

Einführung Internet Computing Teil 2

Medien und Information

Kapitel 2: Medium Bild

Michael Granitzer

Harald Kosch

Universität Passau

Kapitel Medientechnik:IV

I. EIC Teil 2- Medium Bild

- Farbwahrnehmung und Farbmodelle
- Digitalisierung und Kodierung von Bildern
- JPEG Kompression (Überblick, nicht prüfungsrelevant)
- Vektorgrafiken am Beispiel SVG (Scalable Vector Graphics)

Farbwahrnehmung und Farbmodelle

Lernziel

Themen

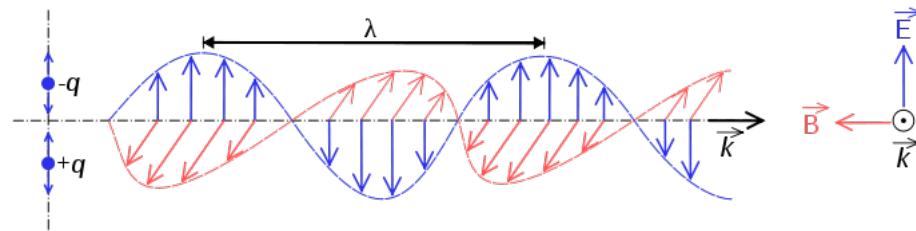
- Licht
 - Welche physikalischen Größen repräsentieren Licht?
- Farbwahrnehmung
 - Wie nehmen wir Licht wahr?
 - Wie wirkt sich die Wahrnehmung auf die Farbgestaltung aus?
- Farbmodelle
 - Was sind Farben?
 - Wie kann man Farben charakterisieren?
 - Sind Farben abhängig vom Gerät und von der Person?
 - Wie spezifizieren wir Farben in digitalen Medien?

Licht

Licht

Charakterisierung von Licht

Licht als eine **elektromagnetische Welle** (oder als Teilchen)



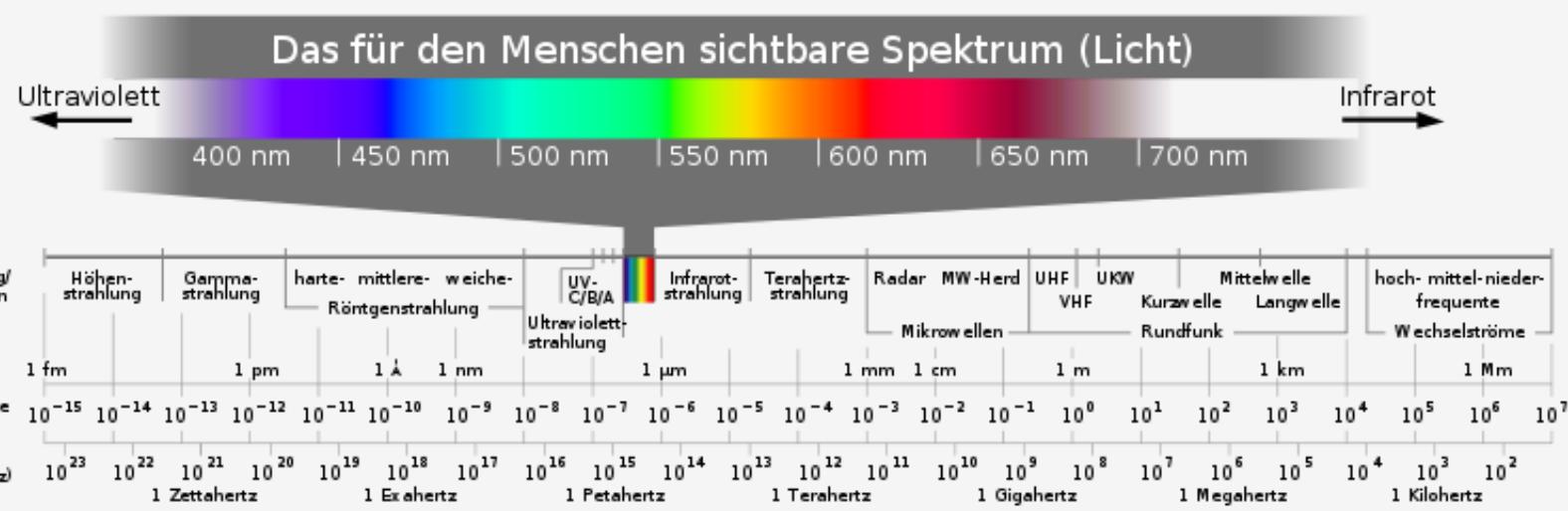
Quelle Wikipedia

Eigenschaften

- Lineare Ausbreitungsrichtung
- Wellencharakter: Brechung, Beugung, Dispersion, Streuung
- Teilchen (Photonen): Absorption, Emission
- Polarisation (Ausbreitungsebene): linear, zirkular, elliptisch
- Charakterisiert durch Wellenlänge λ , Frequenz f , Periodendauer T
- $f = 1/T$ [Hz]
- $T = \lambda/c$ [s]
- $f = c/\lambda$ [Hz]
- $c = 2,998 \cdot 10^8$ m/s

Licht

Spektrum der elektromagnetischen Strahlung



Quelle Wikipedia

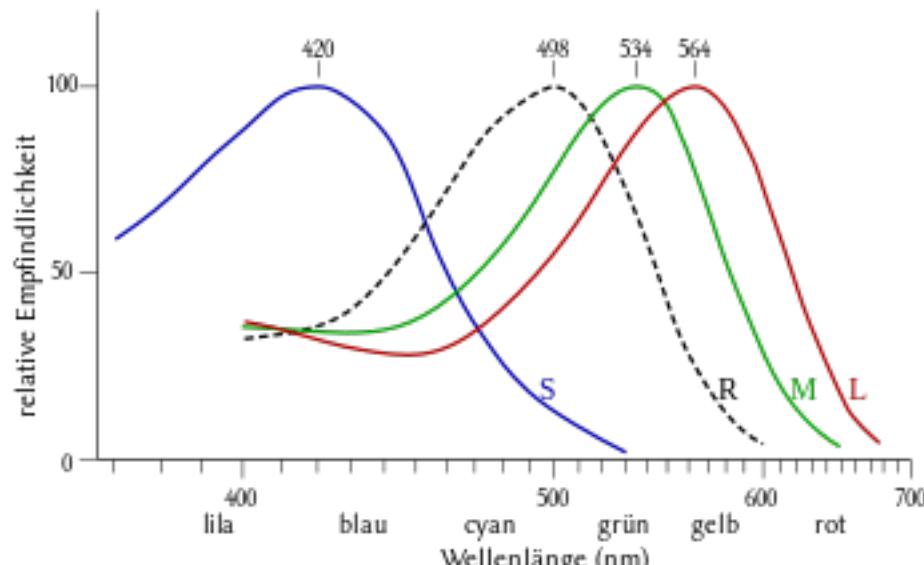
- Reale Strahlung i. A. Überlappung verschiedener Frequenzen
- Wie nehmen wir nun diese elektromagnetischen Strahlung wahr?

Farbwahrnehmung

Farbwahrnehmung

Wahrgenommene Frequenzbereiche

- Stäbchen: Hell/Dunkelwahrnehmung (120 Millionen, skotopisches Sehen, Nachtsehen)
- Zapfen: Farbwahrnehmung (7 Millionen, photopisches Sehen/Tagessehen)
 - 3 Typen: Rot, Grün, Blau
 - Dunkeladaption: Änderung der Farbwahrnehmung bei Dunkelheit
 - Chromatische Adaption: Weißabgleich im Auge bei geänderter Farbtemperatur



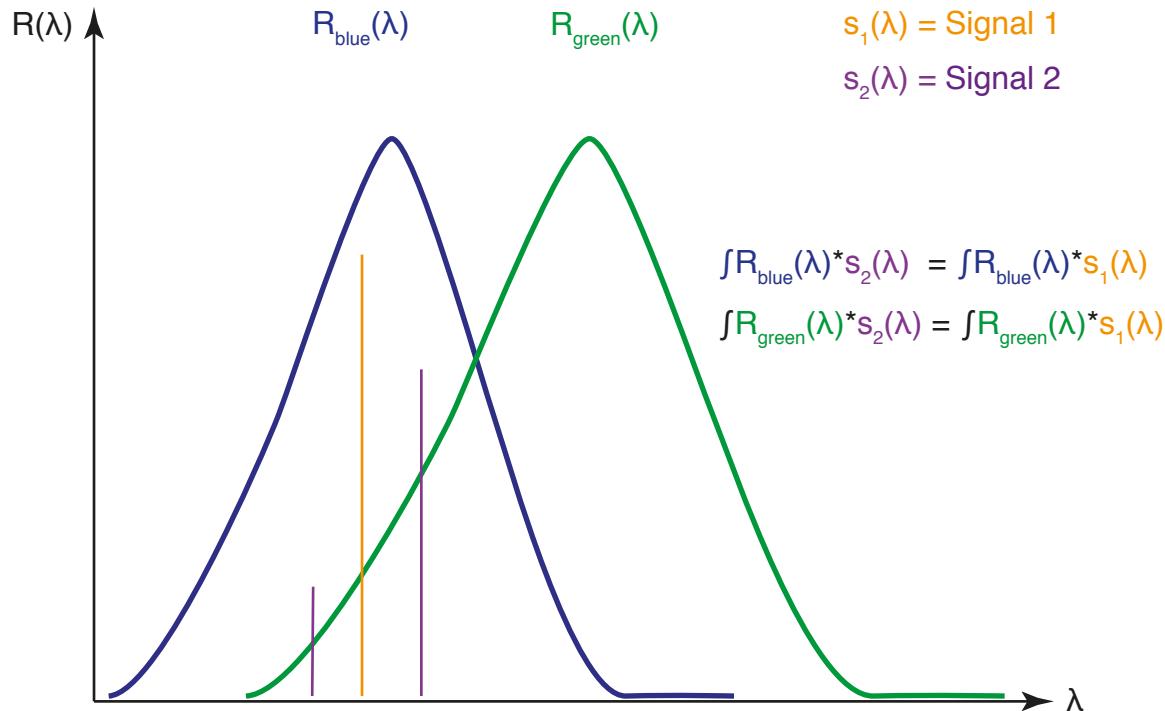
Bildquelle Wikipedia

- 3 Farbentheorie nach Young-Helmholtz

Farbwahrnehmung

Wahrgenommene Frequenzbereiche

- Wahrgenommene Farbe hat keine ein-eindeutige Abbildung zum Frequenzspektrum
- Beispiel: Blaue und grüne Rezeptorkurven (R_{blue} bzw. R_{green}) und zwei unterschiedliche Signale. Die Aktivierung beider Rezeptoren ist für beiden Signale s_1 und s_2 gleich gross.



Farbwahrnehmung

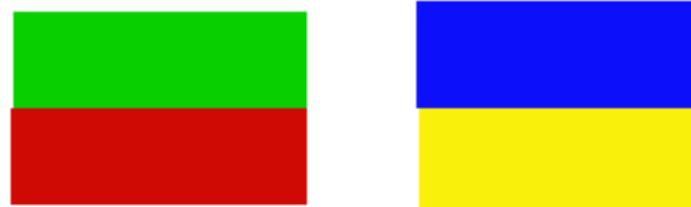
Farbsignalverarbeitung im menschlichen Gehirn

Neuronale Verarbeitung der Signale R,G, B.

- Summensignal Helligkeit (Gelb) $Y = R + G$
 - Differenzsignal Rot/Grün Unterscheidung $R - G$
 - Differenzsignal Blau/Gelb Unterscheidung $Y - B$
 - Y (Yellow) wird als Luminanzsignal bezeichnet und die Paare (R, G) und (Y, B) als Gegenfarben
- ⇒ Gegenfarbentheorie nach Hering

Konsequenzen

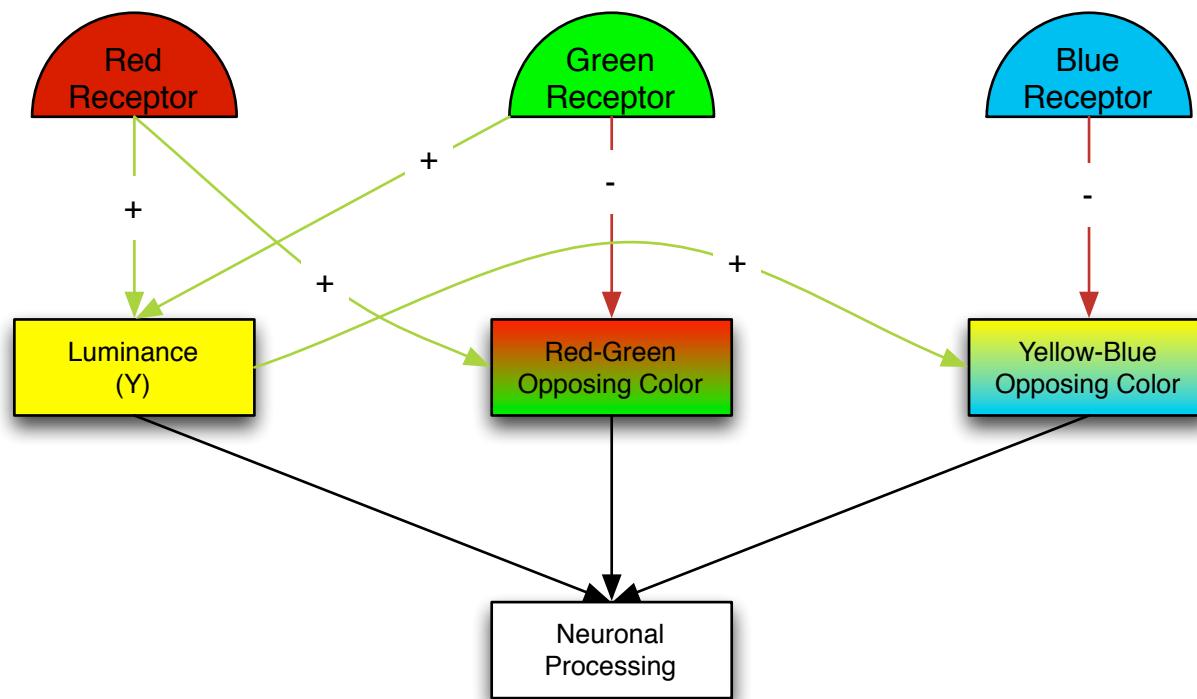
- Gelb-Anteil ist wesentlich für Helligkeitswahrnehmung
- Blau-Anteil spielt keine Rolle bei Helligkeitswahrnehmung (kühle Farbe)
- Farbkontraste Rot/Grün und Blau/Gelb besonders klar erkennbar



Farbwahrnehmung

Farbsignalverarbeitung im menschlichen Gehirn

Schematischer Überblick



Farbwahrnehmung

Anzahl wahrnehmbarer Farben

Unterscheidung zwischen

- 128 verschiedenen Farbtönen (hues)
- 130 verschiedenen Farbsättigungen (Farbreinheit)
- 16 (im gelben) - 26 (im Blauen) verschiedene Helligkeitswerte

ca. 380 000 verschiedene Farben

Sichere Unterscheidung gleichzeitig dargestellter Farben in Experimenten ca. bei 15 Farben

Farbmodelle

Farbmodelle

Begriffsbestimmungen Farbe

Farbe ist eine **individuelle** Wahrnehmung des Lichts durch das menschliche Auge

Definition 1 (Farbe)

Farbe ist diejenige **Gesichtsempfindung** eines dem Auge des Menschen **strukturlos erscheinenden Teiles** des Gesichtsfeldes, durch die sich dieser Teil bei **einäugiger Beobachtung mit unbewegtem Auge** von einem gleichzeitig gesehenen, ebenfalls **strukturlosen angrenzenden Bezirk allein unterscheiden kann.** (Nach Din 5033)

- Farbe entsteht durch Reizung der RGB Rezeptoren
- **Farbreiz** - Licht emittiert von einer Lichtquelle (unabhängig vom Betrachter)
 - Schwarzkörperstrahlung; Spektrum einer Lichtquelle
- **Farbvalenz** - Aufnahme des Farbreizes durch die Augen
 - Rezeptorkurven multipliziert mit Farbreiz
- **Farbempfindung** - Aufnahme der Farbvalenz im Gehirn
 - Wahrnehmung; Neuronale Adaption

Farbmodelle

Subjektivität der Farbwahrnehmung



Welche der Karten hat in beiden Bildern die gleiche Farbe?

