf einen Frame zuzugreifen, müssen alle anderen Frames davor auch decodiert werden



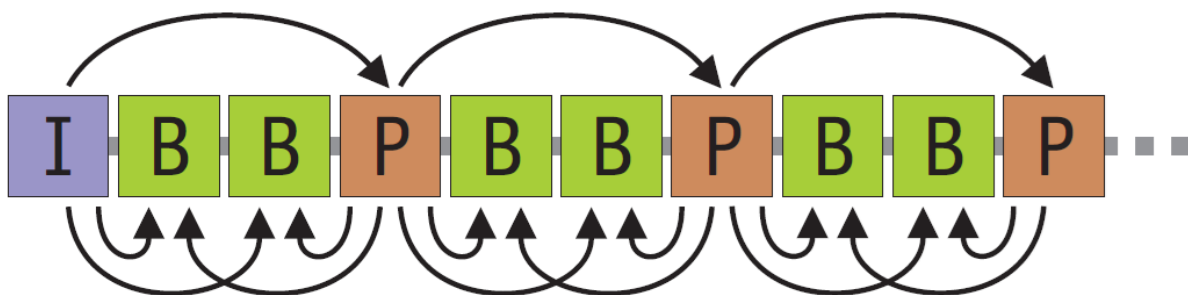
- Ein P-Frame fehlerhaft → Rest vom Video kann nicht decodiert werden



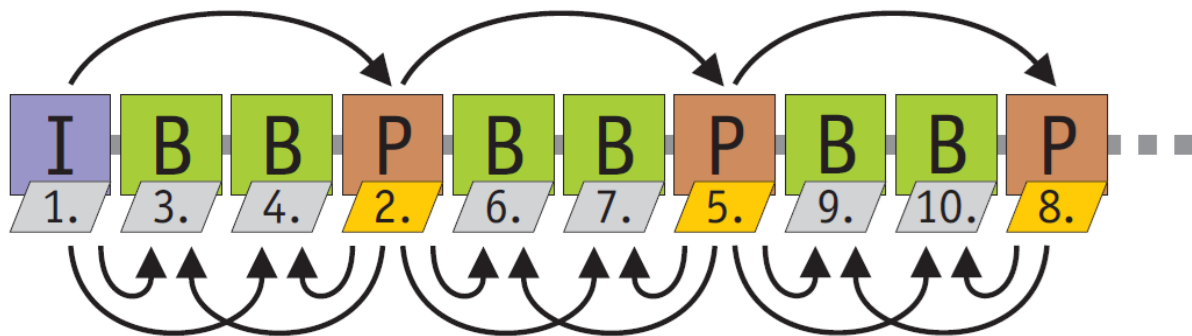
- Teillösung: Periodische Einstreuung von I-Frames; Jedoch bei zu großer Periode nicht hilfreich

B-Frames

- Bi-Directional Predicted Frames
- Unterschied zum vorherigen und nächsten I- oder P-Frame wird codiert
- Bilden keine Grundlage für andere Frames
- Typischerweise 2-3 B-Frames zwischen zwei P-Frames



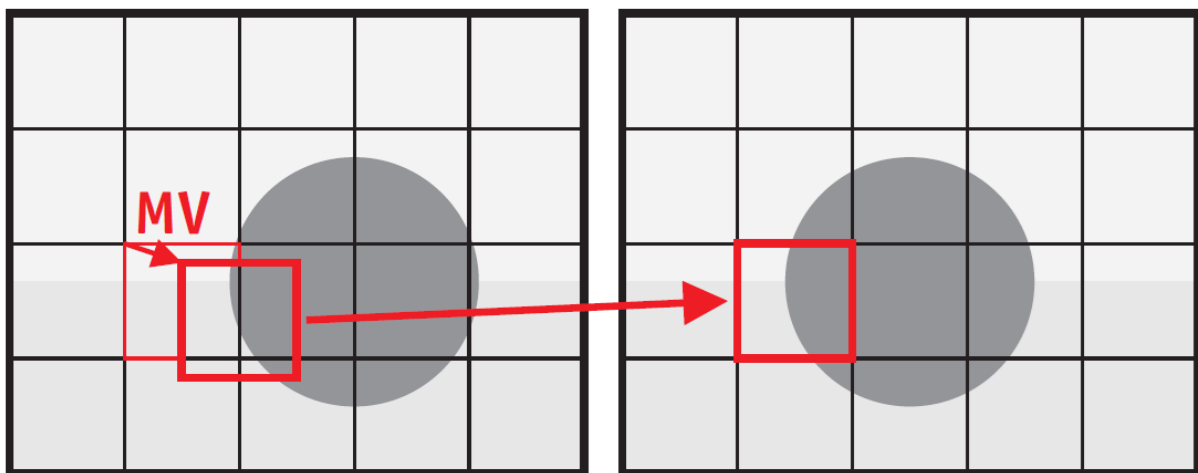
- Vorteile:
 - Höhere Codiereffizienz
 - Schnellere Zugriff auf beliebige Bilder
 - Können bei Probleme auch ausgelassen werden
- Nachteile:
 - Leisten keinen Beitrag zu anderen Bildern
 - Benötigt mehr Speicherplatz & Bandbreite
 - Codierreihenfolge ≠ Anzeigereihenfolge



Die Reihenfolge bezieht sich aufs codieren.

MPEG-1

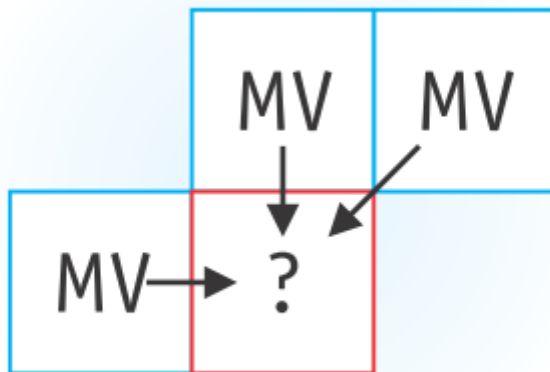
- Wurde 1992 standardisiert
- Entwickelt für CDs
- Unterstützt Video bis zu 4096x4096; 60p
- Motion Compensation wurde eingeführt:



In Motion Vektoren (MV) werden nur die Unterschiede zum letzten Bild pro Makroblock gespeichert. Gleiche Makroblöcke werden übernommen.

MPEG-4

- 2000 standardisiert
- Motion Vector Prediction wurde verbessert (mit MPEG-2 eingeführt)
- MV der umliegenden Makroblöcke → Median Wert für x- und y- Koordinaten gebildet; Abweichung zu eigentlichen Makroblock wird codiert
 - Vorteil: MV werden kleiner, da Prediction besser wird



H.264

- 2003 standardisiert
- In MPEG-4 implementiert
- Anwendungsbereiche:
 - Blu-ray
 - HDTV
 - Standardcodec für mobile Applikationen
 - Standardcodec für Internet-Streaming Videos
- Bei DCT 4x4 Transformation anstatt von 8x8