

# 6. Licht, Farbe und Bilder

6.1 Licht und Farbe: Physikalische und physiologische Aspekte

6.2 Farbmodelle

6.3 Raster-Bilddatenformate 

– Grundbegriffe für Bildspeicherung und -Bearbeitung

– Verlustfrei komprimierende Formate 

6.4 Verlustbehaftete Kompression bei Bildern

Literatur:

John Miano: Compressed Image File Formats, Addison-Wesley 1999

# GIF wins Oxford's 'Word of the Year'

Oxford American Dictionaries elects the verb form of GIF as the word of the year. JPG and PNG declined to comment.



by Christopher MacManus | November 13, 2012 11:55 AM PST



"GIF celebrated a lexical milestone in 2012, gaining traction as a verb, not just a noun," Katherine Martin, head of U.S. dictionaries at Oxford Press, said of the image format.



<http://awesomegifs.com/2012/09/9-pretty-rad-gangnam-style-gifs/>

As an aside to the award, an Oxford press release also gave a lesson in how to pronounce GIF by saying it's appropriate to pronounce it with "a soft g (as in giant) or a hard g (as in graphic)."

"The programmers who developed the format preferred a pronunciation with a soft g (in homage to the commercial tagline of the peanut butter brand Jiff, they supposedly quipped 'choosy developers choose GIF')."

# Portable Network Graphics PNG („Ping“)

- Geschichte:
  - Ausgelöst durch Lizenzforderungen für GIF-Format (1994)
  - Arbeitsgruppe beim W3C für PNG, standardisiert 1996, offen und lizenzfrei
- Ziel:
  - Besserer Ersatz für GIF, teilweise auch Ersatz für JPEG
  - Langsam zunehmende praktische Verbreitung, z.B. durch Wikipedia
- Farbtiefen:
  - 24 oder 48 bit „TrueColor“, 8 oder 16 bit Graustufen, Paletten bis 256 Farben (optional)
- Hauptvorteile:
  - Völlig verlustfrei ("Deflate"-Algorithmus: Lempel-Ziv + Huffman)
  - Echter Alpha-Kanal
  - Gamma-Korrektur (Gamma-Wert der Quellplattform speicherbar)
  - Verbessertes Interlacing (7-Pass-Algorithmus „Adam7“)
  - Bessere Kompression (Kompressionsfilter)
  - Integritätstest für Dateien (*magic signature*, CRC-32)

# Echter Alpha-Kanal in PNG

- Alpha-Werte pro Pixel gespeichert
  - 4 Bytes pro Pixel: „RGBA“-Farbmodell
  - Ermöglicht elegante Schatten und Übergänge zwischen Grafik und Hintergrund
- Vermeidet Wechselwirkungen zwischen Anti-Aliasing und Transparenzfarbe
  - Bei „binärer Transparenz“ wie in GIF oft „weißer Rand“ um transparente Grafiken aufgrund von Anti-Aliasing (erzeugt nicht-transparente Farben)



# Kompressionsverbesserung durch Filter in PNG

- Beispiel:
  - Wertfolge 1, 2, 3, 4, 5, 6, 7, 8, 9, ...
  - Komprimiert extrem schlecht mit LZ-artigen Algorithmen
- Filter (Prädiktion):
  - Ersetze alle Zahlen (außer der ersten) durch die Differenz zur vorhergehenden
  - Wertfolge: 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
  - Komprimiert exzellent! (viele Wiederholungen)
- Filter in PNG:
  - Sub: Differenz zum linksstehenden Byte
  - Up: Differenz zum darüberstehenden Byte
  - Average: Differenz zum Durchschnitt der Sub- und Up-Bytes
  - Paeth: Differenz zum *Paeth-Prediktor* (siehe nächste Folie)
    - » Benutzt linksstehendes, darüberstehendes und "links oben" stehendes Byte
  - Heuristiken zur Wahl des passenden Filters

# Paeth-Prediktor

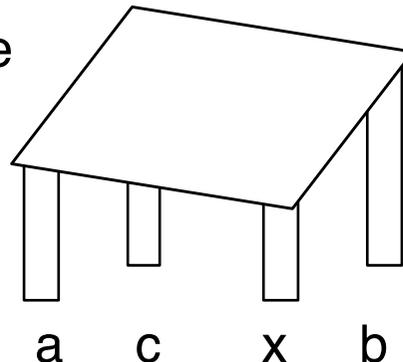
	$c$	$b$	
	$a$	$x$	

$$Px = a + b - c$$

Erfinder: Alan W. Paeth

- Den Prädiktor  $a+b-c$  kann man sich am einfachsten algebraisch erklären:
- Seien  $Ra = f(x_1, y_1)$ ,  $Rb = f(x_2, y_2)$ ,  $Rc = f(x_1, y_2)$ ,  $Rx = f(x_2, y_1)$ .
- Sei  $f$  linear in  $x$  und  $y$ , d.h.  $f(x, y) = Ax + By$ .
- $Ra + Rb - Rc = Ax_1 + By_1 + Ax_2 + By_2 - Ax_1 - By_2 = Ax_2 + By_1 = Rx$

Geometrische  
Erklärung:



# 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)

# 6. Licht, Farbe und Bilder

- 6.1 Licht und Farbe: Physikalische und physiologische Aspekte
- 6.2 Farbmodelle
- 6.3 Raster-Bilddatenformate
- 6.4 Verlustbehaftete Kompression bei Bildern 

Literatur:

