


4. Licht, Farbe und Bilder

- 4.1 Licht und Farbe: Physikalische und physiologische Aspekte
- 4.2 Farbmodelle
- 4.3 Raster-Bilddatenformate
- 4.4 Verlustbehaftete Kompression bei Bildern 
JPEG
- 4.5 Bewegte Bilder: Grundbegriffe der digitalen Videotechnik

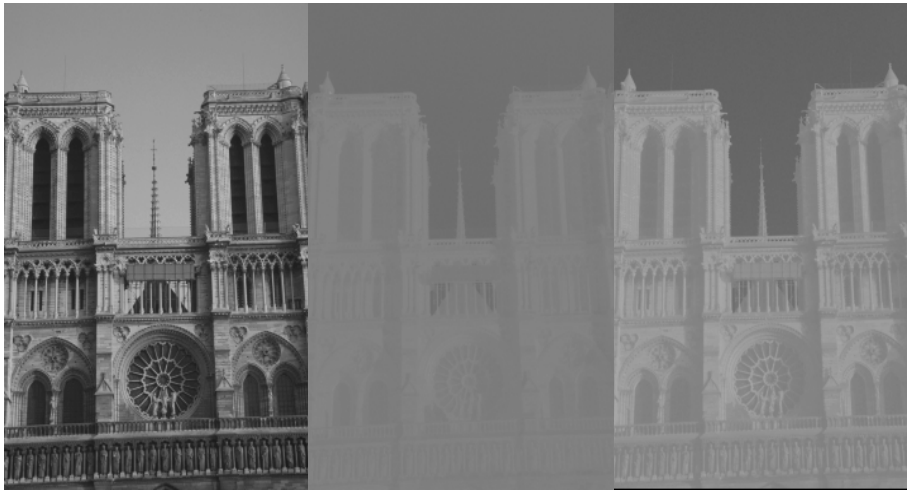
Weiterführende Literatur:

John Miano: Compressed Image File Formats - JPEG, PNG, GIF, XBM, BMP,
Addison-Wesley 1999

Warum und wann verlustbehaftet komprimieren?

- Durch Aufnahme aus der realen Welt erzeugte Bilder (v.a. Fotos) sind sehr groß
 - z.B. 4 Mio. Pixel mit je 24 bit = 12 MByte
- Das menschliche Auge wertet nicht alle Informationen des Bildes gleich gut aus
 - z.B. Helligkeit vs. Farbigkeit
 - z.B. Feinabstufungen von Verläufen
- Mit verlustbehafteten Kompressionsverfahren wird
 - ein oft sehr hoher Gewinn an Speicherplatz erzielt
 - der subjektive Eindruck des Bildes kaum verändert
- Bekanntestes Verfahren: JPEG
- Achtung: Für Archivierung von hochwertigen Bild-Originalen eignet sich JPEG nur bedingt (bei Einstellung von geringen Kompressionsgraden)
 - Alternativen z.B.: TIFF, PNG

Luma- und Chromainformation: Vergleich

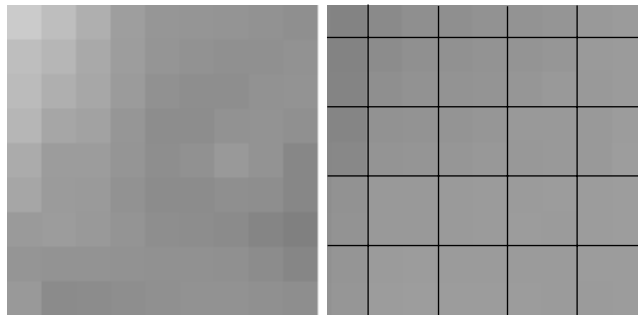


Helligkeit (L-Kanal)

Rot/Grün-Differenz
(a-Kanal)

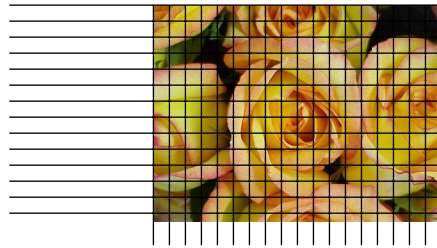
Blau/Gelb-Differenz
(b-Kanal)

