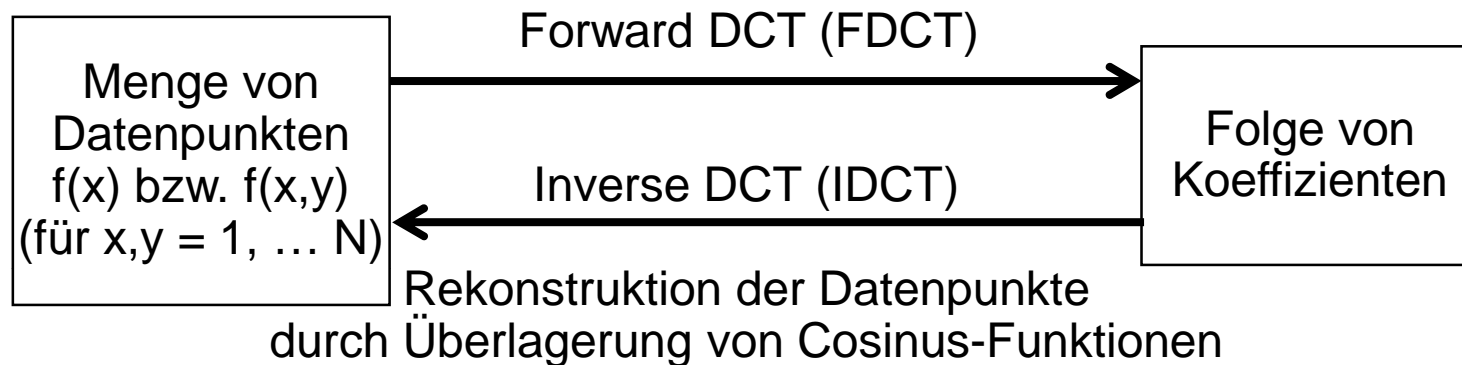


Diskrete Cosinus-Transformation (DCT)

- Prinzip von DCT:
 - (in einer oder zwei Dimensionen...)



Datenpunkte und Koeffizienten sind bei JPEG jeweils 8 x 8 - Integer - Blöcke

(Forward) DCT: Mathematische Definition

$$F(u,v) = \frac{1}{4} c_u c_v \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

wobei

x, y Koordinaten für die Datenpunkte einer Quell-Dateneinheit
 ($x, y = 0, \dots, 7$)

u, v Koordinaten für die Ziel-Koeffizienten ($u, v = 0, \dots, 7$)

$f(x,y)$ Datenwert (Sample)

$F(u,v)$ Koeffizientenwert

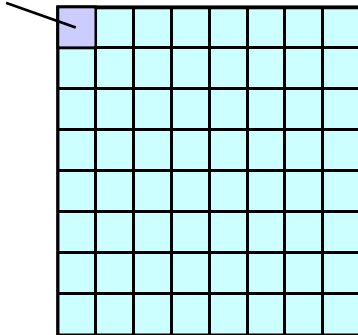
$$c_u, c_v = \frac{1}{\sqrt{2}} \quad \text{falls } u, v = 0$$

$$c_u, c_v = 1 \quad \text{sonst}$$

- Die Berechnung der Formel lässt sich auf eine einfache Matrixmultiplikation mit konstanten Matriceinträgen reduzieren.
- Aus technischen Gründen Sample-Wertebereich zuerst in $(-128, +127)$ verschoben

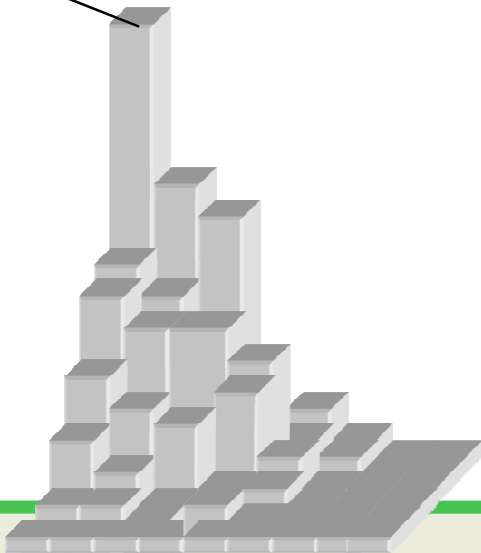
Interpretation der DCT-Koeffizienten

DC-Koeffizient



Alle anderen:
AC-
Koeffizienten

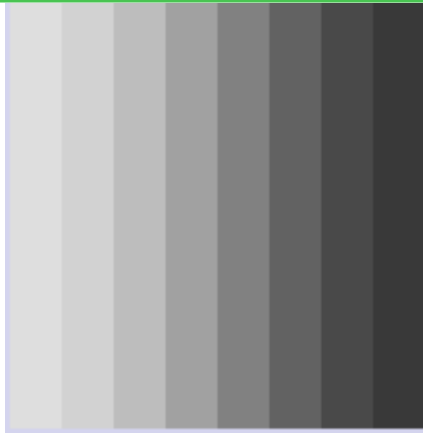
DC-Koeffizient $F(0,0)$



- Der DC-Koeffizient gibt den Grundton des beschriebenen Bereichs (8x8) im Bild an (in der aktuellen Komponente)
Die AC-Koeffizienten geben mit aufsteigenden Indizes den Anteil „höherer Frequenzen“ an, d.h. die Zahl der (vertikalen bzw. horizontalen) Streifen
- Z.B.:
 - $F(7,0)$ gibt an, zu welchem Anteil extrem dichte waagerechte Streifen vorkommen;
 - $F(0,7)$ gibt an, zu welchem Anteil extrem dichte senkrechte Streifen vorkommen

