

# Intelligente Sehsysteme

**Fortsetzung:**

## **1 Einführung & Histogrammbasierte Bildverarbeitung**

Organisatorische & thematische Einführung

Histogrammbasierte Bildverbesserungen

Entropiemaximierende Bildverbesserungen

*Henry Hölzemann*

# Nachtrag: Tutorienvergabe

---

- Die Vergabe der Tutorien findet durch das TVS bis 21. Oktober um 18 Uhr statt (siehe eCampus)
- Die Programmieraufgaben der Übungsblätter bauen erstmalig auf Python auf, nicht mehr auf der ImageToolBox und Java
- Die Abgabe der Übungen erfolgt in Gruppen von drei Studierenden
- Die Übungsblätter werden jeweils Dienstag nach der VL auf eCampus bereitgestellt
- Die Abgaben sind jeweils bis Sonntag, 12 Uhr, einzureichen
- 50% der Gesamtpunkte zur Zulassung erforderlich

# Lineare Grauwertspreizung

Die **lineare Grauwertspreizung** bzw. Intensitätsspreizung optimiert den globalen Kontrast  $C_{\text{global}}$  eines Einkanalbildes mit der folg. linearen Transferfunktion  
 $T(I) = (I + c_1) \cdot c_2$ :

- $c_1 = -I_{\min\text{Given}}$

- $c_2 = I_{\max} / (I_{\max\text{Given}} - I_{\min\text{Given}})$

$$\rightsquigarrow T(I) = [ (I - I_{\min\text{Given}}) \cdot (I_{\max} / (I_{\max\text{Given}} - I_{\min\text{Given}})) ] ^*$$

mit maximalen und minimalen Intensitätswerten  $I_{\max\text{Given}}$ ,  $I_{\min\text{Given}}$  im geg. Bild  $I[x,y]$  und maximal darstellbarem Intensitätswert  $I_{\max}$  \*\*

Wenn  $I_{\min} = 0$ . Bei  $I_{\min} > 0$ :

$$c_1 = I_{\min} - I_{\min\text{Given}}$$

$$c_2 = (I_{\max} - I_{\min}) / (I_{\max\text{Given}} - I_{\min\text{Given}}).$$



Bildquelle: Abteilung Fernerkundung der Univ.Trier: Kursbegleitung *Digitale Bildbearbeitung*

\* Äußerste Klammer steht für (kaufmännische) Rundung auf die nächstliegende ganze Zahl

\*\* Z.B.  $I_{\max} = 255$  für 1-Byte-Grauwertbilder

# Lokaler Kontrast (1)

---

- Die Grauwertspreizung berücksichtigt nur den globalen Kontrast  $C_{\text{global}}$ , nicht aber die lokale Verteilung der Intensitäten im Bild
  - ↪ zwei Pixel  $p_1$  und  $p_2$  mit  $I[p_1] = I_{\min}$  und  $I[p_2] = I_{\max}$  reichen aus, um eine Grauwertspreizung zu verhindern – dennoch kann ein Großteil der Pixel ähnliche Intensitäten zeigen und das gesamte Bild daher kontrastarm sein
  - ↪ Der lokale Kontrast  $C_{\text{local}}$  wird als weitere Kenngröße definiert. Für ein Einkanalbild  $I[I(x,y)]$  mit  $Z$  Zeilen und  $S$  Spalten:

$$C_{\text{local}}(I) = \frac{1}{Z \cdot S} \sum_{x=0}^{Z-1} \sum_{y=0}^{S-1} |I(x, y) - \bar{I}(x, y)|$$

mit  $\bar{I}$  für den durchschnittlichen Wert der Intensitätsfunktion  $I$  in einer lokalen Nachbarschaft (z.B. der 8-Nachbarschaft) um  $(x,y)$

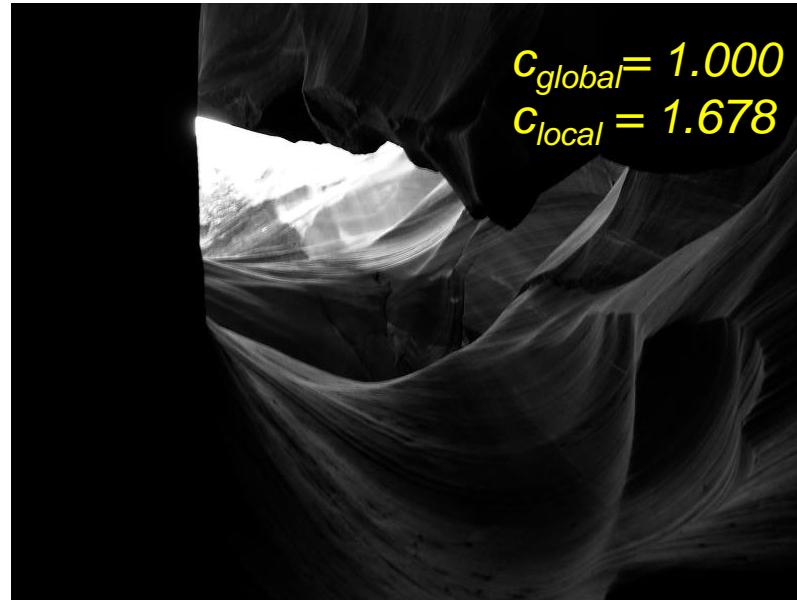
# Lokaler Kontrast (2)

---

Für ein Einkanalbild  $I[(x,y)]$  mit  $Z$  Zeilen und  $S$  Spalten:

$$C_{global} = (I_{maxGiven} - I_{minGiven}) / (I_{max} - I_{min})$$

$$C_{local}(I) = \frac{1}{Z \cdot S} \sum_{x=0}^{Z-1} \sum_{y=0}^{S-1} |I(x, y) - \bar{I}(x, y)|.$$



Bildquelle: Klaus Tönnies

## Lokaler Kontrast (3)

---

Nach Definition nimmt der lokale Kontrast  $C_{local}$  Bezug auf lokale Nachbarpixel.

Lokale Pixelnachbarschaften sind aber nicht in Histogrammen abgetragen!



