

# **7. Anwendungen:**

## **7.2 Farbadaption in Videos**

### **Videoanalyse**

Dr. Stephan Kopf

# Übersicht

- Motivation
- Adaptionungsverfahren
- Adaption und Standardisierung
- Adaption der Farbtiefe
- High-Dynamic-Range-Bilder (HDR-Bilder)
- Zusammenfassung

# Motivation (I)

- Wiedergabe eines Videos ist heute auf einer Vielzahl von Geräten möglich, die hinsichtlich ihrer Ausstattungsmerkmale deutlich variieren (Größe der Displays, unterschiedliche Übertragungskapazitäten der Netzwerke).
- Besonders schwierig ist die Wiedergabe von Videos auf mobilen Geräten (Notebooks, Tablet-PCs, Handheld-PCs (PDAs), Mobiltelefone).
- Wesentliche Eigenschaften: Auflösung und Farbtiefe des Displays, Größe des Arbeitsspeichers, Leistungsfähigkeit des Prozessors, verfügbare Software zur Dekodierung und Darstellung eines Videos.

# Motivation (II)

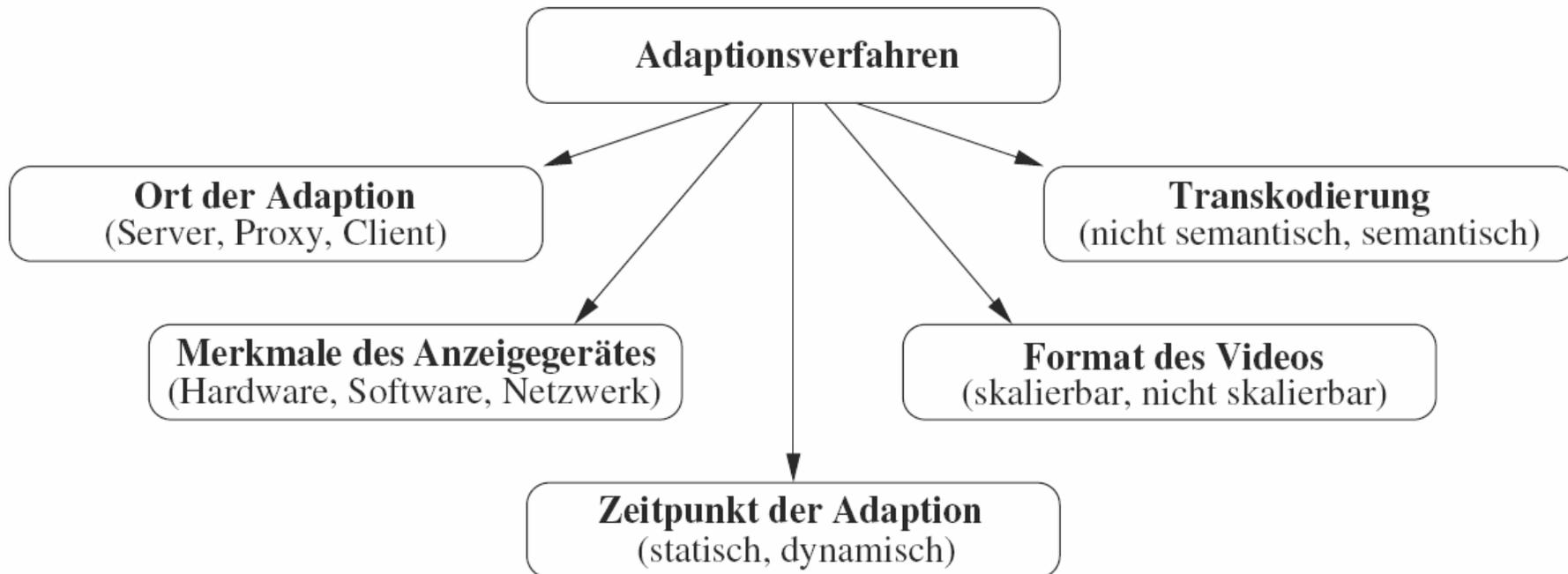
- Für eine gute Darstellung sollten bestehende Videos möglichst genau an die unterschiedlichen Eigenschaften der Anzeigegeräte angepasst werden. Eine manuelle Festlegung der Parameter für alle Kombinationen von Videos und Anzeigegeräten verursacht einen sehr hohen Aufwand.
- Verfahren zur automatischen Adaption von Videos ermöglichen die Wiedergabe bestehender Videos auch auf mobilen Geräten ohne zusätzlichen Aufwand.
- Das zentrale Ziel der Adaption ist der Erhalt der semantischen Informationen eines Videos unabhängig von der Ausstattung eines Anzeigegerätes.

# Adaptionsverfahren (I)

- Wesentliche Parameter eines Videos, die bei der Adaption geändert werden müssen:
  - Bitrate
  - **Farbtiefe**
  - Bildauflösung
  - Bildwiederholrate
  - Verbesserung der Bildqualität des ursprünglichen Videos (insbesondere bei historischen Videos oder Amateurvideos)
  - Bessere Darstellung der gleichzeitig dargestellten Farben bzw. Helligkeitswerte (HDR-Bilder/HDR-Videos)

# Adaptionsverfahren (II)

## Verfahren zur Adaption von Videos



# Adaptionsverfahren (III)

## Verfahren zur Adaption von Videos

- Ort der Adaption:
  - Server → Performanceengpässe
  - Client → große Datenmengen
- Zeitpunkt der Adaption:
  - dynamisch: Die Berechnung und Transkodierung des Videos erfolgt in Echtzeit.
  - statisch: Im Vorfeld werden mehrere statische Versionen eines Videos für ausgewählte Geräteprofile berechnet und gespeichert.

# Adaption und Standardisierung (I)

- Die beiden Standards *MPEG-7* und *MPEG-21* unterstützen die semantische Beschreibung der Inhalte eines Videos.

## **MPEG-7**

- Datenbeschreibungssprache zum vereinfachten Austausch multimedialer Daten.
- Netzbasierter Zugriff von beliebigen Geräten auf multimediale Daten (*Universal Multimedia Access*).
- Regeln für die Transkodierung von Videos
- Nutzerhistorie und individuelle Nutzerpräferenzen (*user preference description*).

# Adaption und Standardisierung (II)

## *MPEG-21*

- Beschreibung der Gerätemerkmale (*usage environment description*)
- Daten sind zur Charakterisierung des Displays, der Systemkonfiguration sowie der verfügbaren Hardware und Software vorgesehen.
- Modellierung von Nutzeranfragen und Nutzerpräferenzen
- Für alle digitalen Elemente (*digital item*) innerhalb von MPEG-21 können spezielle Adaptionsverfahren definiert werden (*digital item adaptation*).

# Adaption der Farbtiefe



# Adaption der Farbtiefe

- Bei einer Verringerung der Farbtiefe auf wenige Helligkeitswerte können große Regionen mit gleichen Helligkeitswerten entstehen, so dass der Bildinhalt in Teilen des Videos nicht mehr erkannt werden kann.
- Besondere Herausforderung: Adaption eines Videos für monochrome Displays mit zwei unterschiedlichen Helligkeitswerten.
- Umwandlung der Farbpixel in Graustufenwerte: ohne zusätzlichen Rechenaufwand möglich, da die Helligkeit unabhängig von den Farbinformationen ähnlich dem YUV-Farbmodell gespeichert wird.

# Adaption der Farbtiefe

- Bei einer Verringerung der Anzahl der unterschiedlichen Helligkeitswerte gehen Details des Bildes verloren (z.B. bei fließenden Übergängen, Helligkeitsverläufen und feinen Texturen).
- Analyse des Histogramms eines Bildes  
→ Die Verteilung der Pixel ist in der Regel nicht gleichmäßig (viele Pixel liegen innerhalb weniger Intervalle)

# Lineare Adaption der Farbtiefe

- Intervalle gleicher Größe werden definiert. Da alle Werte innerhalb eines Intervalls die gleiche Helligkeit zugewiesen bekommen, wird der Kontrastumfang des Displays nicht ausgeschöpft und viele Details gehen verloren.

$$I_{lin}(i) = \lfloor \frac{N_C}{256} \cdot i \rfloor \in [0, N_C - 1].$$

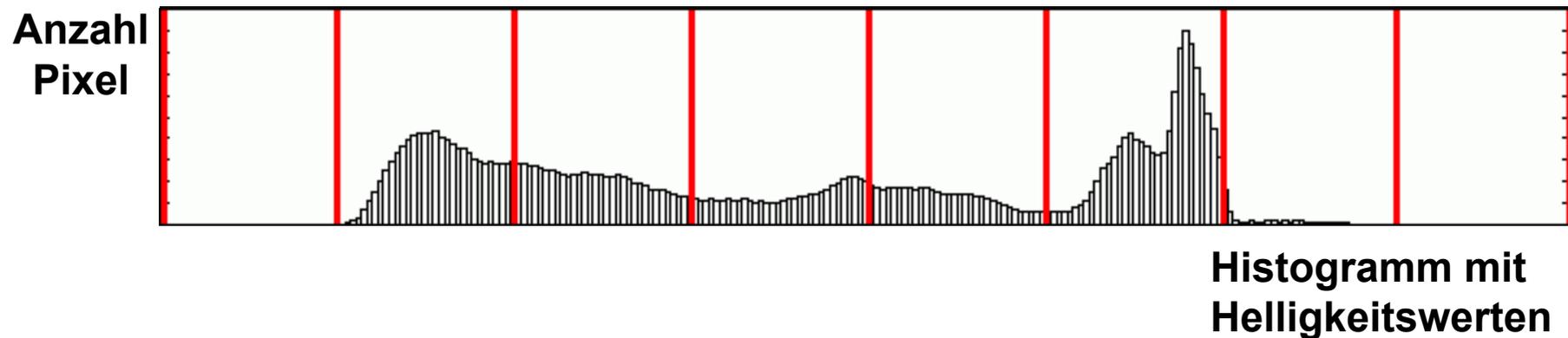
$N_C$ : Anzahl der Helligkeitswerte im adaptierten Bild

$i$ : Helligkeit im ursprünglichen Bild

$I_{lin}(i)$ : Helligkeit im adaptierten Bild

# Lineare Adaption der Farbtiefe

- Wegen der gleichgroßen Intervalle gehen viele Details verloren.