Chroma-Subsampling



- In vielen Fällen genügt eine geringere Auflösung für die Farbinformation (Chroma) als für die Helligkeit (Luma).
 - Passende Farbmodelle: YUV, YIQ, Lab
 - Teilweise aber abhängig vom Darstellungsinhalt
- Chroma-Subsampling = niedrigere Abtastrate für Farbinformation
 - Speicherplatzersparnis im Beispiel 1/3 (bei gleichem Subsampling für b)

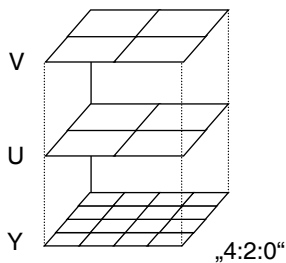
Abtastraten für Bilder



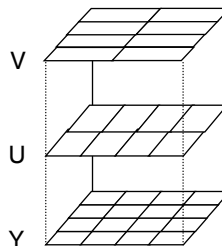
- Abtastrate: Wieviele Pixel pro Längeneinheit des Bildes?
- Mehrdimensionalität:
 - Horizontale Abtastrate (H)
 - Vertikale Abtastrate (V)
- Bei Sub-Sampling:
 - Verschiedene Abtastraten für verschiedene Komponenten des Bildes (Farben, evtl. Alphakanal)

Abtastraten und Subsampling

Y: $H = 4, V = 4$
U: $H = 2, V = 2$
V: $H = 2, V = 2$



Y: $H = 4, V = 4$
U: $H = 4, V = 2$
V: $H = 2, V = 4$



- Subsampling wird bei verschiedenen digitalen Bildverarbeitungstechniken benutzt
 - in JPEG (optional)
 - auch in diversen digitalen Video-Aufzeichnungs-Standards

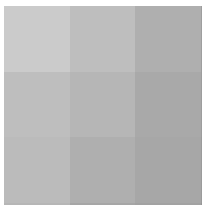


Notation für Subsampling

- Übliche Notation für Subsampling von Farben:
x:y:z
– z.B.: 4:2:2
- Ursprüngliche Bedeutung: Horizontales Frequenzverhältnis für Luma (x) zu den beiden Chroma-Kanälen (y, z)
 - Vertikales Subsampling?
 - Beide Chroma-Kanäle haben in der Praxis immer die gleiche Abtastrate !
- Heutige Bedeutung:
 - x: In der Praxis meist „4“, historische Reminiszenz
 - y: Faktor für das Subsampling von Chroma gegenüber Luma
 - z: Falls z=y: kein vertikales Subsampling
Falls z=0: vertikales Subsampling von 2:1 für beide Chroma-Kanäle
- Beispiele:
 - 4:2:0 $H_V=4, V_V=4, H_U=2, V_U=2, H_V=2, V_V=2$
(erstes Beispiel letzte Folie; bei JPEG im praktischen Einsatz)
 - 4:2:2 $H_V=4, V_V=4, H_U=2, V_U=2, H_V=4, V_V=4$