DC = Gleichstrom
AC = Wechselstrom

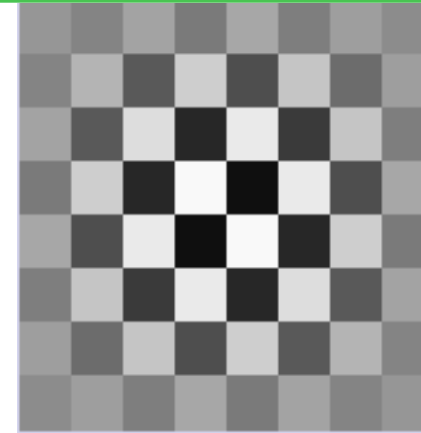
Beispiele für DCT-Transformation



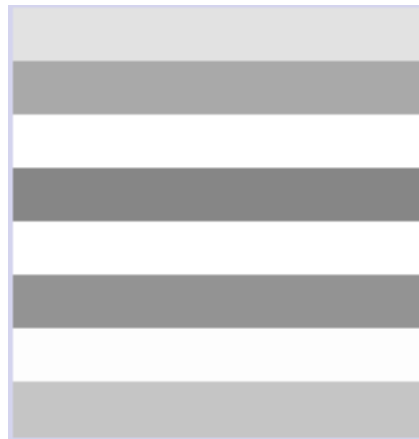
$F(0,1) = 500$,
alle anderen $F(u, v) = 0$



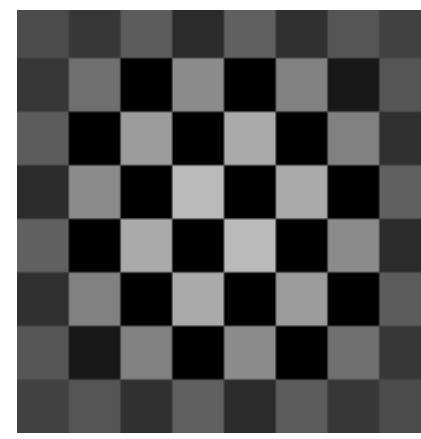
$F(7,0) = 500$,
alle anderen $F(u, v) = 0$



