

## Bildverbesserung

- Grauwerttransformation (Punktoperation)
- Histogrammausgleich
- Nachbarschaftsoperatoren
- Bildglättung
- Bildverschärfung
- Korrektur von uneinheitlichem Hintergrund

“The principal objective of *image enhancement* is to process an image so that the result is more suitable than the original image for a *specific application*.”

## Allgemeine Ziele:

- Verbesserung des subjektiv-visuellen Eindrucks
- Vereinfachung nachfolgender Verarbeitungsschritte

## Spezielle Ziele:

- Kontrastverstärkung bzw. -verminderung
- Glättung (Reduzierung des Rauschanteils)
- Hervorhebung von Konturen (Sharpening)
- Hintergrundkompensation (Korrektur inhomogenen Hintergrunds)

## Realisierung:

- im Ortsraum: Operation direkt am Bild
- im Frequenzraum: Modifikation der Fourier-Transformierten und Rücktransformation in den Ortsraum (Kapitel “Bildverarbeitung im Frequenzraum”)

# Grauwerttransformation

Ein homogener Punktoperator

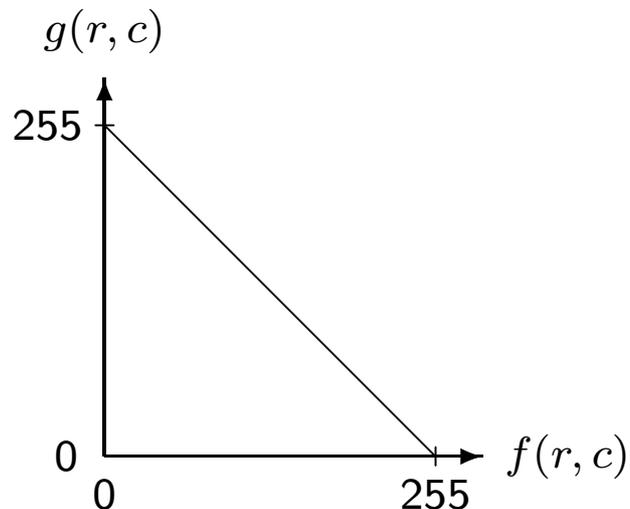
$$g(r, c) = \Omega(f(r, c))$$

definiert eine Grauwerttransformation.  $\Omega$  hat Intervall  $[0, 255]$  als Definitionsbereich und Wertebereich und ist häufig monoton.

Geometrisch wird  $\Omega$  auch eine *Transformationskennlinie* genannt. Für die Praxis unerheblich, ob die Kennlinie stetig ist.

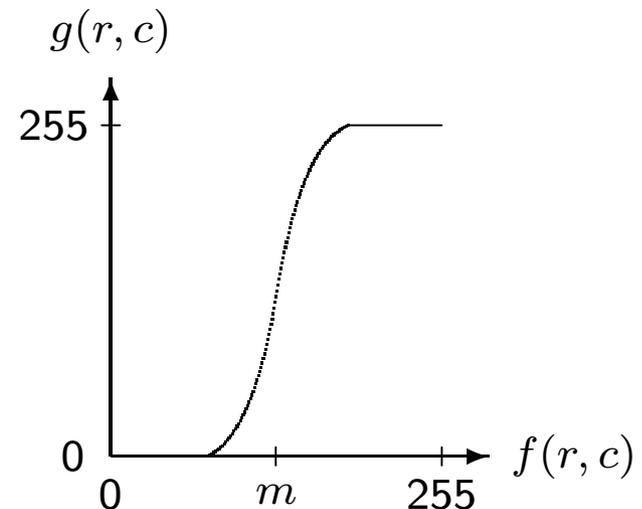
## Beispiel:

Negativ-Bild  $g(r, c) = 255 - f(r, c)$



## Beispiel:

Kontraststreckung (um Grauwert  $m$ )



# Grauwerttransformation: Log-Transformation

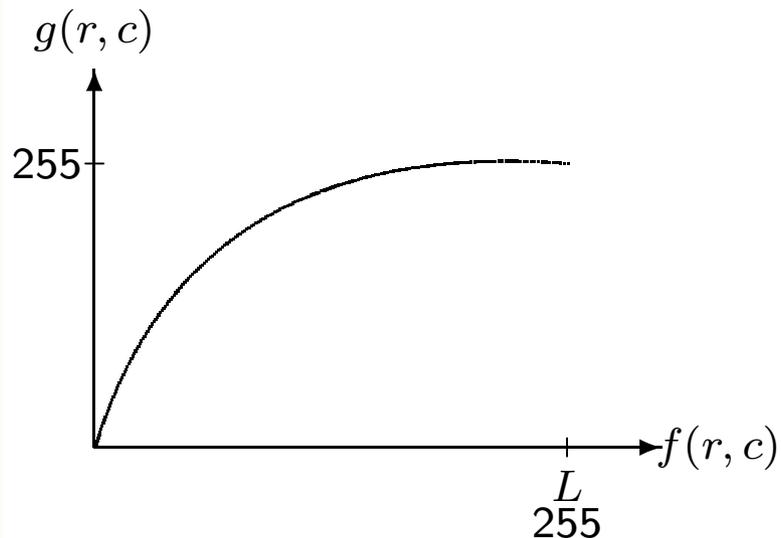
Die nichtlineare Log-Transformation:

- Variante 1:  $g(r, c) = 255 \cdot \frac{\log(1+f(r,c))}{\log 256}$
- Variante 2:  $g(r, c) = 255 \cdot \frac{\log(1+f(r,c))}{\log(1+L)}$ , wobei  $L$  den maximalen Grauwert von  $f(r, c)$  darstellt

bewirkt

- Dehnung der Grauwertdynamik im unteren Grauwertbereich
- gleichzeitige Stauchung im oberen Grauwertbereich

⇒ Details im dunklen Bereich besser sichtbar gemacht



Original



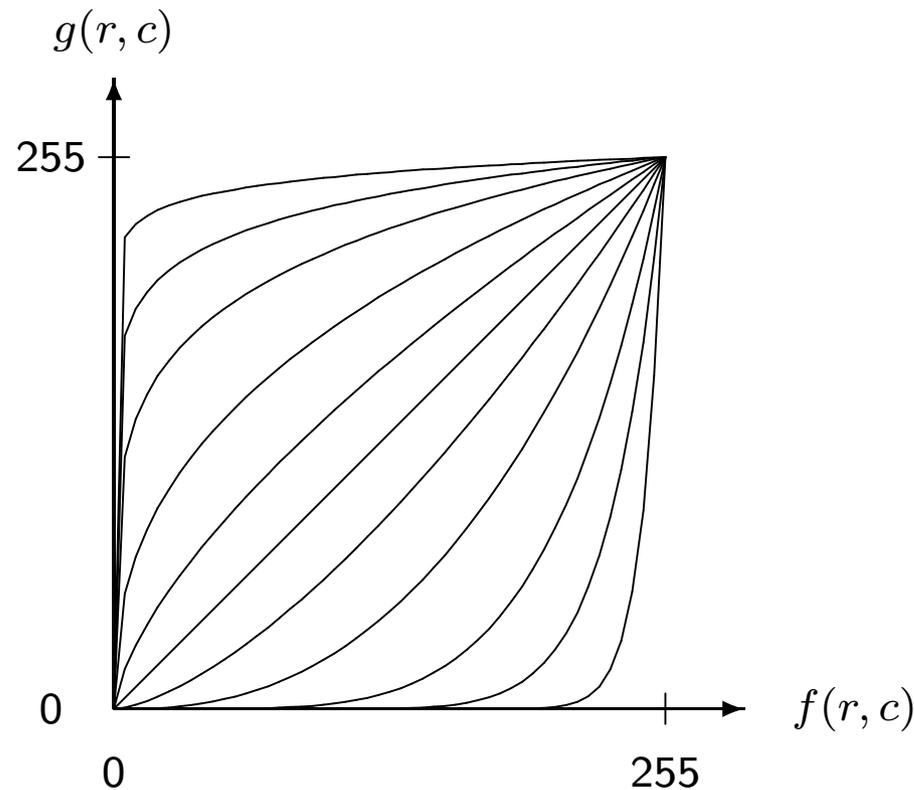
Log-Transformation

# Grauwerttransformation: Exponentielle Transformation (1)

Die allgemeine Form:

$$g(r, c) = 255 \cdot \left( \frac{f(r, c)}{255} \right)^\gamma, \quad \gamma > 0$$

definiert eine Reihe exponentieller Transformationen



Kennlinien für  $\gamma = 0.04, 0.1, 0.2, 0.4, 0.67, 1, 1.5, 2.5, 5, 10, 25$

# Grauwerttransformation: Exponentielle Transformation (2)

**Fall 1:**  $\gamma < 1$

Kennlinie ist derjenigen der Log-Transformation ähnlich

⇒ erhöht Helligkeit und ermöglicht insbesondere bessere Sichtbarkeit im dunklen Bildbereich



Original



$\gamma = 0.6$



$\gamma = 0.4$

# Grauwerttransformation: Exponentielle Transformation (3)

## Fall 2: $\gamma > 1$

Das Verhalten exakt umgekehrt im Vergleich zu  $\gamma < 1$ . Das transformierte Bild wirkt dunkler. Der untere Grauwertbereich wird zusammengestaucht, was bei zu hellen Bildern eine bessere Betrachtung ermöglicht.



Original



$\gamma = 2$



$\gamma = 3.5$



$\gamma = 5$

# Grauwerttransformation: Exponentielle Transformation (4)

## $\gamma$ -Korrektur:

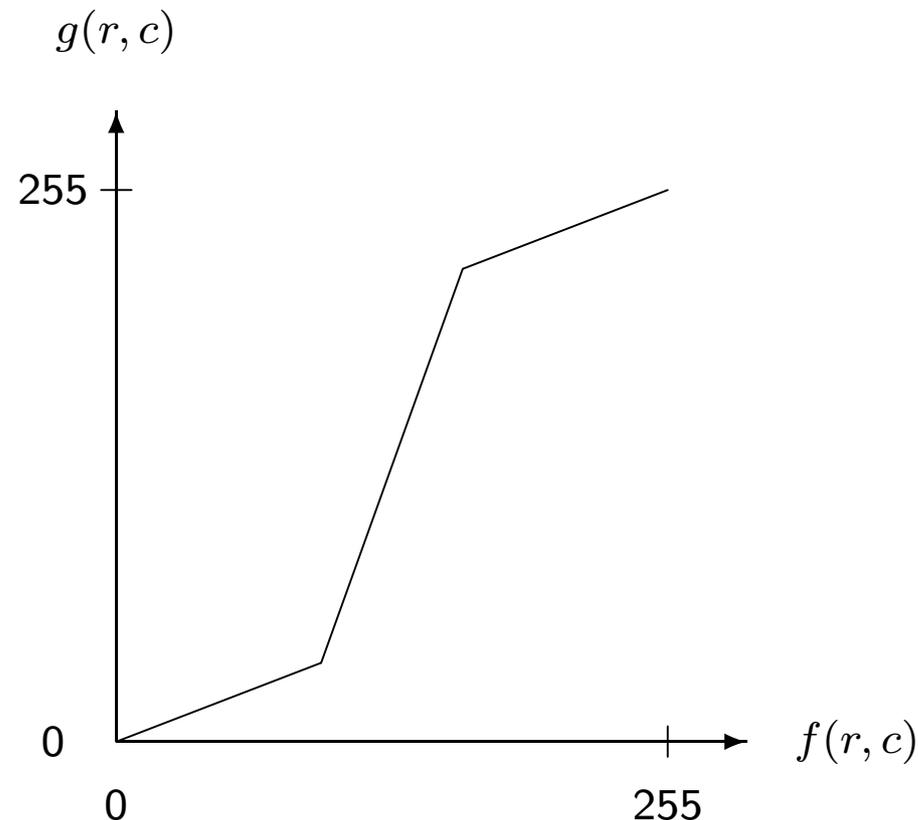
Viele Ausgabe-Geräte zeigen kein lineares, sondern exponentielles Verhalten. Bildschirm:  $1.8 \leq \gamma \leq 2.5$ . Mit  $\gamma = 2.5$  z.B. erscheint ein Bild tendentiell zu dunkel. Vorgeschaltete  $\gamma$ -Korrektur:

$$g(r, c) = 255 \cdot \left(\frac{f(r, c)}{255}\right)^{\frac{1}{2.5}} = 255 \cdot \left(\frac{f(r, c)}{255}\right)^{0.4}$$

kompensiert diesen Effekt und erlaubt natürlichere Betrachtung.