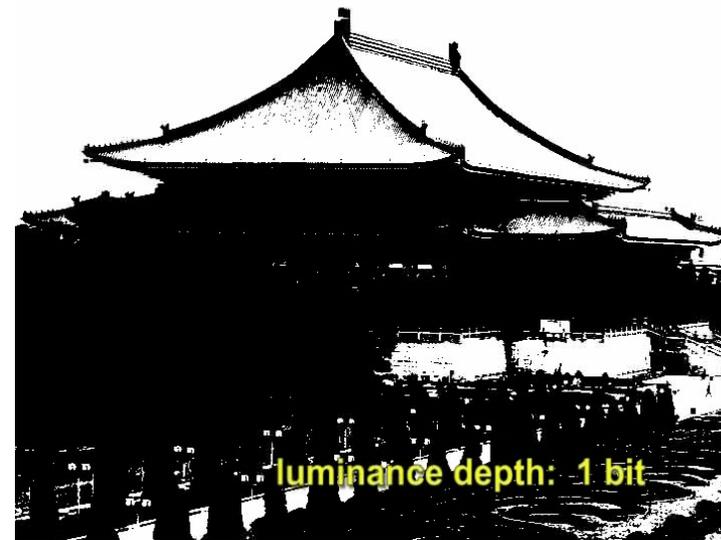


# Lineare Adaption der Farbtiefe

- Bei einer Verringerung der Farbtiefe gehen insbesondere feine Strukturen verloren.



Farbtiefe: 24-Bit



Graustufen

# Nicht-Lineare Adaption der Farbtiefe

- Eine variable Größe der Intervalle abhängig von der Verteilung der Helligkeitswerte liefert mehr Detailinformationen im adaptierten Bild, insbesondere für Bilder mit einem geringen Kontrast.

$$I_{var}(i) = \lfloor \frac{N_C}{S_X \cdot S_Y + 1} \cdot H_{kum}(i) \rfloor \in [0, N_C - 1].$$

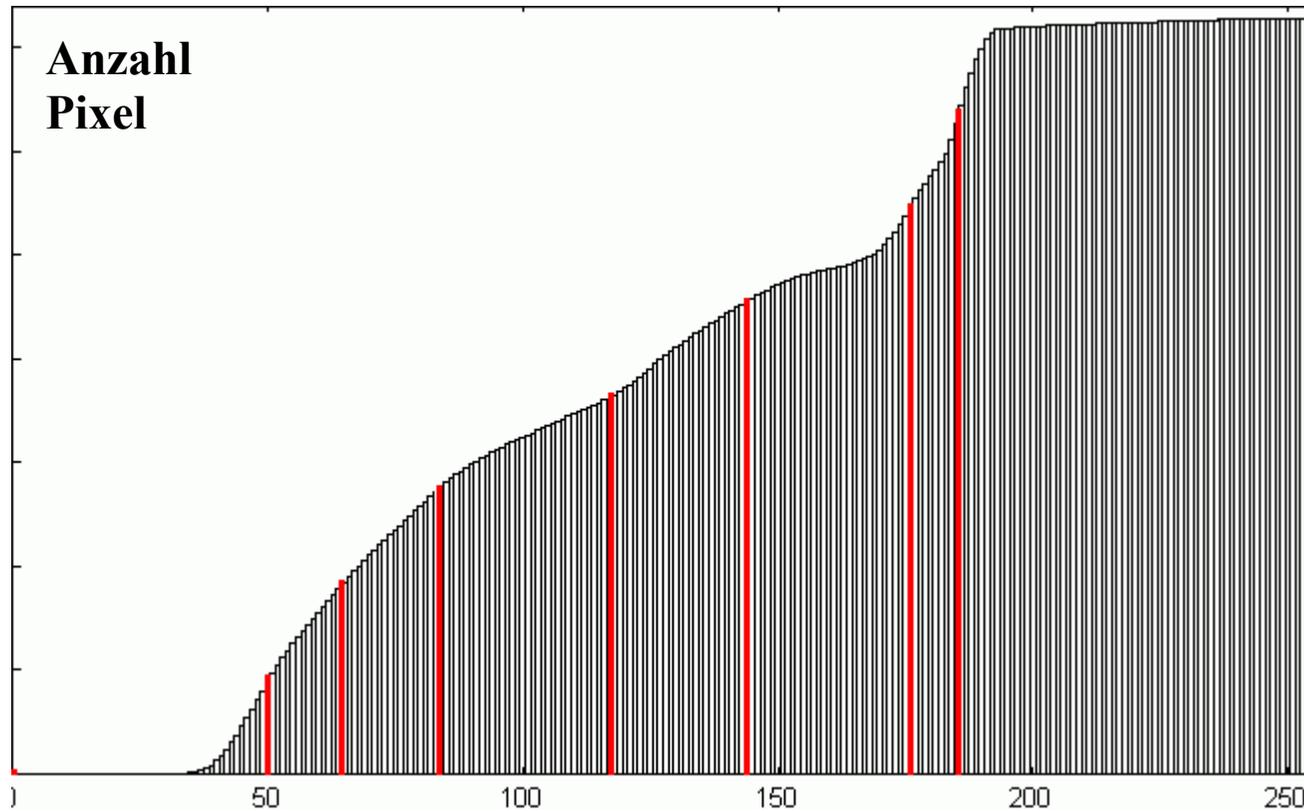
$H_{kum}(i)$ : kumuliertes Histogramm

$S_X, S_Y$ : Bildbreite, Bildhöhe

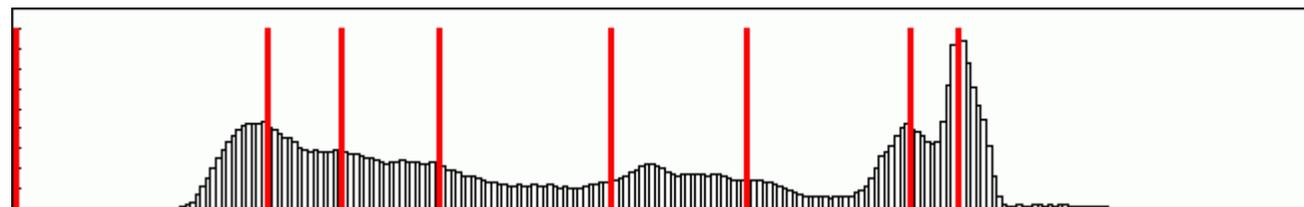
$i$ : Helligkeitswert

$I_{var}(i)$ : neuer Helligkeitswert

# Nicht-Lineare Adaption der Farbtiefe



Kumuliertes  
Histogramm



Ursprüngliches  
Histogramm

# Nicht-Lineare Adaption der Farbtiefe



8 Graustufenwerte  
(lineare Adaption)



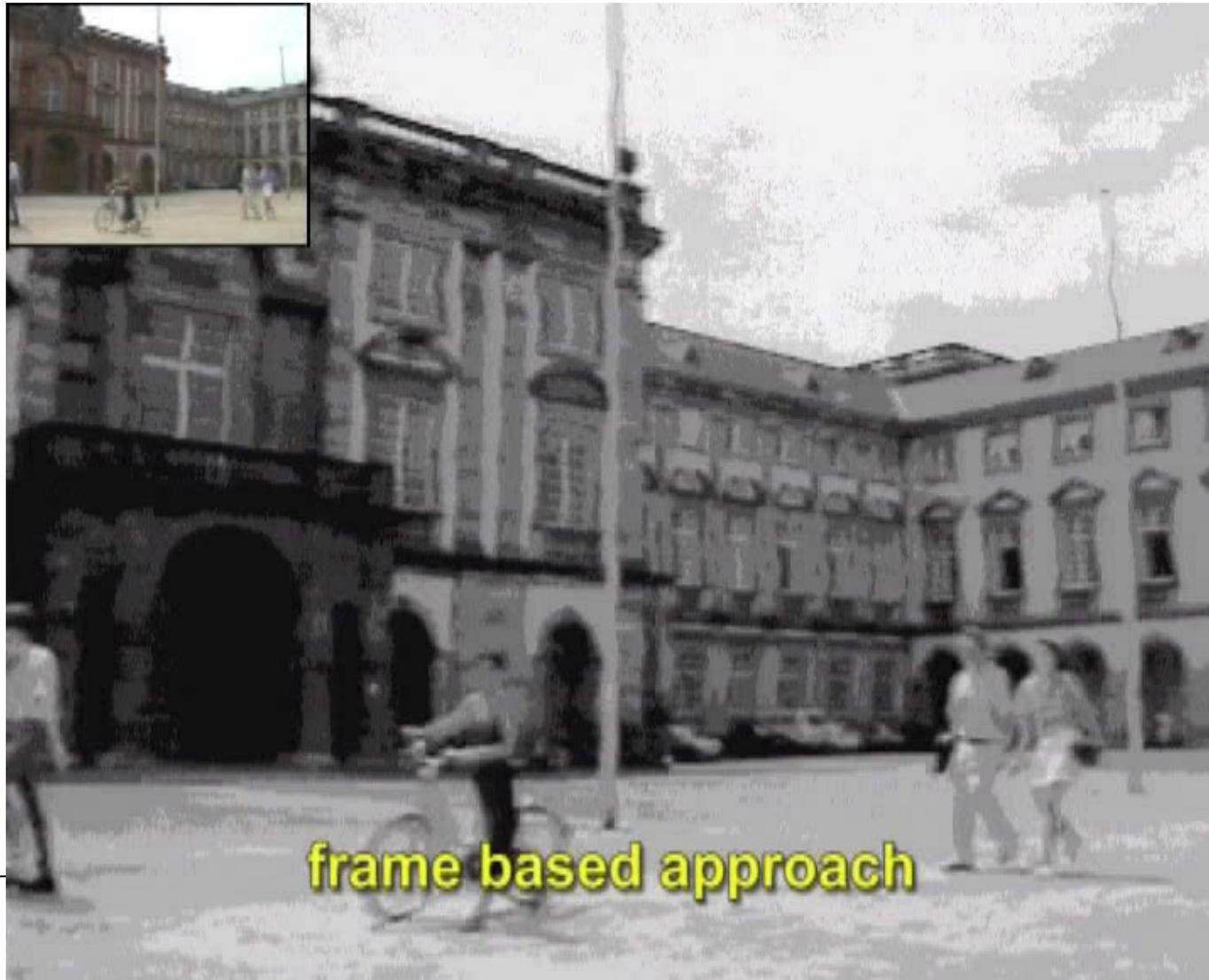
8 Graustufenwerte  
(nicht-lineare Adaption)