↪ Somit scheint eine *histogrammbasierte* Verbesserung von  $C_{local}$  nicht möglich

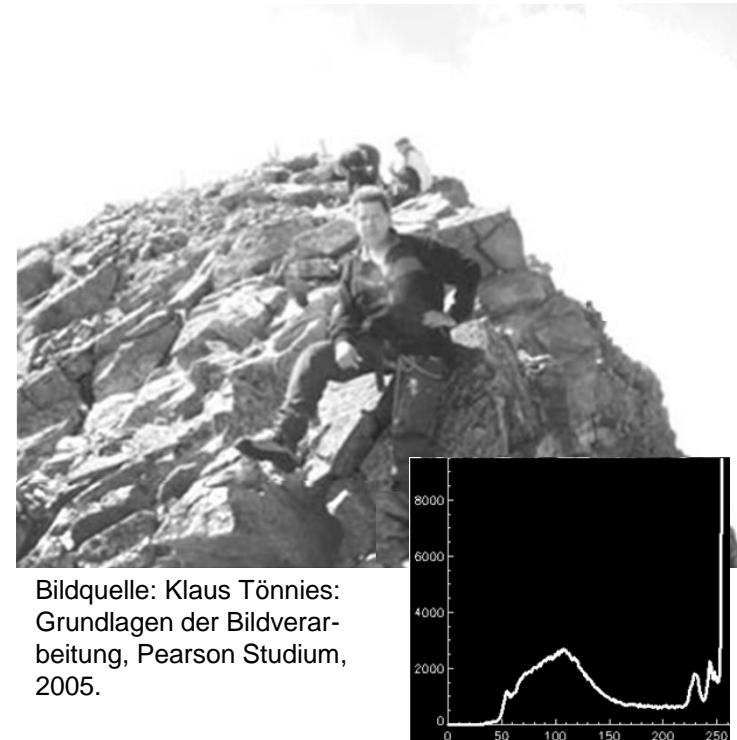
**Aber:**

Ausgangshypothese sei, dass Pixel mit häufig auftretenden Intensitätswerten auch häufig zueinander benachbart sind.

↪ Dann muss die Transferfunktion  $T(I)$  von der Häufigkeit der Intensitätswerte abhängig sein.

# Gamma-Korrektur (1)

- Annahme: die häufigsten Intensitätswerte finden sich entweder vollständig im niedrigen oder im hohen Wertebereich
  - ~ z.B. bei Unter- bzw. Überbelichtung
- Unter dieser Annahme kann die Gamma-Korrektur zu einer Verbesserung des lokalen Kontrastes führen
- Die **Gamma-Korrektur** ist eine nichtlineare Intensitäts- bzw. Grauwertspreizung auf  $I_{min}$  (i.A. = 0) bis  $I_{max}$  (mit  $N_G = I_{max} + 1$ ):



Bildquelle: Klaus Tönnies:  
Grundlagen der Bildverarbeitung, Pearson Studium,  
2005.

$$T_\gamma(I) = \left[ N_G \left( \frac{I - I_{min}}{I_{max} - I_{min}} \right)^\gamma + I_{min} \right]$$

\*

\* Äußerste Klammer steht für (kaufmännische) Rundung auf die nächstliegende ganze Zahl.  
Resultiert das Ergebnis mit  $N_G$ , so wird  $N_G$  ersetzt durch  $I_{max}$

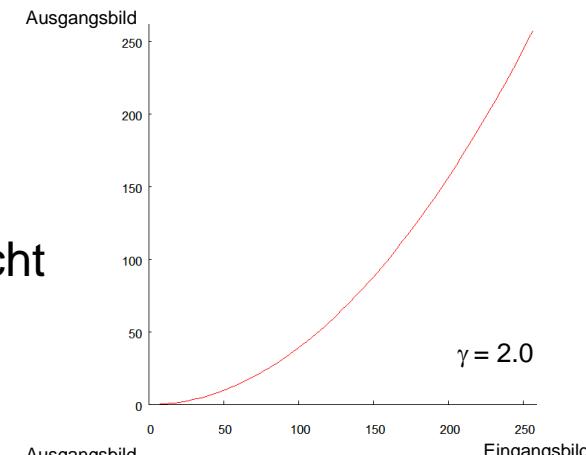
# Gamma-Korrektur (2)

## Anwendung der Gamma-Korrektur

$$T_{\gamma}(I) = \left[ N_G \left( \frac{I - I_{min}}{I_{max} - I_{min}} \right)^{\gamma} + I_{min} \right]$$

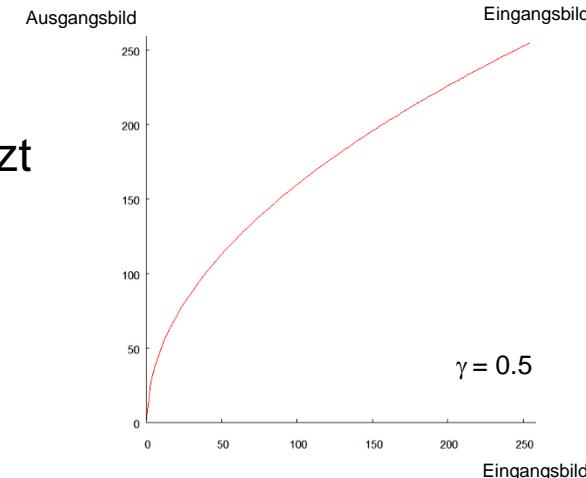
- Bei  $\gamma > 1$ :

- hohe Intensitätswerte werden gespreizt
- niedrige Intensitätswerte werden gestaucht
- ↗ Anwendung bei überbelichtetem Bild



- Bei  $\gamma < 1$ :