Prädiktive Codierung: Grundidee



| | | |
|----|----|----|
| 45 | 48 | 60 |
| 50 | 55 | 65 |
| 52 | 60 | 70 |

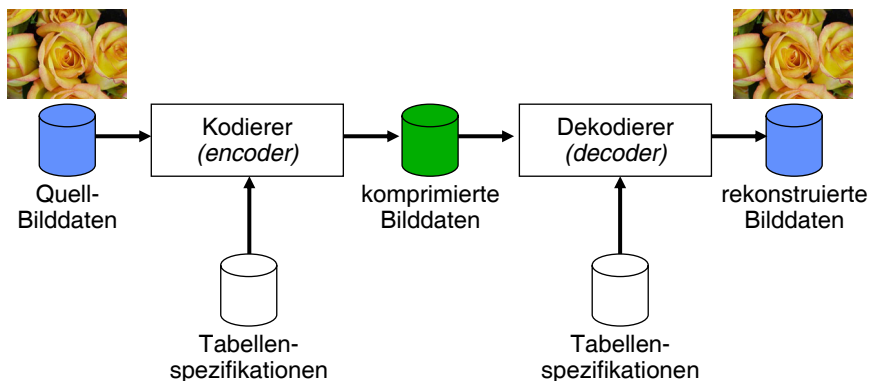
| | | |
|-----|----|-----|
| -10 | -7 | +5 |
| -5 | 55 | +10 |
| -3 | +5 | +15 |

- 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
- Prinzipiell verlustfrei, in JPEG (u. a.) verfeinerte verlustbehaftete Variante der gleichen Idee

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



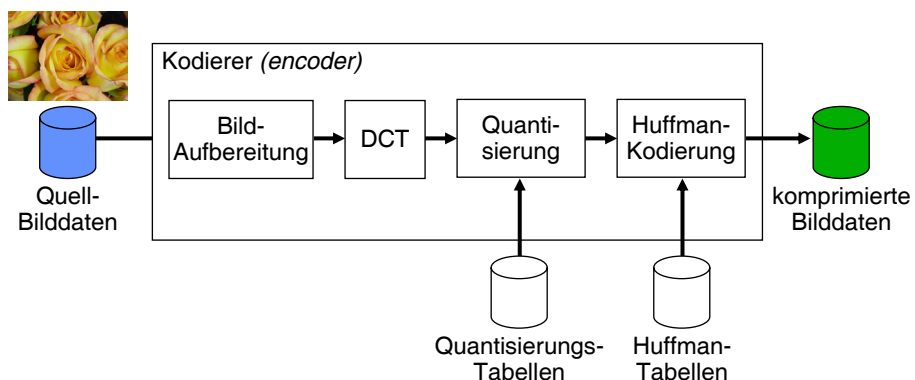
JPEG-Modi

- Charakteristika:
 - Verlustbehaftet oder verlustfrei
 - sequentiell, progressiv oder hierarchisch
 - Abtasttiefe (für bis zu 4 Komponenten)
 - (Entropie-)Kompressionsverfahren: Huffman- oder arithmetische Kodierung
- **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 wird generell in 8 x 8-bit-Blöcke (*data units*) eingeteilt
 - Am Rand wird „aufgefüllt“
- Bild kann theoretisch aus bis zu 255 Komponenten (*components*) bestehen
 - Verbreitet: 3 oder 4, nach Farbmodell
- Verzahnte (*interleaved*) oder nicht-verzahnte Reihenfolge:
 - Ablage der Komponenten nacheinander nicht immer ideal:
Z.B. könnten 3 Farbkomponenten *nacheinander* erscheinen
 - Verzahnte Ablage: Einheiten, die je eine *data unit* jeder Komponente enthalten: *Minimum Coded Units (MCU)*
 - Maximal vier Komponenten können verzahnt werden

JPEG-Kodierung: Bildaufbereitung (2)

- Bildung von MCUs bei Subsampling und Interleaving?
 - Beispiel-Annahme: Typisches Subsampling nach „4:2:0“
 - Die gleiche Bildregion wird abgedeckt von:
 - » 4 Dateneinheiten von Y
 - » 1 Dateneinheit von U
 - » 1 Dateneinheit von V
 - MCU = 4 Y-Dateneinheiten aus Y, 1 U-Dateneinheit, 1 V-Dateneinheit
 - Damit werden MCU und abgedeckte Region größer (dennoch 50% weniger Information pro Pixel als ohne subsampling)
 - » subsampling: Region 32 x 32 pixel, MCU 6 x 8 samples = 48 samples
 - » Standard: Region 8 x 8 pixel, MCU 3 x 8 samples = 24 samples
- Nach dem gleichen Prinzip lassen sich beliebige Subsampling-Verhältnisse mit Interleaving kombinieren.