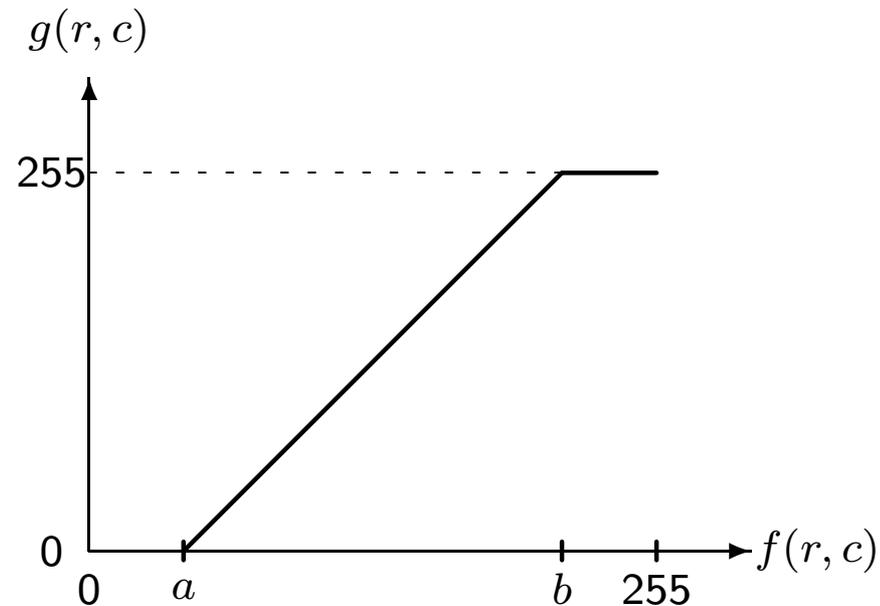
# Stückweise-lineare Transformation (1)

Vielfältige Grauwertveränderungen sind durch stückweise-lineare Transformationen modellierbar. Im Vergleich zur Log- bzw. exponentiellen Transformation sind hierbei allerdings mehr Parameter zu spezifizieren.



## Stückweise-lineare Transformation (2)

**Anwendung:** Dehnung der Grauwerte zur Kontrastverstärkung

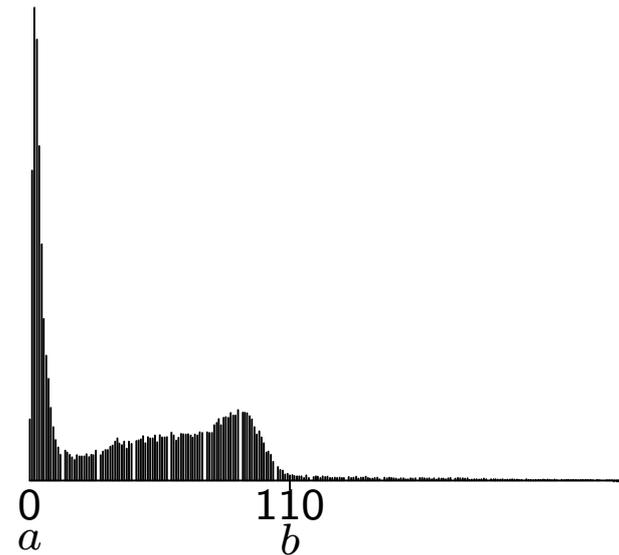
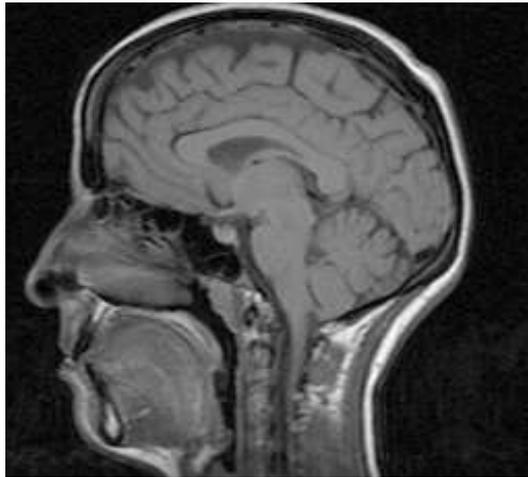


$$g(r, c) = \begin{cases} 0, & \text{falls } f(r, c) < a \\ \frac{f(r, c) - a}{b - a} \cdot 255, & \text{falls } a \leq f(r, c) \leq b \\ 255, & \text{falls } f(r, c) > b \end{cases}$$

Anwendbar, falls  $f(r, c)$  nur Grauwerte aus  $[a, b]$  belegt oder falls Bildinhalte aus dem Teilbereich  $[a, b]$  hervorgehoben werden sollen

# Stückweise-lineare Transformation (3)

**Beispiel:** Parameter  $a$  und  $b$  werden durch Betrachtung des Histogramms bestimmt



Kontrastverstärkung durch Dehnung der Grauwerte:  $[0,110] \rightarrow [0,255]$



**Quiz:** Warum die Spitze rechts im Histogramm?

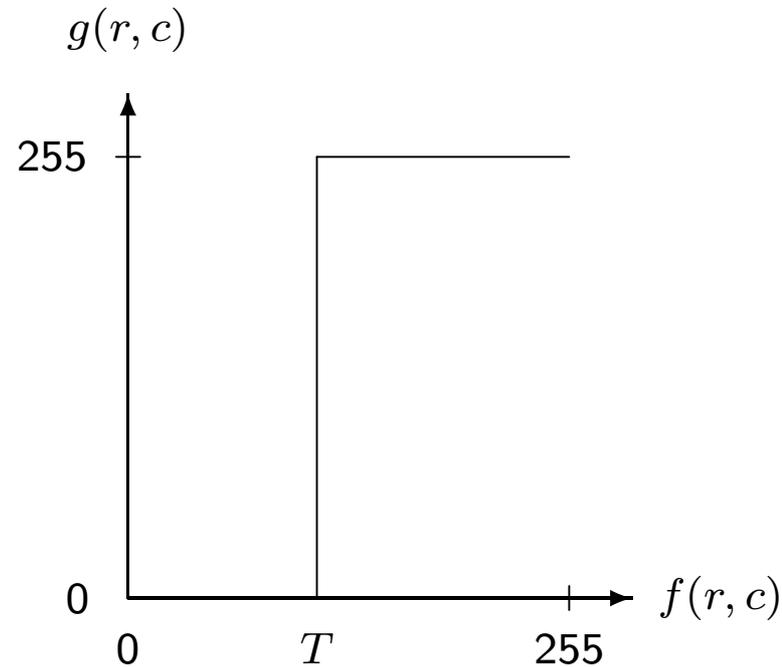
## Stückweise-lineare Transformation (4)

Kontrastverstärkung durch Dehnung der Grauwerte:  $[0,50] \rightarrow [0,255]$



# Stückweise-lineare Transformation (5)

**Anwendung:** Binärisierung



Trennung von Hintergrund und Objekten (z.B. Text). Eine Reihe von Verfahren zur automatischen Bestimmung des Schwellwertes  $T$  bekannt.

# Stückweise-lineare Transformation (6)

## Anwendung: Binärisierung (Fort.)

Grauwertbild  $\implies$  Histogramm  $\implies$  Binärbild

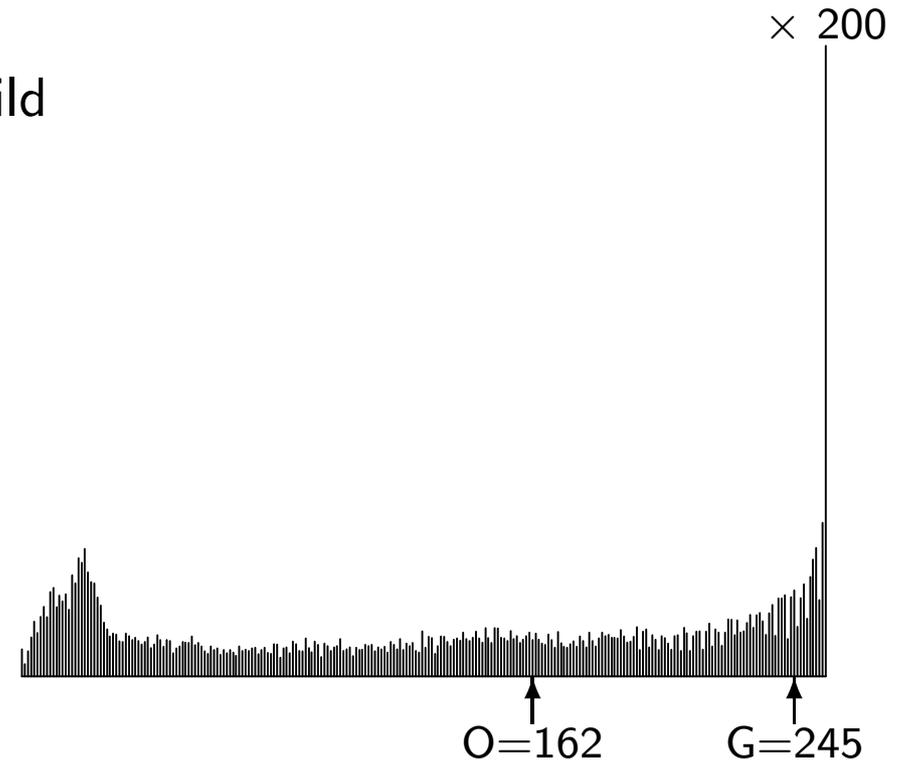
### 5.1.2 Schwellwertbestimmung durch Diskriminanzanalyse

Grauwerte:  $0, 1, \dots, M$ .

$H(i)$ : relative Häufigkeit(Wahrscheinlichkeit) von Grauwert  $i$ ,

$$\sum_{i=0}^M H(i) = 1$$

Ein möglicher Schwellwert  $S$ ,  $0 < S < M$ , trennt die Grauwerte in zwei Klassen,  $C_1$  und  $C_2$ . Die Auftretswahrscheinlichkeiten für die Elemente dieser Klassen sind:



### 5.1.2 Schwellwertbestimmung durch Diskriminanzanalyse

Grauwerte:  $0, 1, \dots, M$ .

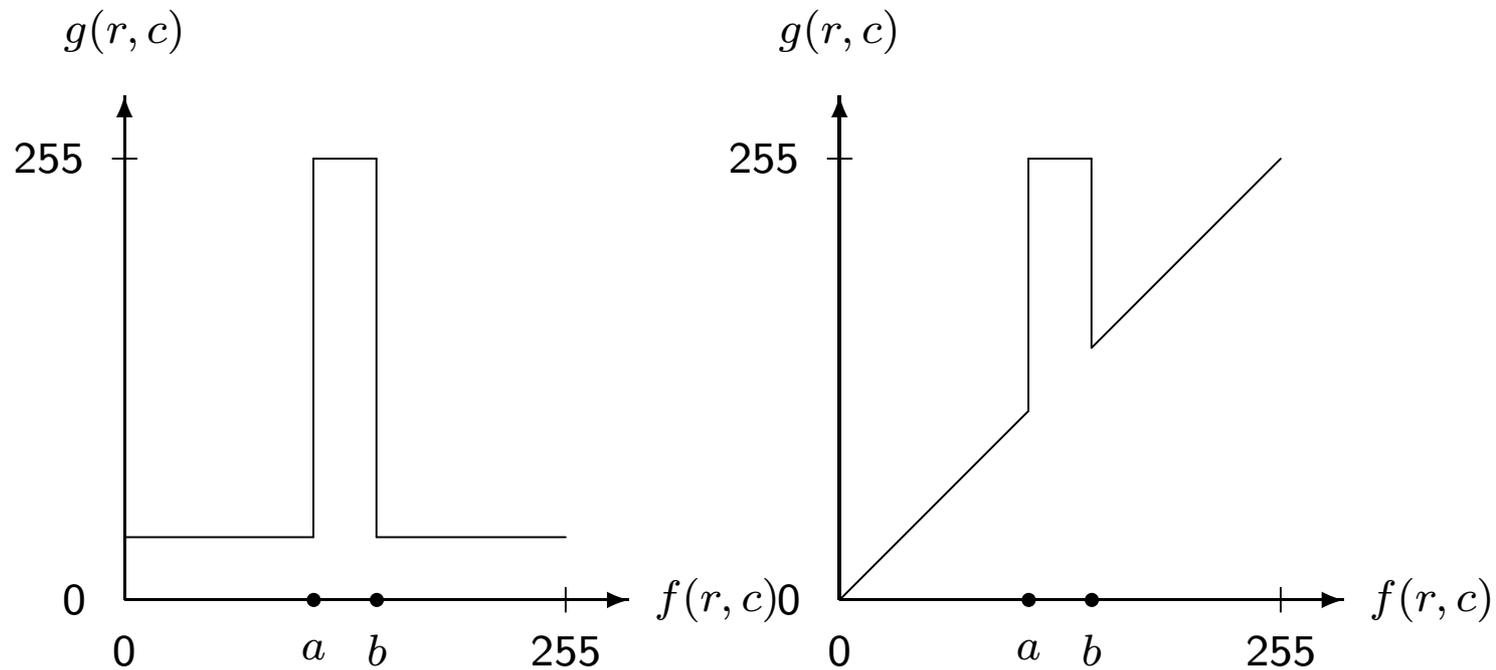
$H(i)$ : relative Häufigkeit(Wahrscheinlichkeit) von Grauwert  $i$ ,

$$\sum_{i=0}^M H(i) = 1$$

Ein möglicher Schwellwert  $S$ ,  $0 < S < M$ , trennt die Grauwerte in zwei Klassen,  $C_1$  und  $C_2$ . Die Auftretswahrscheinlichkeiten für die Elemente dieser Klassen sind:

# Stückweise-lineare Transformation (7)

## Anwendung: Graylevel Slicing

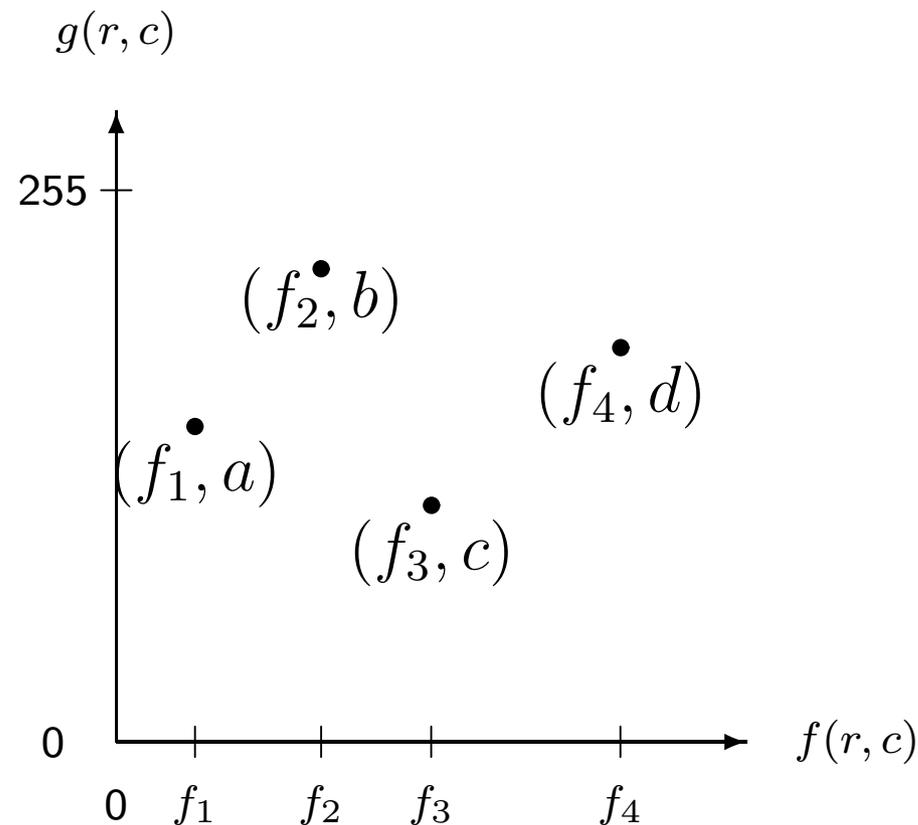


Einen bestimmten Grauwertbereich hervorheben

# Spezifikation der Transformation (1)

Angabe der entsprechenden Abbildungen für einige Grauwerte

Wird eine bestimmte Form von  $\Omega$  angenommen, so lassen sich die Parameter aus der Spezifikation errechnen



## Spezifikation der Transformation (2)

**Beispiel:** Polynom 3. Ordnung

$$g(r, c) = w_0 + w_1 f(r, c) + w_2 [f(r, c)]^2 + w_3 [f(r, c)]^3$$

Spezifikation: vier Paare  $(f_1, a)$ ,  $(f_2, b)$ ,  $(f_3, c)$ , und  $(f_4, d)$

⇒ lineares Gleichungssystem:

$$a = w_0 + w_1 f_1 + w_2 f_1^2 + w_3 f_1^3$$

$$b = w_0 + w_1 f_2 + w_2 f_2^2 + w_3 f_2^3$$

$$c = w_0 + w_1 f_3 + w_2 f_3^2 + w_3 f_3^3$$

$$d = w_0 + w_1 f_4 + w_2 f_4^2 + w_3 f_4^3$$

Dessen Lösung liefert die gewünschte Transformation

> 4 Paare angegeben

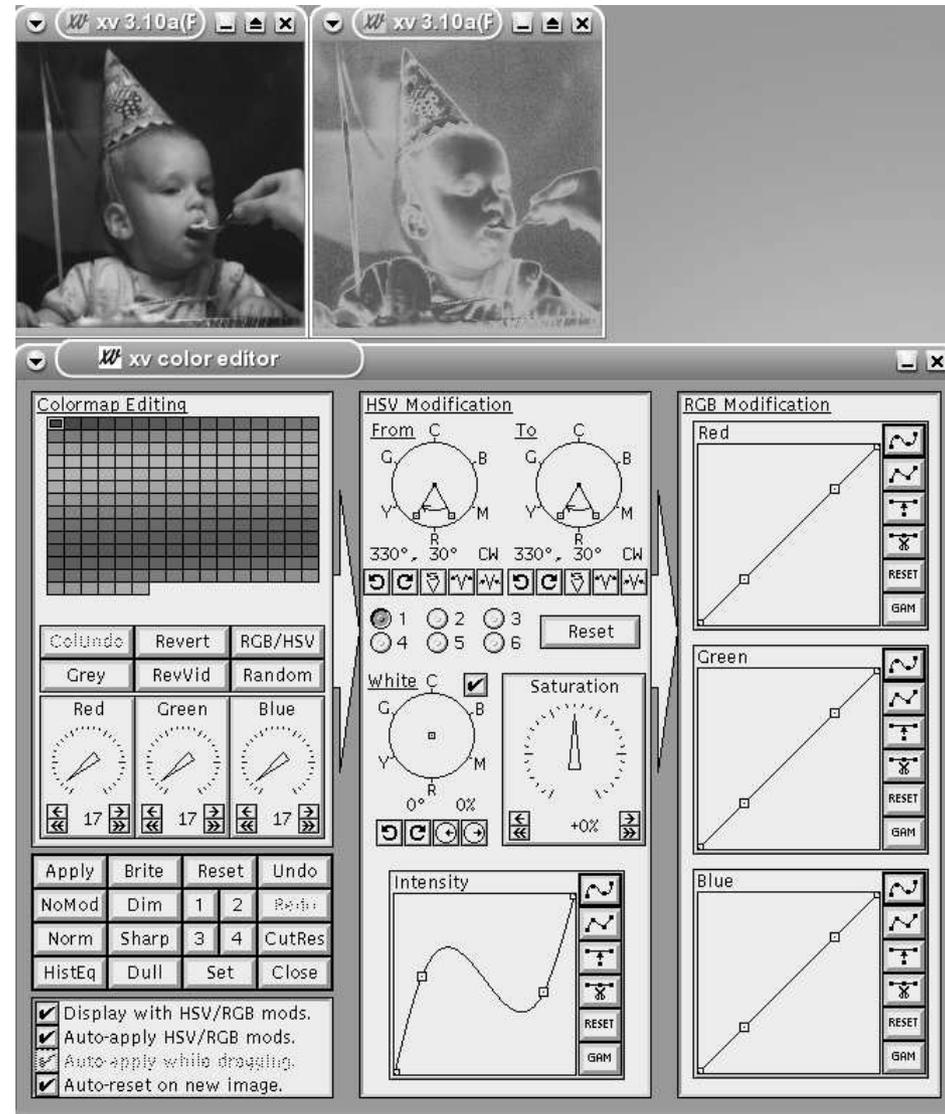
⇒ überbestimmtes Gleichungssystem

⇒ approximative Lösung mit Methode der kleinsten Quadrate

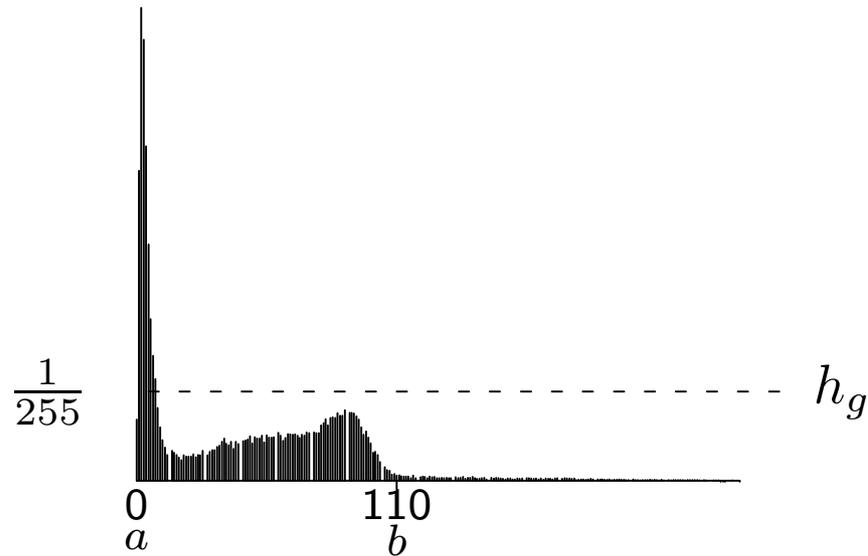
# Spezifikation der Transformation (3)

Viele Bildverarbeitungsprogramme bieten die Möglichkeit, die Transformationskennlinie manuell zu spezifizieren

**Beispiel:** xv



# Histogrammausgleich (1)



## Idee:

- Streckung großer Anhäufungen im Histogramm  $h_f(k)$  von  $f(r, c)$
- Stauchung wenig genutzter Grauwertbereiche (Kompensation)

**Ziel:**  $g(r, c)$  hat gleichverteiltes Histogramm  $h_g(k)$

## Histogrammausgleich (2)

Gesucht: *monotone* Abbildung  $\Omega : [0, 255] \rightarrow [0, 255]$ , so dass  $g(r, c) = \Omega(f(r, c))$  ein gleichverteiltes Histogramm  $h_g(k)$  hat

**Lösung:**

$$\begin{aligned}\int_0^p h_f(s) ds &= \int_0^{\Omega(p)} h_g(t) dt \quad (\text{wegen Monotonie von } \Omega) \\ &= \frac{\Omega(p)}{255}\end{aligned}$$

$$\implies \Omega(p) = 255 \cdot \int_0^p h_f(s) ds$$

**Diskretisierung:**  $\Omega(p) = (\text{int}) 255 \cdot \sum_{k=0}^p h_f(k), \quad p = 0, 1, 2, \dots, 255$

# Histogrammausgleich (3)

**Beispiel:** Bild mit  $M \cdot N = 128$  Bildpunkten; nur die kleinsten 16 Grauwerte (0–15) im Bild belegt

Jedem Grauwert  $p$  wird der Summe der bis zu  $p$  aufsummierten relativen Häufigkeiten zugeordnet, multipliziert mit dem maximalen Grauwert 255

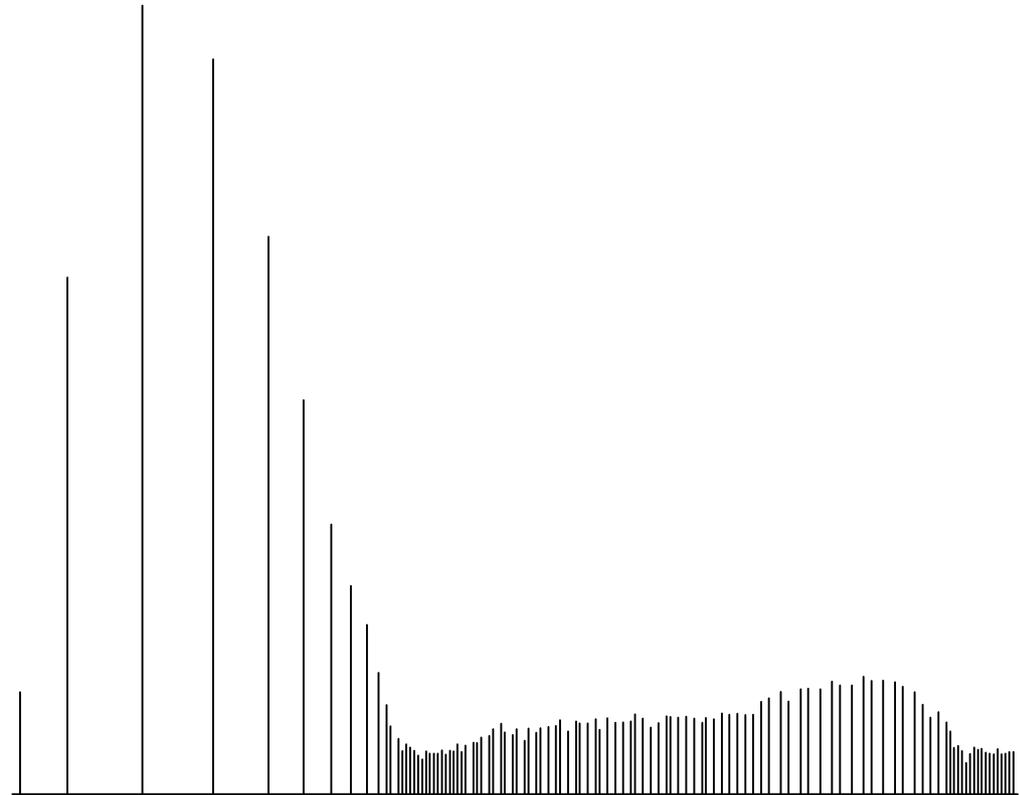
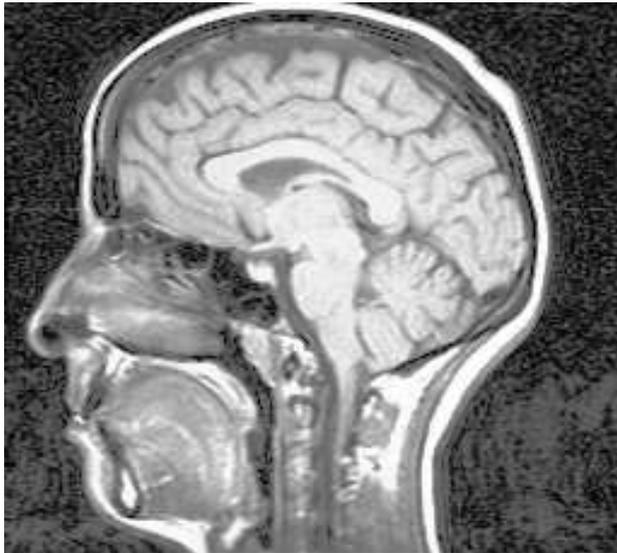
Rechenvorschrift:

$$\Omega(p) = (\text{int}) \frac{255}{M \cdot N} \cdot \sum_{k=0}^p H_f(k)$$

$p$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$H_f(p)$	1	3	4	11	18	14	8	3	1	1	3	8	20	18	12	3
$\sum_{k=0}^p H_f(k)$	1	4	8	19	37	51	59	62	63	64	67	75	95	113	125	128
$\Omega(p)$	1	7	15	37	73	101	117	123	125	127	133	149	189	225	249	255

# Histogrammausgleich (4)

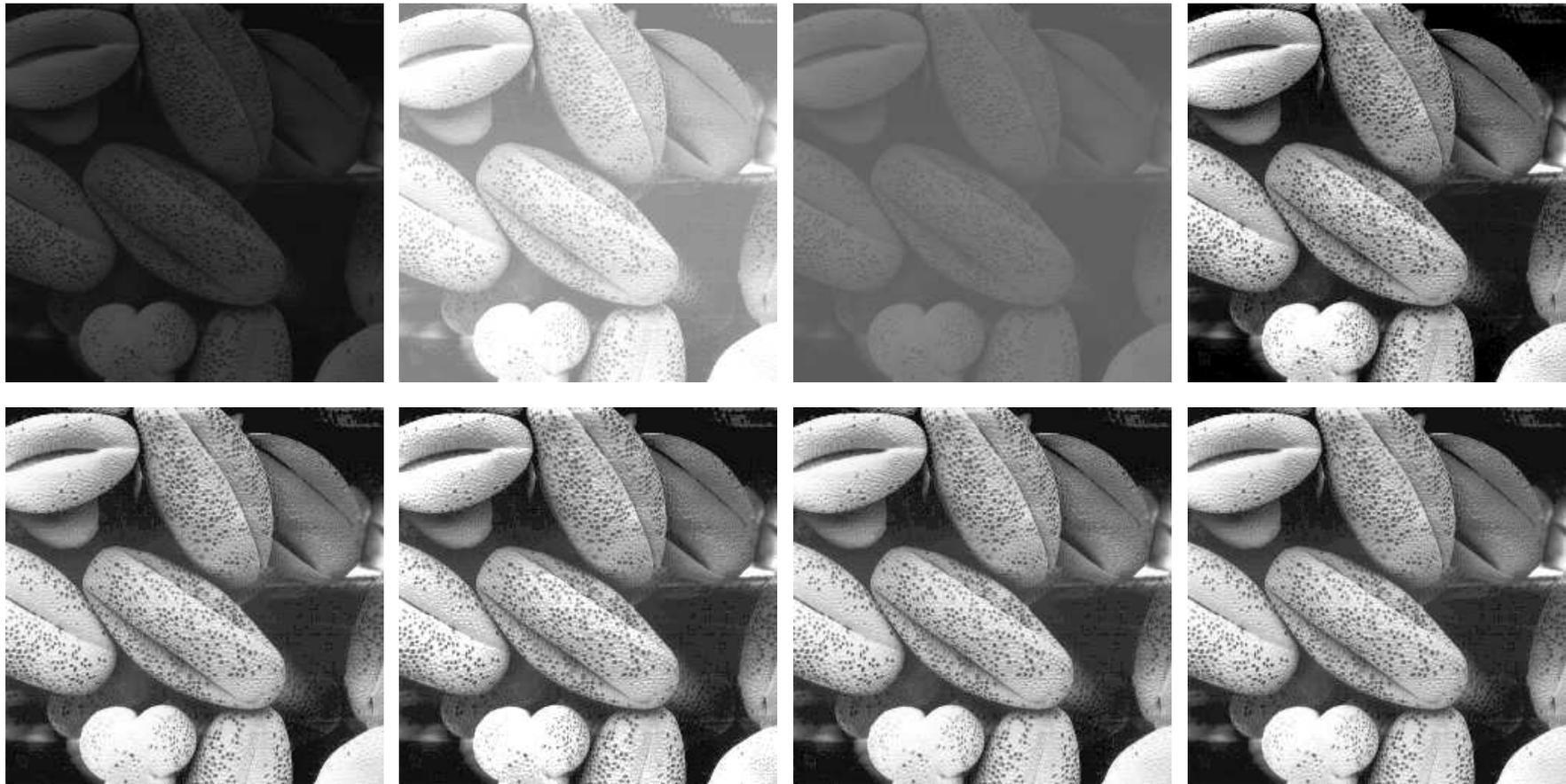
**Beispiel:** Kontrastverstärkung durch Histogrammausgleich



**Quiz:** Ist das resultierende Histogramm rechts gleichverteilt?

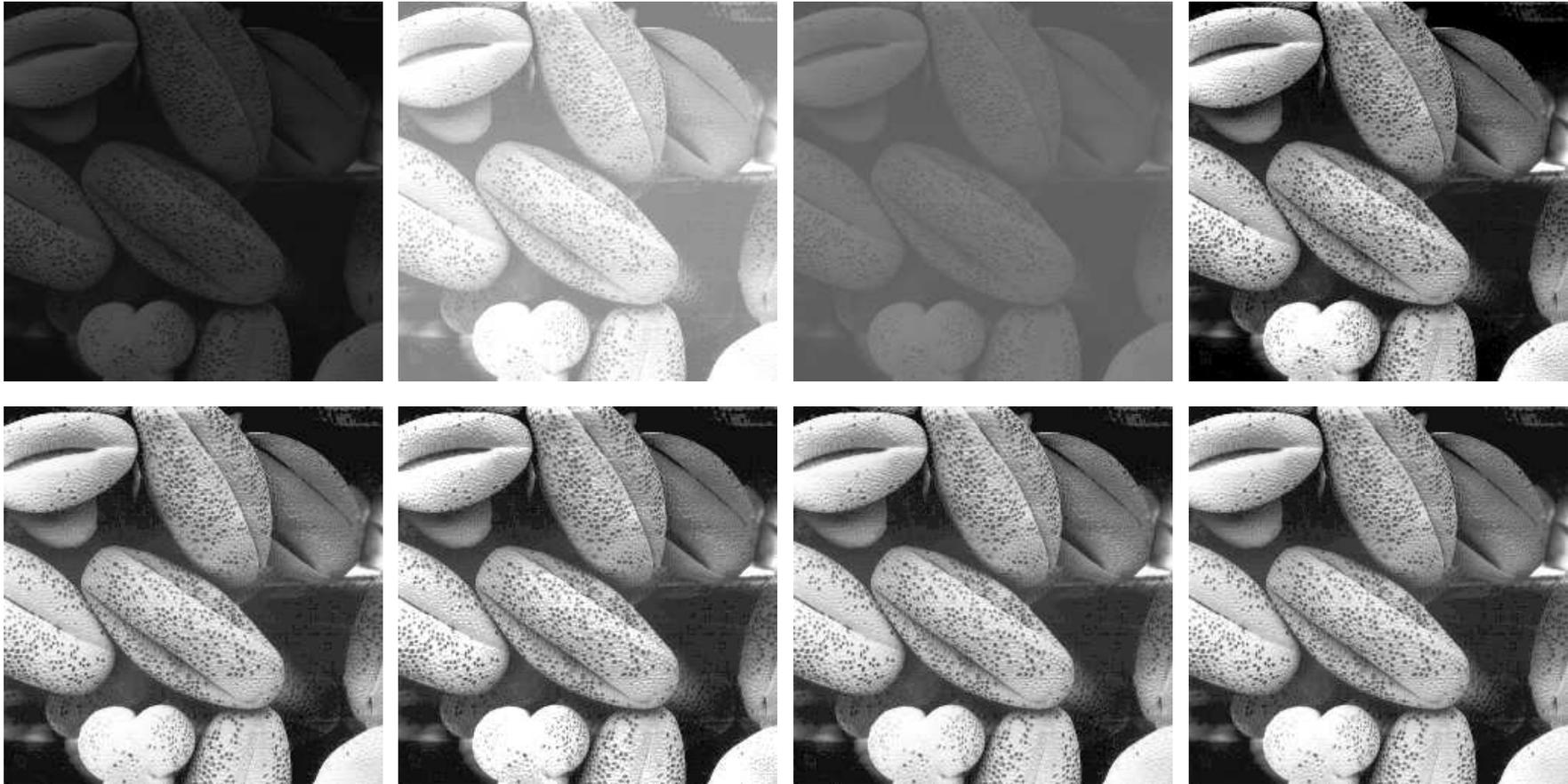
# Histogrammausgleich (5)

**Beispiel:** Histogrammausgleich zur Helligkeitsnormierung



# Histogrammausgleich (5)

**Beispiel:** Histogrammausgleich zur Helligkeitsnormierung



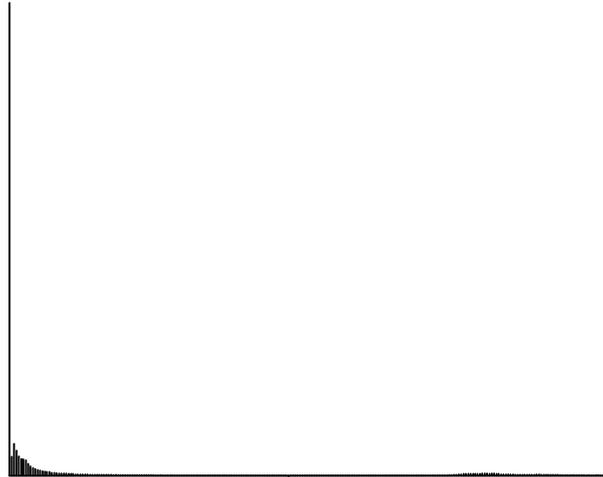
**Quiz:** Wie soll der Histogrammausgleich an Farbbildern durchgeführt werden?

# Histogrammausgleich (6)

**Beispiel:** Histogrammausgleich + Nachverarbeitung



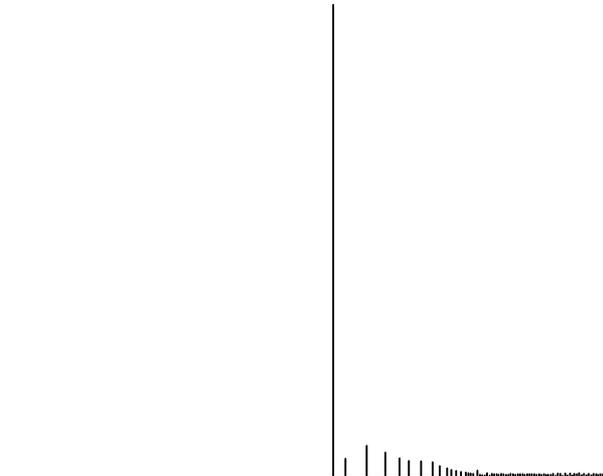
Original



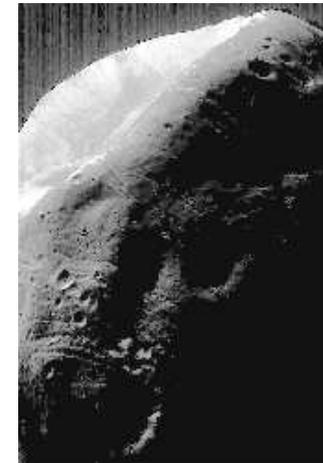
Histogramm



Histogrammausgleich



Histogramm



[128, 255] → [0, 255]

# Exakte Histogrammspezifikation

**Ziel:** Eingabebild  $f(r, c)$  wird so transformiert, dass das Ergebnisbild  $g(r, c)$  ein vorgegebenes Zielhistogramm  $H_g(k)$  *exakt* annimmt

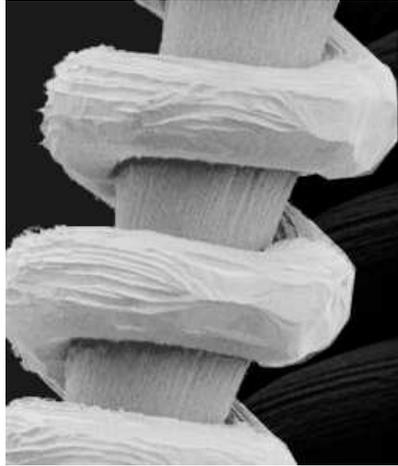
## Algorithmische Skizze:

- alle Pixel von  $f(r, c)$  nach ihren Grauwerten sortieren
- die ersten  $H_g(0)$  Pixel von  $f(r, c)$  bekommen neu den Grauwert 0
- die nächsten  $H_g(1)$  Pixel von  $f(r, c)$  bekommen den Grauwert 1
- .....
- die letzten  $H_g(255)$  Pixel von  $f(r, c)$  bekommen den Grauwert 255

Optimierung zur Effizienzsteigerung nötig!