# Nicht-Lineare Adaption der Farbtiefe

- Mehr Details können im Bild erkannt werden.
- Bei sehr hellen oder sehr dunklen Bildern führt die nichtlineare Adaption zu einer deutlichen Veränderung der durchschnittlichen Helligkeit.
- Eine maximal zulässige Änderung der durchschnittlichen Helligkeit kann durch eine Kombination beider Verfahren garantiert werden:

$$L_w(i) = \lfloor \alpha \cdot L_{lin}(i) + (1 - \alpha) \cdot L_{var}(i) \rfloor \in [0, N_C - 1].$$

# Probleme bei der nicht-linearen Adaption von Videos



# Probleme bei der nicht-linearen Adaption von Videos

## **Probleme**

- Die bei der Umwandlung verwendeten Intervallgrenzen sollten sich nicht innerhalb einer Kameraeinstellung ändern.
- Die nicht-lineare Adaption führt sehr verrauschten Bildern bei sehr dunklen oder hellen Aufnahmen.

## **Lösung: erweiterte nicht-lineare Adaption**

- Aggregiere kumulierte Histogramme für Kameraeinstellungen.
- Sehr dunkle Einzelbilder sollen unberücksichtigt bleiben.

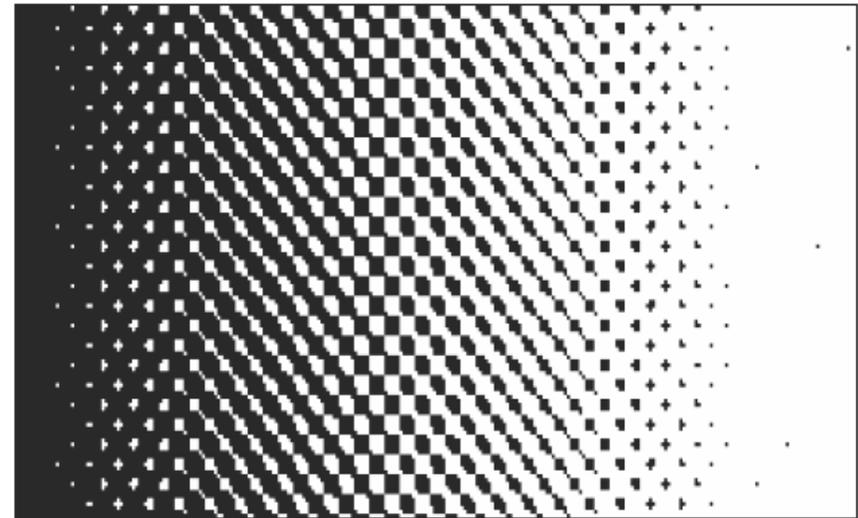
# Erweiterte nicht-lineare Adaption



# Adaption in der Drucktechnik

## Adaption in Binärbilder (Schwarz-Weiß-Druck)

- Bekanntes Problem aus der Drucktechnik:  
Mischung von Farben  
(subtraktives Farbmodell)
- Halbton-Technik (*halftoning*)
- Eine Illusion von kontinuierlichen Farbverläufen wird aus wenigen Farben durch eine spezielle Verteilung der Pixel erzeugt.

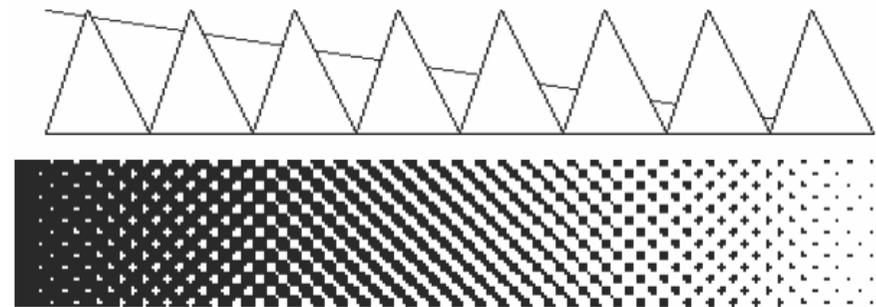


Quelle: Gabriel Marcu, Apple.

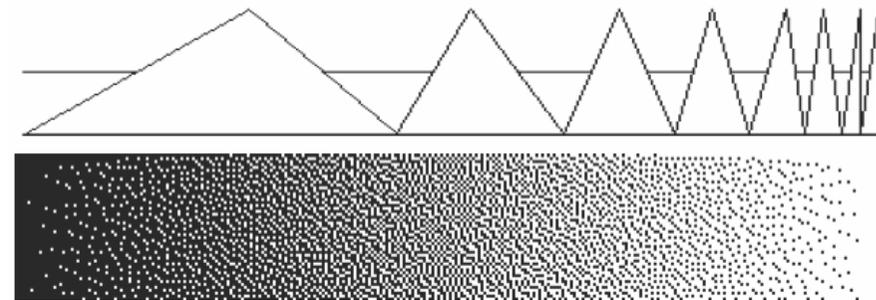
# Adaption in der Drucktechnik

## *Halbton-Verfahren*

- Amplitudenmodulation
  - Pixelgröße ist variabel (die Pixelgrößen werden moduliert)
  - Pixelposition liegen auf einem festen Gitter
- Frequenzmodulation
  - Feste Pixelgröße
  - Die Positionen der Pixel sind nicht an ein spezieller Raster gebunden (die Entfernung der Pixel untereinander wird moduliert)



Halftone: amplitude modulation

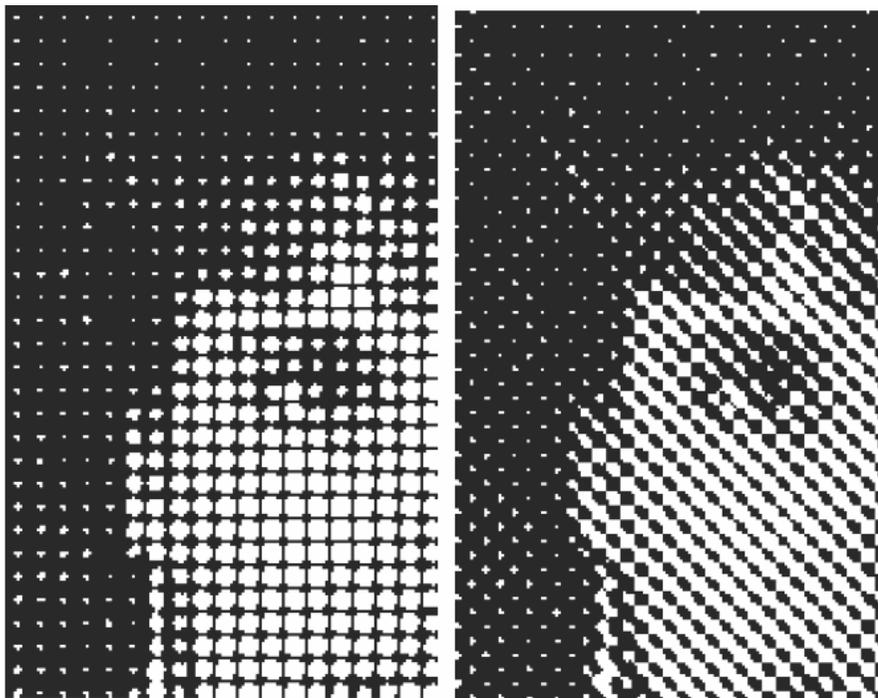


Halftone: frequency modulation

Quelle: Gabriel Marcu, Apple.