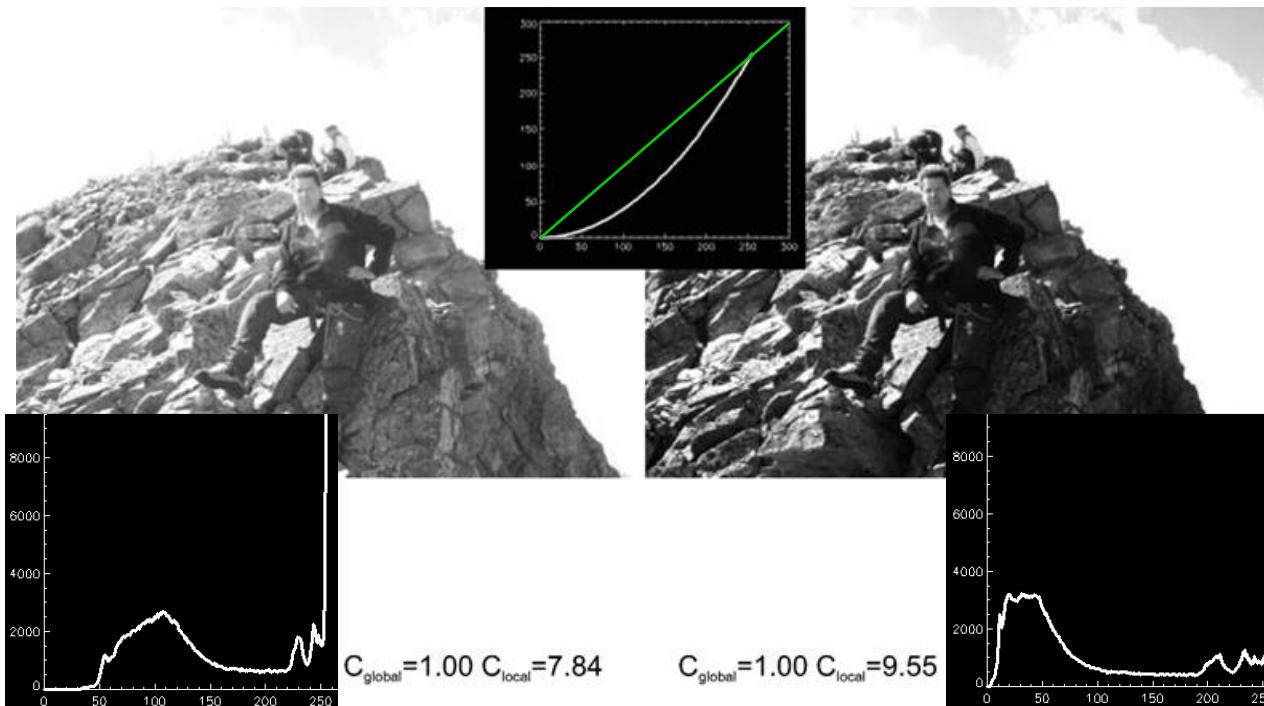
- niedrige Intensitätswerte werden gespreizt
- hohe Intensitätswerte werden gestaucht
- ↗ Anwendung bei unterbelichtetem Bild



# Gamma-Korrektur (3)

---

Bspl.: Gamma-Korrektur mit  $\gamma = 2$



Bildquelle: modifiziert nach Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Entropie von Einkanalbildern (1)

---

Die Entropie

- ist ein Maß für den mittleren Informationsgehalt eines Bildes
- aus dem normalisierten Histogramm ableitbar

Für ein Einkanalbild  $I = [I(x,y)]$

- mit Intensitätsspektrum  $\{0, \dots, I_{\max}\}$  und
- normalisierten Histogramm mit relativen Häufigkeiten  $p_I(I)$

ist die Entropie definiert als

$$H_I = - \sum_{I=0}^{I_{\max}} p_I(I) \cdot \log_2 p_I(I)$$

mit  $0 \cdot \log_2 0 = 0$

# Entropie von Einkanalbildern (2)

---

Beispiele:

$$\text{Geg.: } I = [I(x,y)], \quad H_I = - \sum_{I=0}^{I_{\max}} p_I(I) \cdot \log_2 p_I(I).$$

mit  $0 \cdot \log_2 0 = 0$

- homogenes Bild  $I = [I(x,y)] = I^*$ :

$$H = - \sum_{I=I^*} p_I(I) \cdot \log_2(p_I(I)) - \sum_{I \neq I^*} p_I(I) \cdot \log_2(p_I(I)) = -1 \cdot 0 - 0 = 0$$

- Zweipegelbild  $I = [I(x,y)]$  mit  $p_I(I_1) = 50\%$  und  $p_I(I_2) = 50\%$

$$H = - \sum_{I=I_1} p_I(I) \cdot \log_2(p_I(I)) - \sum_{I=I_2} p_I(I) \cdot \log_2(p_I(I)) = - \frac{1}{2} \cdot \log_2(\frac{1}{2}) - \frac{1}{2} \cdot \log_2(\frac{1}{2}) = 1$$

- gleich verteiltes Grauwertbild  $I = [I(x,y)]$ , z.B. mit  $p_I(I) = 1/256$  für  $I = 0, \dots, 255$

$$H = - \sum_{I=0, \dots, 255} p_I(I) \cdot \log_2(p_I(I)) = 256 \cdot (-1/256 \cdot \log_2(1/256)) = -1 \cdot -8 = 8$$



# Maximierung der Entropie (1)

---

Die Entropie  $H_I = -\sum_{I=0}^{I_{\max}} p_I(I) \cdot \log_2 p_I(I)$

- bildet neben globalem und lokalem Kontrast ein drittes Maß für den Kontrast
- eignet sich zur Spreizung von Intensitätswerten entsprechend ihrer Häufigkeit

Der Zusammenhang zwischen Entropie und Kontrast verwundert nicht, da

- die Entropie ein Maß für die Bildinformation darstellt ...
- ... und Kontrast für die Wahrnehmbarkeit von Objekten wichtig ist

## Maximierung der Entropie (2)

---

Die Kontraststeigerung basiert jetzt also auf einer Maximierung der Entropie

$$H_I = - \sum_{I=0}^{I_{\max}} p_I(I) \cdot \log_2 p_I(I).$$

Die Entropie ist maximal, wenn die Häufigkeitseinträge für alle Intensitätswerte denselben konstanten Wert zeigen (vgl. Folie 42)



## Maximierung der Entropie (3)

---

Die entspr. Transferfunktion ist einfacher ableitbar unter der Annahme, dass die Intensitätswerte *kontinuierlich* auf das Intervall  $[0,1]$  skaliert sind

Dann folgt:

- 1) die Gesamtanzahl  $N$  aller Bildpixel ist das Integral über dem Histogramm
- 2) Für ein gleichverteiltes Histogramm ( $h_I(I) = N/I_{\max}$ ) ist nach der Normalisierung das Integral von 0 bis zu einem beliebigen Intensitätswert  $I$  gerade  $I$  selbst:

$$\int_0^I p_I(i) di = \frac{1}{N} \cdot h_I(I) \cdot I = \frac{1}{N} \cdot \frac{N}{I_{\max}} \cdot I = I, \text{ da } I_{\max} = 1.$$