# Lokale (adaptive) Grauwerttransformation: Ein Beispiel (1)

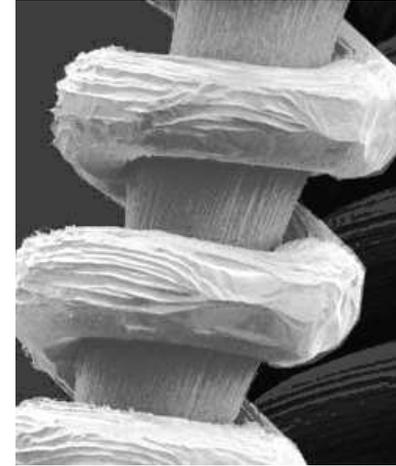
Gewisse Ortsabhängigkeit ist erwünscht:



Original



Exponentiell ( $\gamma = 0.2$ )



Histogrammausgleich

Gesucht ist ein Verfahren, bei dem

- der dunkle Grauwertbereich erhellt wird
- der hellere Grauwertbereich nach Möglichkeit unverändert bleiben soll

Adaptive Transformation  $\Omega_{rc}$ ,  $1 \leq r \leq M$ ,  $1 \leq c \leq N$ , die sich der jeweiligen Umgebung eines Bildpunktes  $f(r, c)$  anpasst

# Lokale (adaptive) Grauwerttransformation: Ein Beispiel (2)

Globale und lokale Bildcharakteristiken:

- $\bar{f}$ : Mittelwert des gesamten Bildes
- $\sigma$ : Standardabweichung des gesamten Bildes
- $\bar{f}_{rc}$ : Mittelwert der  $k \times k$  Umgebung eines Bildpunktes  $f(r, c)$
- $\sigma_{rc}$ : Standardabweichung der  $k \times k$  Umgebung eines Bildpunktes  $f(r, c)$

Diese vier Kennzahlen erlauben eine Beurteilung jedes Bildpunktes, ob er transformiert werden oder unverändert bleiben soll

Die Bildpunkte lassen sich mit drei Kriterien beschreiben:

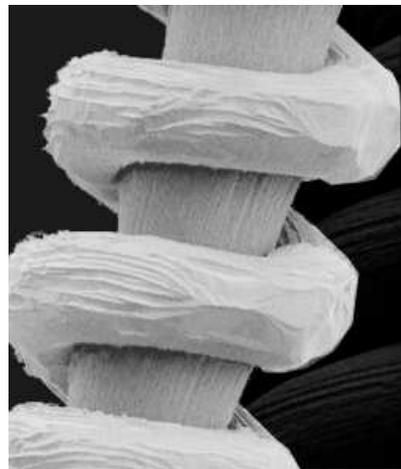
- $\bar{f}_{rc} \leq l_0 \bar{f}$ ,  $l_0 < 1$  (dunklere Bildpunkte)
- $\sigma_{rc} \leq l_2 \sigma$ ,  $l_2 < 1$  (niedriger Kontrast)
- $\sigma_{rc} \geq l_1 \sigma$ ,  $0 < l_1 < l_2$  (minimaler Kontrast; sonst würde auch ein konstanter Bildbereich berücksichtigt)

# Lokale (adaptive) Grauwerttransformation: Ein Beispiel (3)

Adaptive Grauwerttransformation:

$$g(r, c) = \Omega_{rc}(f(r, c))$$
$$= \begin{cases} E \cdot f(r, c); & \text{falls } (\bar{f}_{rc} \leq l_0 \bar{f}) \wedge (l_1 \sigma \leq \sigma_{rc} \leq l_2 \sigma) \\ f(r, c); & \text{sonst} \end{cases}$$

Ergebnis mit  $E = 4.0$ ,  $k = 3$ ,  $l_0 = 0.4$ ,  $l_1 = 0.02$ , und  $l_2 = 0.4$ :



Original



Markierung: ja/nein



Ergebnis

In vielen Situationen ist ein gewisses adaptives Verhalten unabdingbar. Dieses Beispiel zeigt das typische Muster beim adaptiven Vorgehen. Jeder Bildpunkt wird bewertet und entsprechend unterschiedlich bearbeitet.

# Nachbarschaftsoperatoren (1)

Grauwerttransformationen (Punktoperatoren) sind nicht in der Lage, die räumlichen Beziehungen der Grauwerte einer kleinen Umgebung zu erkennen. Zu diesem Zweck sind eine andere Klasse von Operationen notwendig, welche die Bildpunkte einer Nachbarschaft in geeigneter Weise kombinieren:

$$g(r, c) = \Omega(N(f(r, c)))$$

Sie werden *Nachbarschaftsoperatoren* oder *Filter* genannt.

Eine grundlegende Kombinationsart der Bildpunkte in einer Nachbarschaft erfolgt durch eine lineare Operation mit einer Filtermaske:

$$h(u, v), \quad -k \leq u \leq k, \quad -k \leq v \leq k$$

wobei sie jedem Bildpunkt im Bereich der Bildmaske den Gewichtungsfaktor der Maske multipliziert, die Punkte addiert, und die Summe an die Position des zentralen Bildpunktes schreibt:

$$g(r, c) = \sum_{u=-k}^k \sum_{v=-k}^k h(u, v) \cdot f(r + u, c + v)$$

## Nachbarschaftsoperatoren (2)

**Beispiel:**

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad f = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}, \quad g = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & 18 & 27 & \cdot \\ \cdot & 18 & 27 & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

## Nachbarschaftsoperatoren (2)

**Beispiel:**

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad f = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}, \quad g = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & 18 & 27 & \cdot \\ \cdot & 18 & 27 & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

**Quiz:** Was bewirkt die Operation mit folgender Maske?

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

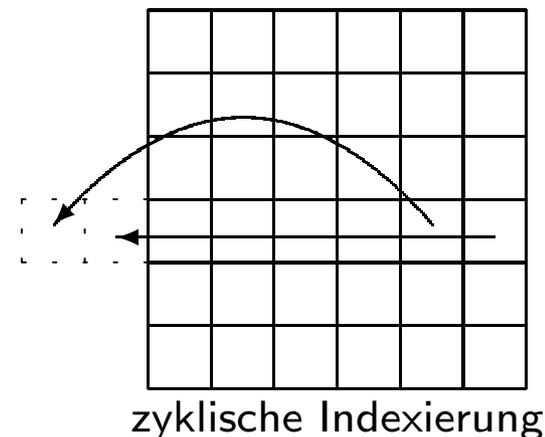
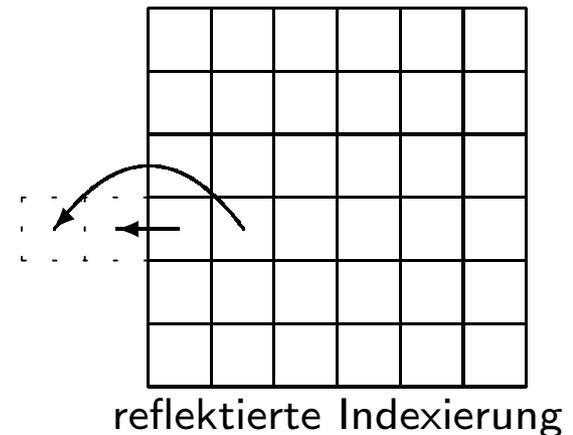
# Nachbarschaftsoperatoren (3)

## Randbehandlung:

Bei einer Filtermaske der Größe  $(2k + 1) \times (2k + 1)$  ist das Ergebnis für  $k$  Randzeilen bzw. -spalten nicht direkt definiert. Eine Nichtberücksichtigung der Randpunkte führt dazu, dass das Bild bei iterativer Anwendung des Operators immer weiter schrumpft. Um das Ergebnisbild in derselben Größe zu erhalten, müssen diese Zeilen und Spalten gefüllt werden.

Für  $k$  Randzeilen und -spalten bei  $(2k + 1) \times (2k + 1)$  Masken

- Randpixel unverändert lassen
- Randpixel auf Konstante setzen
- Faltungskern anpassen
- reflektierte Indexierung
- zyklische Indexierung



# Bildglättung: Störungsmodellierung (1)

---

Störungen (Rauschen; noise): benachbarte Bildpunkte, deren Grauwerte schwach schwanken sollten, weisen stark unterschiedliche Werte auf

Gewünscht: Rückgängigmachung (Reduzierung) der Störungen

Annahme: Störungen  $n(r, c)$  unabhängig von Position und vom Grauwert der Bildpunkte

$$f(r, c) = f^*(r, c) + n(r, c)$$

wobei  $f^*(r, c)$  das (unbekannte) Idealbild darstellt

Es sind verschiedene Störungsmodells in Form von Dichtefunktion bekannt

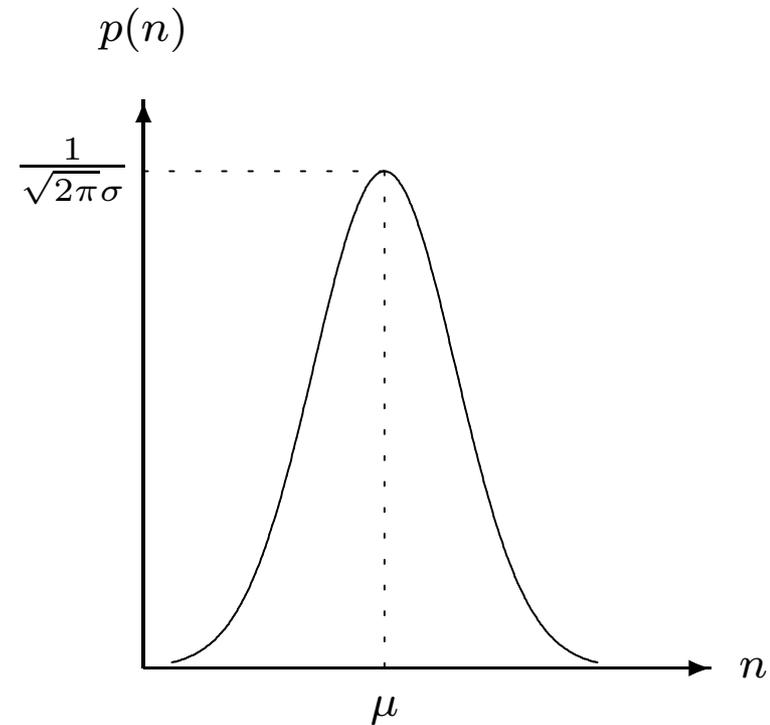
# Bildglättung: Störungsmodellierung (2)

## ● Normalverteilung (Gausssche Verteilung)

$$p(n) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(n-\mu)^2}{2\sigma^2}}$$

Ca. 70% der Werte fällt unter  $[\mu - \sigma, \mu + \sigma]$  und 95% unter  $[\mu - 2\sigma, \mu + 2\sigma]$ .

“Because of its mathematical tractability in both the spatial and frequency domains, Gaussian noise models are used frequently in practice. In fact, this tractability is so convenient that it is often results in Gaussian models being used in situations in which they are marginally applicable at best.”