$F(7,7) = 500$,
alle anderen $F(u, v) = 0$

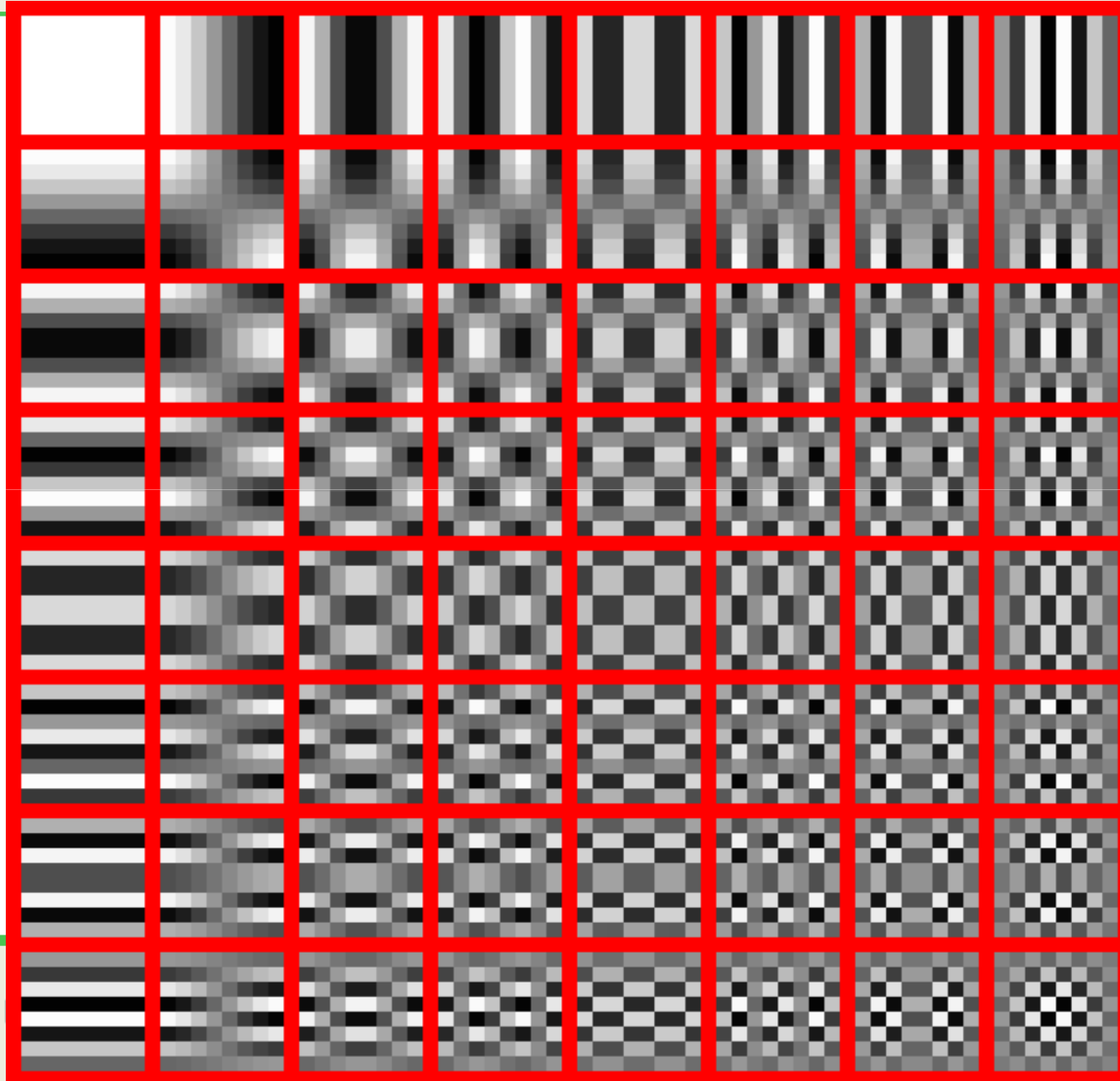


$F(7,0) = 500$, $F(0,0) = 600$
alle anderen $F(u, v) = 0$



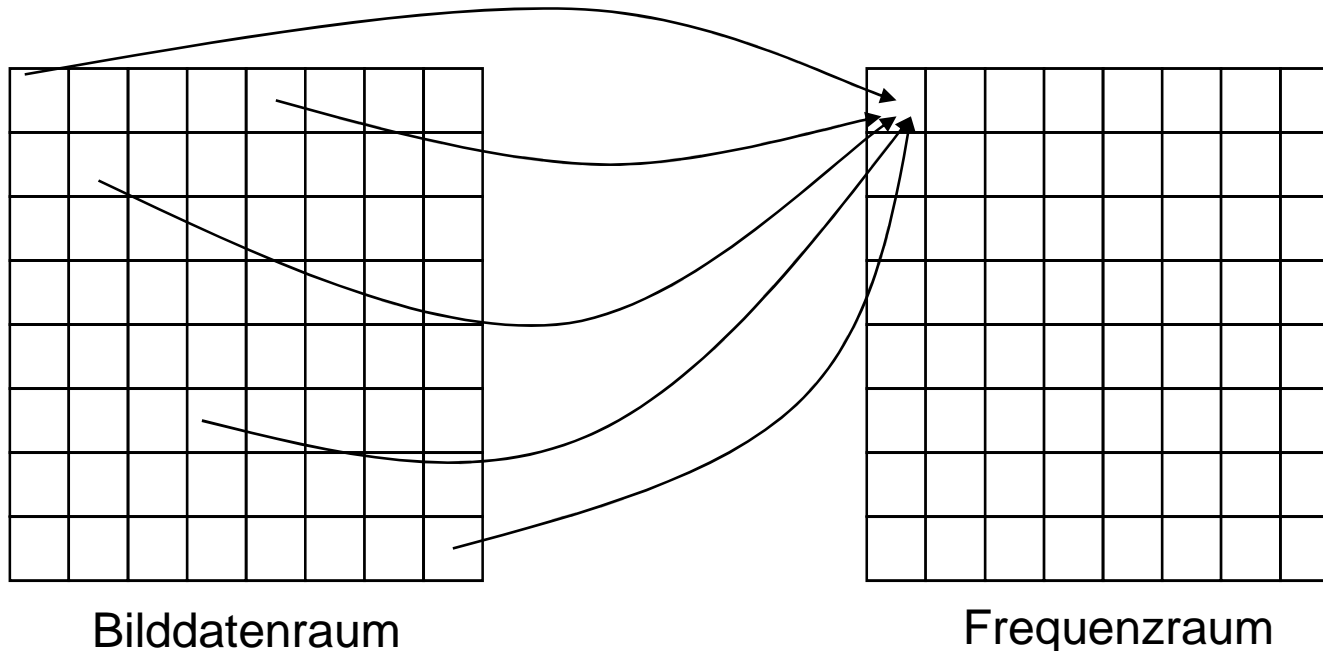
$F(7,7) = 500$, $F(0,0) = -600$
alle anderen $F(u, v) = 0$

Alle DCT Basisvektoren



Quelle:
wikipedia

DCT: Zusammenhang Datenraum - Frequenzraum



- Ein Punkt im Frequenzraum fasst die Informationen aus dem aktuell betrachteten Bildraum (8x8 Pixel) zusammen.
- Kanten erscheinen als Anteile hoher Frequenzen; bei Flächen sind die hohen Frequenzen fast Null
 - Gute Voraussetzung für spätere Kompression der Null-nahen Werte durch Entropiekodierung