Bildquelle Wikipedia

Farbmodelle

Subjektivität der Farbwahrnehmung



- Bilder mit unterschiedlicher Farbtemperatur (Annahme unterschiedlicher Lichtquellen)
- Die zweite Karte von Links hat in beiden Bildern die gleiche Farbe
- Die Wahrnehmung betrügt uns, wegen dem höheren Rotanteil im unteren Bild. Das Auge führt automatisch eine Farbanpassung durch (bezeichnet als **Chromatische Adaption**).

Bildquelle Wikipedia

Farbmodelle

Motivation Farbmodelle

Betrachtung des Zusammenhangs zwischen den physikalischen Kenngrößen des Lichtes (Spektrum und Stärke) und der Wahrnehmung.

Physikalische Eigenschaften

- **Luminanz** - Helligkeit der Farbe (Stärke des Signals) cd/m^2
- **Chromatizität** - der aktuelle Farbwert der Farbe (Frequenzspektrum)

Überführung in die menschliche Wahrnehmung:

- Für Mensch sind **Luminanz plus 2 Chrominanzsignale** ausreichend (für z.B. Bienen nicht)
- Farbmischung durch Überlagerung der drei Grundfarben (Dreifarbtentheorie/Tristimulustheorie nach Helmholtz und Young).
- Andere Parameter möglich: Farbton (Hue), Helligkeit (Brightness), Sättigung (Saturation)

Mehrdeutigkeit der Wahrnehmung:

- Wahrgenommene Luminanz eines Teilbildes ist abhängig von Luminanz **UND** Chromatizität des gesamten Bildes!
- Metamere - Farbreize unterschiedlicher Spektralverteilung aber mit gleicher wahrgenommener Farbe

⇒ **Übersetzung durch Farbmodelle notwendig**

Farbmodelle

Farbmodelle

Definition 2 (Farbmodell, Farbraum)

Ein Farbmodell ist **ein mathematisches Modell**, im Allgemeinen basierend auf 3 Parametern, zur Beschreibung des durch den Menschen wahrgenommen Farbreizes. Die konkrete Instanzierung eines Farbmodells nennt man **Farbraum**. Die technische, numerische Realisierung nennt man **Farbprofil**.

Eigenschaften

- Abbildung von verschiedenen Geräteeigenschaften:
 - Wie repräsentiert sich die Farbe “Rot” am LCD Bildschirm, am Röhrenbildschirm?
 - Das Auge als Gerät: Welchen Farbreiz nehmen wir als “Rot” wahr?
- Keine Behandlung wahrnehmungsspezifischer Eigenschaften wie chromatischer Adaption.

Farbmodelle

Kategorisierung von Farbmodellen

3 Klassen von Farbmodellen:

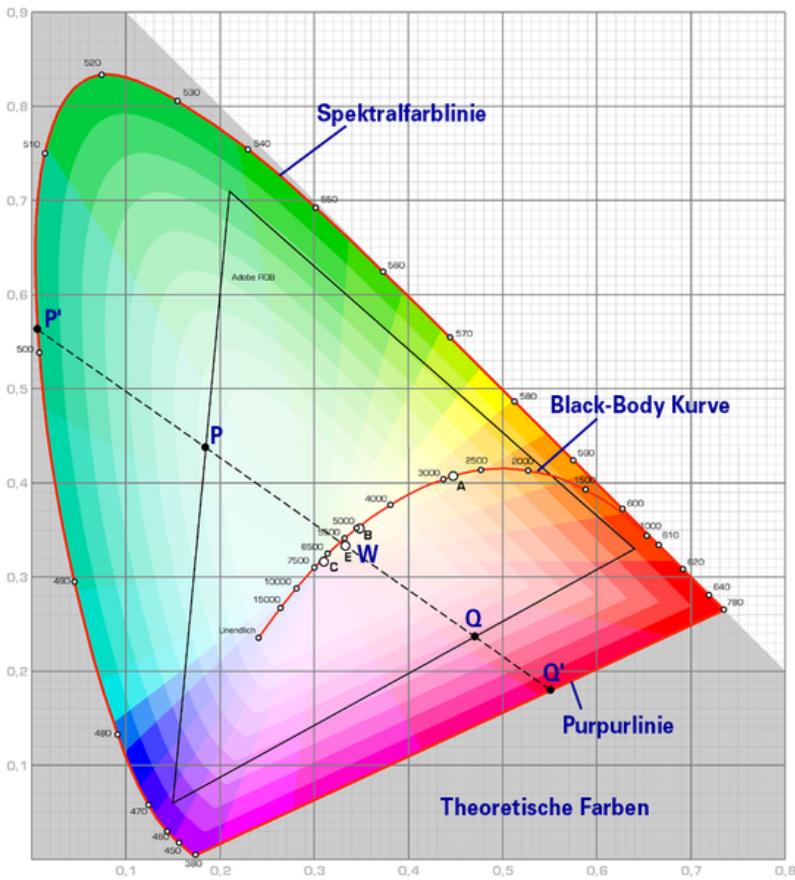
- **Allgemeine Farbmodelle:** CIE-Farbraum, CIE-L*a*b (kurz Lab)
Vollständige Abbildung aller wahrnehmbaren Farben; Referenzfarbraum für Transformationen
- **Präsentationsmedium bezogene Farbmodelle:** RGB, CMY, CMYK, YUV, YIQ
Abbildung der durchs Medium abbildbaren Farben und entspr. Medieneigenschaften
- **Physiologisch orientierte Farbmodelle:** HLS, HSV
Abbildung der Wahrnehmungseigenschaften von Farben

Standardisierungsgremium: CIE (Commision on Illumination)

Applet für Experimente <http://dcssrv1.oit.uci.edu/~wiedeman/cspace/>

Farbmodelle

Allgemeine Farbmodelle - CIE-Normvalenzsystem (CIE-xyY-Modell)



CIE Normtafel, Bildquelle Wikipedia

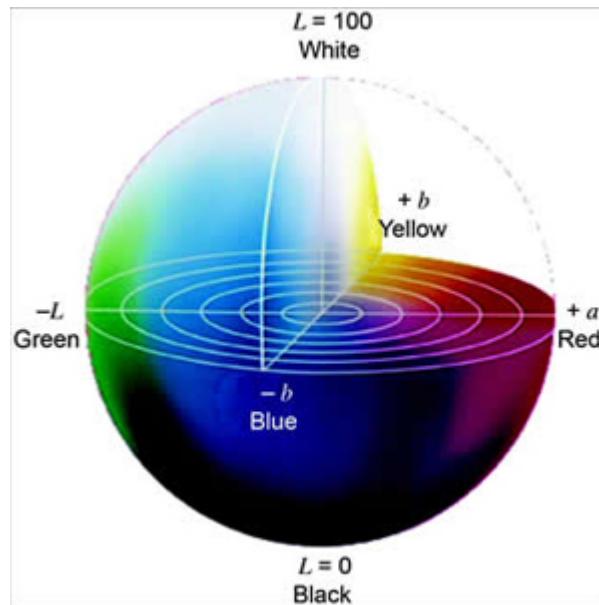
- Allgemeines Farbmodell: experimentell entwickelter Zusammenhang zwischen Farbreiz und Farbwahrnehmung
- Standardisiert durch CIE (Commission on Illumination)
- Hufeisenform der wahrnehmbaren Farben
- umgrenzt durch Spektralfarblinien und Purpurlinie
- **Weißenpunkt** ($1/3, 1/3, 1/3$)
- **Black-Body-Kurve**: Kurve eines Schwarzen Strahlers
- **Farbton**: W-P gerade gleicher Farbtöne bzw. W-Q als Komplementärfarbe
- **Sättigung**: Abstand Weißpunkt-Farbort ($W-P = 100\%$)
- **Gamut**: Farbumerraum eines Geräts repräsentiert als Dreieck von X,Y,Z Werten des Gerätes

Farbmodelle

Allgemeine Farbmodelle - L*a*b*-Farbraum (DIN EN ISO 11664-4)

Dreidimensionales Farbmodell der CIE entwickelt 1976

- Luminanz plus zwei Chrominanz-Werte (L = Luminanz, a =Grün/Rot, b =Blau/Gelb)
- 3D Darstellung: Gleiche Abstände entsprechen empfindungsgemäß gleichen Farbabständen (nichtlineare Darstellung)
- Keine einfache Umrechnung zu RGB Farträumen



- Verwendet z.B. intern in Adobe Produkten (z.B. Photoshop) zur Umrechnung von Farbräumen

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle

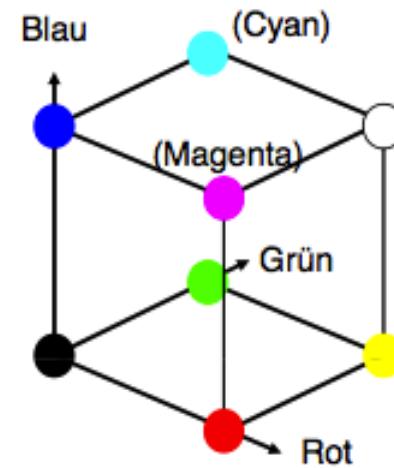
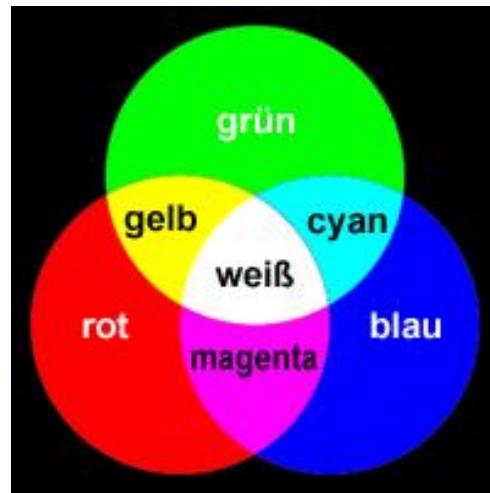
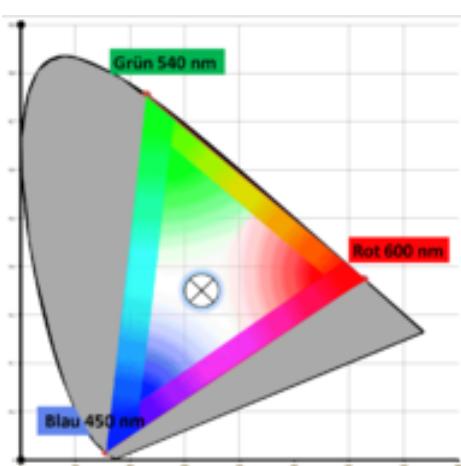
Betrachtung der Eigenschaften von Präsentationsmedien und der mit dem Medium verbundenen Farbwahrnehmung (e.g. Drucker, Bildschirm)

- Betrachten im Allgemeinen nur einen Ausschnitt allgemeiner Farbmodelle (**Gamut**)
- Umrechnung zwischen Präsentationsmedium-spezifischen Farbmodellen i.a. nicht verlustfrei. Die Verwendung von allgemeinen Farbmodellen ist notwendig
- Geräte spezifizieren entsprechende **Farbprofile (International Color Consortium - ICC)**
 - ICC Profile definieren eine Abbildung zwischen Geräte-spezifischen Farben v. Ein/Ausgabegeräten und den CIE LAB oder CIE xyY Farbraum.
- Einbettung in Bilddaten möglich (z.B. als EXIF Metadaten - siehe Digitalisierung)

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - RGB

RGB Farbmodell - Additives Farbmodell



Quelle Butz

- Meist verwendetes Modell für aktive Lichterzeugende Ausgabemedien
- Spektrale Intensität der Komponenten werden addiert
- Gamut als Dreieck definiert die erzeugbaren Farben, welche (i) von den Primärquellen abhängen und (ii) i.a. nicht alle wahrnehmbaren Farben abdecken.

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - RGB

Beispiele v. RGB Farträumen und Farbprofilen:

- **sRGB:** Standard Format definiert v. HP und Microsoft 1996.

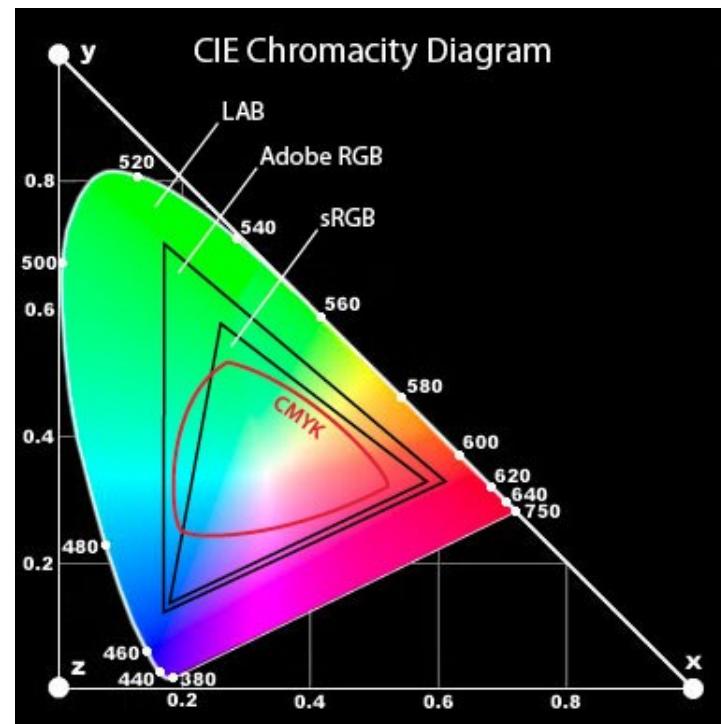
Kleiner Farbraum, der oft den kleinsten gemeinsamen Nenner zwischen Geräten darstellt. Standardeinstellung bei den meisten Photo

- **Adobe RGB:** Umfangreicher als sRGB. Bilder erscheinen dunkler wenn keine Umrechnung in sRGB erfolgt.