John Miano: Compressed Image File Formats, Addison-Wesley 1999

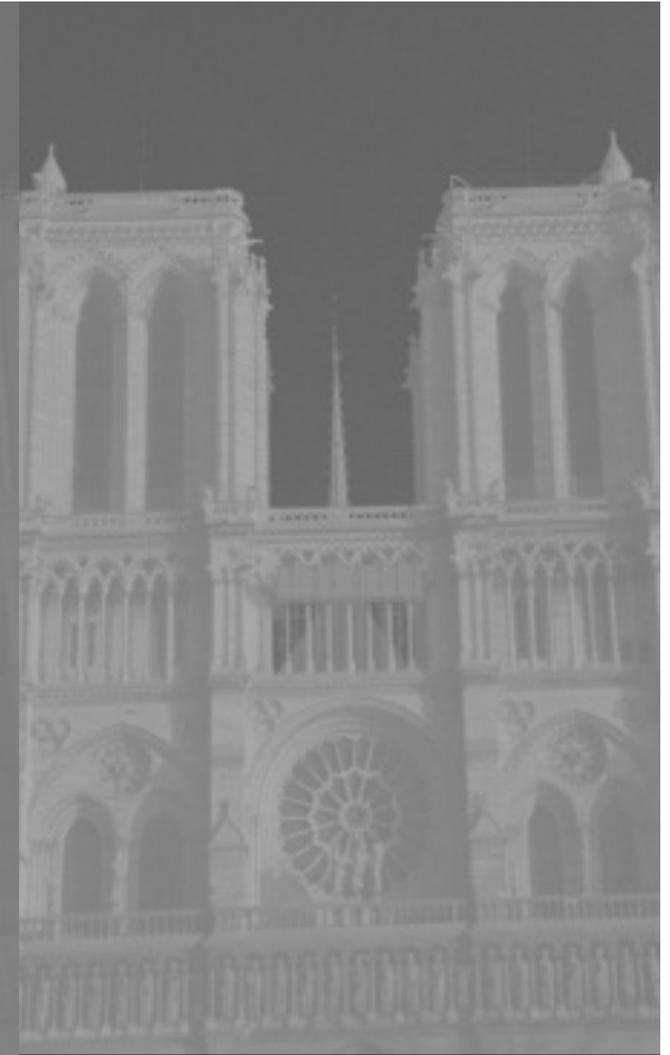
# Luma- und Chromainformation: Vergleich



Helligkeit (L-Kanal)

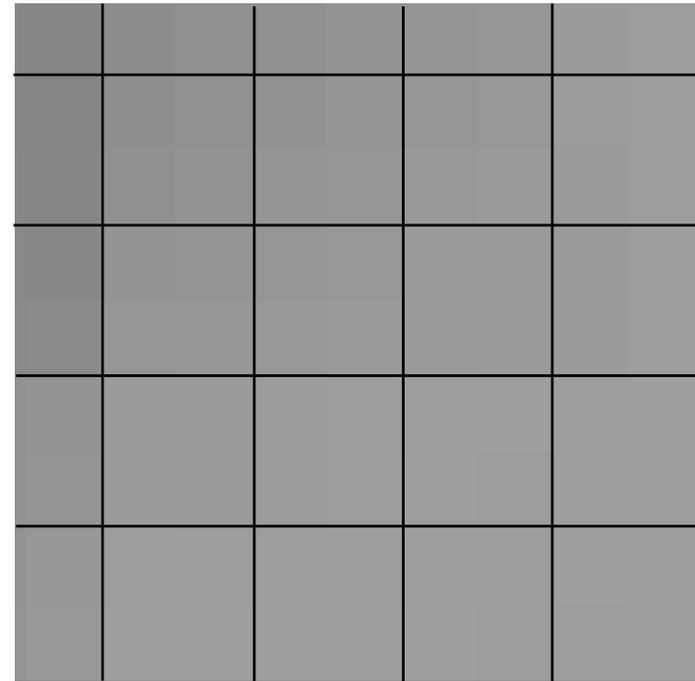
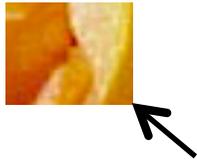


Rot/Grün-Differenz  
(a-Kanal)



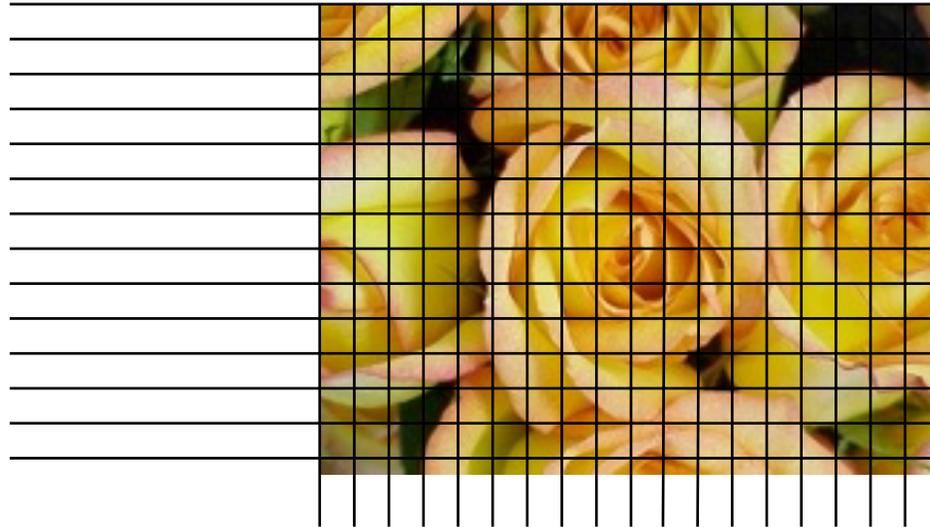
Blau/Gelb-Differenz  
(b-Kanal)

# Chroma-Subsampling



- Differenzierte Auflösung
  - niedrig für die Farbinformation (Chroma, Cr+Cb)
  - hoch für die Helligkeit (Luma, Y)
- Passende Farbmodelle: YCrCb, YUV, YIQ, Lab
- Speicherplatzersparnis im Beispiel 50% (bei gleichem Subsampling für b)

# Abtastraten für Bilder



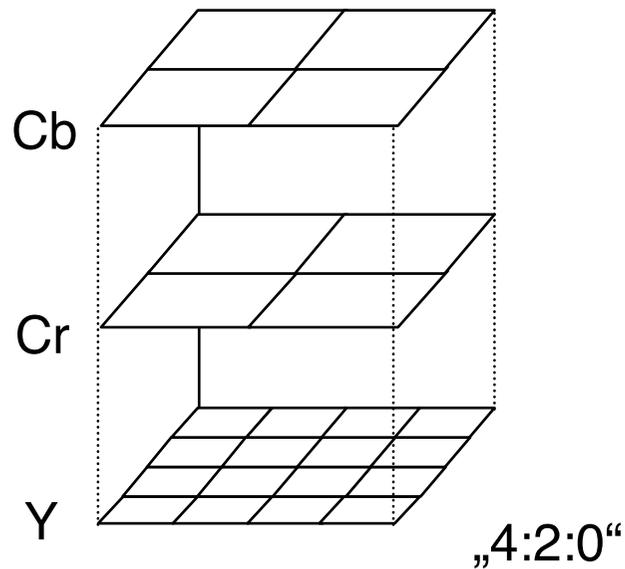
- Abtastrate: Wieviele Pixel pro Längeneinheit des Bildes?
- Mehrdimensionalität:
  - Horizontale Abtastrate ( $H$ )
  - Vertikale Abtastrate ( $V$ )
- Bei Sub-Sampling:
  - Verschiedene Abtastraten für verschiedene Komponenten des Bildes (Farben, evtl. Alphakanal)