Gaussian

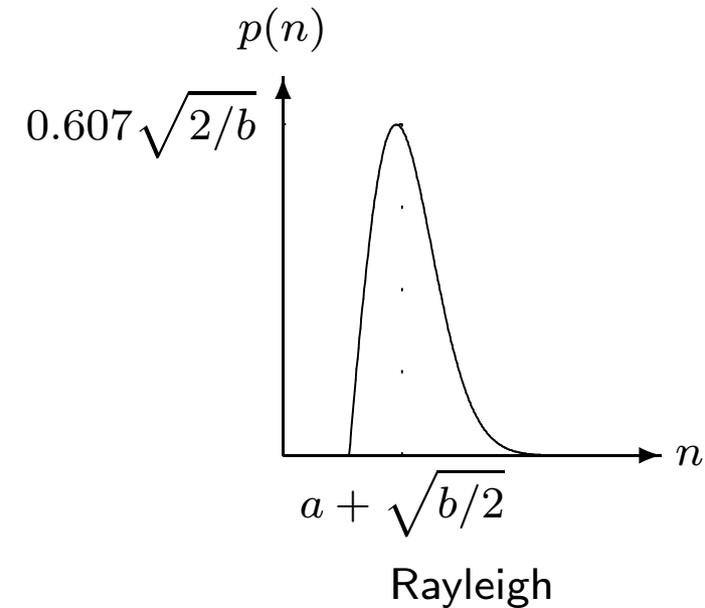
# Bildglättung: Störungsmodellierung (3)

## ● Rayleigh-Verteilung

$$p(n) = \begin{cases} \frac{2}{b}(n-a)e^{-(n-a)^2/b}, & \text{falls } n \geq a \\ 0, & \text{falls } n < a \end{cases}$$

mit

$$\mu = a + \frac{\sqrt{\pi b}}{2}, \quad \sigma = \frac{\sqrt{b(4-\pi)}}{2}$$

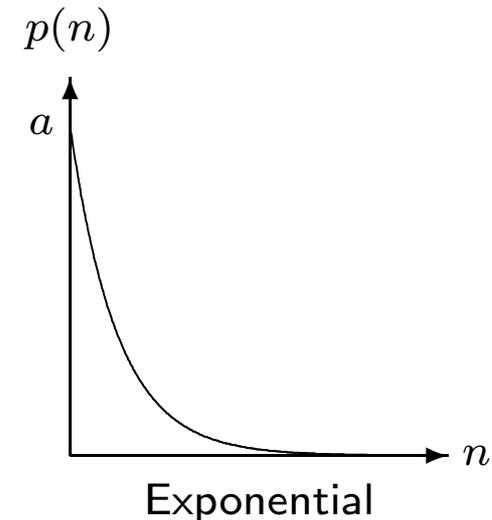


## ● Exponential-Verteilung: $a > 0$

$$p(n) = \begin{cases} ae^{-az}, & \text{falls } n \geq 0 \\ 0, & \text{falls } n < 0 \end{cases}$$

mit

$$\mu = \frac{1}{a}, \quad \sigma = \frac{1}{a}$$



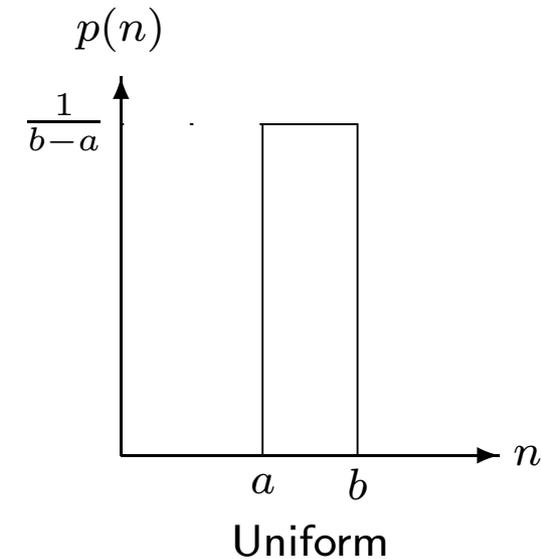
# Bildglättung: Störungsmodellierung (4)

## ● Gleichverteilung

$$p(n) = \begin{cases} \frac{1}{b-a}, & \text{falls } a \leq n \leq b \\ 0, & \text{sonst} \end{cases}$$

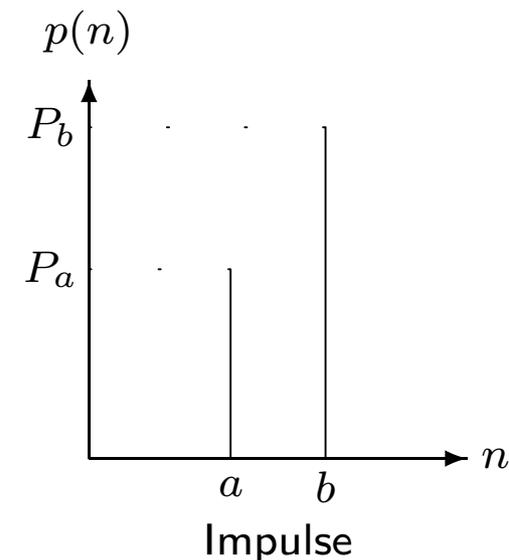
mit

$$\mu = \frac{a+b}{2}, \quad \sigma = \frac{b-a}{\sqrt{12}}$$



## ● Salt-and-pepper (impulse) Störung: $P_a + P_b \ll 1$

$$p(n) = \begin{cases} P_a, & \text{falls } n = a \\ P_b, & \text{falls } n = b \\ 0, & \text{sonst} \end{cases}$$



# Bildglättung: Störungsmodellierung (5)

Beispiel:



Original



Gaussian



Impulse

# Bildglättung: Filter (1)

**Mittelwertfilter:**

$$g(r, c) = \frac{1}{(2k + 1)^2} \sum_{u=-k}^k \sum_{v=-k}^k f(r + u, c + v)$$

Maske:

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (k = 1)$$

Nachteil: Kontrastverminderung (Kantenverwischung)

# Bildglättung: Filter (1)

**Mittelwertfilter:**

$$g(r, c) = \frac{1}{(2k + 1)^2} \sum_{u=-k}^k \sum_{v=-k}^k f(r + u, c + v)$$

Maske:

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (k = 1)$$

Nachteil: Kontrastverminderung (Kantenverwischung)

**Gauss-Filter:**

Gewichtete Mittelwertfilterung; Gauss-Funktion als Gewichte:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Masken:  $\sigma = 0.391 \rightarrow \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$        $\sigma = 0.625 \rightarrow \frac{1}{121} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 7 & 11 & 7 & 2 \\ 3 & 11 & 17 & 11 & 3 \\ 2 & 7 & 11 & 7 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$

Bemerkung:

- Wirkungsbereich der Filterung durch Parameter  $\sigma$  kontrollierbar
- andere Approximationen von  $G(x, y)$  möglich

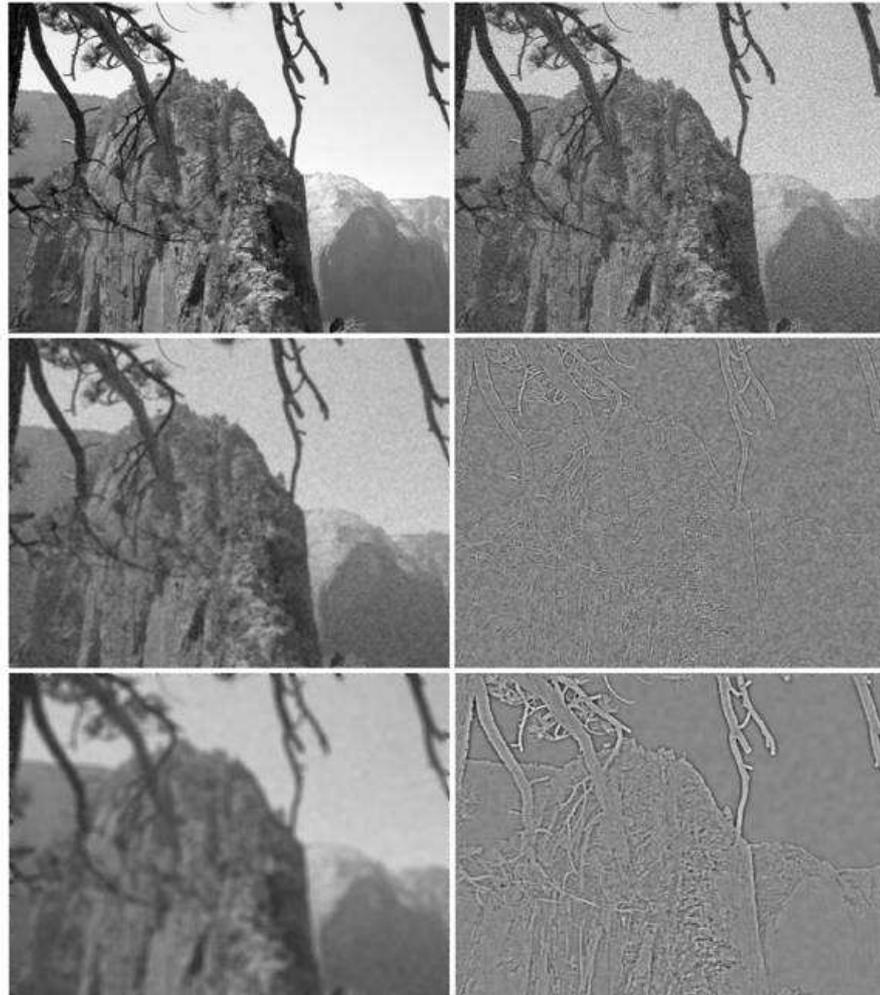
## Bildglättung: Filter (2)

### Beispiel: Mittelwertfilter

Oben: Originalbild (links) und verrauschtes Bild  $f$  (rechts)

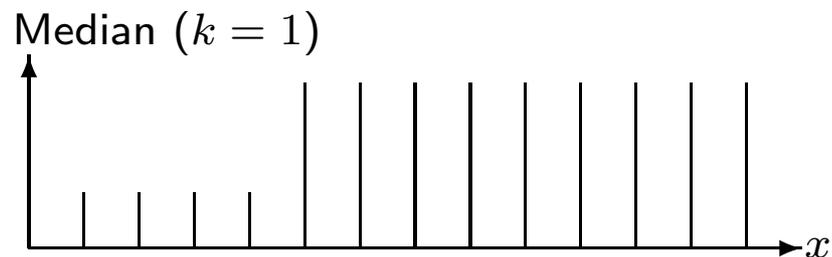
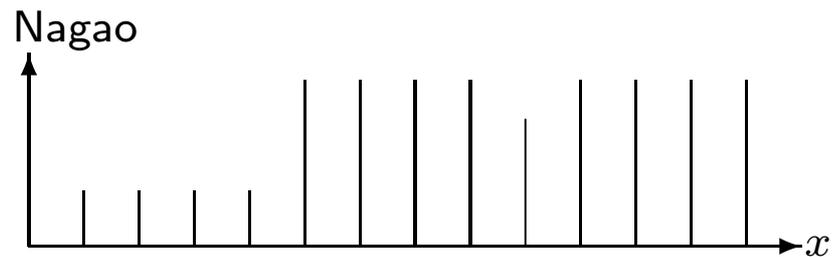
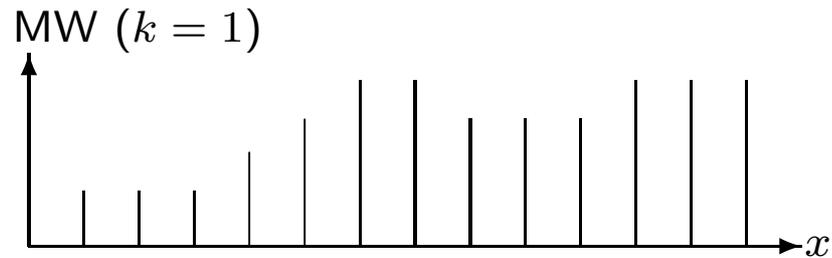
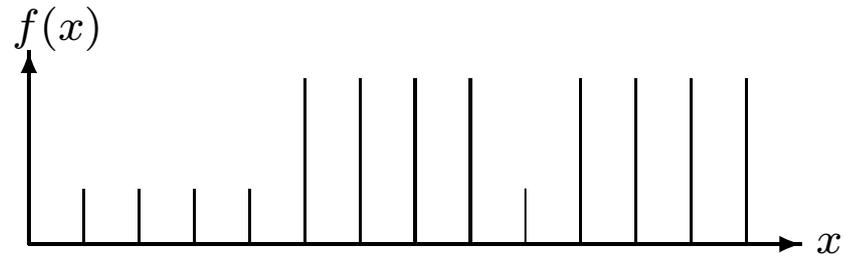
Mitte:  $3 \times 3$  Filtering (links) und Differenz zwischen Ergebnis und  $f$  (rechts)

Unten:  $5 \times 5$  Filtering (links) und Differenz zwischen Ergebnis und  $f$  (rechts)



# Bildglättung: Filter (3)

Illustration: Glättungsfilter (3er-Nachbarschaft)



# Bildglättung: Filter (4)

Beispiel:



Idealbild



Gestörtes Bild



Mittelwertfilter ( $k = 2$ )



Gauss-Filter ( $\sigma = 0.625$ )