- **Adobe wide RGB:** Umfangreicher als Adobe RGB mit puren Spektralfarben als Primärfarben.

- **RGBA:** RGB mit Alpha Kanal. Kein eigentliches Farbmodell

- **UHDTV:** RGB Farbraum f. Ultra High Definition TV.

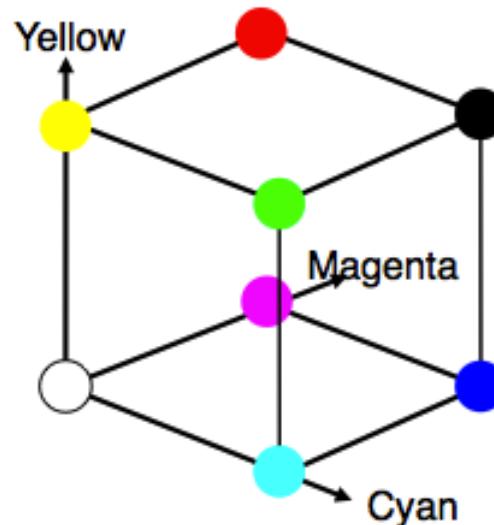
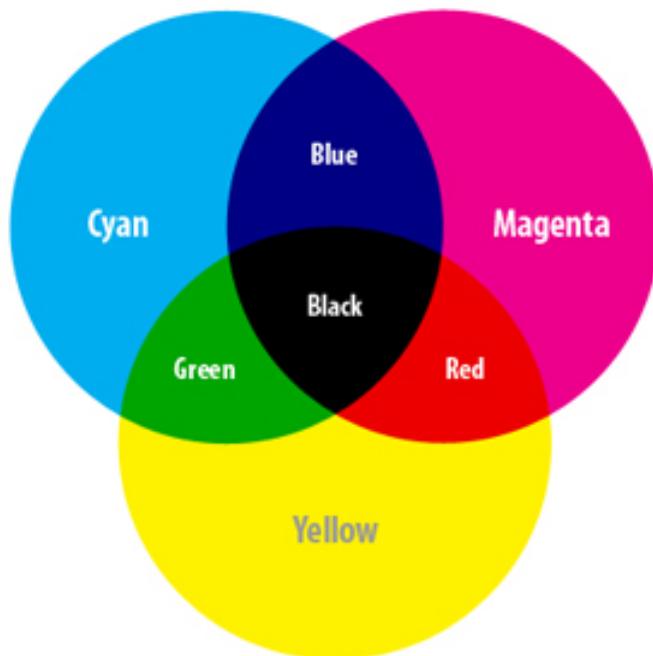


^a<http://photo.net/learn/digital-photography-workflow/advanced-photoshop-tutorials/using-lab-color-adjustments/>

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - CMY(K)

CMY(K) Farbmodell - Subtraktives Farbmodell



Quelle Butz

- Meist verwendetes Modell zur Ausgabe auf reflektierenden Ausgabemedien (z.B. Drucker)
- Einfallendes Licht wird gefiltert (e.g. Magenta filtert Grün)
- Für Tintendrucker oft 4. Farbe K = black (aus rein praktischen Gründen)

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - CMY(K)/RGB

RGB zu CMY transformation

$$f_{RGB} = \begin{pmatrix} f_R \\ f_G \\ f_B \end{pmatrix} = \begin{pmatrix} f_{max} - f_C \\ f_{max} - f_M \\ f_{max} - f_Y \end{pmatrix} = \begin{pmatrix} f_{max} \\ f_{max} \\ f_{max} \end{pmatrix} - \begin{pmatrix} f_C \\ f_M \\ f_Y \end{pmatrix} = white_{RGB} - f_{CMY}$$

Entsprechend gilt:

$$f_{CMY} = \begin{pmatrix} f_C \\ f_M \\ f_Y \end{pmatrix} = white_{RGB} - \begin{pmatrix} f_R \\ f_G \\ f_B \end{pmatrix} = white_{RGB} - f_{RGB}$$

Beispiel bei 8 Bit Farbiefel, Türkiston

$$f_{CMY} = \begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix} - \begin{pmatrix} 80 \\ 200 \\ 130 \end{pmatrix} = \begin{pmatrix} 175 \\ 55 \\ 125 \end{pmatrix}$$

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - YCrCb

YCrCB/YPbPr Farbmodell - Helligkeitssignal und zwei Chrominanz Signale

- Y bezeichnet die Helligkeit
- CrCB/PbPr repräsentieren Chrominanz Signale
- Entsteht aus einfacher linearer Transformation aus dem RGB Modell
- YCrCb Digitaltechnik
- YPbPr analoge Version von YCrCb

RCA connector (Chinch) for Component Video.



Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - YCrCb

YCrCB Konvertierung von RGB

$$f_{YCbCr} = \begin{pmatrix} f_Y \\ f_{Cb} \\ f_{Cr} \end{pmatrix} = \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{pmatrix} \begin{pmatrix} f_R \\ f_G \\ f_B \end{pmatrix}$$

- Y (Helligkeit) Kanal beinhaltet die meiste Information (nutzbar für Kompression oder effiziente Fernsehübertragung)
- Entspricht eher dem natürlichen Farbempfinden
- Farbkanäle enthalten wenig Information (und daher haben wir ein angepasstes Sehsystem)
- Entspricht der Abbildung in der Natur: mehr Flächen gleicher Farbe im Vergleich zu kontrastreichen Licht/Schatten

Farbmodelle

Präsentationsmedium-spezifische Farbmodelle - YCrCb

Beispielvergleich YCrCB mit RGB



Bildquelle Malaka, Butz, Hussmann

Farbmodelle

Physiologische Farbmodelle - HSV + HSL

Physiologische Farbmodelle

- entsprechen menschlicher Wahrnehmung
- relativ einfache Selektion von Farben
 - Wähle zuerst den Farnton
 - Passe dann Helligkeit und Sättigung an

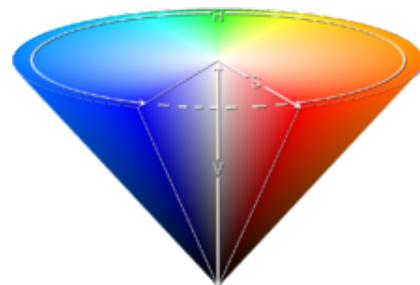
HSB, HSV, HSI:

- Hue, Saturation, Value bzw. Brightness (absolute Helligkeit)
- Hue, Saturation, Lightness (relative Helligkeit)
- Hue, Saturation, Intensity (Lichtintensität)
- Unterschied bezogen auf Anwendungsbereich (Phototechnik)

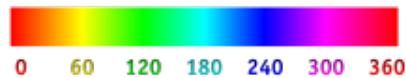
Farbmodelle

Physiologische Farbmodelle - HSV + HSL

HSV Farbmodell im Detail



HSV Kegel - Quelle Wikipedia



HSV Farbton Skala - Quelle Wikipedia

Polar-Koordinaten

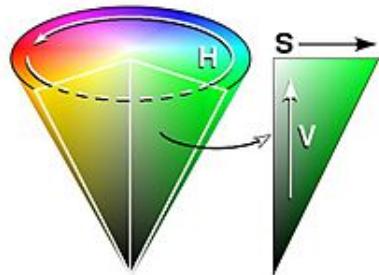
- Farbton als Farbwinkel H auf dem Farbkreis
 - Wertebereich: 0° Rot, 120° Grün, 240° Blau)
 - Physikalisch Interpretation: dominante Wellenlänge
- Sättigung in Prozent
 - Wertebereich: 0% = Neutralgrau, 100% = reine Farbe)
 - Physikalische Interpretation: Hintzmischung von Weiß
- Hellwert V als Prozentwert
 - Wertebereich: 0% = keine Helligkeit, 100% = volle Helligkeit
 - Physikalische Interpretation - Gesamtenergiegehalt

Farbmodelle

Physiologische Farbmodelle - HSV + HSL

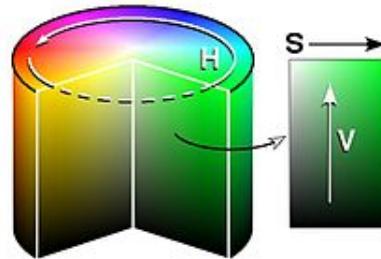
Visualisierung des Farbraums

Als Kegel



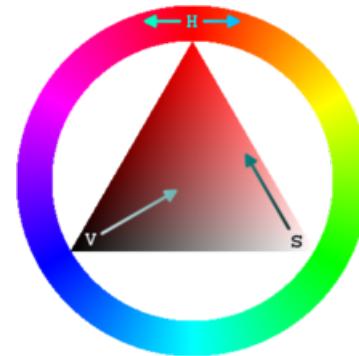
Bildquelle Wikipedia

Als Zylinder



Bildquelle Wikipedia

Farbwähler f. Programme



Bildquelle Wikipedia

Farbmodelle

Farben in HTML

Spezifikation von Farben im RGB Modell

- jeweils 8 bit, d.h. zweistellige Hexadezimalzahl: #rrggbbaa
- Beispiel: “Kiefer” #006633

Anbindung an HTML Tags durch CSS

- Hintergrundfarben, Farben für Pseudovermäte (e.g. Verweise)
- Beispiel <body style="background-color: #CCFFFF">

Websichere Farben (mit festgelegter Palette für Geräte mit geringerer Farbtiefe):

- Standardpalette von 216 RGB Farben
- RGB-Werte durch 51 teilbar
- Eingeführt von Netscape

Farbmodelle

Farben in HTML

black	#000000	gray	#808080
maroon	#800000	red	#FF0000
green	#008000	lime	#00FF00
olive	#808000	yellow	#FFFF00
navy	#000080	blue	#0000FF
purple	#800080	fuchsia	#FF00FF
teal	#008080	aqua	#00FFFF
silver	#C0C0C0	white	#FFFFFF

Farbtabelle und HTML Namen

Zusammenfassung

Farbwahrnehmung und Farbmodelle

Licht

- Elektromagnetische Welle mit physikalischen Kenngrößen
- Unabhängig von Wahrnehmung

Farbwahrnehmung

- Erfolgt über drei Zapfen-Rezeptortypen (RGB bzw. SML) und Mischung dieser
- Farben sind subjektiv, individuell und entstehen durch Mischung der 3 Grundfarben

Farbmodell

- Mathematisches Modell für die Definition von Farben
- Allgemeine Modelle (Alle Farben), Präsentationsmediums spezifische Modelle (Drucker, Monitor etc.), Physiologisch orientierte Farbmodelle (beruhen auf Wahrnehmung)

Zusammenfassung

[Bibliographie]

- **Malaka, Butz, Hussmann (2009)** - Medieninformatik: Eine Einführung
(Pearson Studium - IT)
- **Kerr (2010)**- The CIE XYZ and xyY Color Spaces
[http://graphics.stanford.edu/courses/cs148-10-summer/
docs/2010--kerr--cie_xyz.pdf](http://graphics.stanford.edu/courses/cs148-10-summer/docs/2010--kerr--cie_xyz.pdf)

Kapitel Medientechnik:IV

I. EIC Teil 2- Medium Bild

- Farbwahrnehmung und Farbmodelle
- Digitalisierung und Kodierung von Bildern
- JPEG Kompression (Überblick, nicht prüfungsrelevant)
- Vektorgrafiken am Beispiel SVG (Scalable Vector Graphics)

Digitalisierung und Kodierung digitaler Bilder

Lernziel

Unterthemen

- Digitalisierung - Grundlegende Eigenschaften von Bildern nach der Digitalisierung
- Kodierung - Bilddateiformate (Digital), Aufbau und Anwendungsbereich

Fragestellungen

- Was sind die Eigenschaften digitaler Bilder?
- Was ist Rasterung und welche Farbtiefen werden verwendet?
- Was versteht man unter Dithering?
- Welche (wichtigen) Bildformate gibt es?