# Subsampling

$$Y: H_Y = 4, V_Y = 4$$

$$Cr: H_{Cr} = 2, V_{Cr} = 2$$

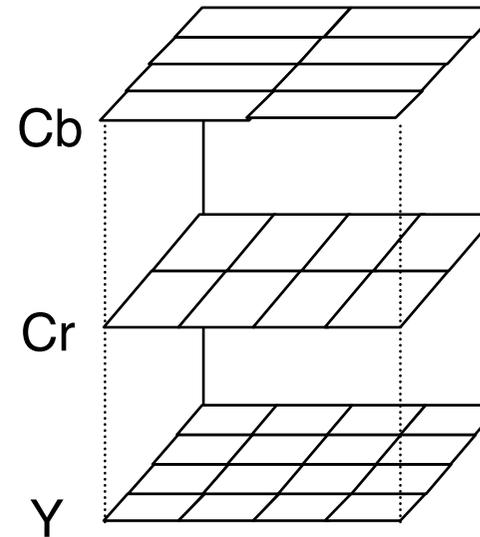
$$Cb: H_{Cb} = 2, V_{Cb} = 2$$



$$Y: H_Y = 4, V_Y = 4$$

$$Cr: H_{Cr} = 4, V_{Cr} = 2$$

$$Cb: H_{Cb} = 2, V_{Cb} = 4$$

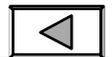


- $H$  und  $V$ : Zahl der berücksichtigten Pixel je 4x4-Block (*subsampling rate*)
  - horizontal und vertikal
- Subsampling bei verschiedenen digitalen Bildverarbeitungstechniken benutzt
  - in JPEG (optional)
  - auch in diversen digitalen Video-Aufzeichnungs-Standards

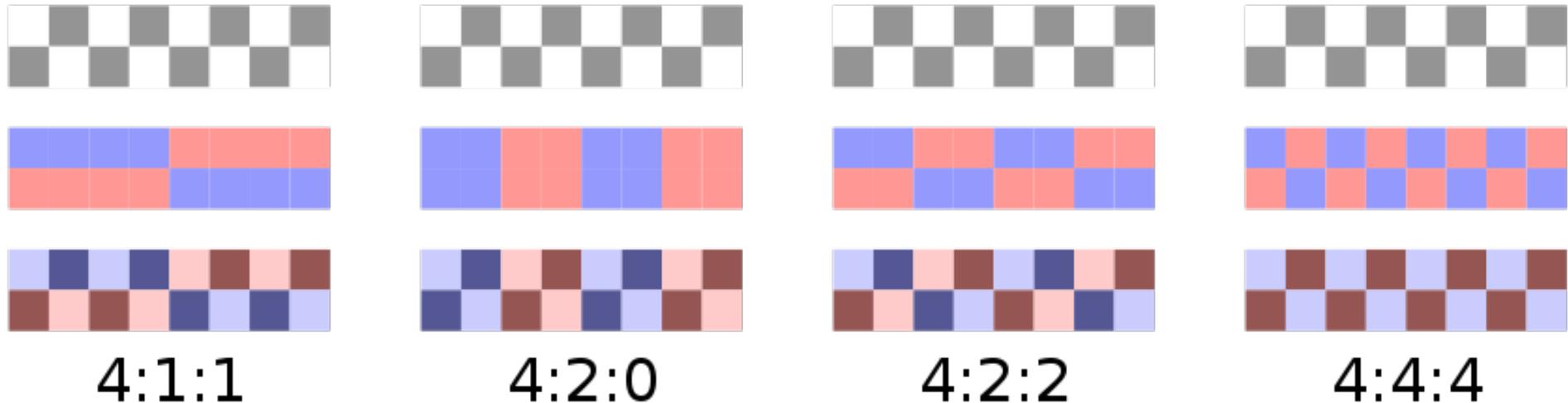


# Notation für Subsampling

- Übliche Notation für Subsampling von Farben:  
x:y:z
- Meist benutzte Bedeutung:
  - Beide Chroma-Kanäle immer gleich abgetastet
  - x: Anzahl der Luma-Samples; in der Regel „4“
  - y: Anzahl der Cr/Cb-Chroma-Samples, horizontal, in der ersten Zeile (horizontales Subsampling)
  - z: Anzahl der Cr/Cb-Chroma-Samples, horizontal, die in der zweiten Zeile *nicht* aus der ersten übernommen werden (indirekt: vertikales Subsampling)
- z=y: kein vertikales Subsampling,
- z=0: vertikales Subsampling 2:1
- Beispiele :
  - 4:2:2  $H_Y=4, V_Y=4, H_{Cr}=2, V_{Cr}=4, H_{Cb}=2, V_{Cb}=4$
  - 4:1:1  $H_Y=4, V_Y=4, H_{Cr}=1, V_{Cr}=4, H_{Cb}=1, V_{Cb}=4$
  - 4:2:0  $H_Y=4, V_Y=4, H_{Cr}=2, V_{Cr}=2, H_{Cb}=2, V_{Cb}=2$  (bei JPEG weit verbreitet)
- Bandbreitenformel: Summe der drei Zahlen geteilt durch 12