Inverse DCT: Mathematische Definition

$$f(x, y) = \frac{1}{4} \sum_{x=0}^7 \sum_{y=0}^7 c_u c_v F(u, v) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

wobei

x, y Koordinaten für die Datenpunkte einer Quell-Dateneinheit
($x, y = 0, \dots, 7$)

u, v Koordinaten für die Ziel-Koeffizienten ($u, v = 0, \dots, 7$)

$f(x, y)$ Datenwert (Sample)

$F(x, y)$ Koeffizientenwert

$$c_u, c_v = \frac{1}{\sqrt{2}} \quad \text{falls } u, v = 0$$

$$c_u, c_v = 1 \quad \text{sonst}$$

- Die Berechnung ist fast identisch mit der Vorwärts-Transformation.
- Mathematisch gesehen, ist der Prozess verlustfrei!
 - Verluste entstehen aber durch Rundungsfehler

JPEG-Kodierung: Quantisierung

- Entscheidender Schritt zum **Informationsverlust** und damit zur starken Kompression !
 - Runden der Koeffizienten erzeugt viele Null-Werte und ähnliche Werte
 - Damit besser mit nachfolgenden verlustfreien Verfahren komprimierbar
- Quantisierungstabelle:
 - Enthält 64 vorgegebene und konstante Bewertungs-Koeffizienten $Q(u, v)$
 - Bedeutung: Bewertung der einzelnen Frequenzanteile des Bildes
 - Größere Tabelleneinträge bedeuten stärkere Vergröberung
 - Konkrete Tabellen nicht Bestandteil des Standards (nur zwei Beispiele)
 - Typisch: Verschiedene Bewertung für hohe und niedrige Frequenzen
 - Benutzte Quantisierungstabellen werden als Bestandteil der komprimierten Daten abgelegt und bei Dekompression benutzt
- Berechnung:
 - Division Frequenz-Koeffizient /
Bewertungskoeffizient und Rundung

$$F'(u, v) = \text{Round}\left(\frac{F(u, v)}{Q(u, v)}\right)$$

Beispiel:

$$Q = \begin{bmatrix} 10 & 15 & 25 & 37 & 51 & 66 & 82 & 100 \\ 15 & 19 & 28 & 39 & 52 & 67 & 83 & 101 \\ 25 & 28 & 35 & 45 & 58 & 72 & 88 & 105 \\ 37 & 39 & 45 & 54 & 66 & 79 & 94 & 111 \\ 51 & 52 & 58 & 66 & 76 & 89 & 103 & 119 \\ 66 & 67 & 72 & 79 & 89 & 101 & 114 & 130 \\ 82 & 83 & 88 & 94 & 103 & 114 & 127 & 142 \\ 100 & 101 & 105 & 111 & 119 & 130 & 142 & 156 \end{bmatrix}$$

$$F = \begin{bmatrix} 782,91 & 44,93 & 172,52 & -35,28 & -20,58 & 35,93 & 2,88 & -3,85 \\ -122,35 & -75,46 & -7,52 & 55,00 & 30,72 & -17,73 & 8,29 & 1,97 \\ -2,99 & -32,77 & -57,18 & -30,07 & 1,76 & 17,63 & 12,23 & -13,57 \\ -7,98 & 0,66 & 2,41 & -21,28 & -31,07 & -17,20 & -9,68 & 16,94 \\ 3,87 & 7,07 & 0,56 & 5,13 & -2,47 & -15,09 & -17,70 & -3,76 \\ -3,77 & 0,80 & -1,46 & -3,50 & 1,48 & 4,13 & -6,32 & -18,47 \\ 1,78 & 3,28 & 4,63 & 3,27 & 2,39 & -2,31 & 5,21 & 11,77 \\ -1,75 & 0,43 & -2,72 & -3,05 & 3,95 & -1,83 & 1,98 & 3,87 \end{bmatrix}$$

$$F^Q = \begin{bmatrix} 78 & 3 & 7 & -1 & 0 & 1 & 0 & 0 \\ -8 & -4 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Entnommen aus Wikipedia

Informationsverlust durch Quantisierung



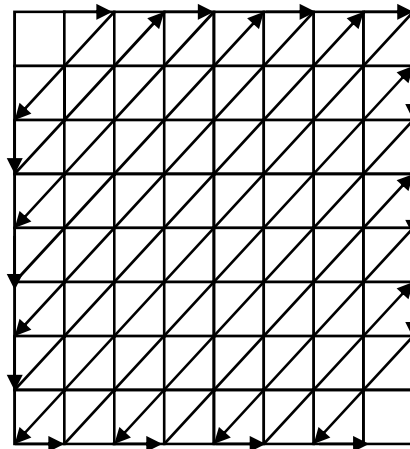
Bei JPEG-Kompressions-Algorithmen ist der Grad der Quantisierung wählbar: „Trade-Off“ zwischen Speicherplatzersparnis und Bildverfälschung (Artefakten)