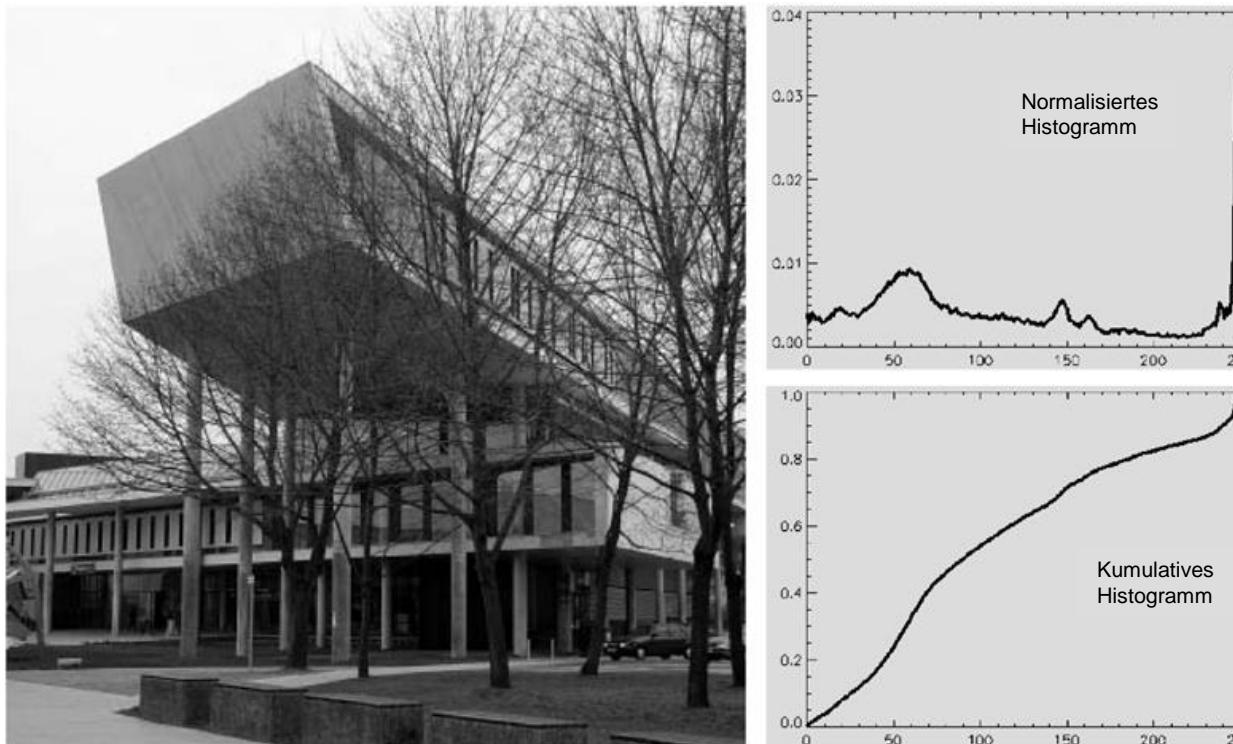
Um dies aus dem ursprünglichen Histogramm abzuleiten, muss diese Bedingung durch ebendiese Transferfunktion erzwungen werden:

$$T_H(I) = \int_0^I p_I(i) di.$$



## Maximierung der Entropie (4)

Diese Transferfunktion  $T_H(I) = \int_0^I p_I(i) di$  entspricht also einer kontinuierlichen Variante des kumulativen Histogramms und bewirkt eine Spreizung für besonders häufig vorkommende Intensitätswerte, weil das Integral mit steigendem  $I$  dort besonders rasch zunimmt.



Beispiel: Um die Entropie zu maximieren, wird das kumulative Histogramm (unten rechts) als Transferfunktion verwendet.

Bildquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005 (Bild von K. Rink).

# Maximierung der Entropie (5)

---

- Für die tatsächlichen Grauwerte
  - ist die Skalierung auf [0,1] rückgängig zu machen, indem mit der tatsächl. Anzahl  $N_G$  ( $= I_{\max} + 1$ ) der Grauwerte multipliziert wird
  - ist die Integration wegen der Rasterung durch eine Summation zu ersetzen
  - sind die resultierenden reellen Intensitätswerte auf ganze Zahlen abzubilden:

$$T_H(I) = \left[ N_G \cdot \sum_{i=0}^I p_I(i) \right]^*$$

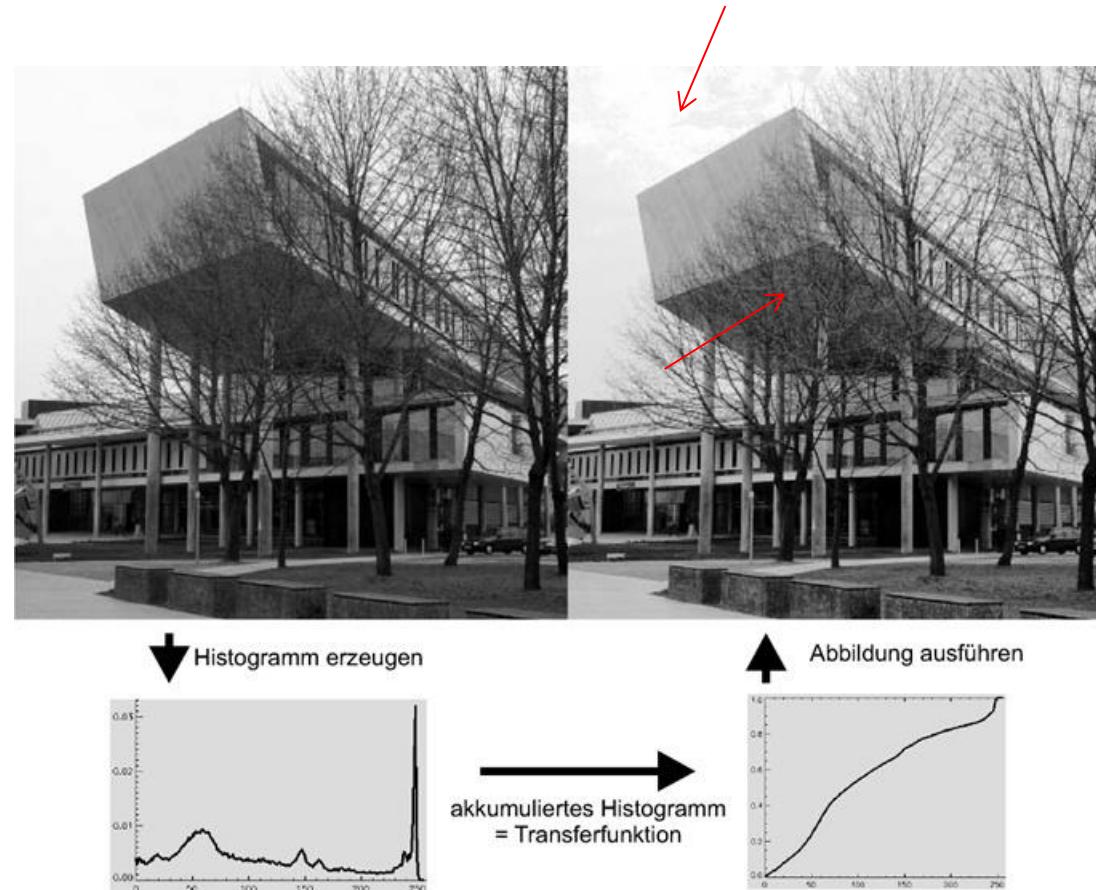
- Diese Operation heißt Histogrammlinearisation (engl. histogram equalisation)
  - das Ergebnis ist kein konstantes Histogramm, da die Summation die Integration nur approximiert
  - vielmehr kommt es zu einer Umverteilung im Histogramm: sehr häufige Intensitätswerte werden gespreizt, selten auftretende Werte werden zusammen gestaucht (z.T. in einzelne Werte überführt)