Digitalisierung

Digitalisierung

Klassifikation von Bilddatenformaten

Rastergrafik (Bitmap)

- Speicherung der Abtastung eines Bilds (Pixel-Bild/Rasterisierung).
- Kompression
 - Verlustfreie Kompression: BMP, TIFF
 - Verlustbehaftete Kompression: JPEG
 - Erweiterte Bitmap-Format mit Zusatzfunktionalität (e.g. Animationen): GIF, PNG

Vektorgrafik

- Beschreibung von Einzelobjekten (z.B. Linien, Kreis etc.)
- Beispiel SVG (Scalable Vector Graphics)

Meta-Files

- Kombination von Vektor- und Rastergrafik
- Beispiele: WMF (Windows Meta File), Macintosh PICT, PDF, EPS, RTF

Digitalisierung

Räumliche Auflösung: Bildgröße und Auflösung

Pixel (Picture Element): Kleinste Einheit eines Bildes; Bildpunkt

- tatsächliche Größe eines Pixels hängt vom Ausgabegerät ab
- Seitenverhältnis muss nicht 1 sein

Bildgröße: Größenangabe eines Bildes in Pixel (100 x 200 Pixel)

Auflösung: Anzahl der Pixel die auf einer bestimmten Strecke zur Darstellung zu Verfügung stehen

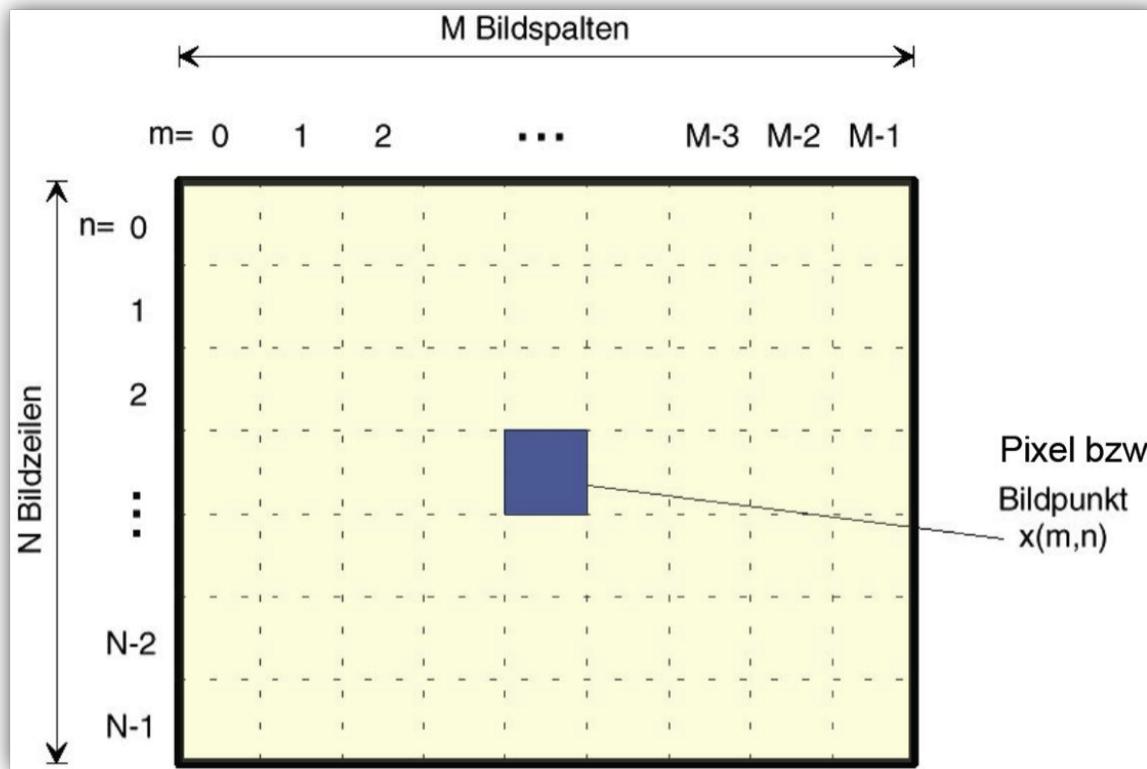
- Angabe in ppi (pixel per inch), Standardwert 72 ppi
- $Breite[\text{pixel}] = Breite[\text{in}] * \text{Auflösung}[ppi]$

Skalierung: Konversion des Bildes auf andere Auflösung (resampling)

- Abwärtsskalierung durch Bildung von Mittelwerten
- Aufwärtsskalierung durch Interpolation der Bildpunkte (e.g. "Bi-kubische Interpolation); nur eingeschränkt automatisierbar

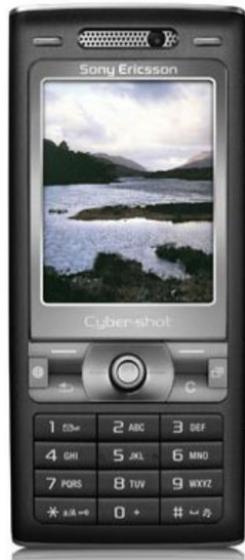
Digitalisierung

Rasterbild - Räumliche Auflösung

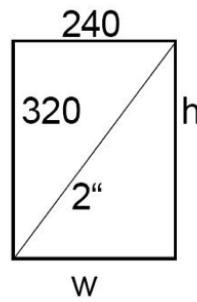


Digitalisierung

Rasterbild - Räumliche Auflösung



W800i
240x320 Pixel 2“ Diagonale
262144 Farben



Breite w:

$$w^2 + h^2 = 2^2$$

$$w^2 + \left(\frac{4}{3}w\right)^2 = 4$$

$$w^2 \cdot \left(1 + \frac{16}{9}\right) = 4$$

$$w^2 \cdot \left(\frac{25}{9}\right) = 4$$

$$w^2 = \frac{9 \cdot 4}{25}$$

$$w = \frac{6}{5} \text{ inch}$$

Auflösung:

$$\frac{6}{5} \text{ inch} \wedge = 240 \text{ Pixel}$$

$$1 \text{ inch} \wedge = 200 \text{ Pixel}$$

$$\Rightarrow 200 \text{ ppi}$$

Digitalisierung

Räumliche Auflösung: Bildgröße und Auflösung

Beispiel Anzahl Pixel



100



50



25



12

Bildquelle Butz - Medieninformatik

Digitalisierung

Farbtiefen und Farbkanäle

Farbtiefe (color resolution): Anzahl der Farben, die pro Pixel gespeichert werden können

- 2 Farben (1-bit) für schwarz-weiß Bild
- 16 Farben (4-bit)
- 256 Farben (8-bit)
- 16,7 Millionen Farben (24 bit)
- 24 bit Farbtiefe (1 Byte je Grundfarbe RGB) - “True Color” ausreichend für menschliche Wahrnehmung
- Moderne Kameras liefern oft mehr 16 Bit/Kanal
- High-Dynamic-Range-Bilder: 32 Bit/Kanal

Digitalisierung

Farbtiefen und Farbkanäle

Farbkanal: Teil der gespeicherten Information der sich auf eine der Primärkomponenten des gewählten Farbmodells bezieht

- Bei Rohdaten meist Rot, Grün und Blau (RGB-Modell)
- Bei Druckvorbereitung auch CMY bzw. CMYK (“Vierfarbendruck”)

Alpha Kanal als zusätzlicher Kanal: Spezifikation der Transparenz (Durchlässigkeit)

- Transparenzabstufung abhängig von Auflösung
 - 1 Bit: Pixel ist transparent oder nicht
 - 8 Bit: Grad der Transparenz
- Formate: PNG, TIFF, PSD (8-bit und mehr), GIF definiert lediglich eine Farbe als Hintergrund (1-Bit Kanal)

Digitalisierung

Farbpaletten und indizierte Farben

Farbpalette die Menge der in einem konkreten Bild tatsächlich enthaltenen Farben
(meist Teilmenge aller möglichen Farben)

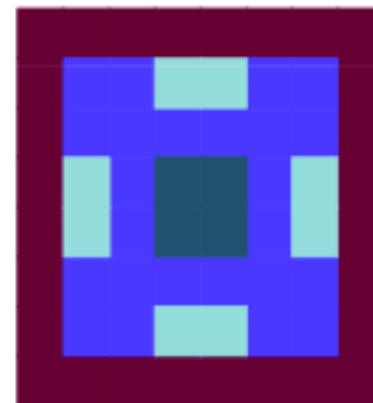
Indizierte Speicherung

- Farbpalette (Tabelle) enthält die im Bild vorkommenden Farben
- Pro Pixel wird nur der Index in die Palettentabelle gespeichert
- Änderung der Farben bei Änderung der Palette

1	1	1	1	1	1	1	1	1
1	2	2	3	3	2	2	1	
1	2	2	2	2	2	2	1	
1	3	2	0	0	2	3	1	
1	3	2	0	0	2	3	1	
1	2	2	2	2	2	2	1	
1	2	2	3	3	2	2	1	
1	1	1	1	1	1	1	1	

	R	G	B	
0	35	101	128	
1	128	0	64	
2	99	92	254	
3	156	227	227	

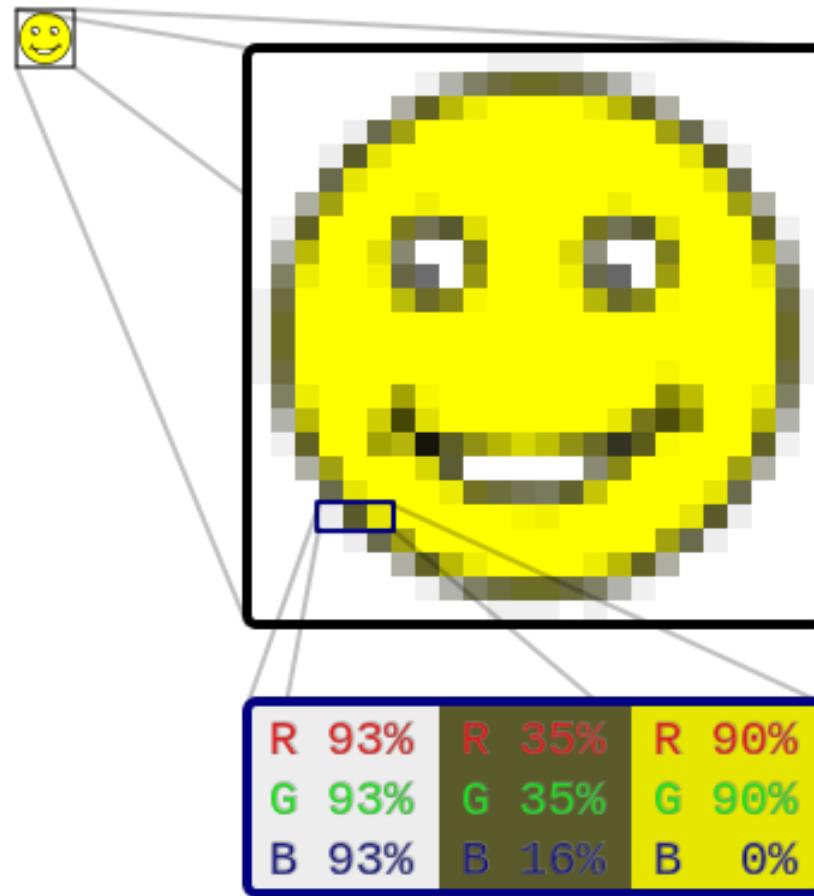
Speicherplatz:
8 x 8 x 2 bit +
4 x 3 x 8 bit + 4 x 2 bit = 232 bit
(statt 1536 bit)



Bildquelle Malaka - VO Digitale Medien

Digitalisierung

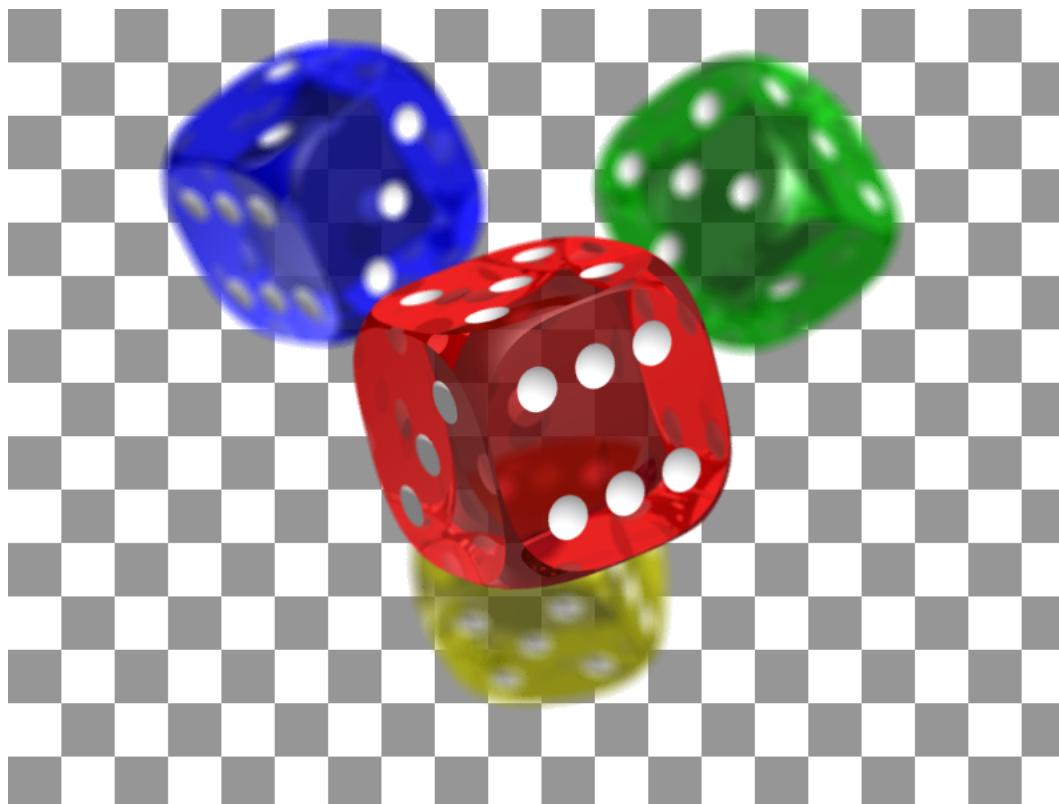
Beispiel Rastergrafik



Bildquelle Wikipedia

Digitalisierung

Beispiel Alphakanal



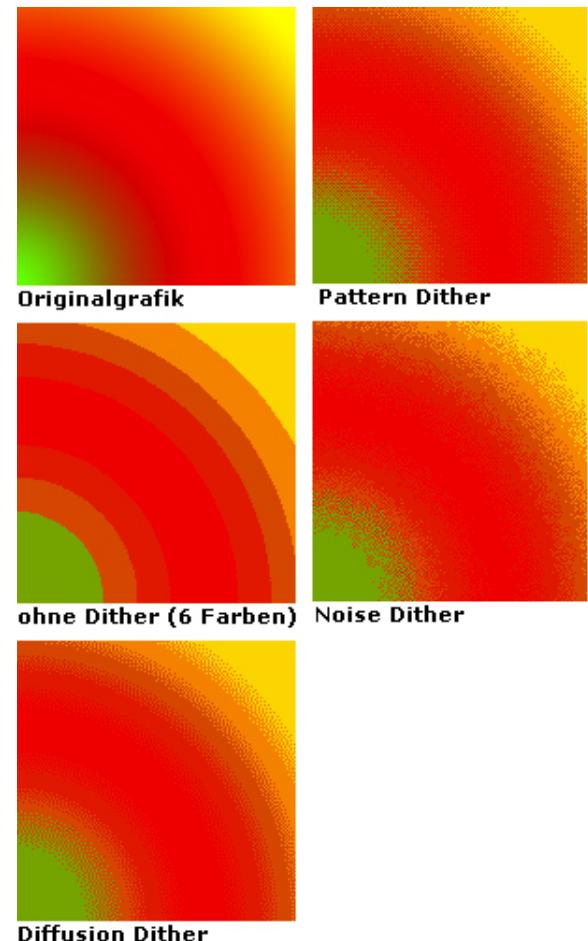
Bildquelle Wikipedia

Digitalisierung

Dithering

Dithering (Fehlerdiffusion) simuliert Farbverläufe durch bestimmte Pixel-Anordnung (Trade-off Räumliche Auflösung vs. Farbauflösung)

- Bei zu geringer Farbtiefe lassen sich Farbverläufe schwer darstellen
- Simulation einer Farbe ähnlich eines Röhrenmonitors
- Verschiedene Algorithmen
 - Floyd-Steinberg
 - Jarvis-Algorithmus
 - Stucki
- **Anti-Aliasing** ist die gegensätzliche Operation. Hier wird ein zu geringe räumliche Auflösung durch Verwendung von Farbeauflösung kompensiert, indem benachbarte Pixel eine ähnliche Farbe bekommen.