# Adaption in der Drucktechnik

## Amplitudenmodulation



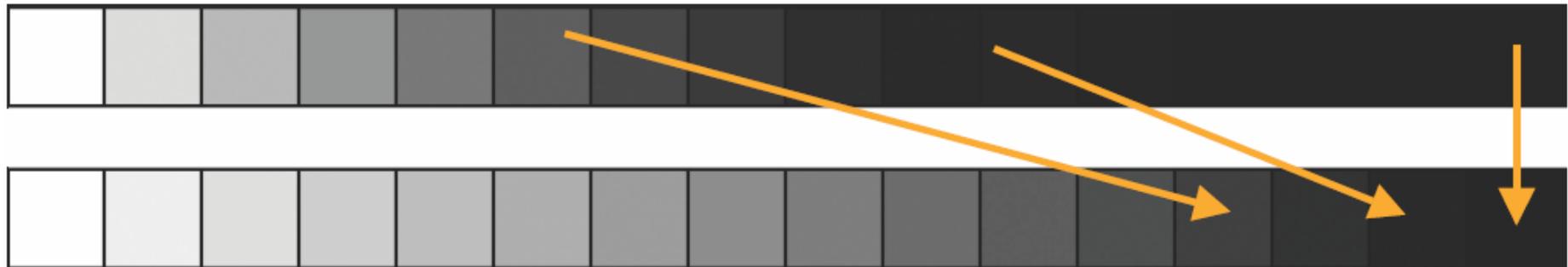
## Frequenzmodulation



Quelle: Gabriel Marcu, Apple.

# Adaption in der Drucktechnik

- Beim Drucken erscheinen dunkle Töne schnell gesättigt und können schlecht unterschieden werden.
- Vor dem Ausdrucken werden Graustufenwerte nicht-linear abgebildet:



Quelle: Gabriel Marcu, Apple.

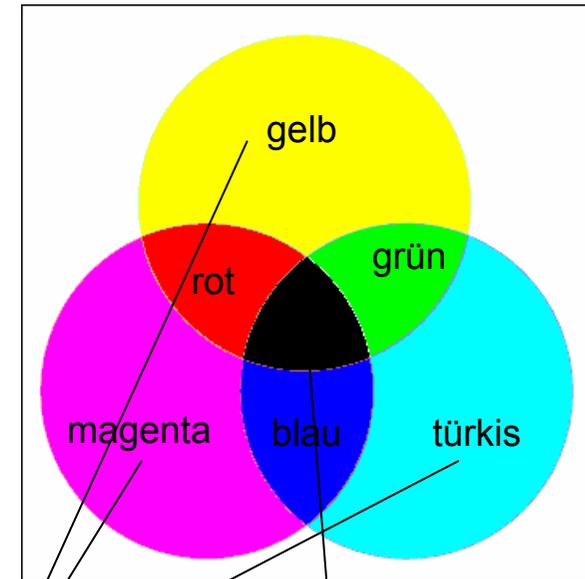
# Adaption in der Drucktechnik

## *Farbdruck*

- Farbig bedruckte Bereiche wirken wie ein Filter, der bestimmte Wellenlängen herausfiltert.

## *Ablauf*

1. Identifiziere 3 (oder 4) Farbkanäle in einem Bild.
2. Berechne ein Halbtonbild für jeden Farbkanal.
3. Drucke die Halbtonbilder übereinander.

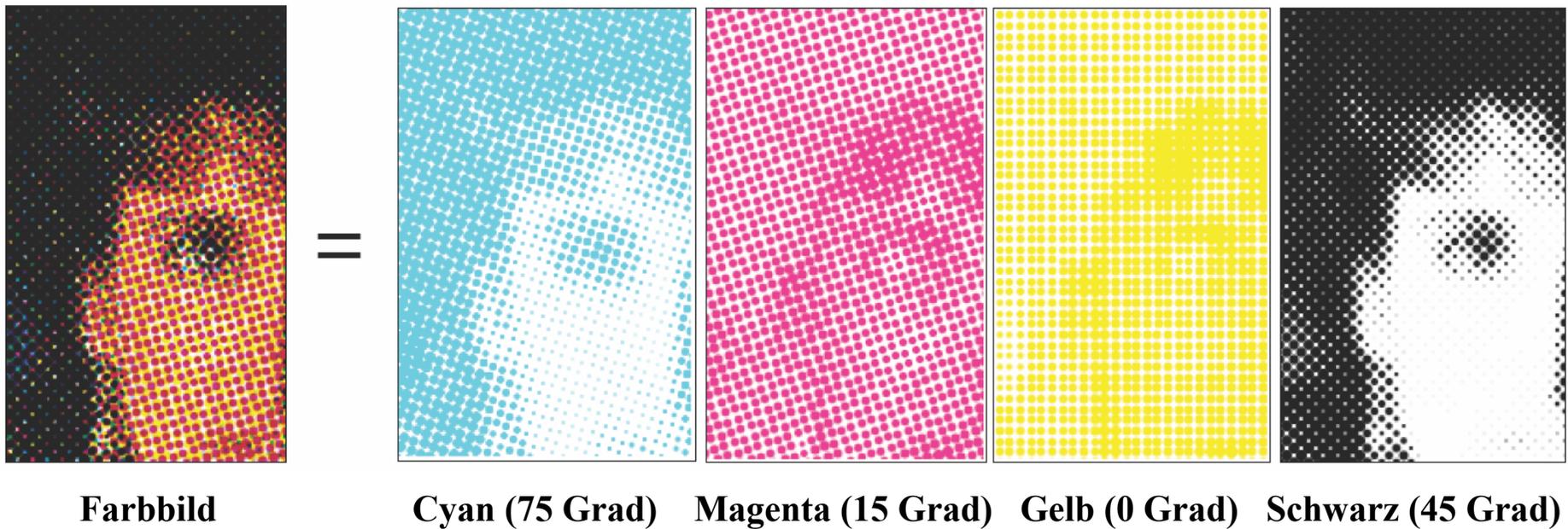


3 Grundfarben

vierte Farbe (schwarz)  
(optional)

# Adaption in der Drucktechnik

## *Halbton-Bilder der einzelnen Farbkanäle*



Quelle: Gabriel Marcu, Apple.

# Adaption in der Drucktechnik

## *Vorteile bei der Verwendung einer vierten Farbe*

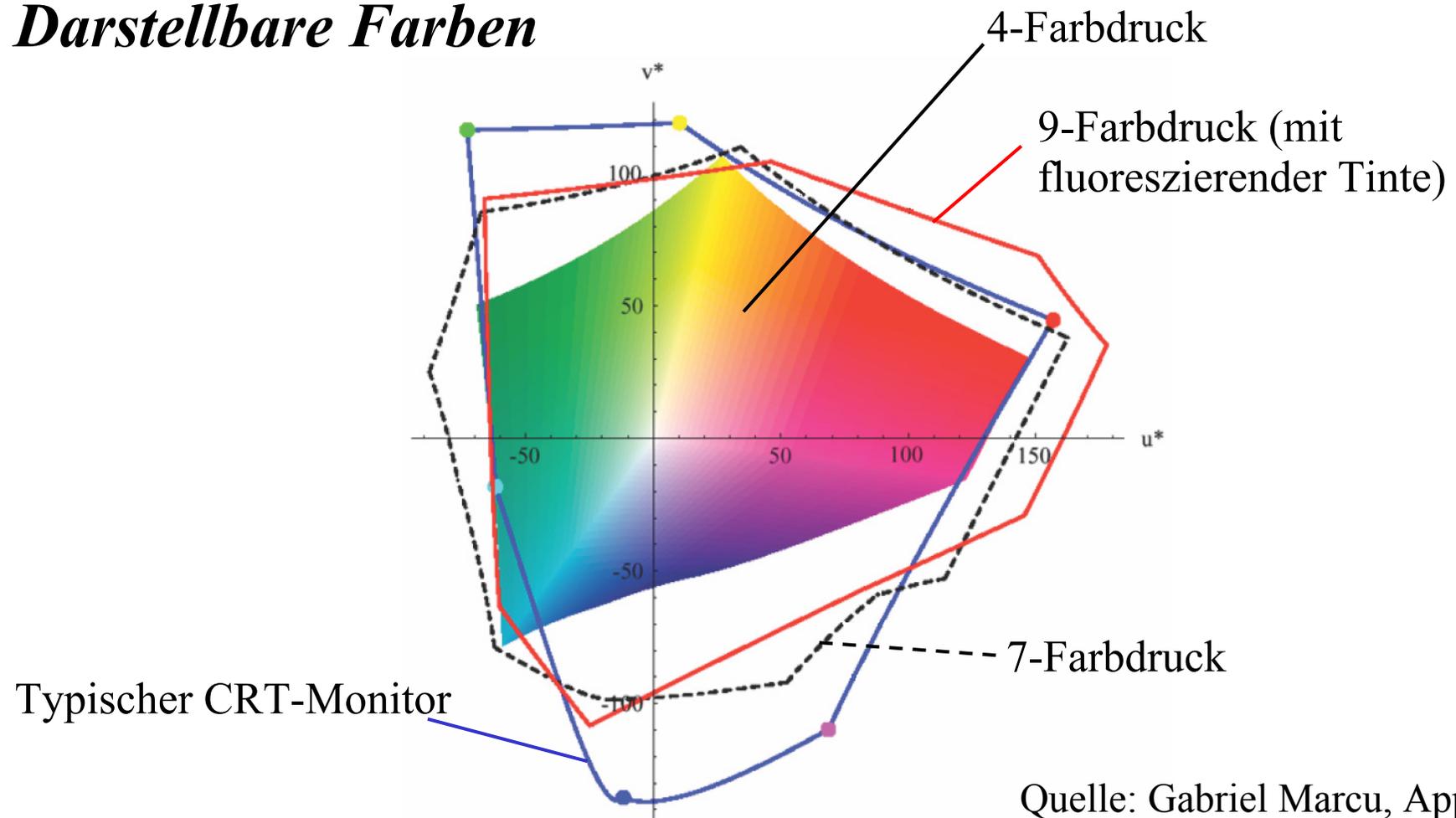
- Erhöhung des Kontrastes (dunkle Farben können besser gedruckt werden)
- Einsparung von Tinte
- Papier wird nicht nass und wellt sich (Papier kann nur begrenzt Tinte aufnehmen)
- Graustufenbilder haben keinen Farbstich