---

\* Äußerste Klammer steht für (kaufmännische) Rundung auf die nächstliegende ganze Zahl.  
Resultiert das Ergebnis mit  $N_G$ , so wird  $N_G$  ersetzt durch  $I_{\max}$

# Maximierung der Entropie (6)

Beispiel für die Histogrammlinearialisierung:



Bildquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005 (Bild von K. Rink).

# Intelligente Sehsysteme

## 2 Bildstörungen und lineare Operatoren

Deterministische und stochastische Einflüsse,  
verschiebungsinvariante, lineare Operatoren,  
Konvolution, Bewegungsunschärfe, Rauschen,  
lineare Glättungsfilter

*Henry Hölzemann*

- Deterministische Einflüsse und stochastische Einflüsse bei der Bildaufnahme
  - Photonenrauschen
  - Signal-Noise-Ratio
  - Impulsrauschen
- Lineare Filter zur Glättung
  - Mittelwertfilter
  - Gauß-Filter
  - Binomialfilter

# Einflüsse bei der Bildaufnahme

---

Die Bildaufnahme geht i.A. mit **deterministischen und stochastischen Einflüssen** auf die abgebildete Information einher:

- **Deterministische Einflüsse** basieren auf berechenbaren und wiederholbaren Einwirkungen bei der Bildaufnahme
  - ~ Ziel ist hier, die Einwirkungen zu beschreiben, ggf. geeignet zu parametrisieren und rückgängig zu machen
- **Stochastische Einflüsse**, verändern **das Bild in statistisch beschreibbarer Weise**
  - ~ Ziel ist hier, die Parameter der statistischen Beschreibung zu ermitteln, damit der Einfluss bei der Interpretation berücksichtigt und näherungsweise rückgängig gemacht werden kann

# Lineare Operatoren (1)

---

Viele determinist. Störungen sind sich durch lineare Operatoren beschreibbar:

- Ein Operator  $O(\cdot)$  heißt linear, wenn für Funktionen  $f$  und  $g$  sowie  $\lambda \in \mathbb{R}$  die Homogenität und Additivität gelten:
  - Homogenität:  $O(\lambda \cdot f) = \lambda \cdot O(f)$
  - Additivität:  $O(f + g) = O(f) + O(g)$
- Daraus folgt:
  - 1) ein linearer Operator ist unabhängig von den Funktionswerten
  - 2) ein linearer Operator kann nur die **gewichtete Summe aller Funktionswerte** sein kann

# Lineare Operatoren (2)

---

Damit ist jeder lineare Operator als lineares Gleichungssystem formulierbar:

$$\mathbf{y} = \mathbf{A} \mathbf{x}^T$$

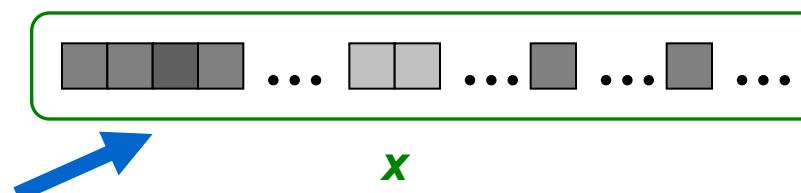
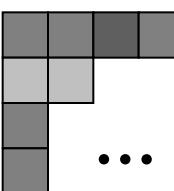
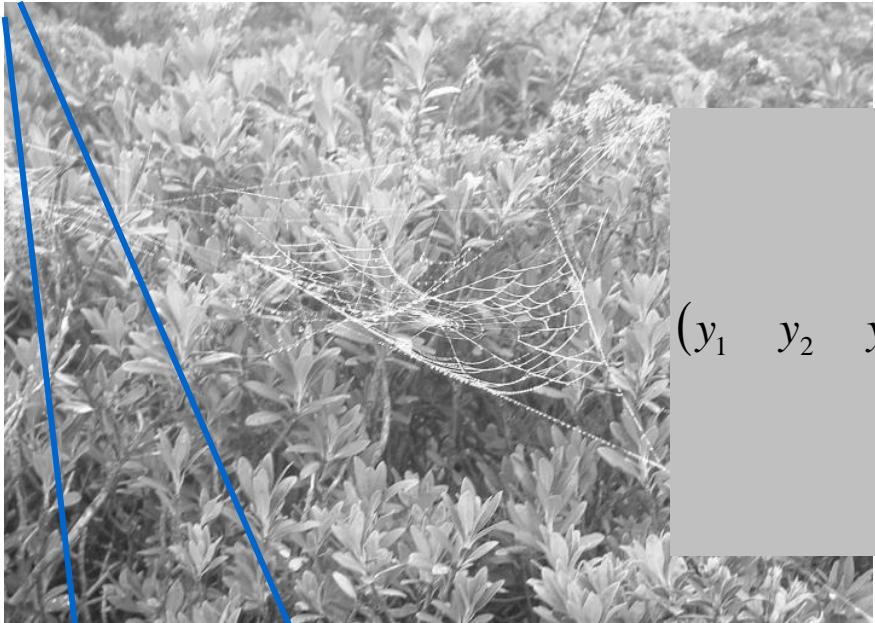
mit