Subsampling

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

(Forward) DCT: Mathematische Definition

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

wobei

x, y Koordinaten für die Datenpunkte einer Quell-Dateneinheit ($x, y = 1, \dots, 8$)

u, v Koordinaten für die Ziel-Koeffizienten ($u, v = 1, \dots, 8$)

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

$F(x, y)$ Koeffizientenwert

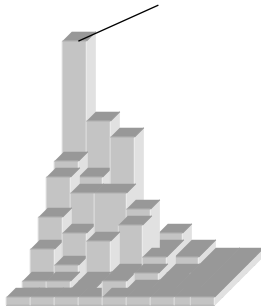
$$c_u, c_v = \frac{1}{\sqrt{2}} \quad \text{falls } u, v = 0$$

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

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

Interpretation der DCT-Koeffizienten

DC-Koeffizient (0,0)

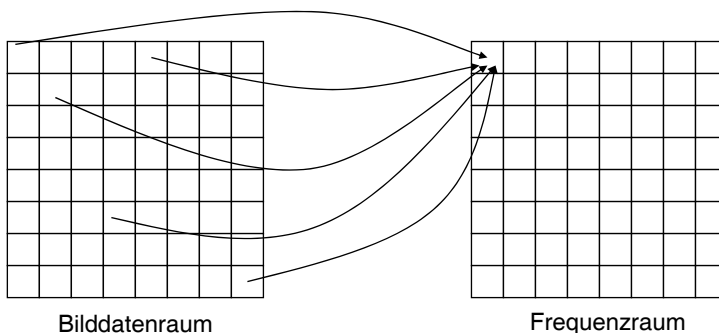


Rest: AC-Koeffizienten

DC = Gleichstrom
AC = Wechselstrom

- 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

DCT: Zusammenhang Datenraum - Frequenzraum



- Ein Punkt im Frequenzraum faßt die Informationen aus dem aktuell betrachteten Bilddatenraum (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
($x, y = 1, \dots, 8$)

u, v Koordinaten für die Ziel-Koeffizienten ($u, v = 1, \dots, 8$)

$f(x, y)$ 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

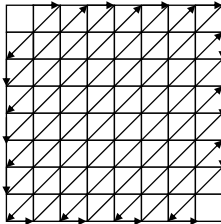
- Entscheidender Schritt zum Informationsverlust und damit zur starken Kompression !
- Quantisierungstabelle:
 - Gibt 64 Bewertungs-Koeffizienten $Q(u, v)$ vor
 - Bedeutung: Bewertung der einzelnen Frequenzanteile des Bildes
 - Größere Tabelleneinträge bedeuten stärkere Vergrößerung
 - Konkrete Tabellen nicht Bestandteil des Standards (nur zwei Beispiele)
 - Benutzte Quantisierungstabellen werden als Bestandteil der komprimierten Daten abgelegt
- Berechnung:
 - Division Frequenz-Koeffizient / Bewertungskoeffizient und Rundung
 - Macht mehr oder minder viele der Koeffizienten zu Null

JPEG-Schritte

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

Vorbereitung zur Weiterverarbeitung

- Quantisierte Frequenzwerte:
 - werden in linearer Reihenfolge ausgegeben
 - unterschiedliche Behandlung DC- und AC-Koeffizienten
- DC-Koeffizienten:
 - Benachbarte Dateneinheiten haben oft ähnlichen Grundton
 - Prädiktionsverfahren: *Differenz* zum DC-Wert der vorgehenden Dateneinheit wird weiter verwendet
- AC-Koeffizienten:
 - Ausgabe nach absteigender Frequenz („Zick-Zack“)



JPEG-Kodierung: Entropie-Kompression

- Vorletzter Schritt: „Statistische Modellierung“
 - » DC-Koeffizienten: Umsetzung in Werte, die Größen- und Amplitudenunterschied zeigen
 - » 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 Standard gegeben
 - Alternative: Durch zusätzlichen Pass über die Daten errechnen

JPEG-Schritte

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)
 - ...

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 – Abstatfrequenz