Bildquelle Wikipedia

Kodierung

Kodierung

Überblick

Für die Speicherung von Bilddaten sind unterschiedliche Formate definiert, die unterschiedlichen Zwecken dienen.

- Unterschiedliche Einsatzbereiche/Eigenschaften (e.g. Kompression, Farbtiefe, Auflösung, Metadaten)
- Wir betrachten: TIFF, EXIF, BMP, PNG, GIF, JPG im Detail

Kodierung

Rasterformate - TIFF (Tagged Image File Format)

- Weit verbreitetes Format im Amateur und Profi Bereich für hohe Farbtiefe
- Entwickelt von Aldus (jetzt Adobe) 1986 als einheitliches Format für Scans; Derzeitige Version 6.0 (von 1992)
- Ermöglicht **flexible Kombination** unterschiedlicher Eigenschaften
 - Verlustfreie (LZW) /verlustbehaftete Kompression (JPG)
 - Einbettung von JPG
 - Ebenen und Seiten
 - Einbettung von Metadaten
- Unterscheidung unterschiedlicher Ausprägungen aufgrund der Flexibilität (Thousands of Incompatible File Formats)
 - Baseline TIFF (max. 4 GB große Files)
 - TIFF Extensions
 - BigTiff - 64-bit Variante
- **Mime type:** image/tiff

Kodierung

Rasterformate - BMP (bitmap; Device Independent Bitmap)

Format zur Geräte-unabhängigen Speicherung von Rasterbildern

- Windows/MS DOS Standardformat
- Farbtiefen: 1, 4, 8 und 24 bit
- Verwendung einer Farbpaletten bei Farbtiefe < 24 Bit
- RLE (verlustfreie) Kompression

Format

- File Header mit fixer Größe (Signatur, Größe, Offset zu Pixel Array)
- DIB (BMP) Header mit Metainformation zum Bild (Breite, Höhe, Kompression, Farbpalette)
- Farbtabelle (optional)
- Datenbereich (Bitdaten)
- ICC Color Profile

Mime-Type: `image/bmp` oder `image/x-bmp`

Kodierung

Rasterformate - GIF (Graphics Interchange Format)

GIF-Überblick

- Entwickelt 1987 von CompuServe (Version GIF87a; Erweiterung GIF89a)
- Limitierte Farbtiefe von 8-bit (256 Farben)
- Spezifikation der globalen/lokalen Farbpalette aus 24-Bit RGB Farben
- Kompression über LZW. Da LZW von UniSys patentiert war, gab es einen Patentstreit.
Wegen der Lizenzforderungen wurde das Format PNG 1994 entwickelt
- GIF war eines von zwei der ersten Bildformate im WWW
- Animationsfähigkeiten in Version GIF89a
- Interlacing zur schnelleren Darstellung im Web

Format

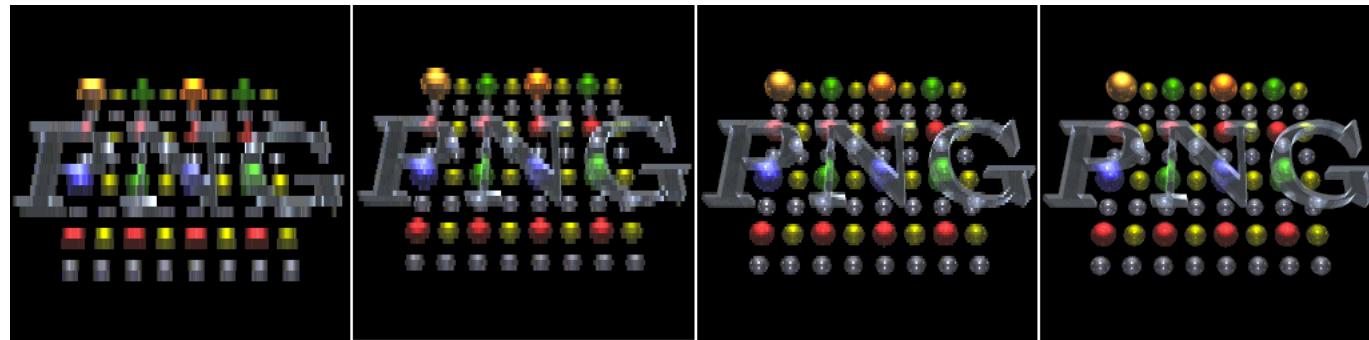
- Header mit Fixer Größe (Version, Logischer Darstellungsbereich)
- Globale Farbpalette
- n Segmente welche jeweils den logischen Darstellungsbereich füllen und die lokale Farbpalette spezifizieren
- Jedes Segment definiert ein Bild (\Rightarrow Animation)
- Definition einer Hintergrundfarbe (1-Bit Alphakanal) in globaler Farbtabelle möglich (keine echte Transparenz/Halo Effekt)

Kodierung

Rasterformate - GIF

Interlacing in GIF

- Ziel: Kürzere empfundene Ladezeit für Betrachter
- Ablauf: Das Bild wird schrittweise in Zeilen aufgebaut
 - 1. Durchlauf: Jede 8. Zeile beginnend mit Zeile 0
 - 2. Durchlauf: Jede 8. Zeile beginnend in Zeile 4
 - 3. Durchlauf: Jede 4. Zeile beginnend in Zeile 2
 - 4. Durchlauf: jede 2. Zeile beginnend in Zeile 1



Bildquelle <http://www.schaik.com/png/adam7.html>

Kodierung

Rasterformate - PNG

Geschichte

- Entwickelt aufgrund GIF Lizenzforderungen für GIF Format (1994)
- Erste Version 1996, Standardisierung März 2004: ISO/IEC 15948:2004

Eigenschaften

- Verlustfreie Kompression (DEFLATE/zlib)
- “Full Color” oder Paletten-basiert (24-Bit RGB oder 32-bit RGBA, Graubilder mit od. ohne Alphakanal)
- Fokus Bilddaten im Internet, keine Unterstützung nicht RGB basierter Farbräume wie CMYK
- Verbessertes Interlacing (7-pass vertikal und horizontales Interlacing. Adam7-algorithm)

Vergleich PNG (Adam7) zu GIF Interlacing: <http://www.schaik.com/png/adam7.html>

- Animationen nur mit MNG (Multiple Image Network Graphics)

Mime Typ:`image/png`

Kodierung

Rasterformate - PNG

Format

- Header mit fixer Größe
- Danach eine Menge von Chunks (Länge, Typ des Chunks als 4-byte ASCII, Data und CRC Code)

Length	Chunk type	Chunk data	CRC
4 bytes	4 bytes	<i>Length</i> bytes	4 bytes

- Critical Chunks:
 - * Metadaten (Typ=IHDR)
 - * Paletten (Typ=PLTE)
 - * Bilddaten (Typ=Data)
 - * Bildende (Typ=IEND)
- Ancillary Chunks:
 - * ICC Profile (iCCP)
 - * Gamma (gAMA)
 - * Text/Metadaten (tEXT)
 - * ...

Kodierung

Rasterformate - PNG

Alpha Kanal

- Alpha Werte mit jedem Pixel gespeichert
- 4 Bytes pro Pixel: “RGBA”-Farbmodell ermöglicht elegante Schatten und Übergänge
- Vermeidet Wechselwirkung zwischen Anti-Aliasing und Transparenzfarbe, wie dies in GIFs bei einem 1-Bit Alphakanal auftreten kann:
 - Anti-Aliasing interpoliert Pixel zur Auflösung von “Unstetigkeiten”
 - Farbwert entspricht nicht der Hintergrundfarbe
- ⇒ Randeffekte (Halo Effekt)



<http://www.lunaloca.com/tutorials/antialiasing/>

Kodierung

Rasterformate - PNG - Kompression

2 Schritte verlustfreier Kompressionsprozess

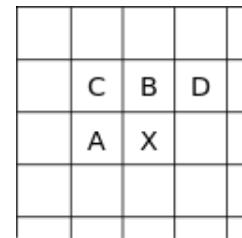
- Vorfilter (prediction):
Überführung der Daten in einen besser komprimierbaren Binär-Stream
- compression:
 - DEFLATE (zlib library unter Linux/Mac)
 - Patentfreier, verlustfreier Kompressionsalgorismus
 - * Basiert auf LZ77 Algorithmus (Vorgänger von LZW) kombiniert mit Huffman Kodierung

Kodierung

Rasterformate - PNG

Filter

- Komprimiere nicht die Rohwerte, sondern die Differenz sequentiell aufeinander folgender Pixelwerte, z.B. $<1, 2, 3, 4, 5, 6, 7, 9, 11> \Rightarrow <1, 1, 1, 1, 1, 1, 1, 2, 2>$
- Heuristische Selektion aus unterschiedlichen Filter je nach Bildzeile:
 - Aktuelles Pixel ist X
 - **Sub** -Differenz zu Pixel A
 - **Up** - Differenz zu Pixel B
 - **Average** - Differenz zu $\frac{B+A}{2}$
 - **Path** - Differenz zu $P_x = A + B - C$



Bildquelle Wikipedia

Beispiel: PNG Bild mit 256 Farben (oben) komprimiert auf 251 Byte durch Vorfilterung (unten)



Bildquelle Wikipedia

Kodierung

Rasterformate - JPEG/JFIF

Geschichte

- Entwickelt von der Joint Photographic Experts Group 1992
- ISO/IE 10918-1 (CCITT) (1994)
- Standard für Digitalbilder

Eigenschaften

- JPEG - Beschreibt die Kompressions CODECS (Codierung/Decodierung. Details später)
- Kombination verlustfreier und verlustbehafteter Kompression bis zu 10:1 ohne sehbaren Verlust
- JPEG File Interchange Format (JFIF) beschreibt das Fileformat (Weitere Varianten: SPIFF, JNG)
- Maximale Auflösung 65535x65365
- Farbraum YCbCr
- Format: Header + Segmente (äquivalent zu Tags in TIFFs)

Mime Typ:image/jpeg

Kodierung

Bildmetadaten - EXIF (Exchangeable Image File Format)

Dateiformat zur Speicherung von Metadaten über digitale Bilder

- Standardisiert durch Japan Electronic and Information Technology Industries Association
- EXIF Daten werden direkt in JPEG oder TIFF im Header verwendet
- Nahezu all Digitalkameras unterstützen das Format

Metadaten

- Datum/Uhrzeit
- Orientierung (Hoch/Quer), Brennweite, Belichtungszeit, Blendeinstellung
- Belichtungsprogramm, ISO-Wert, GPS Koordinaten, Thumbnail
- IPTC-Daten (Kommentar, Name)

Name der gespeicherten Datei	DSC04923.ARW
Kameramodell	Sony Alpha 380
Firmware	Firmware Version 1.1
Aufnahmedatum/-zeit	27.11.2011 15:39:28
Autor/Photographer	xxxx xxxx (Name)
Copyright-Vermerk	Copyright: xxxx xxxx
Name des Besitzers	xxxx xxxx
Aufnahmemodus	A (Blendenpriorität)
Tv (Verschlusszeit)	1/6

Quelle Wikipedia

Kodierung

Anwendungsbereiche

Web-Grafik (klein, geringe Farbanzahl) → GIF od. PNG

Scanner/Bilderzeugung → TIFF

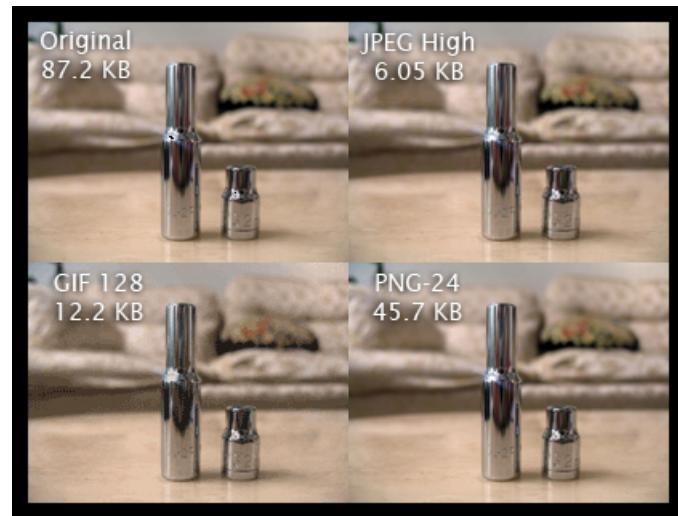
Austausch zwischen Geräten → TIFF

Hochauflösende Bilder mit vielen Farben →

- JPEG - bessere Kompression
- PNG - bessere Qualität, gute Kompression bei grossen einheitlichen Farbflächen

Kodierung

Anwendungsbereiche - Vergleich Kompression



<http://rightyaleft.com/others/png-or-jpeg-or-gif-which-is-best-way-to-choose-your-image-file/>



<http://rightyaleft.com/others/png-or-jpeg-or-gif-which-is-best-way-to-choose-your-image-file/>

Zusammenfassung

Digitalisierung und Kodierung

Digitalisierung

- Rastergrafik, Vektorgrafik, Meta-Files
- Pixel, Bildgröße, Auflösung, Skalierung, Farbtiefe, Farbkanäle, Paletten
- Anti-Aliasing, Dithering

Kodierung

- Generelle Formate: TIFF, BMP
- Web-optimierte Formate: GIF, PNG, JPEG
- Metadaten: EXIF
- Kompression: verlustbehaftet oder verlustfrei

Zusammenfassung

[Bibliographie]

- **Malaka, Butz, Hussmann (2009)** - Medieninformatik: Eine Einführung
(Pearson Studium - IT)
- **Butz (2011)**- Prof. Andreas Butz, Vorlesung Digitalen Medien - LMU München, 2011

Kapitel Medientechnik:IV

I. EIC Teil 2- Medium Bild

- Farbwahrnehmung und Farbmodelle
- Digitalisierung und Kodierung von Bildern
- JPEG Kompression (Überblick, nicht prüfungsrelevant)
- Vektorgrafiken am Beispiel SVG (Scalable Vector Graphics)