Nagao-Filter

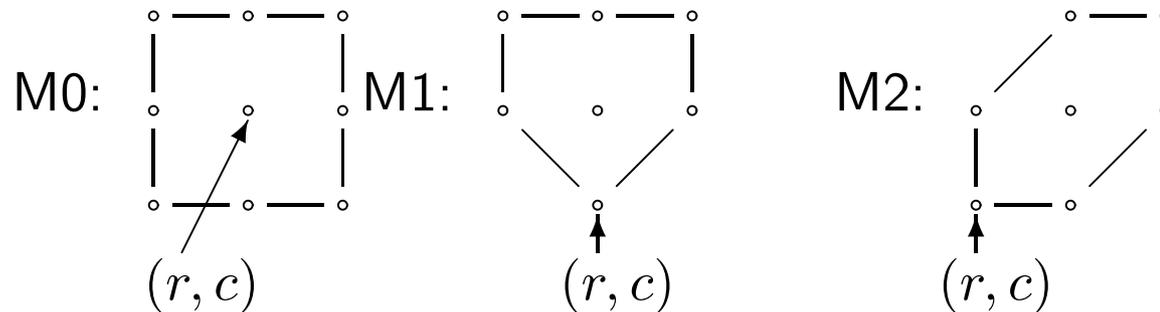


Medianfilter ( $k = 2$ )

## Bildglättung: Filter (5)

**Adaptive** Glättung nach Nagao: Mittelung nur in einem homogenen Teil der Umgebung

Masken:  $M_0, M_1, \dots, M_8$



$M_3, \dots, M_8$  durch Rotation von  $M_1$  und  $M_2$  um  $90^\circ, 180^\circ, 270^\circ$

Rechenvorschrift:

- mittleren Grauwert und Varianz für  $M_0, M_1, \dots, M_8$  berechnen
- Maske  $M$  mit minimaler Varianz bestimmen
- $g(r, c)$  ergibt sich aus dem mittleren Grauwert von  $M$

i.a. mehrere Iterationen

# Bildglättung: Filter (6)

## Medianfilter:

Ergebnis  $g(r, c) = \text{Median}$  der Grauwerte im Bild  $f$  innerhalb eines  $(2k + 1) \times (2k + 1)$  Fensters mit Mittelpunkt  $(r, c)$

Berechnung des Medians: Die  $(2k + 1)^2$  Grauwerte nach der Größe sortieren.

Der  $\lceil \frac{(2k+1)^2}{2} \rceil \equiv (2k^2 + 2k + 1)$ -te Wert ist der Median.

Z.B.  $3 \times 3$  Fenster  $\rightarrow$  5. Wert

$$\left. \begin{array}{ccc} 1 & 1 & 1 \\ 1 & 3 & 2 \\ 5 & 4 & 2 \end{array} \right\} \longrightarrow 1, 1, 1, 1, 2, 2, 3, 4, 5$$

↑  
Median

Bemerkung:

- Gehört zur Klasse der Rangordnungsoperatoren (rank order filtering)
- Höhere Rechenkomplexität als andere Operatoren (Sortieren)
- Kein Verwischen von Kanten, aber Abrunden von Ecken
- Bes. effektiv für salt-and-pepper Störungen; bis zu 50% Ausreißer

## Bildglättung: Filter (7)

**Zusammenhang** zwischen Mittelwert- und Medianfilter: Herleitung  
Mittelwert- und Medianfilter in einem einheitlichen Rahmen

Aufgabe: für die Nachbarschaft

$$f(r + u, c + v), \quad -k \leq u, v \leq k$$

einen Grauwert  $\bar{f}$  suchen, der die  $(2k + 1)^2$  Grauwerte optimal repräsentiert

Optimierung:

$$g(r, c) = \arg \min_{\bar{f}} \sum_{u=-k}^k \sum_{v=-k}^k d(f(r + u, c + v), \bar{f})$$

basierend auf Abstand  $d(g_1, g_2)$  zwischen Grauwert  $g_1$  und  $g_2$

Verschiedene Abstandsfunktionen  $d()$ :

- $d_1(f(r + u, c + v), \bar{f}) = |f(r + u, c + v) - \bar{f}| \Rightarrow$  Medianfilter
- $d_2(f(r + u, c + v), \bar{f}) = (f(r + u, c + v) - \bar{f})^2 \Rightarrow$  Mittelwertfilter

Im Vergleich zu  $d_2$  ist  $d_1$  weit robuster gegenüber Ausreißern,  
dementsprechend auch das Verhalten der beiden Glättungsfilter

## Bildglättung: Filter (8)

---

**Glättung durch Bildmittelung:** Störungsreduktion durch Berücksichtigung einer Bildsequenz

Annahme:

- Störungsmodell:  $f(r, c) = f^*(r, c) + n(r, c)$
- $n(r, c)$  hat Erwartungswert null

Lösung: Störungsreduktion durch Mittelung einer Reihe von Bildern  $f_i(r, c)$  desselben Inhaltes

## Bildglättung: Filter (8)

**Glättung durch Bildmittelung:** Störungsreduktion durch Berücksichtigung einer Bildsequenz

Annahme:

- Störungsmodell:  $f(r, c) = f^*(r, c) + n(r, c)$
- $n(r, c)$  hat Erwartungswert null

Lösung: Störungsreduktion durch Mittelung einer Reihe von Bildern  $f_i(r, c)$  desselben Inhaltes

Begründung: Mittelung von  $k$  verschiedenen gestörten Bildern  $f_i(r, c)$

$$\bar{f}(r, c) = \frac{1}{k} \cdot \sum_{i=1}^k f_i(r, c)$$

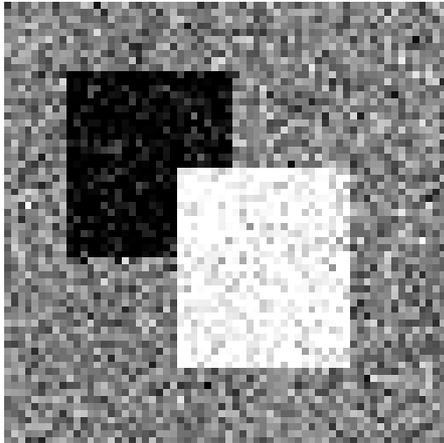
Es gilt:

$$E\{\bar{f}(r, c)\} = f^*(r, c), \quad \sigma\{\bar{f}(r, c)\} = \frac{1}{\sqrt{k}} \sigma_{n(r, c)}$$

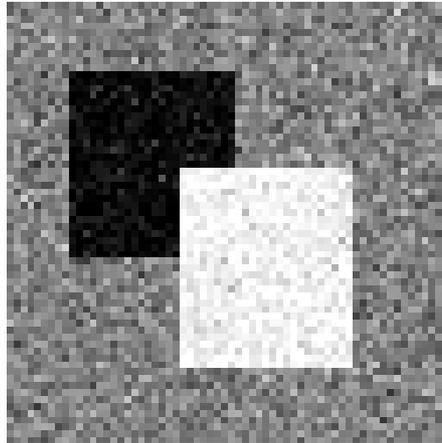
$E\{\bar{f}(r, c)\} = f^*(r, c) \implies \bar{f}(r, c)$  konvergiert zum Idealwert  $f^*(r, c)$   
 $\sigma\{\bar{f}(r, c)\} \implies$  Varibilitätät (Störung) des Grauwertes  $\bar{f}(r, c)$  nimmt ab

# Bildglättung: Filter (9)

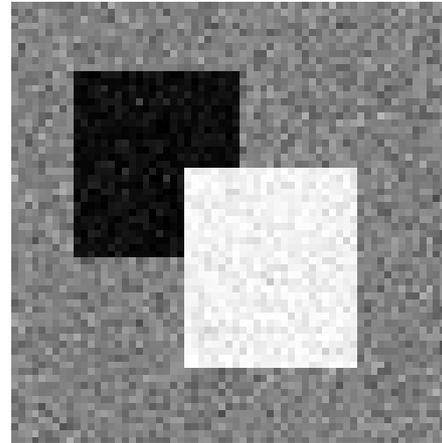
**Beispiel:** Glättung durch Bildmittelung



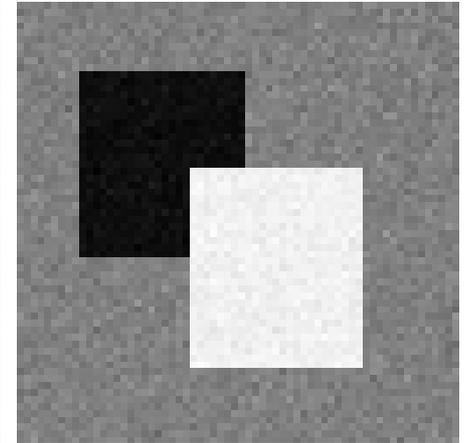
$k = 1$



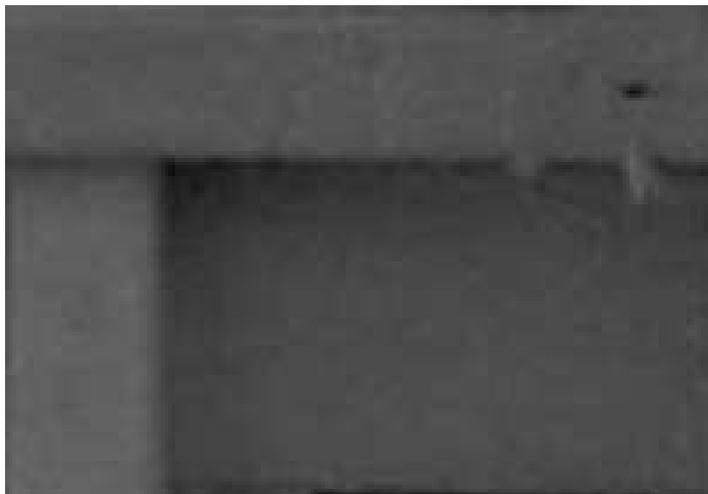
$k = 2$



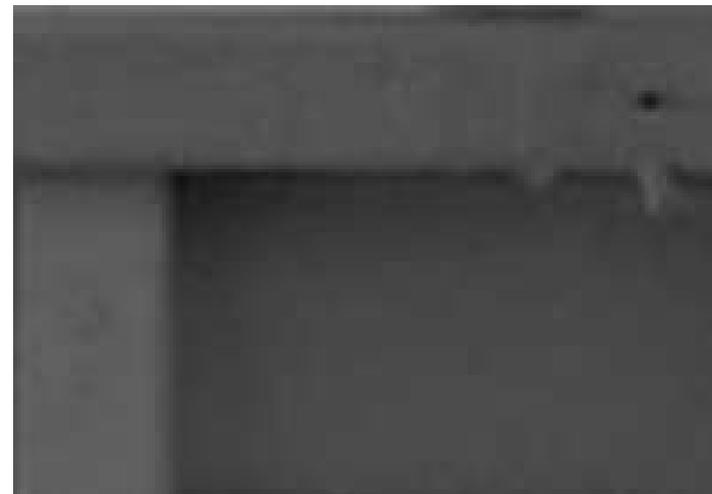
$k = 5$



$k = 20$



$k = 1$

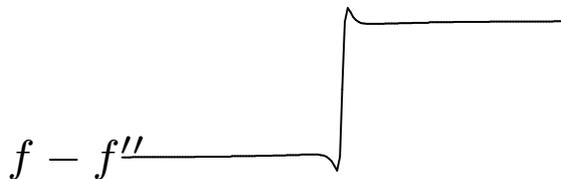
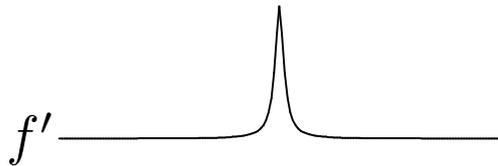
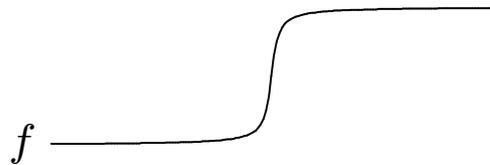


$k = 4$

# Filter zur Bildverschärfung (1)

Bilder mit hervorgehobenen Kanten werden vom Betrachter subjektiv besser bewertet. Dieser Effekt seit langem bekannt / z.B. Einsatz in der Druckindustrie

**Bildverschärfung mittels Laplace-Operator:** Kontrast  $\equiv$  Kantensteilheit



2D-Fall: Laplace-Operator als Entsprechung von  $f''$

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Rechenvorschrift:  $g(r, c) = f(r, c) - k \cdot \nabla^2 f$

## Filter zur Bildverschärfung (2)

Diskrete Approximation:

$$\left. \frac{\partial f}{\partial x} \right|_{\substack{x=c \\ y=r}} = f(r, c+1) - f(r, c)$$

$$\left. \frac{\partial^2 f}{\partial x^2} \right|_{\substack{x=c \\ y=r}} = \left. \frac{\partial f}{\partial x} \right|_{\substack{x=c \\ y=r}} - \left. \frac{\partial f}{\partial x} \right|_{\substack{x=c-1 \\ y=r}} = f(r, c-1) + f(r, c+1) - 2f(r, c)$$

$$\left. \frac{\partial^2 f}{\partial y^2} \right|_{\substack{x=c \\ y=r}} = f(r-1, c) + f(r+1, c) - 2f(r, c)$$

$$g(r, c) = (4k+1) \cdot f(r, c) - k \cdot [f(r, c-1) + f(r, c+1) + f(r-1, c) + f(r+1, c)]$$

$$\text{Maske: } \begin{bmatrix} 0 & -k & 0 \\ -k & 4k+1 & -k \\ 0 & -k & 0 \end{bmatrix} \quad \text{Alternative: } \begin{bmatrix} -k & -k & -k \\ -k & 8k+1 & -k \\ -k & -k & -k \end{bmatrix}$$