Artefakte treten bei Kanten und Details auf, kaum bei Flächen

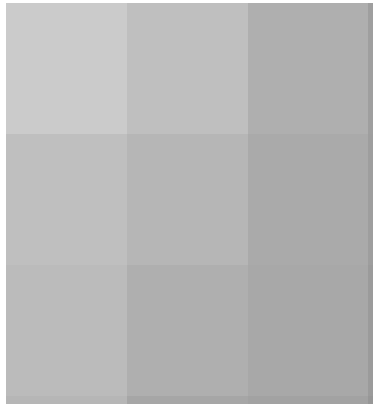
Vorbereitung zur Weiterverarbeitung

- Quantisierte Frequenzwerte:
 - werden in linearer Reihenfolge ausgegeben
 - unterschiedliche Behandlung DC- und AC-Koeffizienten
- DC-Koeffizienten:
 - Benachbarte Dateneinheiten haben oft ähnlichen Grundton
 - Deshalb separat extrahiert (alle DC-Koeffizienten des Bildes in ein „Grob bild“)
- AC-Koeffizienten:
 - Ausgabe nach absteigender Frequenz („Zick-Zack“)



DC-Komponenten: Prädiktive Codierung

- Grundidee:



45	48	60
50	55	65
52	60	70

-10	-7	+5
-5	55	+10
-3	+5	+15

- Feste Werte nur für einige Basispixel definieren
- Für andere Pixel nur Differenz speichern
- Potentielle Vorteile:
 - an sehr vielen Stellen kleinere Bitbreite für Differenzwerte möglich
 - Wiederholung gleicher Differenzwerte ermöglicht effektive Kompression mit verlustfreien Verfahren
- Verlustfrei
(in JPEG kombiniert mit weiteren – auch verlustbehafteten - Techniken)

JPEG-Kodierung: Entropie-Kompression

- Vorletzter Schritt: „Statistische Modellierung“
 - DC-Koeffizienten: Prädiktive Codierung (*Differenzen*)
 - AC-Koeffizienten: im Wesentlichen Lauflängen-Codierung
- Letzter Schritt: Entropie-Kodierung
 - Wahl zwischen Huffman-Algorithmus und arithmetischer Kompression
 - Getrennt für DC- und AC-Koeffizienten
- Woher kommen die Häufigkeitsverteilungen?
 - Zwei Beispielveilteilungen im JPEG-Standard beschrieben
 - Alternative: Durch zusätzlichen Durchlauf über die Daten errechnen