# Beispiele zur Notation für Subsampling



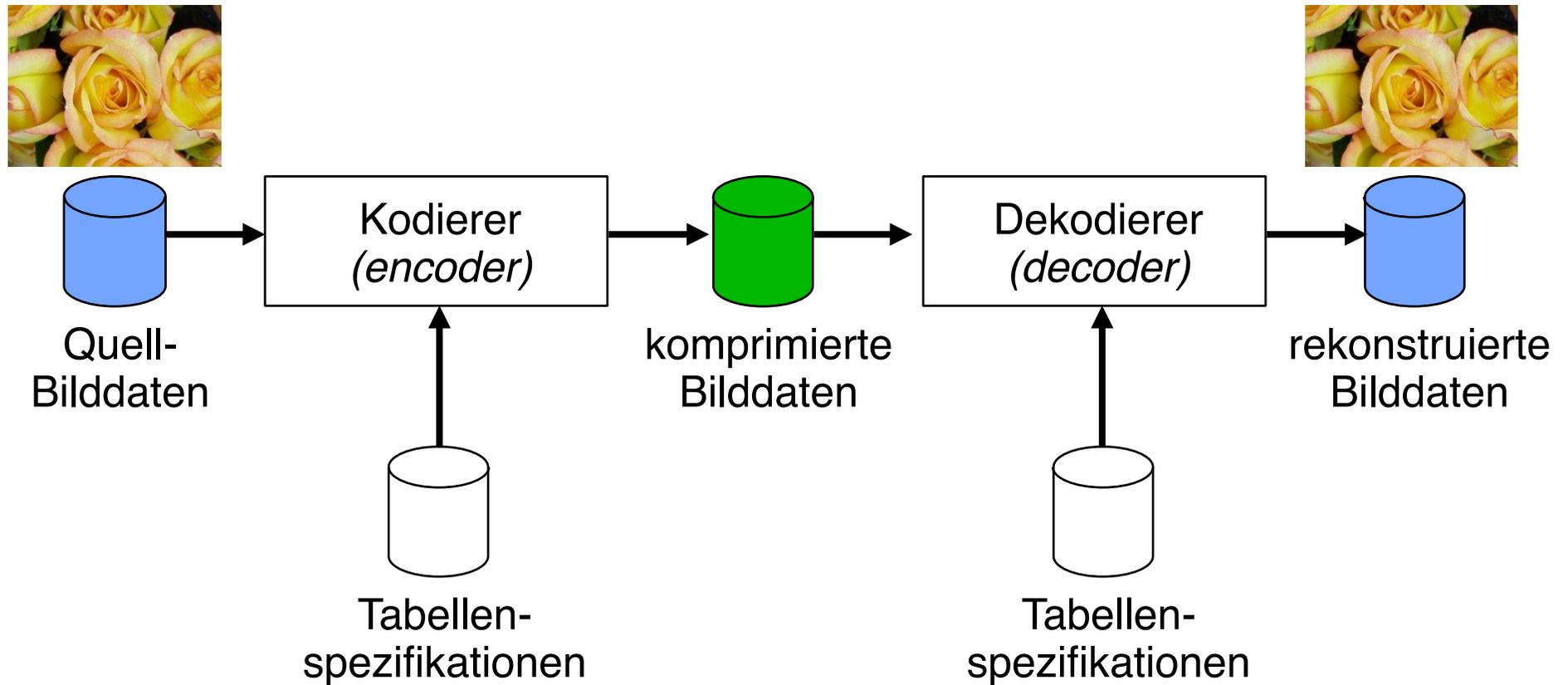
- x: Anzahl der Luma-Samples; in der Regel „4“
- y: Anzahl der Cr/Cb-Chroma-Samples, horizontal, in der ersten Zeile
- z: Anzahl der Cr/Cb-Chroma-Samples, horizontal, in der zweiten Zeile die in der zweiten Zeile *nicht* aus der ersten übernommen werden

Bild: Wikipedia

# JPEG: Hintergrundinformation

- 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



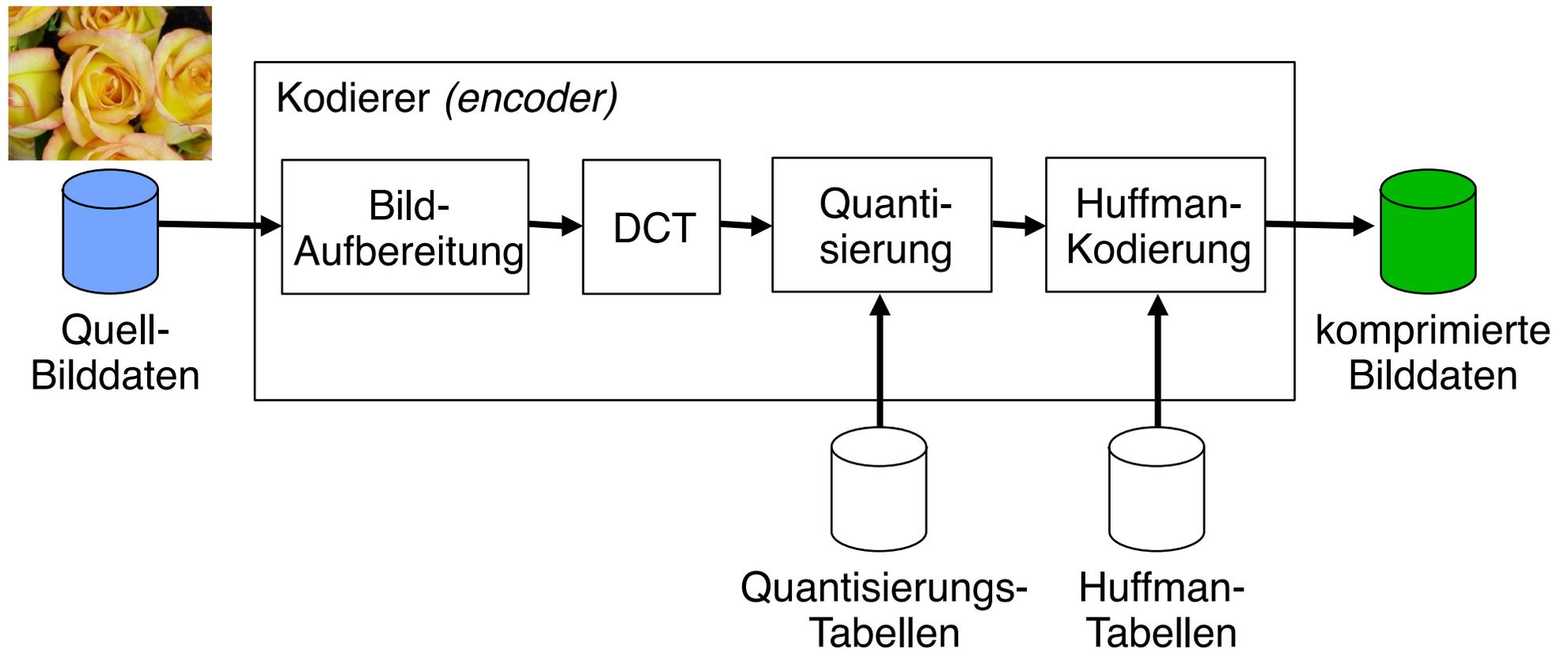
# JPEG-Modi

- Charakteristika:
  - Verlustbehaftet oder verlustfrei
  - sequentiell, progressiv oder hierarchisch
  - Abtasttiefe (für bis zu 4 Komponenten)
  - (Entropie-)Kompressionsverfahren: Huffman- oder arithmetische Kodierung
- **Basismodus (*baseline process*):**
  - Verlustbehaftet (DCT), 8 bit Tiefe, sequentiell, Huffman-Kodierung
- **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