## *Nachteile*

- Wann soll ein Ton schwarz gedruckt werden?  
(*under color removal*)
- Großer Einfluss der Papiersorte

# Adaption in der Drucktechnik

## *Darstellbare Farben*

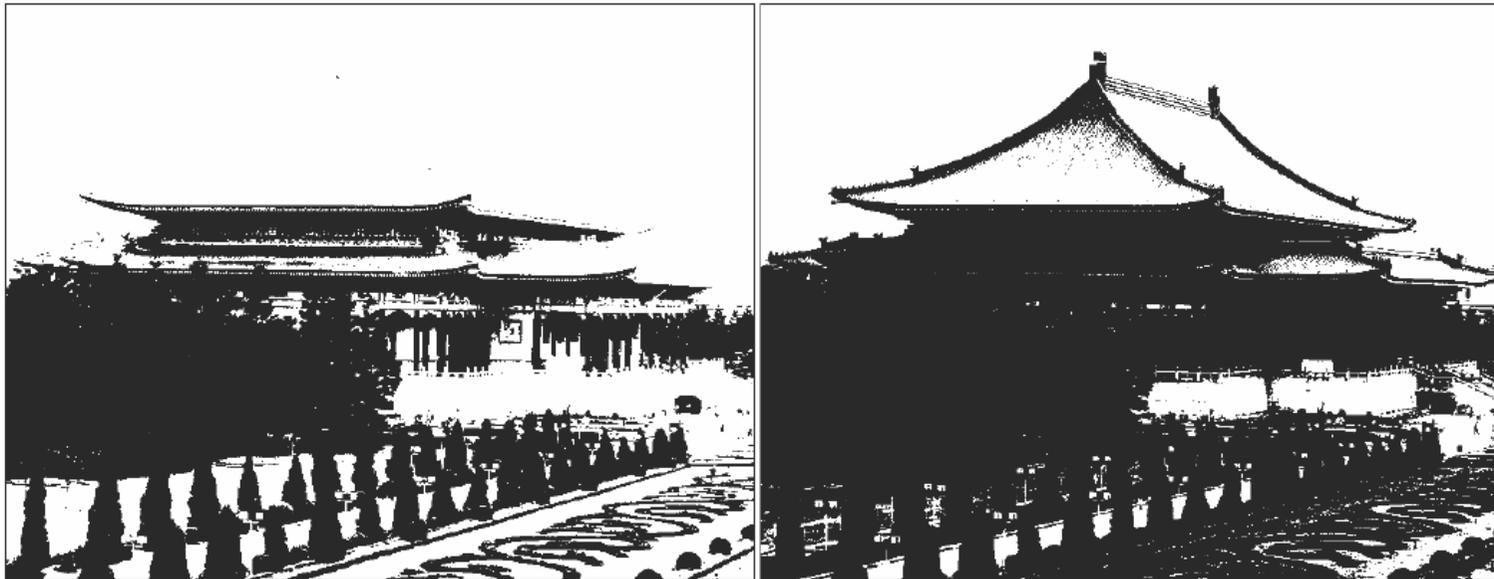


Quelle: Gabriel Marcu, Apple.

# Verfahren zur Anpassung der Farbtiefe

## *Verfahren zur Anpassung der Farbtiefe*

- Um die Druck- oder Displayfarben zu erhalten, werden die Farben eines Bildes auf die neue Farbpalette abgebildet.
- Naiver Ansatz: Zuordnung der Pixel durch den Vergleich mit einem Schwellwert (links:  $T=90$ , rechts:  $T=130$ )



# Verfahren zur Anpassung der Farbtiefe

## Floyd-Steinberg-Algorithmus zur Anpassung der Farbtiefe

- Das Verfahren wurde 1975 von Floyd und Steinberg veröffentlicht.
- Das Bild wird pixelweise von links oben nach rechts unten umgewandelt.
- Das aktuelle Pixel wird auf die ähnlichste verfügbare Farbe oder Helligkeit abgebildet.
- Der durch den neuen Wert des Pixels entstandene Fehler wird auf benachbarte Pixel verteilt (*error diffusion*).
- $\frac{7}{16}$  des Fehlers wird auf das rechte benachbarte Pixel, jeweils  $\frac{3}{16}$ ,  $\frac{5}{16}$  und  $\frac{1}{16}$  auf die angrenzenden Pixel in der folgenden Zeile übertragen.

# Verfahren zur Anpassung der Farbtiefe

## Floyd-Steinberg-Algorithmus zur Anpassung der Farbtiefe

Iteration 7 ↑ aktuelles Pixel

255	255	0	255	255
0	100	230	140	50
80	100	50	90	70

**Fehler:**  $100 - 0 = 100$   
 $\rightarrow 7/16 * 100 = 44$  (rechts)  
 $\rightarrow 3/16 * 100 = 19$  (links-unten)  
 $\rightarrow 5/16 * 100 = 31$  (unten)  
 $\rightarrow 100 - 44 - 19 - 31 = 6$  (rechts-unten)

255	255	0	255	255
0	0	230	140	50
80	100	50	90	70

*(Red arrows and numbers show adjustments: +44 to right, +19 to bottom-left, +31 to bottom, +6 to bottom-right)*

Iteration 8

255	255	0	255	255
0	0	274	140	50
99	131	56	90	70

**Fehler:**  $274 - 255 = 19$   
 $\rightarrow 7/16 * 19 = 8$  (rechts)  
 $\rightarrow 3/16 * 19 = 4$  (links-unten)  
 $\rightarrow 5/16 * 19 = 6$  (unten)  
 $\rightarrow 19 - 8 - 4 - 6 = 1$  (rechts-unten)

255	255	0	255	255
0	0	255	140	50
99	131	56	90	70

*(Red arrows and numbers show adjustments: +8 to right, +4 to bottom-left, +6 to bottom, +1 to bottom-right)*

# Verfahren zur Anpassung der Farbtiefe

## Floyd-Steinberg-Algorithmus zur Anpassung der Farbtiefe

Iteration 9

255	255	0	255	255
0	0	255	148	50
99	135	62	91	70

**Fehler:**  $148 - 255 = -107$   
 $\rightarrow 7/16 * (-107) = -47$  (rechts)  
 $\rightarrow 3/16 * (-107) = -20$  (links-unten)  
 $\rightarrow 5/16 * (-107) = -33$  (unten)  
 $\rightarrow (-107) - (-44) - (-19) - (-31) = -7$  (rechts-unten)

255	255	0	255	255
0	0	255	255	50
99	135	62	91	70

*(Red arrows and numbers indicate error distribution: -47 to right, -20 to bottom-left, -33 to bottom, -7 to bottom-right)*

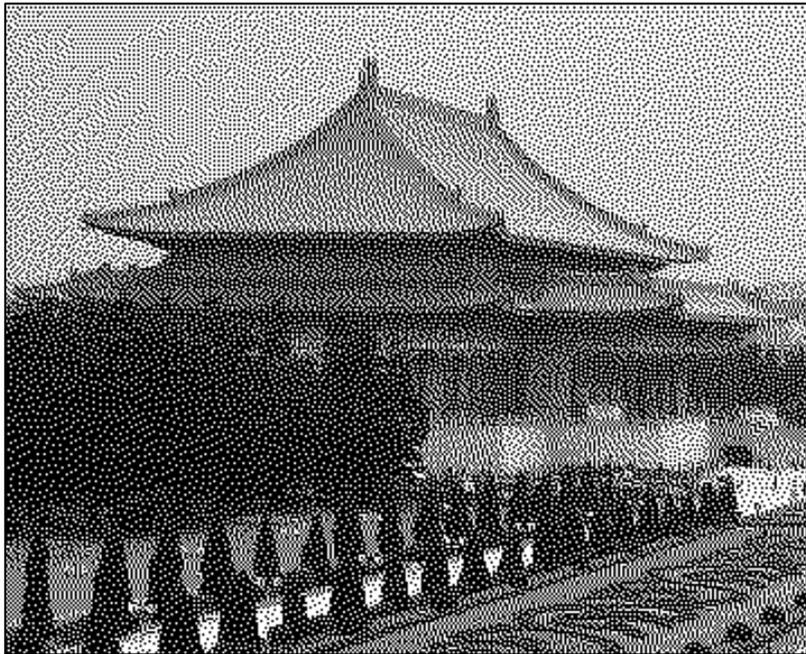
Iteration 10

255	255	0	255	255
0	0	255	255	3
99	135	42	58	63

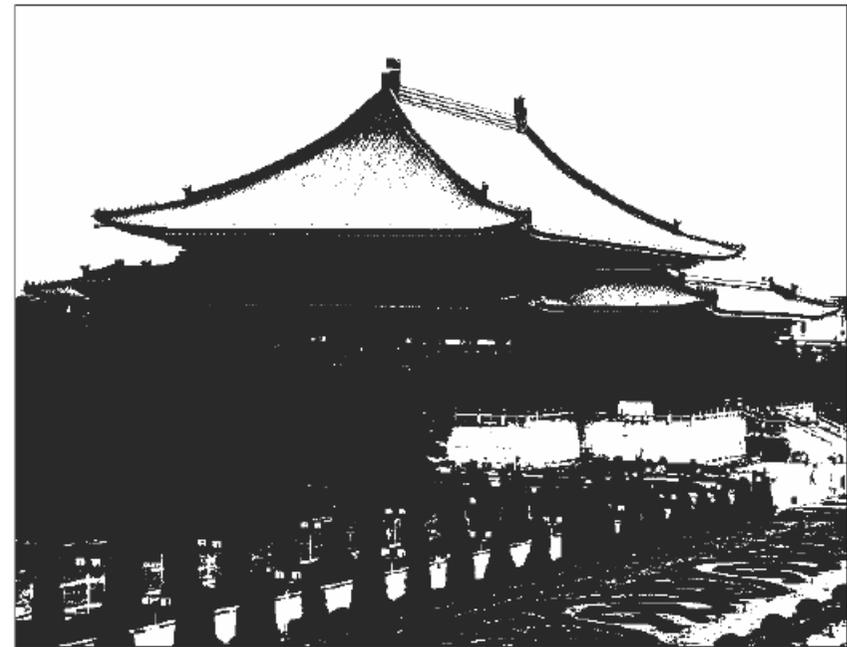
**Iterativ werden die Fehler auf angrenzende Pixel verteilt.**