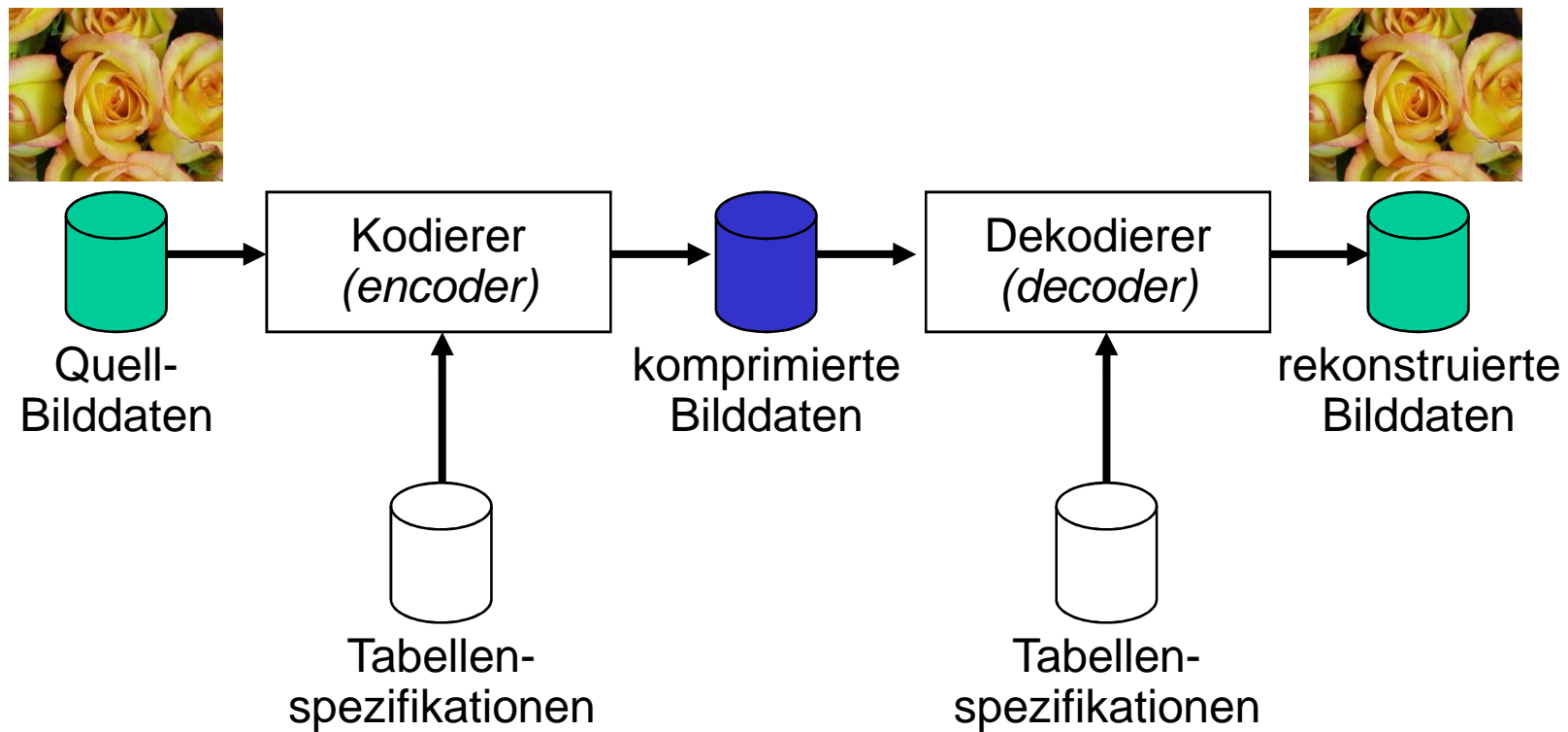
JPEG Datenströme

- Ausgabe der JPEG-Kompression
 - Besteht aus Komponenten
 - Getrennt durch *marker* (2 Bytes, erstes Byte xFF)
- Beispiele für Marker:
 - Start of image (SOI)
 - End of image (EOI)
 - Start of frame, baseline (SOF₀)
 - Start of frame, extended sequential (SOF₁)
 - Start of frame, progressive (SOF₂)
 - Start of frame, lossless (SOF₃)
 - Define Huffman table (DHT)
 - Define quantization tables (DQT)
 - Define restart interval (DRI)
 - Application specific (APP₀ – APP₁₅)
 - ...

JPEG: Hintergrund

- JPEG = „Joint Photographics Expert Group“
 - „Joint“ wegen Zusammenarbeit von Arbeitsgruppen zweier Organisationen (ISO und CCITT/ITU)
 - Arbeit seit 1982, Verfahrensvergleich 1987, Auswahl einer „adaptiven Transformationskodierung basierend auf Diskreter Cosinus-Transformation (DCT)“
 - 1992: ITU-T Recommendation T.81 + Internationaler Standard ISO 10918-1
- Wichtige Eigenschaften/Anforderungen:
 - Unabhängigkeit von Bildgröße, Seitenverhältnis, Farbraum, Farbvielfalt
 - Anwendbar auf jedes digitale Standbild mit Farben oder Grautönen
 - Sehr hohe Kompressionsrate
 - Parametrisierbar in Qualität/Kompression
 - Realisierbar durch Software und Spezial-Hardware: gute Komplexität
 - Sequentielle und progressive Dekodierung
 - Unterstützung von verlustfreier Kompression und hierarchischer Verfeinerung der Bildqualität