Bemerkung:

- Andere diskrete Approximationen von  $\nabla^2 f$  möglich
- Es können Werte  $g(r, c)$  außerhalb von  $[0, 255]$  auftreten; in diesem Fall durch 0 bzw. 255 ersetzen.
- Biologische Motivierung: Mach-Effekt

# Filter zur Bildverschärfung (3)

Beispiel:



Original



Laplace ( $k = 1$ )



Laplace ( $k = 4$ )



Unsharp-Masking ( $c = 0.8$ )



Unsharp-Masking ( $c = 0.6$ )

# Filter zur Bildverschärfung (4)

**Unsharp-Masking Verfahren:** Technik in der Photographie; das Positiv einer unscharfen Belichtung auf dem ursprünglichen Negativ ausgerichtet und erneut belichtet

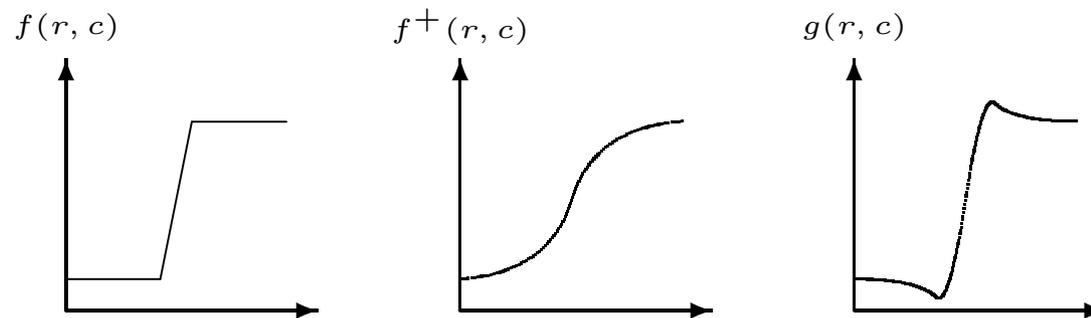
In der digitalen Bildverarbeitung:

$$g(r, c) = \frac{c}{2c - 1} f(r, c) - \frac{1 - c}{2c - 1} f^+(r, c)$$

$f^+(r, c)$ : unscharfe Version des Originals  $f(r, c)$

Gewichtungsfaktor  $c$  liegt in der Praxis zwischen 0.6 und 0.8 und bestimmt den Schärfeeffekt (Zunahme bei einer Verkleinerung von  $c$ )

Effekt der Kanten hervorhebung:



Z.B. Gauss-Filter ( $\sigma = 0.391$ ) zur Erstellung einer unscharfen Version,  $c = 0.75$ :

$$\frac{3}{2} \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{2} \cdot \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \frac{1}{32} \begin{bmatrix} -1 & -2 & -1 \\ -2 & 4 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

# Korrektur von uneinheitlichem Hintergrund (1)

Uneinheitlicher Hintergrund: Beleuchtung, Materialeigenschaft, etc.

Korrekturvorgang:

- Modellierung des Hintergrunds; häufig polynomiale Funktion:

$$b(r, c) = a_{00} + a_{10}r + a_{01}c + a_{20}r^2 + a_{11}rc + a_{02}c^2$$

oder mit höherer Ordnung

- Subtraktion des Hintergrund vom Eingabebild

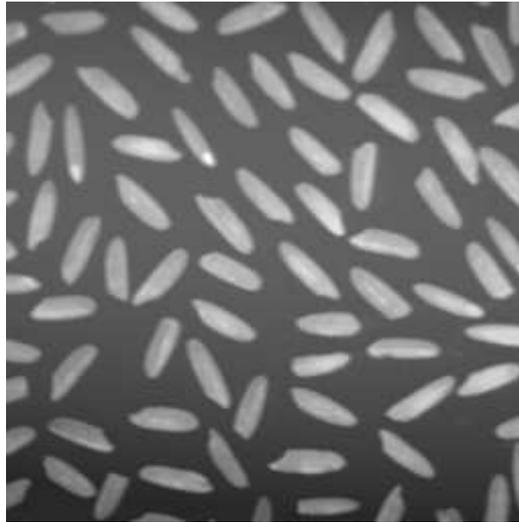
Bestimmung der Parameter  $a_{ij}$ :

Genügende Anzahl ( $\geq 6$ , i.d.R. mehr) von Hintergrundpunkten, manuell oder automatisch, bestimmen. Anschließend Funktionsapproximation.

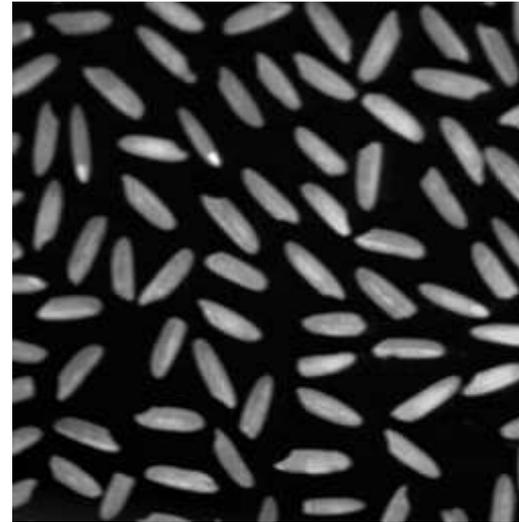
Beispiel: Bild in  $k \times k$  Subbilder aufteilen; für jedes Subbild den dunkelsten Punkt selektieren  $\implies k^2$  Hintergrundpunkte für Funktionsapproximation

# Korrektur von uneinheitlichem Hintergrund (2)

**Beispiel:** Uneinheitlicher Hintergrund

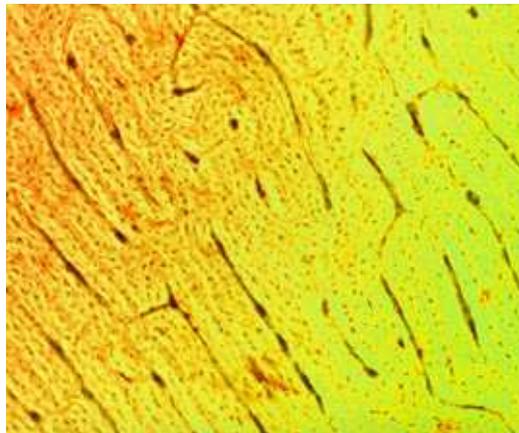


Original

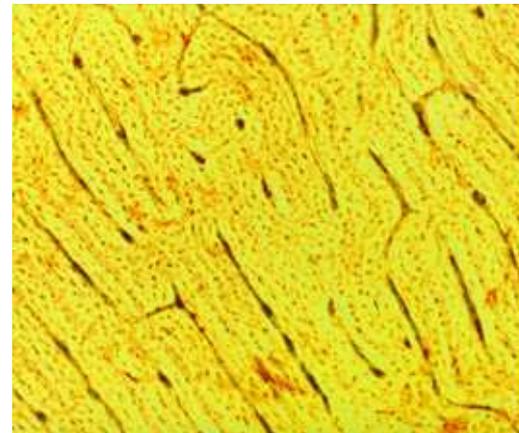


Nach Korrektur

**Beispiel:** Uneinheitlicher Farbton (siehe Kapitel Farbbilder)



Original



Nach Korrektur

# Separierbare Filter (1)

Separierbarer Filter erlaubt die Ausführung der Filteroperation als eine Kette zweier Filter mit jeweils einer 1D Maske

Filterung mit 2D Maske:

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \quad f = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 4 & 5 & 7 & 1 \\ 1 & 2 & 3 & 5 & 1 \\ 1 & 7 & 2 & 4 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$g = h \star f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 19 & 29 & 25 & 0 \\ 0 & 26 & 39 & 29 & 0 \\ 0 & 19 & 26 & 19 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{Randpixel auf null gesetzt})$$

## Separierbare Filter (2)

Filterung als eine Kette zweier 1D Filter:

● Schritt 1:  $g_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \star f = \begin{bmatrix} 0 & 3 & 3 & 3 & 0 \\ 0 & 10 & 16 & 13 & 0 \\ 0 & 6 & 10 & 9 & 0 \\ 0 & 10 & 13 & 7 & 0 \\ 0 & 3 & 3 & 3 & 0 \end{bmatrix}$

● Schritt 2:  $g = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \star g_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 19 & 29 & 25 & 0 \\ 0 & 26 & 39 & 29 & 0 \\ 0 & 19 & 26 & 19 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Beispiele von separierbaren Filtern sind Mittelwertfilter, Gaussfilter, uvm

Separierbare Filter der Größe  $n \times n$ :

Rechenaufwand (Multiplikationen, Additionen) von  $MNn^2$  auf  $2MNn$

# Entwerfen diskreter Filter (1)

Filter häufig durch kontinuierliche Funktion definiert; Umsetzung in diskrete Maske erforderlich

**Beispiel:** 1D Gauss-Filter (2D Gauss-Filter ist separierbar)

$$g(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}}$$

Adequate Wahl der Maskengröße  $2k + 1$ :

$$2\sigma \leq k \leq 3\sigma$$

Gauss-Filter mit real-Zahlen:

$$\begin{aligned} G^*(x, \sigma) &= [g(-k), g(-k+1), \dots, g(-1), g(0), g(1), \dots, g(k-1), g(k)] \\ &= [g(k), g(k-1), \dots, g(1), g(0), g(1), \dots, g(k-1), g(k)] \end{aligned}$$

## Entwerfen diskreter Filter (2)

Gauss-Filter mit integer-Zahlen:

$$G(x, \sigma) = \frac{1}{\text{NormFac}} \cdot [1, \text{round}\left(\frac{g(k-1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(1)}{g(k)}\right), \\ \text{round}\left(\frac{g(0)}{g(k)}\right), \text{round}\left(\frac{g(1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(k-1)}{g(k)}\right), 1]$$

Normierungsfaktor:  $\text{NormFac} = 2 + \text{round}\left(\frac{g(0)}{g(k)}\right) + 2 \cdot \sum_{i=1}^{k-1} \text{round}\left(\frac{g(i)}{g(k)}\right)$

⇒ Summe aller Maskenelemente ist eins!

## Entwerfen diskreter Filter (2)

Gauss-Filter mit integer-Zahlen:

$$G(x, \sigma) = \frac{1}{\text{NormFac}} \cdot [1, \text{round}\left(\frac{g(k-1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(1)}{g(k)}\right), \text{round}\left(\frac{g(0)}{g(k)}\right), \text{round}\left(\frac{g(1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(k-1)}{g(k)}\right), 1]$$

Normierungsfaktor:  $\text{NormFac} = 2 + \text{round}\left(\frac{g(0)}{g(k)}\right) + 2 \cdot \sum_{i=1}^{k-1} \text{round}\left(\frac{g(i)}{g(k)}\right)$

⇒ Summe aller Maskenelemente ist eins!

**Beispiel:**  $\sigma = 1.2$ ,  $k = 2.5 \cdot \sigma = 3$  (i.A. Rundung nötig)

$$G^*(x, \sigma) = [0.015, 0.083, 0.235, 0.332, 0.235, 0.083, 0.015]$$

$$G(x, \sigma) = \frac{1}{69} \cdot [1, 6, 16, 23, 16, 6, 1]$$

## Entwerfen diskreter Filter (2)

Gauss-Filter mit integer-Zahlen:

$$G(x, \sigma) = \frac{1}{\text{NormFac}} \cdot [1, \text{round}\left(\frac{g(k-1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(1)}{g(k)}\right), \text{round}\left(\frac{g(0)}{g(k)}\right), \text{round}\left(\frac{g(1)}{g(k)}\right), \dots, \text{round}\left(\frac{g(k-1)}{g(k)}\right), 1]$$

Normierungsfaktor:  $\text{NormFac} = 2 + \text{round}\left(\frac{g(0)}{g(k)}\right) + 2 \cdot \sum_{i=1}^{k-1} \text{round}\left(\frac{g(i)}{g(k)}\right)$

⇒ Summe aller Maskenelemente ist eins!

**Beispiel:**  $\sigma = 1.2$ ,  $k = 2.5 \cdot \sigma = 3$  (i.A. Rundung nötig)

$$G^*(x, \sigma) = [0.015, 0.083, 0.235, 0.332, 0.235, 0.083, 0.015]$$

$$G(x, \sigma) = \frac{1}{69} \cdot [1, 6, 16, 23, 16, 6, 1]$$

**Quiz:** I.A. ist die Summe der Maskenelemente für alle Glättungsmasken immer eins. Warum?