# Verfahren zur Anpassung der Farbtiefe

## Beispiel zum Floyd-Steinberg-Algorithmus



Floyd-Steinberg-Algorithmus



Vergleich mit festem Schwellwert

# Verfahren zur Anpassung der Farbtiefe

## Probleme des Floyd-Steinberg-Algorithmus

- Für Videosequenzen ist der Algorithmus *nicht anwendbar*, da sich durch die Verteilung des Fehlers viele Pixel in aufeinander folgenden Bildern ändern.
- Das führt zu sehr starkem Rauschen, so dass die Inhalte einer Videosequenz bei der Adaption mit dem Floyd-Steinberg-Algorithmus in sehr schlechter Qualität dargestellt werden.

# Verfahren zur Anpassung der Farbtiefe

## Probleme des Floyd-Steinberg-Algorithmus bei Videos



# Verfahren zur Anpassung der Farbtiefe

## Textur-basierte Adaption der Farbtiefe

- Das Bild wird mit binären Texturen überlagert:

$$I_t(x, y) = \begin{cases} 0 & [(x + S_X \cdot y) \text{ MOD } (T_B + T_W)] < T_B, \\ 1 & \text{else.} \end{cases}$$

$T_B, T_W$  definieren das Verhältnis von schwarzen und weißen Pixeln



$T_B \gg T_W$

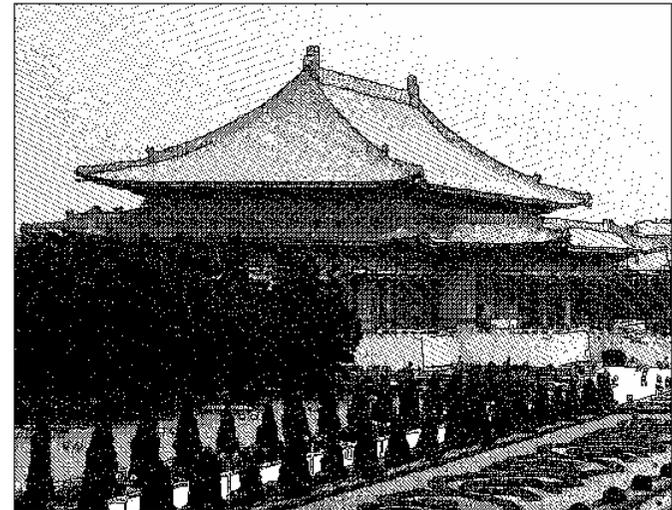
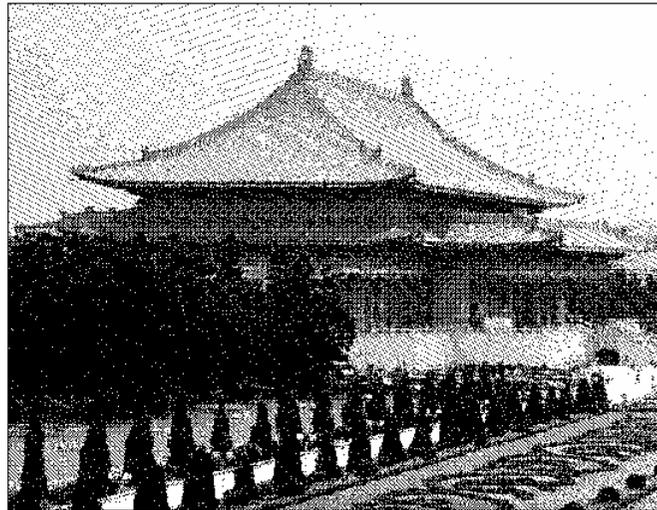
$T_B \ll T_W$

- Texturen mit ähnlichen Helligkeitswerten sollen einfach unterschieden werden können.

# Verfahren zur Anpassung der Farbtiefe

## Textur-basierte Adaption der Farbtiefe

1. Erzeuge ein Graustufenbild mit  $N_C = 16$  unterschiedlichen Helligkeitswerten basierend auf kumulierten Histogrammen.
2. Ersetze jedes Pixel mit einem Pixel der entsprechenden Textur.

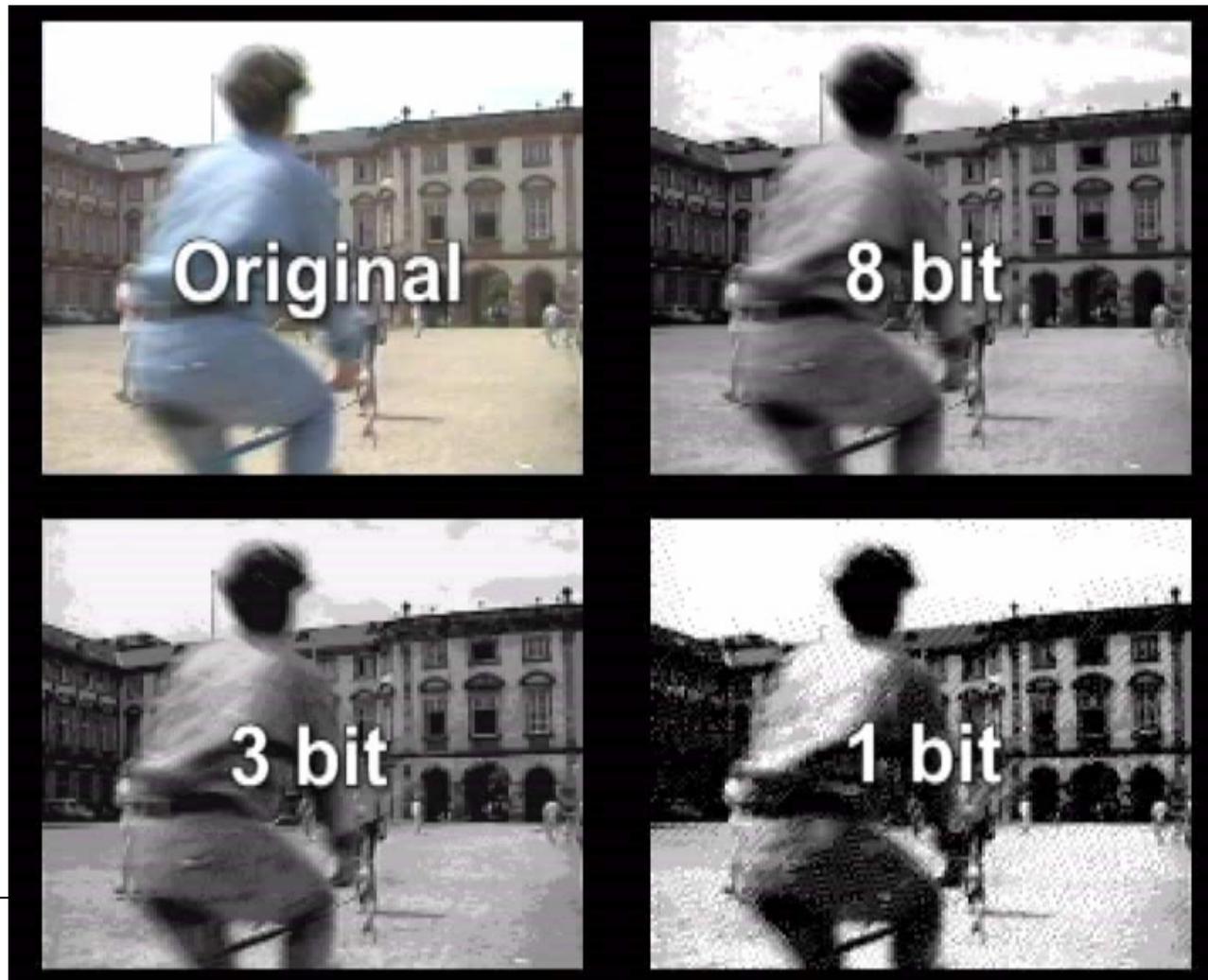


→ starke Kanten gehen verloren

3. Hebe starke Kanten hervor.

# Verfahren zur Anpassung der Farbtiefe

Ergebnis der textur-basierten Adaption von Videos



# Videodemo: Adaption der Farbtiefe



<http://www.informatik.uni-mannheim.de/pi4/projects/moca/Project-ColorAdaptation.html>

# High-Dynamic-Range-Bilder (HDR)



Quelle: wikipedia.org

# High-Dynamic-Range-Bilder (HDR)

- High Dynamic Range bezeichnet ein hohes Kontrastverhältnis in einem Bild
- Kontrastverhältnis
  - bei digitalem Bild: 1.000:1
  - bei analogen Fotos: 10.000:1 (wesentlicher Vorteil analoger Fotos)
  - bei HDR-Bildern: 200.000:1
- Der Umfang der Helligkeitsintensitäten ist so groß, dass nicht alle Werte auf einem Monitor dargestellt oder gedruckt werden können.
- *Tone-Mapping*: Reduziert den HDR-Helligkeitsbereich auf den Wertebereich eines Monitors.