JPEG-Architekturmodell

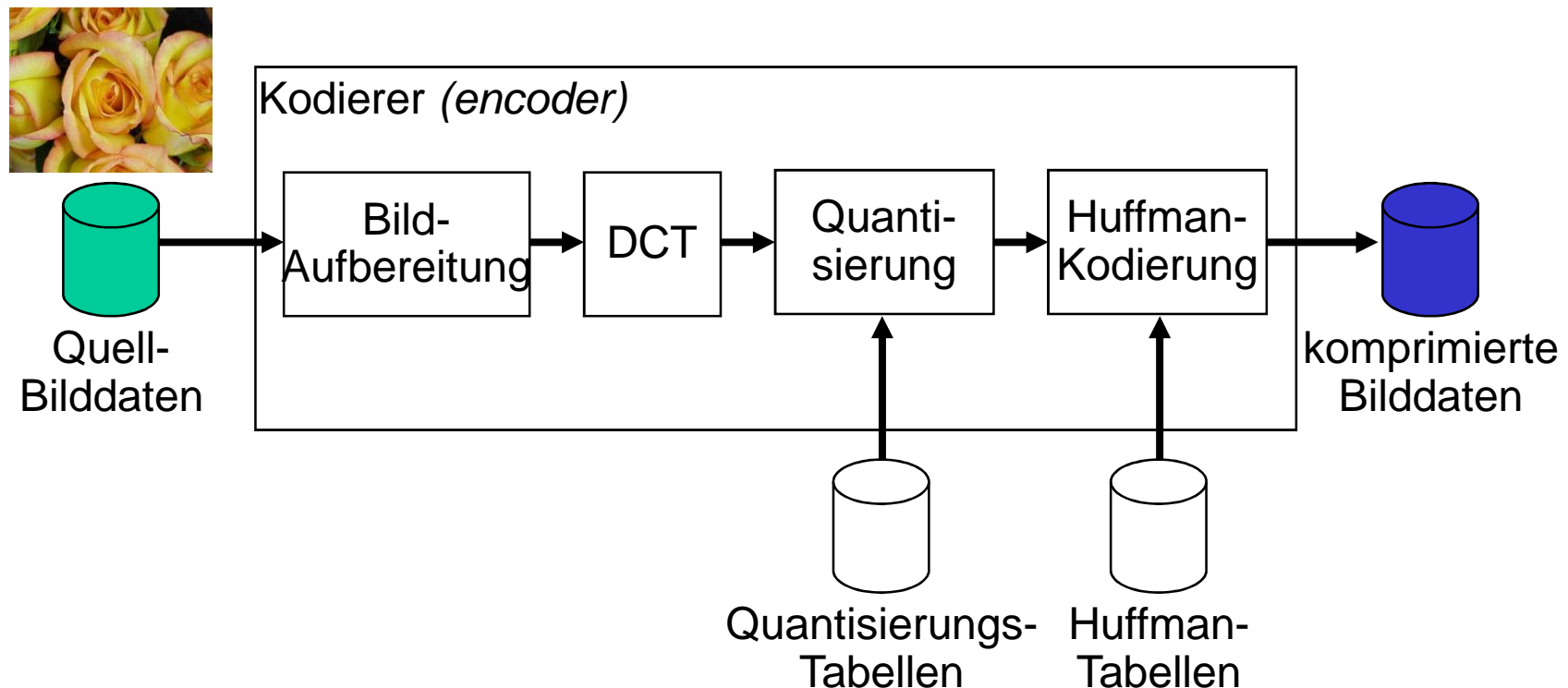


JPEG-Modi

- **Charakteristika:**
 - Verlustbehaftet oder verlustfrei
 - sequentiell, progressiv oder hierarchisch
 - Abtasttiefe (für bis zu 4 Komponenten)
 - (Entropie-)Kompressionsverfahren: Huffman- oder arithmetische Kodierung
- **Meist verwendet: Basismodus (*baseline process*):**
 - Verlustbehaftet (Diskrete Cosinus-Transformation),
 - 8 bit Tiefe, sequentiell, Huffman-Kodierung
- **Weitere Modi:**
 - **Erweiterter Modus (*extended process*):**
 - Verlustbehaftet (DCT), 8 oder 12 bit Tiefe, sequentiell oder progressiv, Huffman-Kodierung oder arithmetische Kodierung, mehr Tabellen
 - **Verlustfreier Modus (*lossless process*):**
 - Verlustfrei (kein DCT), 2 – 16 bit Tiefe, sequentiell, Huffman-Kodierung oder arithmetische Kodierung
 - **Hierarchischer Modus (*hierarchical process*):**
 - Baut auf erweitertem oder verlustfreiem Modus auf, Mehrfach-Frames

Schritte der JPEG-Kodierung

- Hier nur die gebräuchlichste Variante:
verlustbehaftet, sequentiell, 8-bit-Daten, Huffman-Kodierung



DCT = *Discrete Cosinus Transformation*

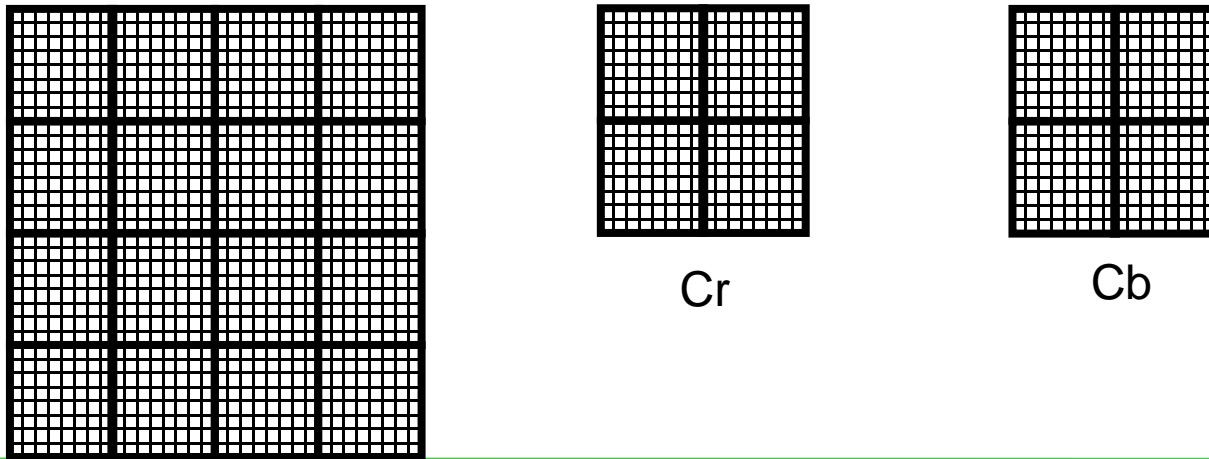
JPEG-Kodierung: Bildaufbereitung (1)