JPEG Kompression im Überblick

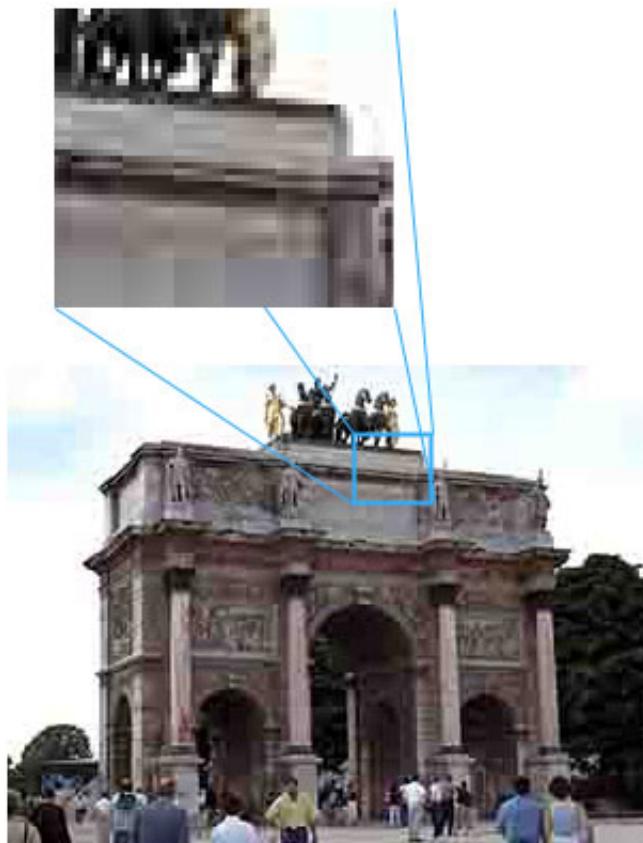
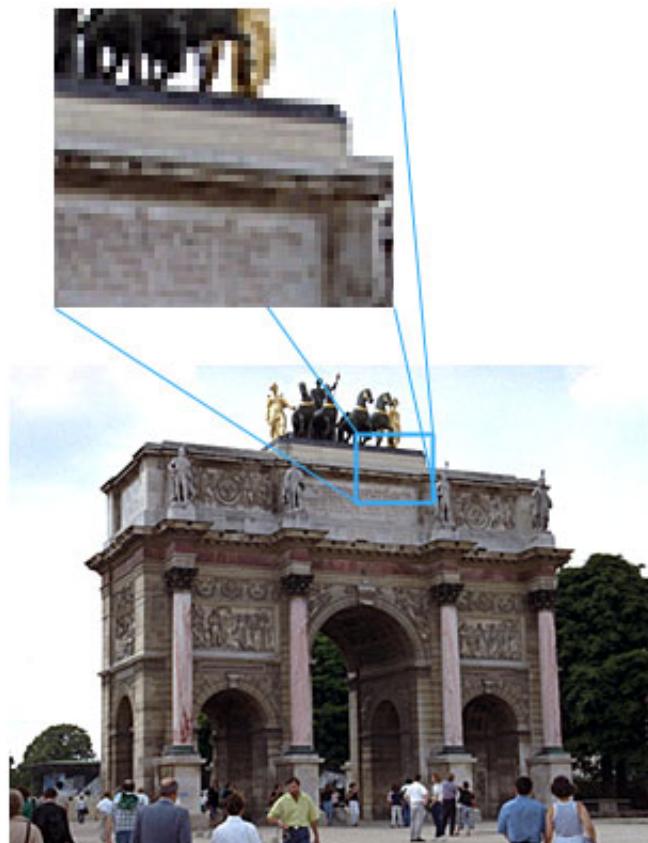
Verlustbehaftete Kompression

- **Problem:** Theoretische und praktische Grenzen von verlustfreien Kompressionsverfahren
- **Lösung:** Zur Erhöhung der Kompressionsraten muss Wissen über die Daten eingebracht werden
- JPEG-Kompression ist ein mehrstufiges Verfahren, welches Wissen über visuelle Wahrnehmung nutzt
 - Die Wahrnehmung wertet nicht alle Informationen des Bildes gleich gut aus
 - Beispiel: Helligkeit vs. Farbigkeit
 - Beispiel: Feinabstufung von Verläufen
 - JPEG eignet sich daher am besten für Bilder realistischer Szenen (Photographie)
 - Weniger gut bei Linienzeichnungen, Icons etc.
 - Mehrmaliges editieren verschlechtert die Qualität
- Interlaced Modus: JPEG Progressiv

Hier: Überblick über das Verfahren. Details werden in Master-Vorlesungen behandelt.

JPEG Kompression im Überblick

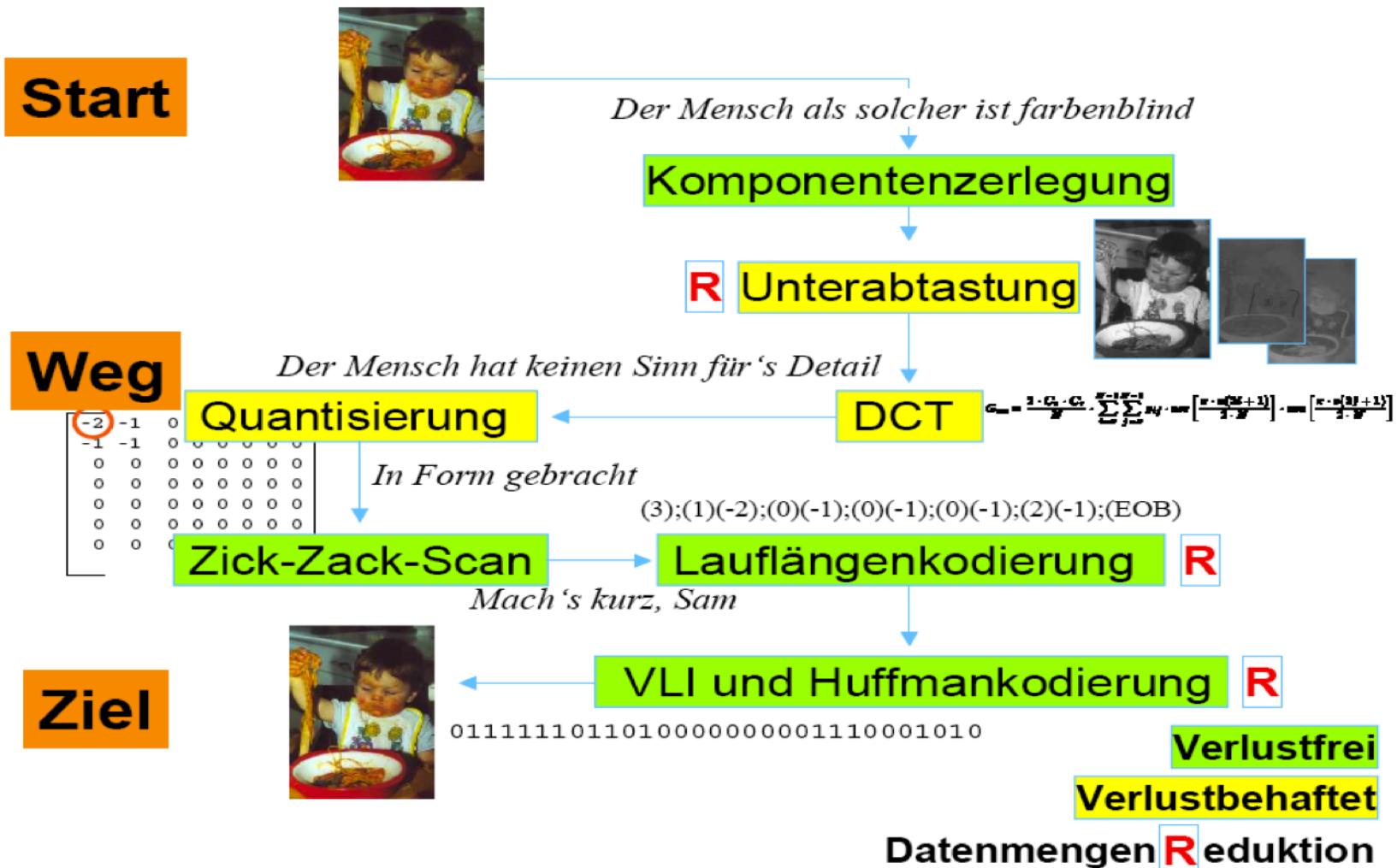
JPEG Kompression: Vergleich



Bildquelle http://www.mathematik.de/spudema/spudema_beitraege/beitraege/rooch/nkap04.html

JPEG Kompression im Überblick

JPEG Kompression: Ablauf



JPEG Kompression im Überblick

JPEG Kompression: Ablauf

Ablauf

1. Konvertierung von RGB nach $Y'C_BC_R$ (Luminanz + 2 Chroma Kanäle)
2. **Chroma Subsampling:** Reduktion der Chroma Kanäle (Farbe)
3. Split in 8x8 Pixel Blöcke und **Discrete Cosinus Transformation** auf jeden Kanal
4. **Quantisierung** der Amplituden der Frequenzen (Hochfrequenz Bereiche ungenauer)
5. **Entropie-Kodierung:** “Zick-Zack” Lauflängenkodierung und Huffman Kodierung

JPEG Kompression im Überblick

JPEG Kompression: Effekte

High Quality, 83k, 2.6:1 Ratio



Bildquelle Wikipedia

Medium Quality, 15kk, 46:1 Ratio



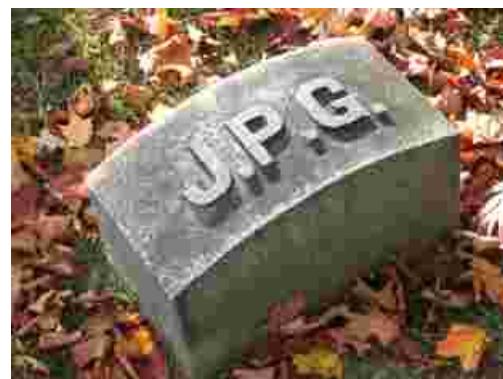
Bildquelle Wikipedia

Medium-High Quality, 9k, 23:1 Ratio



Bildquelle Wikipedia

Low Quality, 4k, 144:1 Ratio



Bildquelle Wikipedia

JPEG Kompression im Überblick

Bibliographie

- **Malaka et.al. 2009** Malaka, Butz, Hussmann (2009) - Medieninformatik:
Eine Einführung (Pearson Studium - IT), Kapitel 2.3
- **Kerr** Chrominance Subsampling in Digital Images,
<http://dougkerr.net/pumpkin/articles/Subsampling.pdf>
- **Poynton 2008** Charles Poynton, Chroma subsampling notation, 2008
http://scanline.ca/ycbcr/Chroma_subsampling_notation.pdf

Kapitel Medientechnik:IV

I. EIC Teil 2- Medium Bild

- Farbwahrnehmung und Farbmodelle
- Digitalisierung und Kodierung von Bildern
- JPEG Kompression (Überblick, nicht prüfungsrelevant)
- Vektorgrafiken am Beispiel SVG (Scalable Vector Graphics)

Vektorgrafik Allgemein

Vektorgrafik Allgemein

Bilder vs. Grafiken

Digitales Bild besteht aus N Zeilen und je M Bildpunkten Anwendungsbereiche:

- Pixeln bzw. Picture Elements (Auflösung, Farbtiefe etc.)
- Bild kann aus der realen Welt kommen oder virtuell sein
- Wie beschreiben wir digital erstellte Grafiken?

(Vektor-)Grafiken: durch grafische Primitive und ihre Attribute spezifiziert

- Primitive 2D Objekte: Linien, Rechtecke, Kreise, Ellipsen, Texte
- 2D Formate: SVG, PostScript, Windoes Metafile, CorelDraw, PDF ...
- Primitive 3D Objekte: Polyeder, Kugeln, ...
- Formate 3D: VRML, X3D
- Attribute: Stil der Linie, Breite, Farbe etc

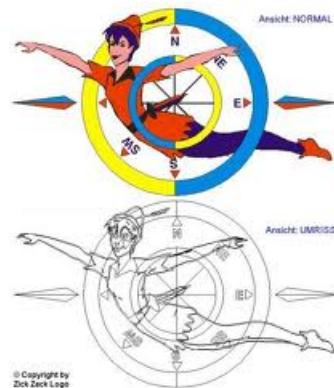
Vektorgrafik Allgemein

Vektorgrafiken

Definition 3 (Vektorgrafik)

Als Vektorgrafiken bezeichnet man **mathematisch und programmatisch** definierte Zeichenanweisungen in einem Koordinatensystem, aus welchen Rastergrafiken generiert, gespeichert und präsentiert werden können.

Eigenschaften



Bildquelle zz-logo.de

- Vektorgrafiken können einfach und exakt geometrisch transformiert werden
 - Rotation, Skalierung, Verschiebung
 - Separierung einzelner Bildelemente (Gruppierung, Ebenen)
 - Änderung der Attribute einzelner Elemente (Farbe einer Fläche etc)
 - Im Grunde ein perfektes mathematisches Modell, welches jedoch in ein Rasterbild überführt werden muss

Vektorgrafik Allgemein

Bilder vs. Grafiken

Unterschiede zu Rasterbild

- Erstellungsformat - Editor, Vektorgrafik-Programm
- Speicherformat - XML, proprietär (nicht in binären Daten)
- Ausgabeformat - Rendering
- Falls keine Konvertierung in ein Standardformat, sind je Editorensystem eigene Previewer nötig
- Für einen Ausdruck sind (fast) immer Konvertierungen nötig (z.B. nach Postscript)

Vektorgrafik Allgemein

Koordinatensystem

Koordinatensystem: Grundlage jeder 2D Vektorgrafik ist ein zweidimensionaler Vektorraum

- X/Horizontal- und Y/Vertikalachse
- i.A. gleicher Abstand auf beiden Achsen

Elemente

- Punkt beschrieben durch X/Y Koordinate
- Gerade beschrieben durch Start- und Endpunkt
- Polygon aus mehreren Geraden
 - Geschlossenes Polygon (Fläche)
 - Offenes Polygon
- Kreis - Radius und Mittelpunkt

SVG - Scalable Vector Graphics

SVG - Scalable Vector Graphics

Lernziel

Unterthemen

- Überblick SVG und ähnliche Formate
- Statische SVG Bilder
- Animationen in SVG
- Beispiele
- Erstellungsprogramme

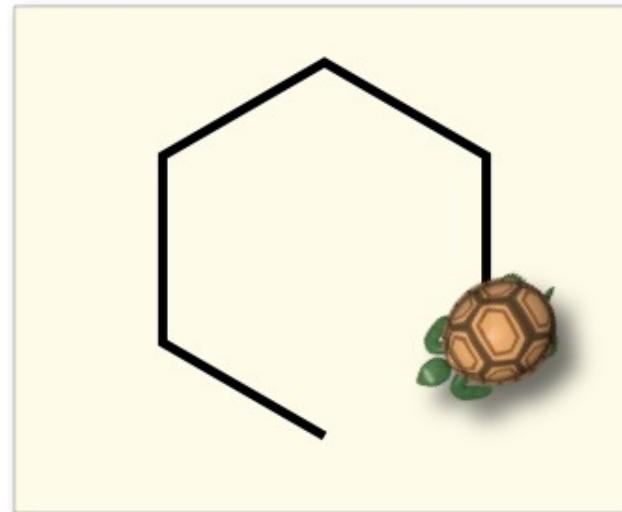
SVG - Scalable Vector Graphics

Grundelemente

Zur **Kodierung** von Vektorgrafiken ist eine entsprechende Sprache zur Definition von Geometrie und Animation notwendig.