meist verwendet
selten verwendet
ungebräuchlich

# Schritte der JPEG-Kodierung

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



DCT = *Discrete Cosinus Transformation*



# JPEG-Kodierung: Bildaufbereitung (1)

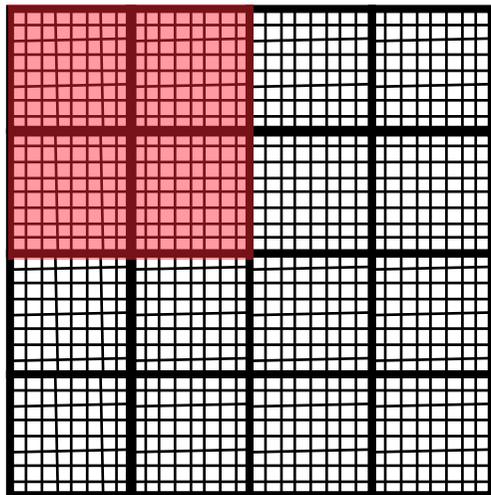
- Bild in 8 x 8-Pixel-Blöcke (*data units*) eingeteilt
  - Am Rand wird „aufgefüllt“
- Theoretisch bis zu 255 Komponenten (*components*)
  - Verbreitet: 3 oder 4, nach Farbmodell
- Verzahnte (*interleaved*) Reihenfolge:
  - Ablage der Komponenten nacheinander:
    - » Z.B. 3 Farbkomponenten *nacheinander*
    - » Problematisch bei *Pipelining*
  - Verzahnte Ablage in *Minimum Coded Units (MCU)*:  
Einheiten enthalten mindestens eine *data unit* jeder Komponente:

# JPEG-Kodierung: Bildaufbereitung (2)

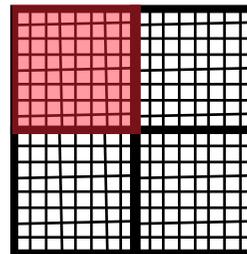
Subsampling

- Interleaving bei gleichzeitigem Chroma-Subsampling:
  - Jede Komponente  $c$  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

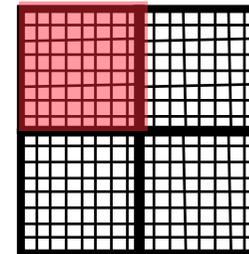
Beispiel: MCU bei 4:2:0-Subsampling  
( $H_Y = 4, V_Y = 4, H_{Cr} = 2, V_{Cr} = 2, H_{Cb} = 2, V_{Cb} = 2$ )



Y



Cr

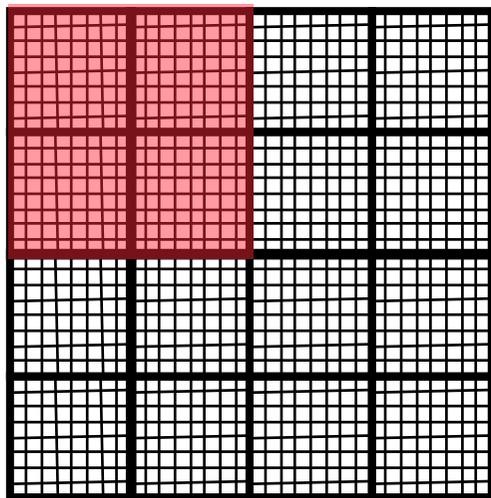


Cb

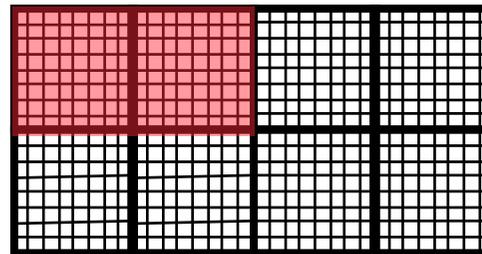
# JPEG-Kodierung: Bildaufbereitung (3)

Subsampling

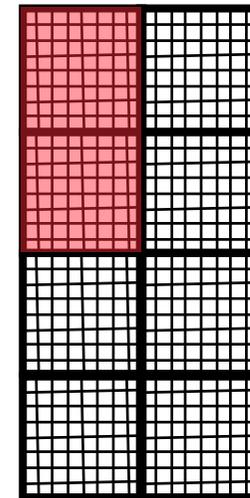
Subsampling für Y:  $H_Y = 4$ ,  $V_Y = 4$ , für Cr:  $H_{Cr} = 4$ ,  $V_{Cr} = 2$ , für Cb:  $H_{Cb} = 2$ ,  $V_{Cb} = 4$



Y



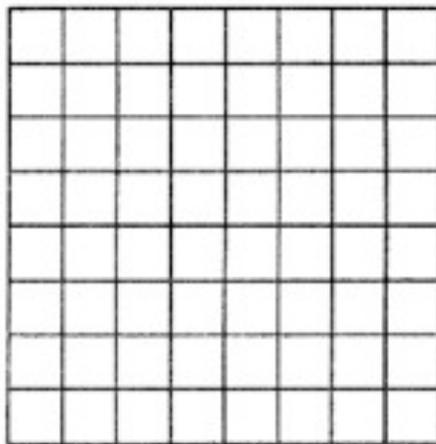
Cr



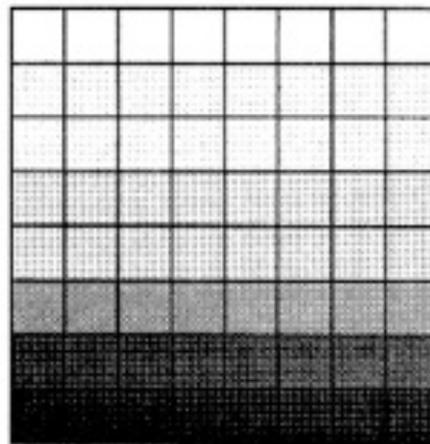
Cb

# Ortsfrequenz

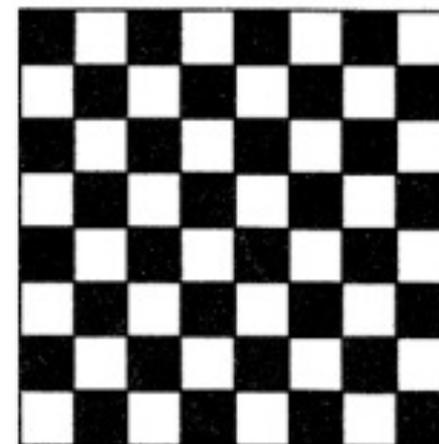
- Ortsfrequenz (oder: räumliche Frequenz, *spatial frequency*)
  - Häufigkeit der Wiederholung einer im Bild erkennbaren Eigenschaft über die räumliche Ausdehnung
  - Maßeinheit: 1/Längeneinheit
  - z.B. Dichte von Linien auf Papier: Anzahl Striche pro cm
- Meist: Anzahl von Helligkeitsschwankungen pro Längeneinheit
- 2-dimensionale Frequenz (horizontal und vertikal)