- 1) Vektor  $\mathbf{x}$  der hintereinander geschriebenen Pixel des **ungestörten Bildes**
  - 2) Vektor  $\mathbf{y}$  der hintereinander geschriebenen Pixel des **gestörten Bildes**
  - 3) quadratischer Matrix  $\mathbf{A}$ , welche die **Störung** kodiert
- ↗ eine durch einen linearen Operator beschreibbare Störung ist **invertierbar**, wenn die Determinante von  $\mathbf{A}$  von Null verschieden ist

# Lineare Operatoren (3)

Damit ist jeder lineare Operator als lineares Gleichungssystem formulierbar:

$$\mathbf{y} = \mathbf{A} \mathbf{x}^T.$$



$$(y_1 \quad y_2 \quad y_3 \quad \dots \quad y_M) = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,M} \\ a_{2,1} & a_{2,2} & a_{2,3} & & \\ a_{3,1} & a_{3,2} & a_{3,3} & & \\ \dots & & & & \dots \\ a_{M,1} & & & & a_{M,M} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_M \end{pmatrix}$$

# Verschiebungsinvariante, lineare Operatoren (1)

---

- Bei linearen *verschiebungsinvarianten* Operatoren  $O(\cdot)$  sind die Operatorergebnisse zusätzlich vom Ort der Operatoranwendung unabhängig:

$$O(S_{s,t}(f)) = S_{s,t}(O(f))$$

für Verschiebung  $S_{s,t}$  mit  $(s,t) \in \mathbb{Z}^2$ . M.a.W.: verschieben wir ein Bild, ist das Ausgabebild identisch (mit Ausnahme der Verschiebung).

- Viele Einflüsse bei der Bildaufnahme sind unabhängig vom Ort der Pixel im Bild und daher durch verschiebungsinvariante, lineare Operatoren beschreibbar.

Beispiele:

- 1) gleichmäßige Bildunschärfe durch Defokussierung
- 2) gleichmäßige Bildunschärfe durch Kamerabewegung  
(Bewegungsunschärfe)

## Verschiebungsinvariante, lineare Operatoren (2)

---

Ein weiterer Vorteil:

- Ein verschiebungsinvarianter, lineare Operator ist auch dann bestimmbar, wenn die Ursache des Einflusses nicht bekannt ist
- Daher sind verschiebungsinvariante, lineare Operatoren für die Modellierung bzw. Beseitigung von Störungen sehr attraktiv

# Konvolution

---

Verschiebungsinvariante, lineare Operatoren werden durch die **Konvolution** oder **Faltung** operationalisiert:

$$(f * g)(x,y) = \sum_{u \in D} \sum_{v \in D} f(u,v) \cdot g(x-u, y-v)$$

mit Menge  $D$  von ganzzahligen Verschiebungsvektoren  $(u,v)$

- Funktion  $f$  heißt **Konvolutionsfunktion** oder **Faltungsfunktion**
- Darstellung der **Konvolution** durch Operatorsymbol „\*“
- Konvolutionsergebnis  $(f * g)(x,y)$  ist also die gewichtete Summe der **Funktionswerte**  $g(x-u, y-v)$  mit den Gewichten der nach  $(x,y)$  verschobenen **Konvolutionsfunktion**  $f$

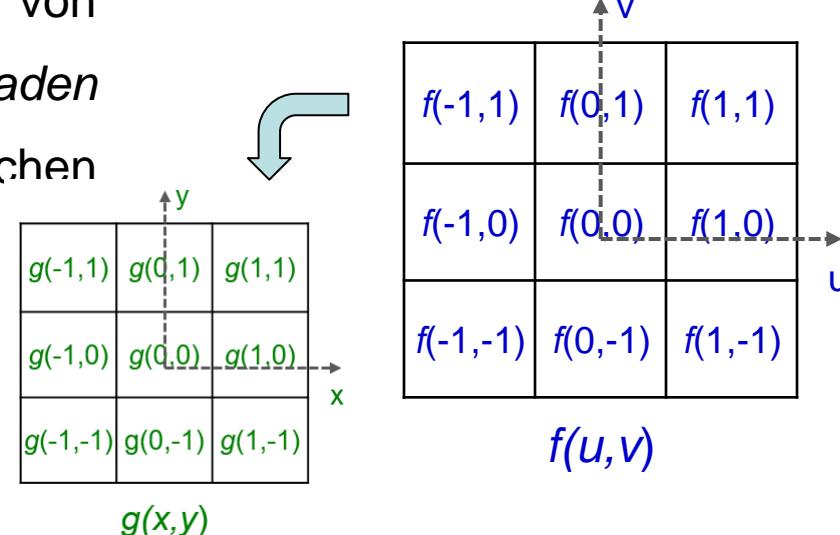
# Konvolution im Computersehen

Bei der Anwendung in der Bildverarbeitung

- wird die Konvolution i. A. umgesetzt als

$$(f * g)(x,y) = \sum_u \sum_v f(u,v) \cdot g(x-u,y-v) \text{ mit } u,v = -(m-1)/2, \dots, (m-1)/2$$

- steht die **Funktion  $g(x,y)$**  für die **Grau- bzw. Intensitätswerte der Bildmatrix**
- definieren die Verschiebungsvektoren  $(u,v)$  einen begrenzten Argumentsbereich der **Konvolutionsfunktion  $f$**  von **quadratischer Form mit ungeraden und in Null zentrierten Wertebereichen**  
(im Bsp. für  $m = 3$ )



# Konvolutionsmaske

---

Die diskrete und begrenzte *Konvolutionsfunktion*  $f(u,v)$  mit  $u,v = -(m-1)/2, \dots, (m-1)/2$  ist also als Matrix von Konvolutions- bzw. Gewichtswerten darstellbar:

- diese werden als **Konvolutions- bzw. Faltungsmasken** oder **Konvolutions- bzw. Faltungskerne** bezeichnet
- der Parameter  $m$  wird als **Größe der Konvolutionsmaske** bezeichnet

Das Beispiel zeigt eine Konvolutionsmaske der Größe  $m = 3$  mit den ganzzahligen Argumentwerten -1, 0, 1 für  $u$  und  $v$

A 3x3 grid representing a convolution mask  $f(u,v)$ . The grid has three columns labeled  $u = -1, 0, 1$  and three rows labeled  $v = -1, 0, 1$ . The values in the grid are:  $f(-1,1), f(0,1), f(1,1)$  in the top row;  $f(-1,0), f(0,0), f(1,0)$  in the middle row; and  $f(-1,-1), f(0,-1), f(1,-1)$  in the bottom row. A dashed line connects the center cell  $f(0,0)$  to the origin of both the  $u$  and  $v$  axes.