# High-Dynamic-Range-Bilder (HDR)

- Maler des Mittelalters verwendeten spezielle Technik, durch die helle und dunkle Bereiche sehr gut in einem Gemälde erkannt werden können:
  - Verwendung von gesättigten Farben, um den dynamischen Bereich des Bildes zu erhöhen.
  - Verstärkung von Konturen, indem schwarze und weiße Linien an die Ränder von Objekten gemalt werden. Der Kontrast erscheint deutlich größer.



El Greco's *La Agoria en el Jardin* (1590)

Quelle: [cybergrain.com](http://cybergrain.com)

# High-Dynamic-Range-Bilder (HDR)

- Beeinflussung der menschlichen Wahrnehmung
- Warum scheint Sonne hervorzustehen?
  - Sonne hat identische Helligkeit wie Hintergrund (geringer Kontrast)
  - Widersprüchliche Interpretation der Sonne im Gehirn:
    - Gehirnregion für einfache Wahrnehmung (Bewegung und Position):  
Sonne ist unsichtbar
    - höhere Gehirnregion (Farbe):  
normale Sonne



Claude Monet, *Impressions at Sunrise*

Quelle: [webexhibits.org](http://webexhibits.org)

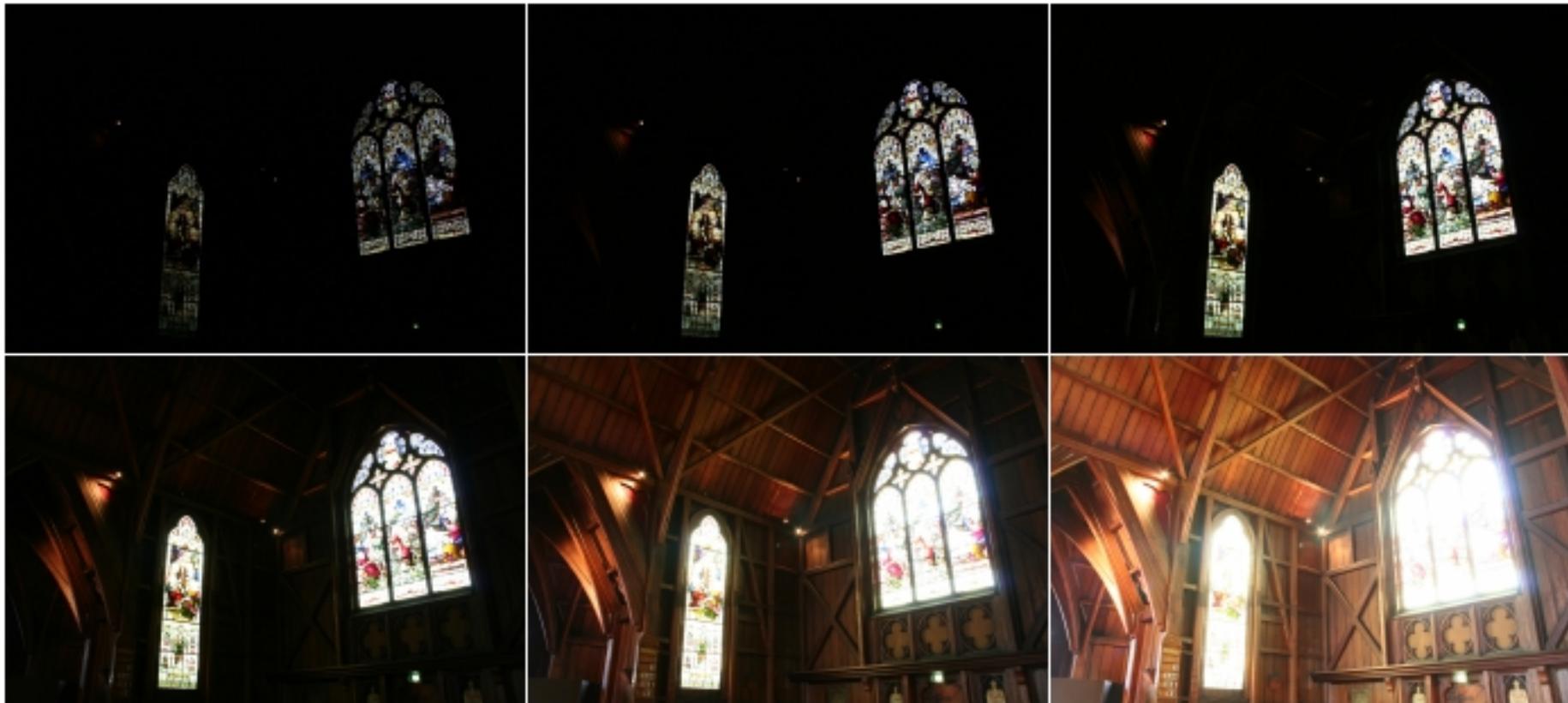
# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Es werden mehrere Aufnahmen der selben Szene mit unterschiedlicher Belichtung gemacht (normal belichtet, sowie unter- und überbelichtet).
- Die einzelnen Bilder werden miteinander zu einem gemeinsamen HDR-Bild kombiniert.
- In den unterbelichteten Bildern können sehr helle Bildbereiche gut unterschieden werden.
- In den überbelichteten Bildern können sehr dunkle Bildbereiche gut unterschieden werden.

# High-Dynamic-Range-Bilder (HDR)

Beispiel unter- und überbelichteter Aufnahmen



Quelle: wikipedia.org

# High-Dynamic-Range-Bilder (HDR)

Ergebnis (Kombination der Aufnahmen)



Quelle: wikipedia.org

# High-Dynamic-Range-Bilder (HDR)

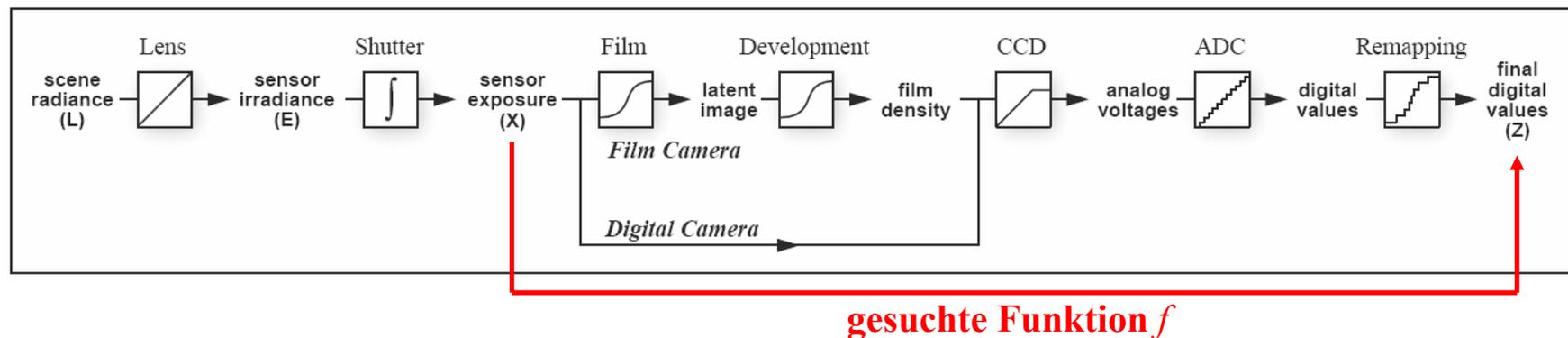
## Erzeugung von HDR-Bildern

- Berechne eine *Übergangsfunktion (response function)*, welche die Helligkeit der Szene in Pixelwerte umwandelt.
- Füge die Bilder mit den unterschiedlichen Belichtungszeiten zu einem HDR-Bild zusammen.
- Die Pixelwerte im HDR-Bild sind proportional zur echten Helligkeit der Szene.

# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Die tatsächliche Helligkeit der Szene wird durch eine nicht-lineare Funktion auf Pixelwerte abgebildet:



- Faktoren: Belichtung eines analogen Films, Entwicklung des Films, Digitalisierung mit einem Scanner

Quelle: Debevec, Malik (University of California at Berkeley)

# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Ermittlung der *charakteristischen Kurve* eines Films (*Übergangsfunktion, response function*).
- Definition *Belichtung*  $X$ :  $X = E \cdot \Delta t$   
 $E$ : Beleuchtungsichte,  $\Delta t$ : Belichtungszeit
- Durch die Verarbeitung erhalten wir einen digitalen Wert  $Z$ , der durch eine nichtlineare Funktion  $f$  aus  $X$  entstanden ist.
- Falls  $f$  bekannt ist, kann die tatsächliche Belichtung  $X$  berechnet werden:  $X = f^{-1}(Z)$
- Da die Belichtungszeit  $\Delta t$  bekannt ist, kann dann die Beleuchtungsichte  $E$  ermittelt werden:  $E = X / \Delta t$ .
- Die Beleuchtungsichte  $E$  ist proportional zum Licht  $L$  der Szene.

# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Gegeben: mehrere Bilder mit unterschiedlicher Belichtungszeit  $\Delta t$
- Die Beleuchtungsichte  $E$  bleibt für jedes Pixel konstant.
- Pixelwerte der Bilder:  $Z_{ij} = f(E_i \Delta t_j)$   
 $i$ : Pixelposition (x/y-Koordinate),  $j$ : Index des Bildes
- Es existiert Umkehrfunktion:  $f^{-1}(Z_{ij}) = E_i \Delta t_j$
- Berechne natürlichen Logarithmus:

$$\ln f^{-1}(Z_{ij}) = \ln E_i + \ln \Delta t_j$$

# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Setze:  $g = \ln f^{-1}$

$$\rightarrow g(Z_{ij}) = \ln E_i + \ln \Delta t_j$$

bekannt: Pixelwerte  $Z_{ij}$ , Belichtungszeit  $\Delta t$

unbekannt: Beleuchtungsichte  $E_i$ , Funktion  $g$

- $g$  bildet eine endliche Zahl von Punkten ab  
( $Z$  beschreibt feste Pixelwerte)

# High-Dynamic-Range-Bilder (HDR)

## Erzeugung von HDR-Bildern

- Bestimme  $E_i$  und  $g$  durch Minimierung der folgenden Funktion:

$$\mathcal{O} = \sum_{i=1}^N \sum_{j=1}^P [g(Z_{ij}) - \ln E_i - \ln \Delta t_j]^2 + \lambda \sum_{z=Z_{min}+1}^{Z_{max}-1} g''(z)^2$$

**Fehler wird minimal**

**Glätte die Funktion**

$N$ : Anzahl der Pixel im Bild

$P$ : Anzahl der Bilder

ermittle  $(Z_{max} - Z_{min} + 1)$  Werte für  $g(Z)$

ermittle  $N$  Werte für  $\ln E_i$

# High-Dynamic-Range-Bilder (HDR)

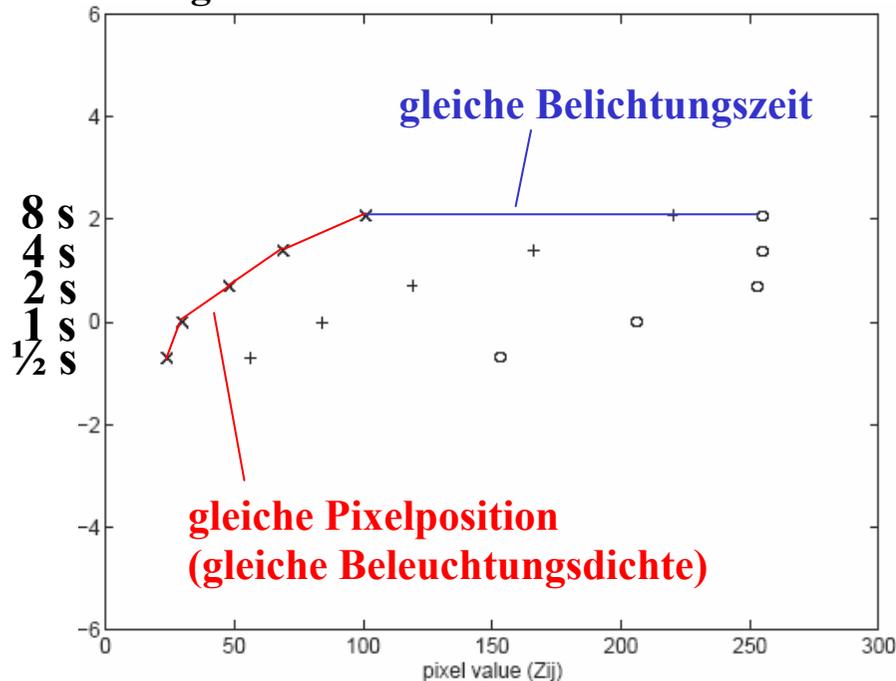
## Erzeugung von HDR-Bildern

- Verfahren zur Bestimmung von  $E_i$  und  $g$ :
  - Stelle überbestimmtes Gleichungssystem auf.
  - Löse dieses mit einem linearen Kleinste-Quadrate-Algorithmus, bei dem die Summe der quadratischen Abweichungen der Funktion  $g$  von den beobachteten Pixelwerten  $Z$  minimiert wird.

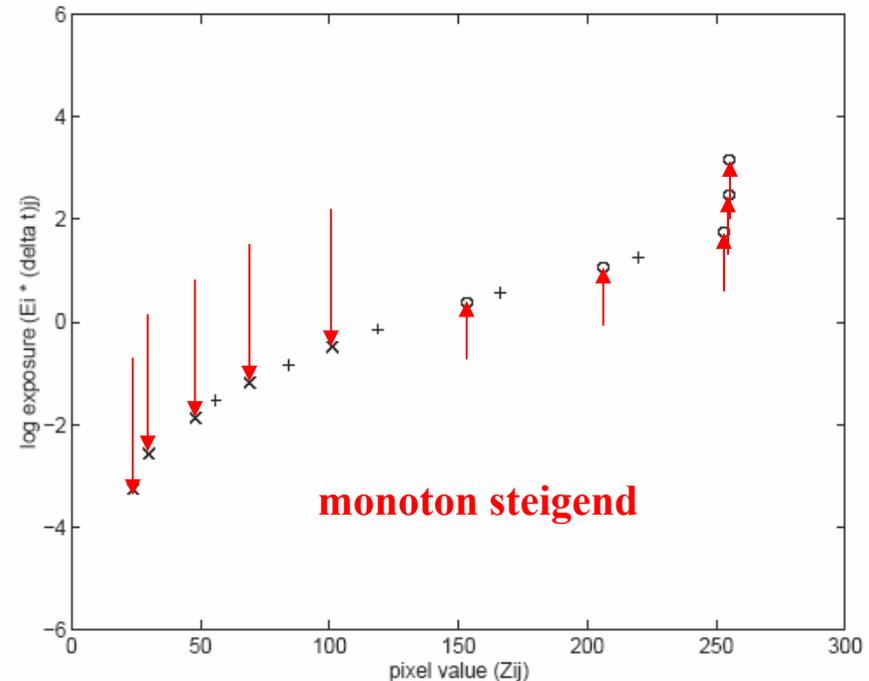
# High-Dynamic-Range-Bilder (HDR)

## Beispiel

Belichtungszeit  $\Delta t$



Zuordnung der Pixelwerte zur Belichtungszeit  
(Beleuchtungsichte  $E_i=0$ )



Berücksichtigung der Beleuchtungsichte  $E_i$

Quelle: Debevec, Malik (University of California at Berkeley)

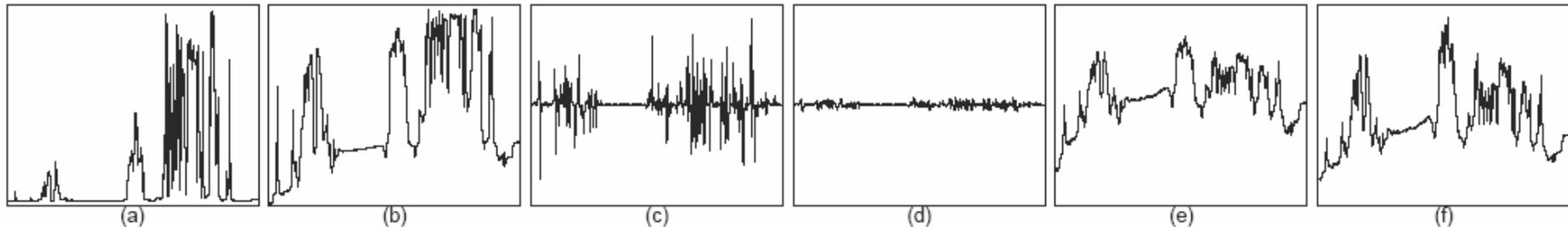
# High-Dynamic-Range-Bilder (HDR)

## Darstellung von HDR-Bildern (Tone Mapping)

- Tone-Mapping reduziert den Dynamikbereich von Bildern
- Reduziere den Kontrastumfang eines HDR-Bildes, um dieses auf Geräten mit beschränktem Kontrastumfang wiederzugeben.
- Zu große oder zu kleine Werte könnten auf den nächsten zulässigen Wert abgebildet werden. Dadurch kann jedoch zwischen den ursprünglich sehr unterschiedlichen Werten nicht mehr unterschieden werden.
- Durch Tone-Mapping werden Details besser wiedergegeben.

# High-Dynamic-Range-Bilder (HDR)

## Darstellung von HDR-Bildern (Tone Mapping)



a) Bildzeile mit einer Dynamik von 2415:1

b)  $H(x) = \log(\text{Bildzeile})$

c) Ableitung  $H'(x)$

d) Verringerte Ableitung  $G(x)$

e) Rekonstruierte Bildzeile  $I(x)$

f)  $\exp(I(x))$  mit einer Dynamik von 7,5:1

Quelle: Fattal, Lischinski, Werman (cs.huji.ac.il)

# High-Dynamic-Range-Bilder (HDR)

## Darstellung von HDR-Bildern (Tone Mapping)

- Verringere den Wertebereich der Pixel im HDR-Bild, indem jedes Pixel durch einen individuellen Normierungsfaktor  $\Phi(x, y)$  geteilt wird.
- Der Faktor wird so bestimmt, dass starke und schwache Kanten gut sichtbar sind.
- Berechne Kanten (Gradienten) für unterschiedlich skalierte Bilder.
- Übertrage die Gradienten des kleinsten Bildes auf das nächst größere Bild und fasse diese zusammen.
- Wiederhole bis die Kanten in das Bild mit voller Auflösung übertragen wurden.

# High-Dynamic-Range-Bilder (HDR)

## Darstellung von HDR-Bildern (Tone Mapping)



**Aggregiertes Kantenbild**



**HDR-Bild**

# Beispiele für HDR-Bilder



Quelle:  
chip.de

Philip Mildner/  
Christian Takacs

# Fragen ?