Ortsfrequenz 0



Ortsfrequenz  
0 horizontal,  
niedrig vertikal

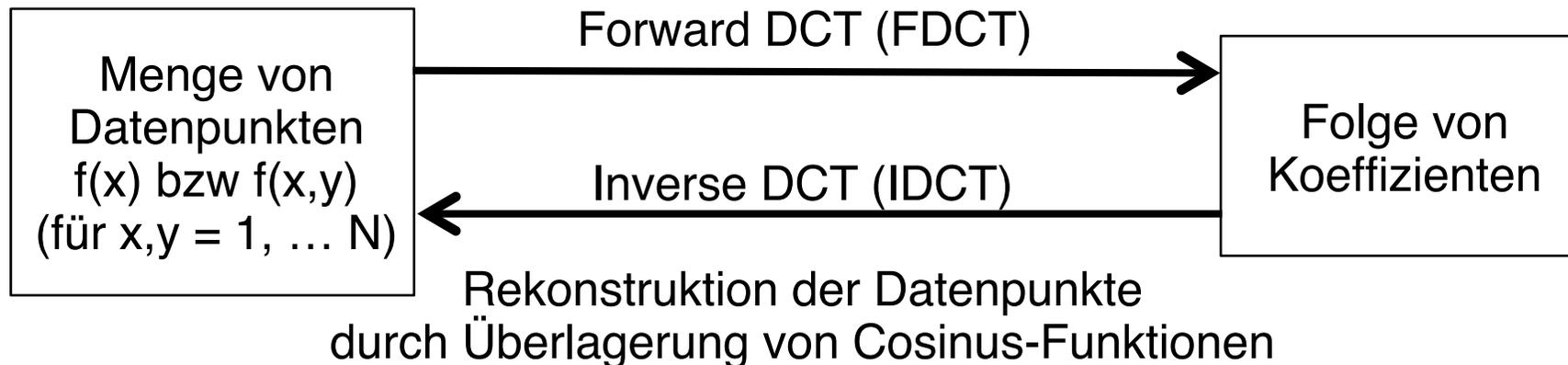


Ortsfrequenz  
hoch  
horizontal und vertikal

# Diskrete Cosinus-Transformation (DCT)

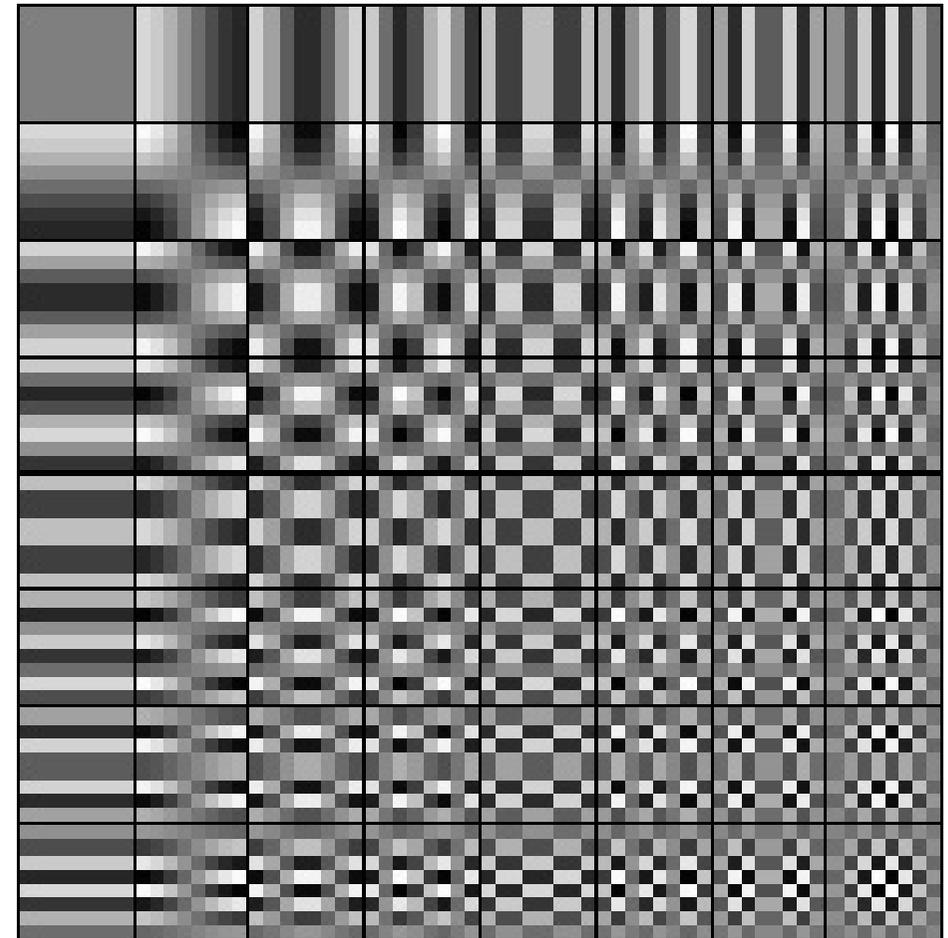
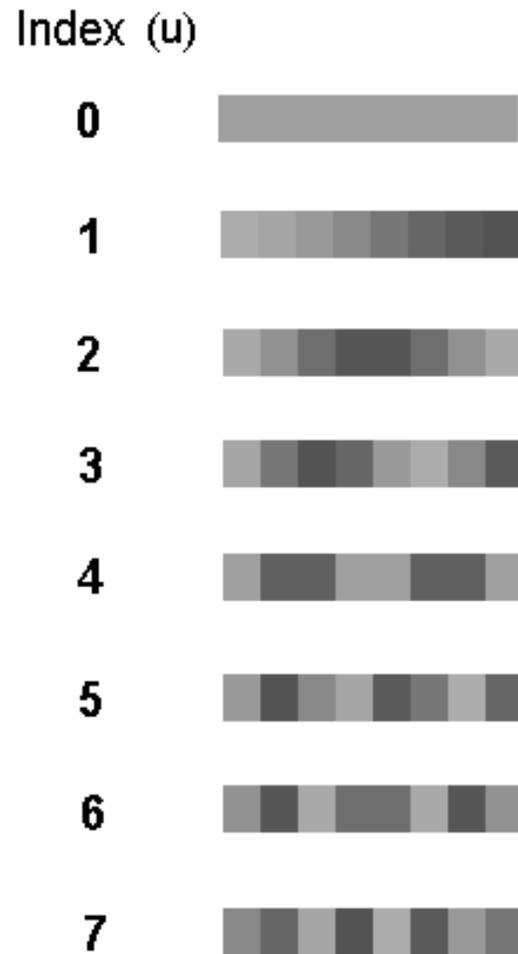
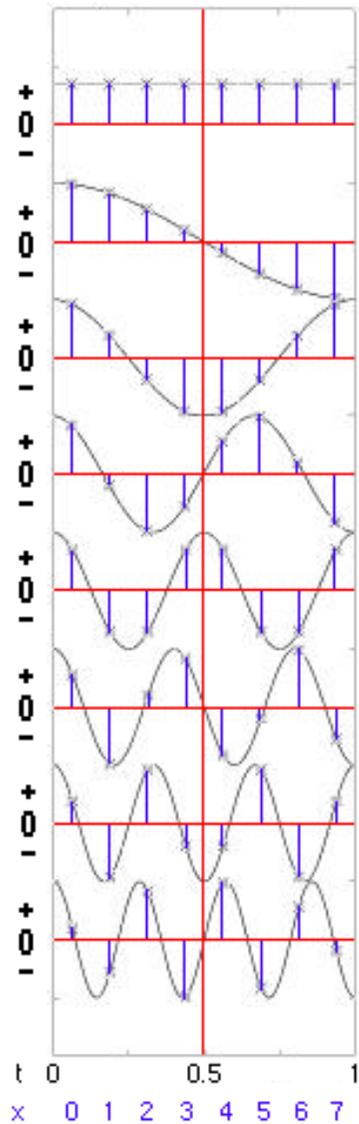
JPEG-Schritte