$f(-1,1)$	$f(0,1)$	$f(1,1)$
$f(-1,0)$	$f(0,0)$	$f(1,0)$
$f(-1,-1)$	$f(0,-1)$	$f(1,-1)$

$f(u,v)$

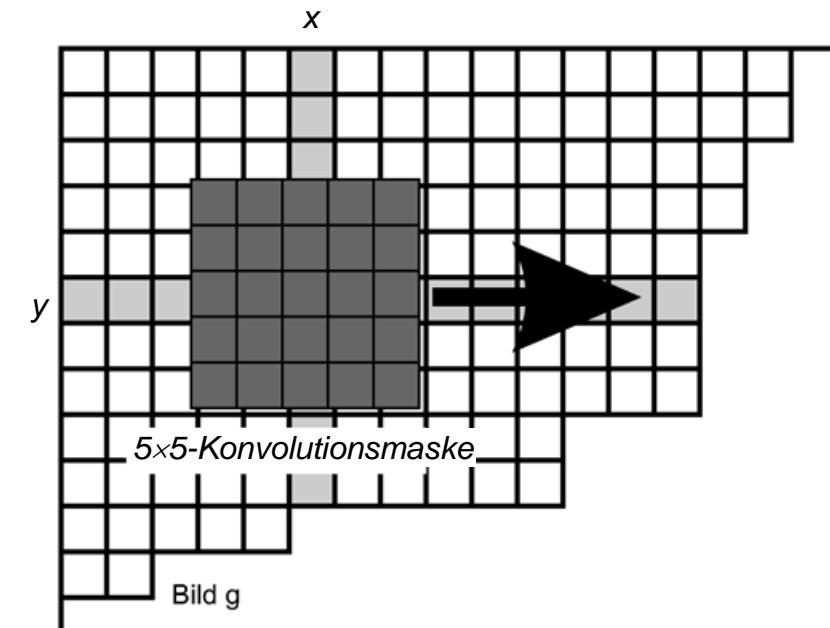
# Anwendung der Konvolution

---

- Die Konvolution  $(f * g)(x,y)$  wird i. A. über zwei geschachtelte Schleifen (engl. *nested loops*) auf alle Pixel eines Bildes abgewendet:

```
for (int x = 0; x < height; x++) {  
    for (int y = 0; y < width; y++) { ...
```

- Für jedes Pixel  $(x,y)$  mit Grauwert  $g(x,y)$  wird das **Konvolutionsergebnis**  $(f * g)(x,y)$  ermittelt \*
- Das Bspl. zeigt eine Konvolutionsmaske der Größe  $m=5$



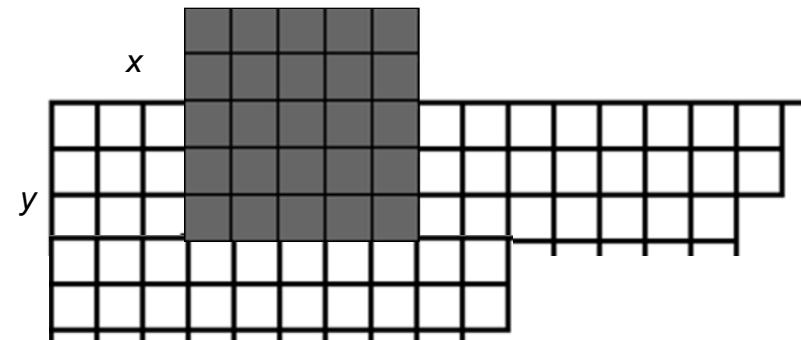
---

\* Die Ergebnisse  $(f * g)(x,y)$  werden i.A. in separates Ausgabebild geschrieben

# Randbehandlung bei der Konvolution

---

- Die naive Anwendung der Konvolution  $(f * g)(x,y)$  führt an den Bildrändern zu undefinierten Intensitätswerten jenseits der Bildränder
- Vier gängige heuristische Lösungen:
  - 1) Speziallösungen für Randpixel, die nur die definierten Anteile der Konvolution nutzen
  - 2) Erweiterung der Bilder durch Kopieren der Randwerte jenseits der Ränder
  - 3) Erweiterung der Bilder durch Spiegelung der Randwerte jenseits der Ränder
  - 4) Vernachlässigung der Randpixel, die zu Ergebnisbildern führt, die in Höhe und Breite um die Größe der Konvolutionsmaske verkleinert sind



Nach Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Eigenschaften der Konvolution

---

Neben Linearität und Verschiebungsinvarianz gilt für die Konvolution:

- 1) die Kommutativität:  $[f_1 * f_2](x,y) = [f_2 * f_1](x,y)$ ,
- 2) die Assoziativität:  $f_1 * ([f_2 * f_3](x,y)) = [f_1 * f_2](f_3(x,y))$

für Operatoren  $f_1, f_2, f_3$ .

Die Assoziativität ermöglicht effiziente Ausführung: anstelle von mehreren sukzessive auszuführenden Konvolutionen  $f_1, f_2, \dots, f_k$  auf ein Bild  $I = [I(x,y)]$  kann ein Operator  $f = f_1 * f_2 * \dots * f_k$  erzeugt werden und auf Bild  $I$  angewandt werden.

# Beispl. für determin. Einfluss (1)

---

Störungen durch **Bewegungsunschärfe** entstehen, wenn die Kamera während der Belichtungszeit relativ zur beobachteten Szene bewegt wird.