- Bild wird generell in 8 x 8-Pixel-Blöcke (*data units*) eingeteilt
 - Am Rand wird „aufgefüllt“
- Normalerweise bestehen die Bilder aus 3 Farbkomponenten
 - theoretisch bis zu 255 Komponenten (*components*) möglich
- Verzahnte (*interleaved*) oder nicht-verzahnte Reihenfolge:
 - Ablage der Komponenten nacheinander nicht ideal:
 - Z.B. könnten 3 Farbkomponenten *nacheinander* erscheinen
 - *Pipelining* in der Verarbeitung erfordert vollständige Information über einen Bildanteil
 - Verzahnte Ablage: Einheiten, die je mindestens eine *data unit* jeder Komponente enthalten: *Minimum Coded Units (MCU)*
 - Maximal vier Komponenten können verzahnt werden

JPEG-Kodierung: Bildaufbereitung (2)

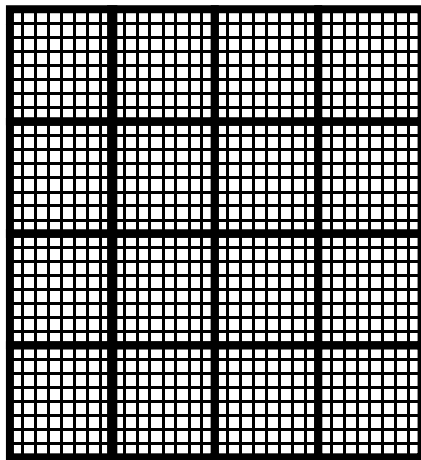
- Interleaving bei gleichzeitigem Chroma-Subsampling:
 - Jede Komponente eingeteilt in *Regionen* aus $H_c \times V_c$ Data Units (H_c und V_c Subsampling-Raten der Komponente c)
 - Jede Komponente von links oben nach rechts unten zeilenweise gespeichert
 - MCUs enthalten Data Units aus allen Komponenten anteilig

Beispiel: MCU bei 4:2:0-Subsampling
($H_Y = 4, V_Y = 4, H_{Cr} = 2, V_{Cr} = 2, H_{Cb} = 2, V_{Cb} = 2$)

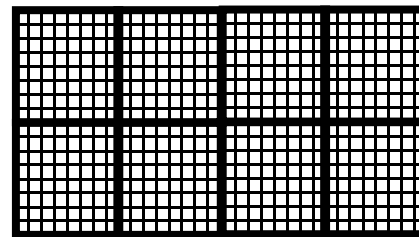


JPEG-Kodierung: Bildaufbereitung (3)

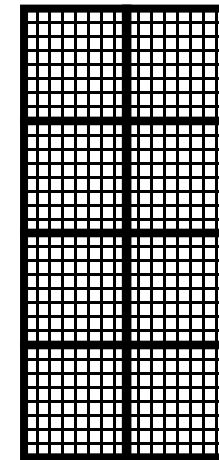
Subsampling für Y: $H_Y = 4$, $V_Y = 4$, für Cr: $H_{Cr} = 4$, $V_{Cr} = 2$, für Cb: $H_{Cb} = 2$, $V_{Cb} = 4$



Y



Cr



Cb

JFIF Dateiformat

- Der JPEG-Standard definiert das Dateiformat nicht im Detail.
- De-Facto-Standard: JFIF (JPEG File Interchange Format)
 - inoffiziell (David Hamilton)!
- Neuer offizieller Standard: SPIFF (Still Picture Interchange File Format)
 - von der JPEG
 - spät eingeführt, kompatibel mit JFIF, aber wesentlich flexibler
- JFIF definiert:
 - „Signatur“ zur Identifikation von JPEG-Dateien („JFXX“)
 - Farbraum
 - Pixeldichte
 - Vorschaubilder („Thumbnails“)
 - Zusammenhang Pixel – Abtastfrequenz

Welches Format wofür?

- Für Web-Grafiken (klein, geringe Farbanzahl)
 - GIF oder PNG
- Für Bilderzeugung mit Scanner oder Austausch über diverse Geräte hinweg:
 - TIFF
- Für hochauflösende Bilder mit vielen Farben (Fotos)
 - JPEG (wegen wesentlich besserer Kompression)
 - Bei großen einheitlichen Farbflächen evtl. auch PNG (beste Qualität)
- Zur Speicherung digitaler Originale
 - Raw-Formate (ggf. 16 Bit), DNG

Wrap Up

- Informationsverarbeitung ist immer Verarbeitung von Repräsentationen
- Bei Digitalen Medien spielen alle semiotische Ebenen eine Rolle
- Repräsentationen können durch Kodierung und Kompression optimiert werden
- Verlustfreie Kompressionsverfahren:
 - Statistische Verfahren (z.B. Huffman)
 - Lauflängenverfahren
 - Wörterbuchverfahren (z.B. LZW)
- Verlustbehaftete Verfahren
 - nutzen pragmatische Ebene aus

- Weiterführende Literatur zum Thema Kompression:

- Taschenbuch Medieninformatik Kapitel 2
- Herbert Klimant, Rudi Piotraschke, Dagmar Schönfeld:
Informations- und Kodierungstheorie, Teubner 2003
- Khalid Sayood: Introduction to Data Compression, 2nd. ed.,
Morgan Kaufmann 2000