- Grundmotivation:
  - Menschliche Sehvernehmung sehr empfindlich für niedrige und mittlere Frequenzen (Flächen, deutliche Kanten), wenig empfindlich für hohe Frequenzen (z.B. feine Detaillinien)
  - Deshalb Zerlegung der Bildinformation in Frequenzanteile (ähnlich zu Fourier-Transformation)
- Prinzip von DCT:
  - (in einer oder zwei Dimensionen...)



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

# Basisfunktionen der DCT in 1D und 2D



# (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

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

$c_u, c_v = 1$  sonst

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

# Matrixdarstellung zur Durchführung einer DCT

$$\begin{bmatrix}
 \frac{1}{4}\sqrt{2} & \frac{1}{4}\sqrt{2} \\
 \frac{1}{2}\cos\left(\frac{1}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{3}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{5}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{7}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{9}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{11}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{13}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{16}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{1}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{3}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{5}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{7}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{9}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{11}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{13}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{8}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{3}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{9}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{21}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{27}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{33}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{39}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{45}{16}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{1}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{3}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{5}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{7}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{9}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{11}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{13}{4}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{4}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{5}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{25}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{35}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{45}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{55}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{65}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{75}{16}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{3}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{9}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{15}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{21}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{27}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{33}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{39}{8}\pi\right) & \frac{1}{2}\cos\left(\frac{45}{8}\pi\right) \\
 \frac{1}{2}\cos\left(\frac{7}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{21}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{35}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{49}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{63}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{77}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{91}{16}\pi\right) & \frac{1}{2}\cos\left(\frac{105}{16}\pi\right)
 \end{bmatrix}$$

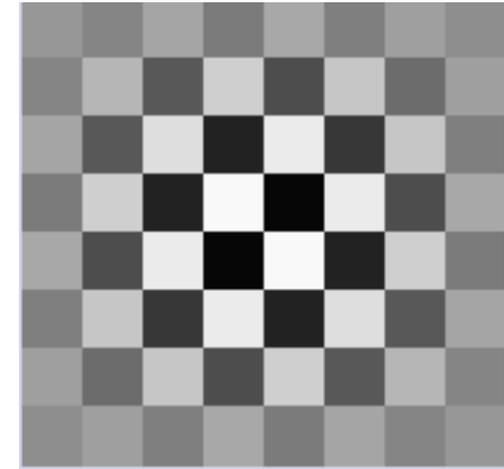
# Beispiele für DCT-Transformation



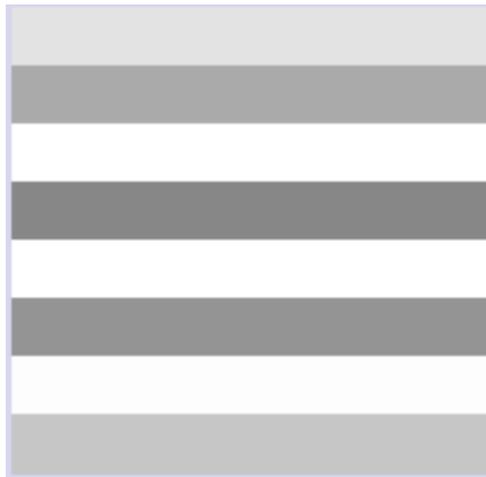
$F(0,1) = 500$ ,  
alle anderen  $F(u, v) = 0$



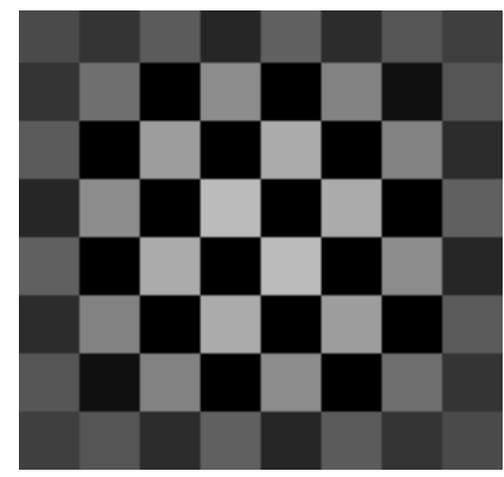
$F(7,0) = 500$ ,  
alle anderen  $F(u, v) = 0$