Beispiel Turtle Grafik

- Bekannt durch Logo Programmiersprache aus den 1970iger Jahren zum Lernen von Programmierung
- Turtle: Position, Orientierung, Stift (Größe, Farbe etc.)
- Sprache beschreibt Weg: “move forward 10 units”; “lift pen”; “turn left 90°”
- Ähnlich dem Turtle Robot (physisch) aus der frühen Robotik Forschung 1960



Bildquelle <http://www.alancsmith.co.uk/logo/>

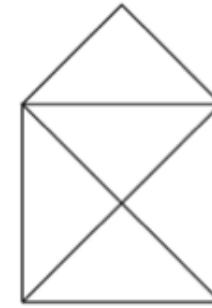
SVG - Scalable Vector Graphics

Grundelemente

Post Script/Encapsulated PostScript (EPS)/Portable Document Format (PDF)

- PostScript entwickelt 1984 von Adobe zur geräteunabhängigen Darstellung formatierter Texte
- Darstellung von Vektorgrafik + Rastergrafik
- Vollständige Programmiersprache
- Wird meist von Druckern (e.g. Laserdrucker) implementiert
- Angabe von Punkten und Pfaden. Pfad wird dann mit Zeichengerät (e.g. Stift, Pinsel) gezeichnet (siehe Beispiel)
- PDF als Nachfolger: Bessere Komprimierung, dafür keine vollständige Programmiersprache mehr

```
%!PS  
100 100 newpath moveto  
100 200 lineto  
200 200 lineto  
150 250 lineto  
100 200 lineto  
200 100 lineto  
100 100 lineto  
200 200 lineto  
200 100 lineto  
stroke  
showpage
```



Bildquelle [?]

Scalable Vektor Grafik - Grundlagen

Scalable Vektor Grafik - Grundlagen

SVG-Überblick

Sprache für 2D-Graphik in XML, welche kombinierbar mit anderen Web-Standards

Drei Typen grafischer Objekte

- Shapes (Pfade aus Kurven und geraden Linien)
- Bilder (Raster-Graphik)
- Text

Grafische Objekte können

- gruppiert
- gestyled (CSS)
- transformiert
- zusammengesetzt werden

Vorteile

- Zusammengesetzte Transformationen
- Clipping paths (Bilder flexibel zuschneiden)
- Alpha-Masken (Durchsichtigkeit von Objekten)
- Filter-Effekte, Objektvorlagen

SVG Zeichnungen sind potenziell

- interaktiv und
- dynamisch

W3C Standard <http://www.w3.org/Graphics/SVG/>

Scalable Vektor Grafik - Grundlagen

Grundelemente SVG

Koordinatensystem:

- Koordinatensystem: (0, 0) links oben
- User Koordinaten System korrespondiert mit Bildschirmkoordinatensystem per Default (100 Pixel in SVG Datei entsprechen 100 Pixel am Bildschirm)
- Änderungen am User Koordinatensystem möglich

- Pfade (ähnlich der Turtle) als grundlegende Zeichenobjekt
- Erweiterung um geometrische Objekte (e.g. Kreis, Rechteck etc.)
- Definition von Attributen (e.g. Strichbreite) pro Pfad/Objekt

⇒ Scalable Vector Graphics (SVG) 1.1 (Second Edition)

<http://www.w3.org/TR/SVG/Overview.html>

Scalable Vektor Grafik - Grundlagen

XML Grundstruktur SVG

XML-Deklaration

DTD-Bezug

SVG-Namespace

Größe der
Ansichtsfläche
&Position im
Browserfenster

```
<?xml version="1.0" standalone="no"?>  
  
<!DOCTYPE svgPUBLIC "-//W3C//DTD SVG 1.1//  
EN" "http://www.w3.org/Graphics/SVG/1.1/  
DTD/svg11.dtd">  
  
<svg xmlns="http://www.w3.org/2000/svg"  
width="300" height="300" x="0" y="0">  
.....  
</svg>
```

Scalable Vektor Grafik - Grundlagen

XML Grundstruktur SVG

□ <svg>

1. Definitionen wieder verwendbarer Bestandteile

- * Pfade

- * Gradienten

- * Filter

2. Zeichnen unter Verwendung der Definitionen und Grundoperationen

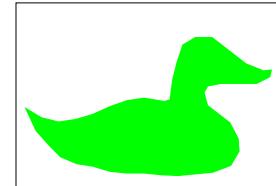
□ </svg>

Scalable Vektor Grafik - Grundlagen

Einfaches SVG Beispiel

```
<?xml version="1.0"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  width="320" height="220">
  <rect width="320" height="220" fill="white" stroke="black""/>
  <g transform="translate(10 10)">
    <g stroke="none" fill="lime">
      <path d="M 0 112 L 20 124 L 40 129 L 60 126 L 80 120
        L 100 111 L 120 104 L 140 101 L 164 105 L 170 103
        L 173 80 L 178 60 L 185 39 L 200 30 L 220 30
        L 260 61 L 280 69 L 290 68 L 288 77 L 272 85
        L 250 85 L 230 85 L 215 88 L 211 95 L 215 110
        L 228 120 L 241 130 L 251 149 L 252 164 L 242 181
        L 221 189 L 200 191 L 180 193 L 160 192 L 140 190
        L 120 190 L 100 188 L 80 182 L 61 179 L 42 171
        L 30 159 L 13 140 Z"/>
    </g> </g>
  </svg>
```

Quelle: Prof. Butz, LMU



Zum

Ausprobieren: http://www.w3schools.com/svg/tryit.asp?filename=trysvg_myfirst

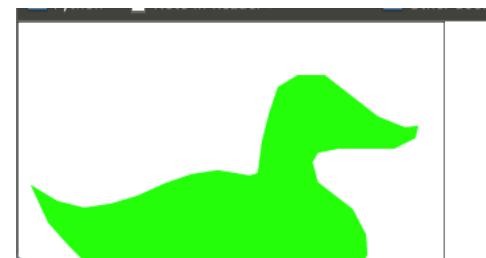
Scalable Vektor Grafik - Grundlagen

SVG - Canvas Größe

Bestimmung der Größe der Zeichenfläche auf 2 Arten möglich

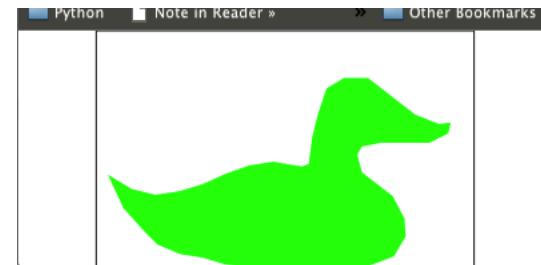
Absolute Größenangabe, d.h. Grafik wird bei Verkleinerung abgeschnitten

```
<svg width="320" height="220">
```



Angabe eines Sichtfensters, d.h. Größe wird bei Änderung des Fensters skaliert

```
<svg viewBox="0 0 320 200">
```



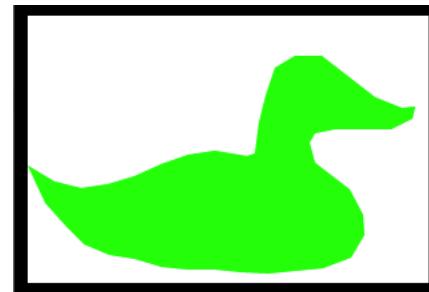
Scalable Vektor Grafik - Grundlagen

SVG - Rendering Attribute

Beeinflussung eines grafischen Objektes mit Attributen

Angabe der Attribute direkt in XML Tag, über `style` Definition in CSS2-Syntax oder über CSS2-Stylesheet

- Füllfarbe `fill`
- Transparenz `opacity`
- Linienfarbe und -stärke `stroke` und `stroke-width`
- Linienenden `stroke-linecap`
- Schriftfamilie und -größe `font-family` und `font-size`



`<rect stroke-width="20"/>`

Scalable Vektor Grafik - Grundlagen

SVG mit Stylsheet

```
<?xml-stylesheet type="text/css" href="renderstyle.css" ?>
<svg viewBox="0 0 300 300">
  <rect class="heavy" width="300" height="300"/>
  <rect class="type1" x="100" y="100" width="100" height="100"/>
  <rect class="type2" x="50" y="50" width="100" height="100"/>
</svg>
```

SVG-Datei

```
rect {stroke:black; fill:white}
rect.type1 {stroke:none; fill:red}
rect.type2 {stroke:black; stroke-width:6; fill:green}

.heavy {stroke:black; stroke-width:10}
```

renderstyle.css

Quelle Butz LMU

Scalable Vektor Grafik - Grundlagen

SVG - Pfad Syntax

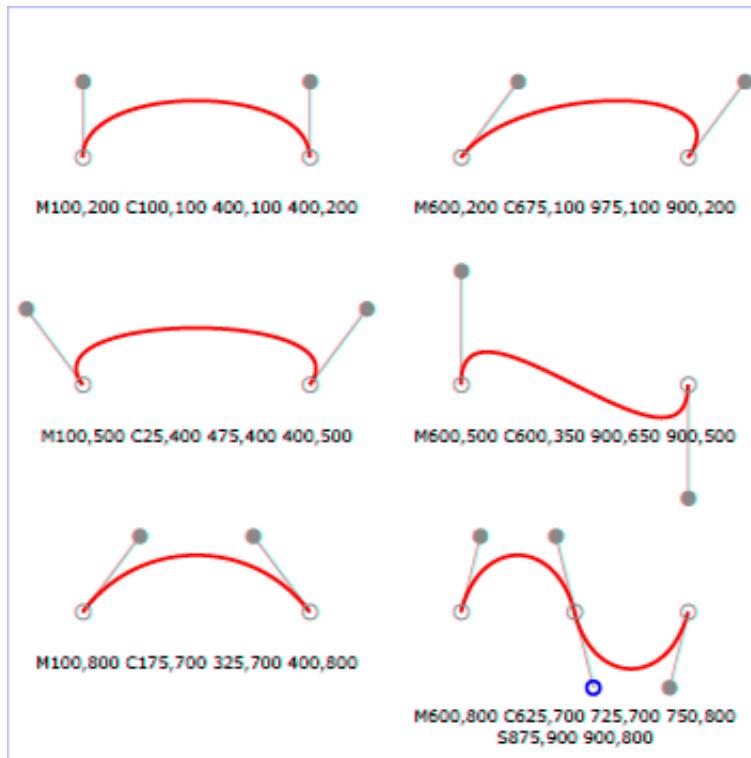
Pfade definieren eine Folge von Zeichenkommandos für einen virtuellen Zeichenstift

- Syntax ist knapp gehalten, um Speicherplatz zu sparen
 - Kommandos mit Zeichenlänge 1, relative Koordinaten, keine Token Separatoren wenn möglich, Berzier-Kurven Formulierung
 - Zusätzlich Möglichkeit der verlustfreien Kompression (z.B. Huffmann)
- Kommandos
 - M X Y Startpunkt auf Koordinate X,Y
 - L X Y Linie nach X Y
 - Z Gerade Linie zurück zum Startpunkt
 - H X Horizontale Linie bis Koordinate X
 - V X Vertikale Linie bis Koordinate X
 - Z Gerade Linie zurück zum Startpunkt
 - Q cx cy x y Quadratische Berzier-Kurve nach X,Y mit Kontrollpunkt cx,cy
 - C c1x c1y c2x c2y x y Kubische Berzier-Kurve nach X,Y mit den beiden Kontrollpunkten (c1x,c1y) und (c2x,c2y)
 - A rx ry x-rot la-flag sweep-flag x y Elliptische Kurve
 - Kleinbuchstaben Versionen des Kommandos stehen für relative Koordinaten (e.g. l x y)

Scalable Vektor Grafik - Grundlagen

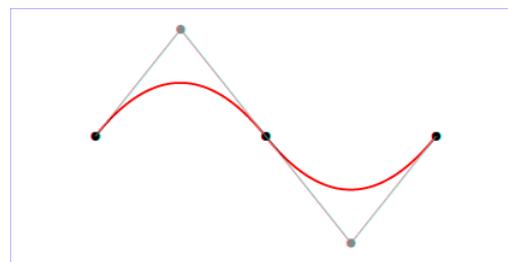
SVG Bezier Kurven

Kubische Bezier Kurven



<http://www.w3.org/TR/SVG/paths.html>

Quadratische Bezier Kurven

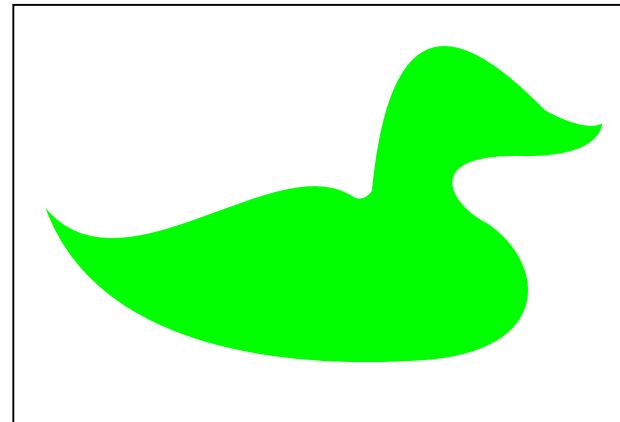


<http://www.w3.org/TR/SVG/paths.html>

Scalable Vektor Grafik - Grundlagen

SVG Bezier Pfad Beispiele

```
<?xml version="1.0"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      width="320" height="420">
  <rect x="0" y="200" width="320" height="220" fill="white" stroke="black"/>
  <g transform="translate(10 10)">
    <g stroke="none" fill="lime">
      <path d="M 0 312
C 40 360 120 280 160 306 C 160 306 165 310 170 303
C 180 200 220 220 260 261 C 260 261 280 273 290 268
C 288 280 272 285 250 285 C 195 283 210 310 230 320
C 260 340 265 385 200 391 C 150 395 30 395 0 312 z"/>
    </g> </g>
  </svg>
```



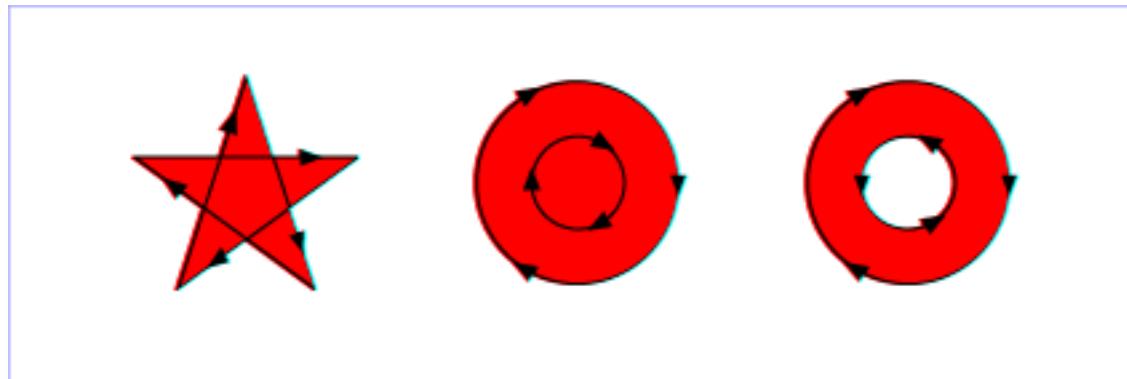
Quelle: [?]