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



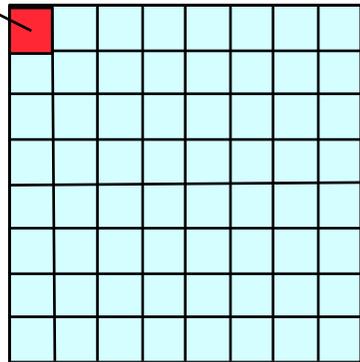
$F(7,0) = 500$ ,  $F(0,0) = 600$   
alle anderen  $F(u, v) = 0$



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

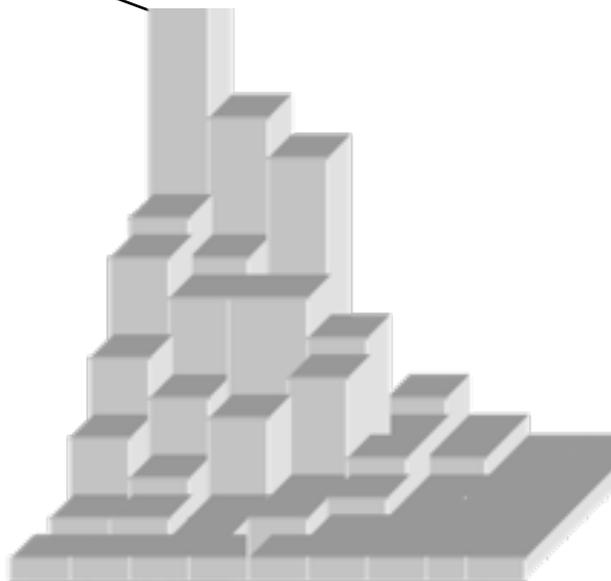
# Interpretation der DCT-Koeffizienten

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 waagrechte Streifen vorkommen;
  - $F(0,7)$  gibt an, zu welchem Anteil extrem dichte senkrechte Streifen vorkommen

DC = Gleichstrom  
AC = Wechselstrom

# 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

JPEG-Schritte

- Entscheidender Schritt zum **Informationsverlust** und damit zur starken Kompression !
  - Runden der Koeffizienten: Verbessert nachfolgenden verlustfreie Kompression
- Quantisierungstabelle:
  - 64 vorgegebene und konstante *Bewertungs*-Koeffizienten  $Q(u, v)$
  - Bewertung = Division, also größerer Wert = stärkere Vergrößerung
  - Konkrete Tabellen nicht Bestandteil des Standards (nur zwei Beispiele)
  - Quantisierungstabellen werden als Bestandteil der komprimierten Daten abgelegt
- Berechnung:
  - $F(u,v)$ : Matrix-Koeffizient vor Quantisierung
  - $Q(u,v)$ : Quantisierungs-Koeffizient
  - $F'(u,v)$ : Matrix-Koeffizient nach Quantisierung

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

Typische  
Tabelle

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

# Rechenbeispiel: Quantisierung

$$\begin{pmatrix} 93 & 2 & -8 & -7 & 3 & 1 & 1 & -2 \\ -38 & -58 & 11 & 17 & -3 & 5 & 5 & -3 \\ -84 & 63 & -1 & -17 & 2 & 7 & -4 & 0 \\ -51 & -37 & -10 & 13 & -10 & 5 & -1 & -4 \\ -85 & -42 & 50 & -8 & 18 & -5 & -1 & 1 \\ -63 & 66 & -13 & -1 & 2 & -6 & -2 & -2 \\ -16 & 14 & -37 & 18 & -12 & 4 & 3 & -3 \\ -53 & 31 & -7 & -10 & 23 & 0 & 2 & 2 \end{pmatrix}$$

DCT-Koeffizienten

$$\begin{pmatrix} 31 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -7 & -8 & 1 & 1 & 0 & 0 & 0 & 0 \\ -12 & 7 & 0 & -1 & 0 & 0 & 0 & 0 \\ -5 & -3 & 0 & 0 & 0 & 0 & 0 & 0 \\ -7 & -3 & 3 & 0 & 0 & 0 & 0 & 0 \\ -4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

quantisierte DCT-Koeffizienten

$$\begin{pmatrix} 3 & 5 & 7 & 9 & 11 & 13 & 15 & 17 \\ 5 & 7 & 9 & 11 & 13 & 15 & 17 & 19 \\ 7 & 9 & 11 & 13 & 15 & 17 & 19 & 21 \\ 9 & 11 & 13 & 15 & 17 & 19 & 21 & 23 \\ 11 & 13 & 15 & 17 & 19 & 21 & 23 & 25 \\ 13 & 15 & 17 & 19 & 21 & 23 & 25 & 27 \\ 15 & 17 & 19 & 21 & 23 & 25 & 27 & 29 \\ 17 & 19 & 21 & 23 & 25 & 27 & 29 & 31 \end{pmatrix}$$

Quantisierungsmatrix

<http://www.mathematik.de/>

# Informationsverlust durch Quantisierung



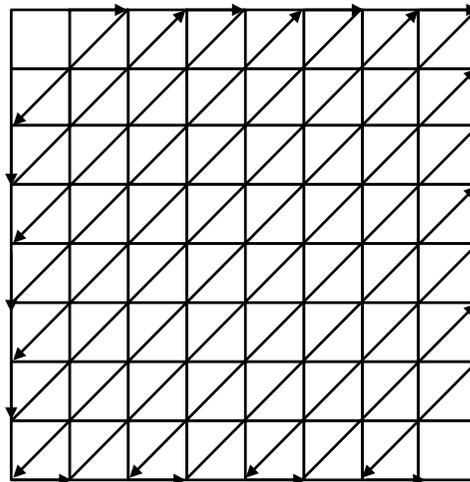
JPEG: Grad der Quantisierung wählbar:  
„Trade-Off“ zwischen Speicherplatzersparnis  
und Bildverfälschung (Artefakten)



Artefakte bei  
Kanten und Details,  
kaum bei Flächen

# Vorbereitung zur Weiterverarbeitung

- Quantisierte Frequenzwerte:
  - in linearer Reihenfolge ausgegeben
  - unterschiedliche Behandlung DC- und AC-Koeffizienten
- DC-Koeffizienten:
  - Separat extrahiert (Grobbild mit 8x8-Blöcken)
- AC-Koeffizienten:
  - Ausgabe nach absteigender Frequenz („Zick-Zack“)



# JPEG-Kodierung: Entropie-Kompression

JPEG-Schritte

- 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

# JFIF Dateiformat

- Der JPEG-Standard definiert das Dateiformat nicht im Detail.
- De-Facto-Standard: JFIF (JPEG File Interchange Format)
  - inoffiziell (David Hamilton 1992)
- 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