Scalable Vektor Grafik - Grundlagen

SVG - Füllregeln

Füllen benötigt die Bestimmung, ob ein Punkt innen liegt oder nicht.

nonzero: Für jeden Punkt sende einen Strahl ins unendliche. Für jede links-rechts (rechts-links) gezeichnete Linie erhöhe (erniedrige) den Zähler. Wenn Zähler ungleich 0, dann liegt der Punkt innen



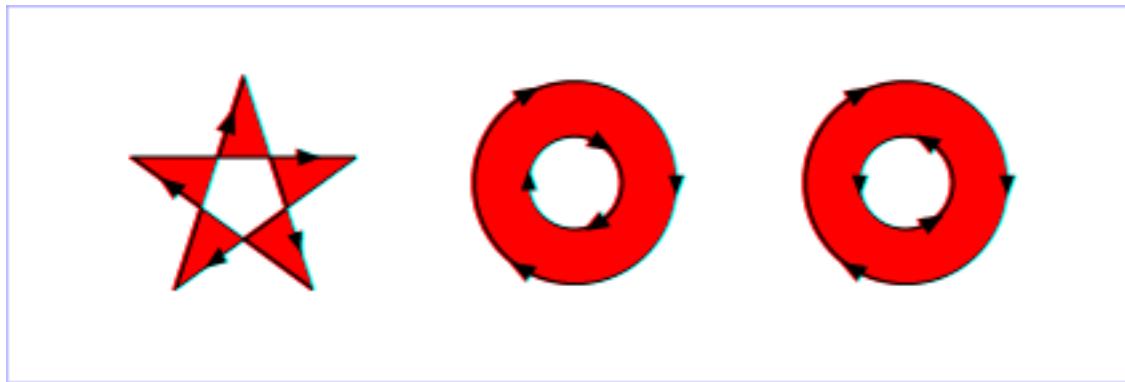
Bildquelle <http://www.w3.org/TR/SVG/painting.html>

Scalable Vektor Grafik - Grundlagen

SVG - Füllregeln

Bestimme, ob ein Punkt innen liegt oder nicht

evenodd: Für jeden Punkt sende einen Strahl ins unendliche. Für jede links-rechts (rechts-links) gezeichnete Linie erhöhe den Zähler. Wenn Zähler ungerade ist, dann liegt der Punkt innen



Bildquelle <http://www.w3.org/TR/SVG/painting.html>

Scalable Vektor Grafik - Grundlagen

SVG - Text

<text>

- Platzierung von Text auf der Leinwand
- Koordinaten-Attribute x und y: Linke untere Ecke des ersten Buchstabens
- Schrift, Größe etc. über Attribute oder Stylesheet

<tspan>

- Untergruppe von Text in einem <text>-Element
- Einheitliche Formatierung (wie in HTML)
- Relative Position zur aktuellen Textposition: Attribute dx und dy

Spezialeffekte

- Drehen einzelner Buchstaben (**rotate**-Attribut)
- Text entlang eines beliebigen Pfades (<textpath>-Element)

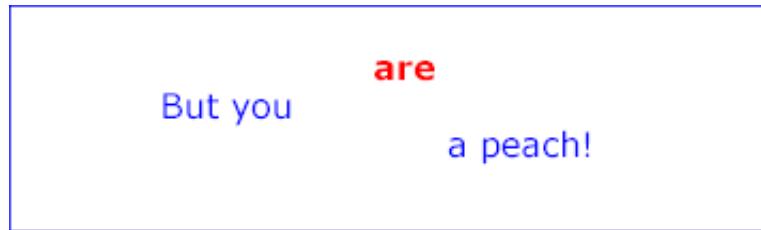
Scalable Vektor Grafik - Grundlagen

SVG Text einfach

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300"
 xmlns="http://www.w3.org/2000/svg" version="1.1">
<desc>Example tspan02 - using tspan's dx and dy attributes
 for incremental positioning adjustments</desc>

<g font-family="Verdana" font-size="45" >
<text x="200" y="150" fill="blue" >
  But you
  <tspan dx="2em" dy="-50" font-weight="bold" fill="red" >
    are
  </tspan>
  <tspan dy="100">
    a peach!
  </tspan>
</text>
</g>

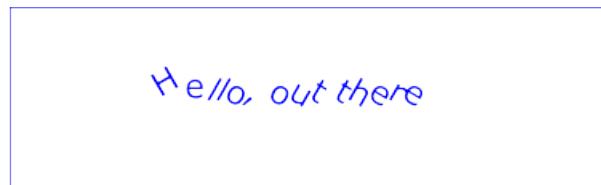
<!-- Show outline of canvas using 'rect' element -->
<rect x="1" y="1" width="998" height="298"
      fill="none" stroke="blue" stroke-width="2" />
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG Text rotiert

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300"
xmlns="http://www.w3.org/2000/svg" version="1.1">
<desc>
  Example tspan04 - The number of rotate values is less than the number of
  characters in the string.
</desc>
<text font-family="Verdana" font-size="55" fill="blue" >
  <tspan x="250" y="150" rotate="-30,0,30">
    Hello, out there
  </tspan>
</text>
<!-- Show outline of canvas using 'rect' element -->
<rect x="1" y="1" width="998" height="298"
fill="none" stroke="blue" stroke-width="2" />
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG - Geometrische Primitive und Transformationen

Pfade definieren i.A. keine geschlossenen Objekte

SVG definiert auch Standard Objekte als geometrische Primitive

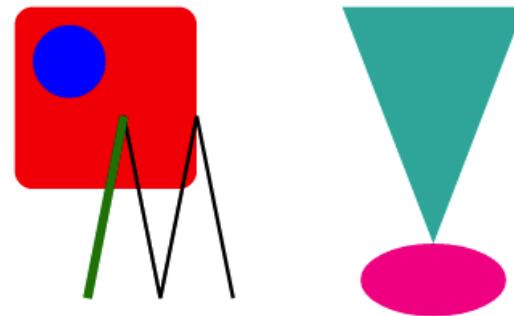
Elementname	Bedeutung	Attribute
<line>	Linie	x1, y1: Erster Punkt x2, y2: Zweiter Punkt
<polyline>	Folge zusammenhängender Linien	points: Folge von x, y
<polygon>	Polygon	points: Folge von x, y
<rect>	Rechteck	x, y: Linke obere Ecke width: Breite, height: Höhe rx, ry: Radien der Ecken
<circle>	Kreis	cx, cy: Zentrum, r: Radius
<ellipse>	Ellipse	cx, cy: Zentrum rx, ry: Radien

Quelle Prof. Butz, LMU

Scalable Vektor Grafik - Grundlagen

SVG Geometrie Beispiel

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="10cm" height="3cm" viewBox="0 0 1000 300"
xmlns="http://www.w3.org/2000/svg" version="1.1">
<rect x="20" y="20" width="100" height="100" rx="10"
ry="10" fill="red" stroke="none"/>
<circle cx="50" cy="50" fill="blue" r="20"/>
<polyline points="80,80 100,180 120,80 140,180"
fill="none" stroke="black" stroke-width="2"/>
<line x1="80" y1="80" x2="60" y2="180" stroke="green"
stroke-width="5"/>
<polygon points="200,20 300,20 250,150"
fill="lightseagreen"/>
<ellipse cx="250" cy="170"
rx="40" ry="20"
fill="deeppink"/>
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG -Gruppen und Transformationen

Gruppen - Geometrische Primitive können zu Gruppen kombiniert werden <g>

- Einheitliche Attributdefinition für element der Gruppe
- Manipulation der gesamten Gruppe (e.g. Transformation)

Transformationen

- Verschieben (translate), drehen (rotate), verzerrn (skew) oder skalieren (scale)
- SVG Attribut transform
- Wert des Attributtes entspricht der Operation plus parameter

Scalable Vektor Grafik - Grundlagen

SVG Geometrie Beispiel

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="400px" height="120px" version="1.1"
      xmlns="http://www.w3.org/2000/svg">
  <g transform="translate(200,30)">
    <g transform="skewY(30)">
      <g fill="none" stroke="red" stroke-width="3" >
        <line x1="0" y1="0" x2="50" y2="0" />
        <line x1="0" y1="0" x2="0" y2="50" />
      </g>
      <text x="0" y="0" font-size="20" font-family="Verdana" fill="blue" >
        ABC (skewY)
      </text>
    </g>
  </g>
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG - Clipping

Clipping - Ausschneiden von Pfaden aus Objekten

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="400px" height="500px" version="1.1"
      xmlns="http://www.w3.org/2000/svg">
  <clipPath id="myclip">
    <circle cx="250" cy="150" r="150"/>
  </clipPath>
  <g clip-path="url(#myclip)">
    <rect width="500" height="100"
          x="0" y="0" fill="black"/>
    <rect width="500" height="100"
          x="0" y="100" fill="red"/>
    <rect width="500" height="100"
          x="0" y="200" fill="gold"/>
  </g>
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG- Alpha Composition, Masking

Opacity - Transparenz von Objekten

Masking - Nutzung von beliebigen Objekten zur Maskierung (Überdeckung) eines anderen Objektes

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="400px" height="500px" version="1.1"
      xmlns="http://www.w3.org/2000/svg">
<g>
    <rect width="500" height="100"
          x="0" y="0" fill="black"/>
    <rect width="500" height="100"
          x="0" y="100" fill="red"/>
    <rect width="500" height="100"
          x="0" y="200" fill="gold"/>
</g>
<circle cx="250" cy="150" r="150" opacity="0.5" fill="green"/>
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG - Links

Hyperlinks können über den Xlink (XML Links) Standard

<http://www.w3.org/1999/xlink> eingefügt werden

- Der Namensraum muss im svg Tag spezifiziert werden
- Beispiel:

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
  "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <a xlink:href="http://mics.fim.uni-passau.de">
    <circle cx="50" cy="50" fill="blue" r="20"/>
  </a>
</svg>
```

Scalable Vektor Grafik - Grundlagen

SVG - Symbole

Symbole können zur wiederholten Verwendung definiert werden

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
 xmlns:xlink="http://www.w3.org/1999/xlink">
<symbol id="sym1">
  <circle cx="50" cy="50" fill="blue" r="20"/>
</symbol>
<use xlink:href="#sym1" x="30" y="30"/>
<use xlink:href="#sym1" x="50" y="50" opacity=".3"/>
</svg>
```



Scalable Vektor Grafik - Grundlagen

SVG - Animationen

SVG-Objekte können zeitabhängig verändert werden.

Elemente

- `animate` - Animation eines einzelnen Attributes über die Zeit. Beispiel für Ausblenden (`opacity`) über die Zeit:

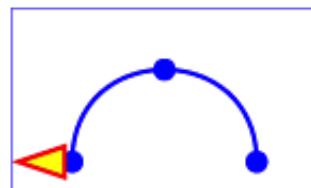
```
<rect> <animate attributeType="CSS"  
attributeName="opacity" from="1" to="0" dur="5s"  
repeatCount="indefinite" /> </rect>
```

- `set` - setzen eines Attributwertes für ein spezifische Zeit
- `animateMotion` - bewegt ein referenciertes Element entlang eines Pfades
- `animateColor` - Farbtransformation über die Zeit
- `animateTransform` - Transformation (Rotation, Skalierung etc.) über die Zeit
- Grundattribute: `from`, `by`, `to` zur Spezifikation der Startwerte, Schrittweite und Endwerte
- Abhängig v. d. Elementart weitere Attribute spezifizierbar

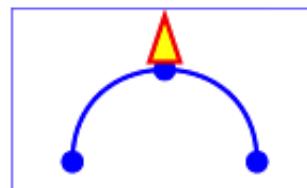
Scalable Vektor Grafik - Grundlagen

SVG - Animationen

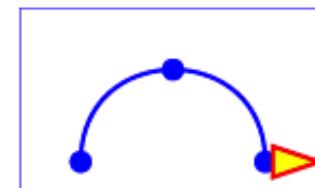
```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
 "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="5cm" height="3cm" viewBox="0 0 500 300"
      xmlns="http://www.w3.org/2000/svg" version="1.1"
      xmlns:xlink="http://www.w3.org/1999/xlink" >
  <rect x="1" y="1" width="498" height="298"
        fill="none" stroke="blue" stroke-width="2" />
  <path id="path1" d="M100,250 C 100,50 400,50 400,250"
        fill="none" stroke="blue" stroke-width="7.06" />
  <circle cx="100" cy="250" r="17.64" fill="blue" />
  <circle cx="250" cy="100" r="17.64" fill="blue" />
  <circle cx="400" cy="250" r="17.64" fill="blue" />
  <path d="M-25,-12.5 L25,-12.5 L 0,-87.5 z"
        fill="yellow" stroke="red" stroke-width="7.06" >
    <animateMotion dur="6s" repeatCount="indefinite" rotate="auto" >
      <mpath xlink:href="#path1"/>
    </animateMotion>
  </path>
</svg>
```



At zero seconds



At three seconds



At six seconds

Scalable Vektor Grafik - Grundlagen

Software zur Darstellung und Erzeugung von SVG

- Direkte Browserunterstützung:
 - Firefox, Safari, Opera, Chrome
 - (derzeit) nicht in Internet Explorer
 - Diverse Plugins für Internet Explorer, z.B.: Adobe SVG Viewer (nicht weiterentwickelt), Google Chrome Frame
- Früher: Spezialsoftware (Standalone Viewer)
- Vektorgrafik-Editoren mit SVG-Import und Export z.B. Adobe Illustrator, CorelDraw
- SVG-orientierte Grafik-Editoren z.B. Inkscape (Open Source), Sketsa
- XML-Editoren, Keine Grafik-Unterstützung, nur Text-Syntax

Auch für 3D Grafiken gibt es solche Formate (z.B. VRML)

Zusammenfassung

Scalable Vector Graphics

- Standardformat für Vektorgrafiken im WWW
- Kombinierbar mit anderen Web-Standards
- Elemente: Pfade, geometrische Primitive, Bezier, Kreis, Elipse,
- Text, Animationen, Transformationen und Füllungen
- Links, Symboldefinitionen etc.