Beispiel: eine ungestörte Aufnahme im Vergleich zu einer Aufnahme, bei der die Kamera in einem Winkel von  $25^\circ$  zur x-Achse des Kamerakoordinatensystems bewegt wurde



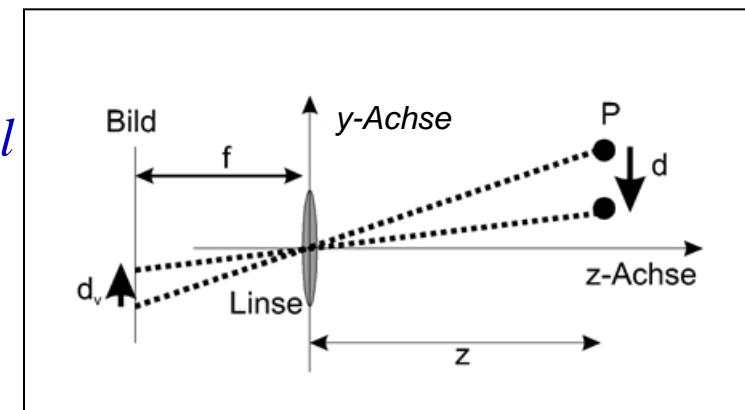
## Beispl. für determin. Einfluss (2)

- Annahme: Bildfläche der digitalen Kamera wird für 30 ms belichtet, die Kamera wird in dieser Zeitspanne um  $d = 2,5 \text{ cm}$  in y-Richtung verschoben
- Der Einfachheit halber sollen sich alle Szenenobjekte im Abstand  $z = 1 \text{ m}$  zum Linsenzentrum befinden. Die Brennweite der Linse betrage  $f = 8 \text{ mm}$ . Die Pixelgröße sei  $0,04 \text{ mm} \times 0,04 \text{ mm}$ .
- Es folgt eine Bewegungsunschärfe durch die Verschiebung  $d_v$  (s. Abb.):

Aus  $\frac{d_v}{d} = \frac{f}{z}$  folgt:

$$d_v = d \cdot \frac{f}{z} = 25 \cdot \frac{8}{10^3} = 0,2 \text{ mm} = 5 \text{ Pixel}$$

↗ dieselbe Lichtenergie, die ohne Verschiebung auf ein Pixel fiele, wird nun auf 5 Pixel verteilt



Nach Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Beispl. für determin. Einfluss (3)

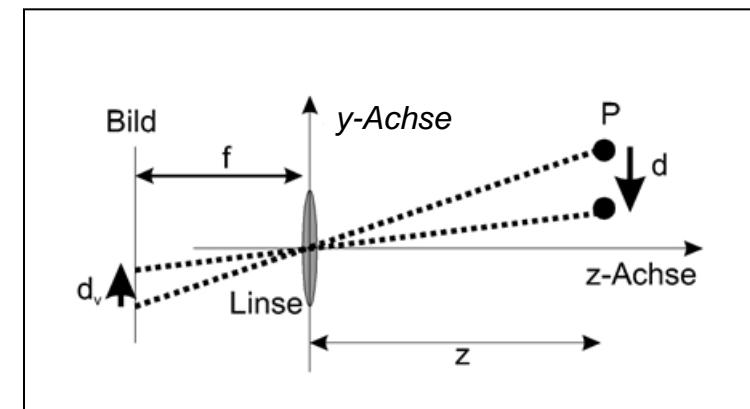
Die Verteilung der Lichtenergie eines Pixels auf 5 Pixel in y-Richtung führt zu folg. Konvolutionsfunktion als Modell dieser Störung einer Bewegungsunschärfe:

$$f(u,v) = \begin{cases} 0,2 & \text{für } -2 \leq v \leq 2 \wedge u = 0 \\ 0 & \text{sonst} \end{cases}$$

mit  $m = 5$  und  $u,v = -(m-1)/2, \dots, (m-1)/2$

bzw.

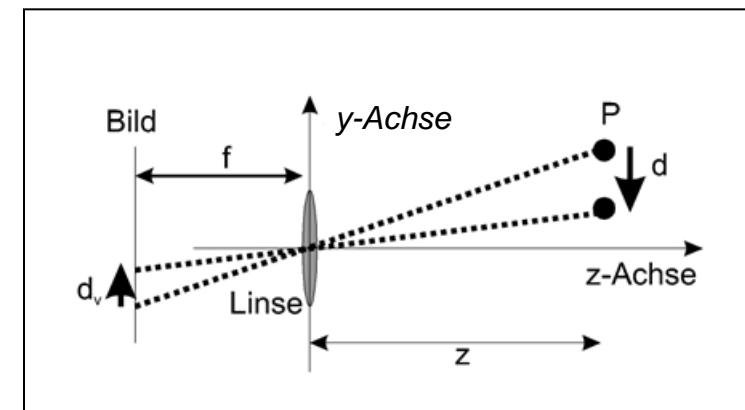
$$f(u,v) = \begin{bmatrix} 0 & 0 & 0,2 & 0 & 0 \\ 0 & 0 & 0,2 & 0 & 0 \\ 0 & 0 & 0,2 & 0 & 0 \\ 0 & 0 & 0,2 & 0 & 0 \\ 0 & 0 & 0,2 & 0 & 0 \end{bmatrix}$$



Nach Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Beispl. für determin. Einfluss (4)

- Die Funktion gilt aber auch approximativ für Szenenpunkte, die geringfügig kleinere oder größere Abstände als 1 m zur Kamera aufweisen
- Damit sind Bewegungsunschärfen für Szenarien mit kleinen Distanzunterschiede zwischen den abgebildeten Szeneobjekten modellierbar
- Zur Unterdrückung von (Bewegungs)-Unschärfen wird die **inverse Filterung** durch das **Wiener-Filter** umgesetzt, das den Fehler zwischen ungestörtem und gestörtem Bild minimiert
  - ↪ K. Tönnies: *Grundlagen der Bildverarbeitung*, 121ff, Pearson Studium, 2005. R. Gonzales, R. Woods: *Digital Image Processing*, 3<sup>rd</sup> Ed., 351 ff., Pearson Education, 2008



Nach Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Stochastische Einflüsse (1)

---

- Stochastische Einflüsse sind nicht deterministisch wiederholbar. Sie sind aber über Wahrscheinlichkeitsverteilungen beschreibbar
- Für Kameraaufnahmen ist insbes. das durch **Quantenrauschen** verursachte Bildrauschen von Interesse
- Prinzipiell wird eine **geradlinige Ausbreitung** von **Photonen** entspr. der **Strahlenoptik** angenommen
- **Quanteneffekte** bedingen eine teilweise Streuung von Photonen, die dann nicht an „der richtigen Stelle“ der Bildfläche auftreffen
- Dieser Effekt wird probabilistisch durch W'verteilungen beschrieben



Verrausches Bild einer Digitalkamera

Bildquelle: [http://en.wikipedia.org/  
wiki/Image\\_noise](http://en.wikipedia.org/wiki/Image_noise) (24/10/2011)

# Stochastische Einflüsse (2)

---

- Das **Photonenrauschen** röhrt i.W. aus Poisson-verteiltem Quantenrauschen und ist für hohe Photonenzahlen **durch eine Gauß-Verteilung approximierbar**

↗ Die Aufnahme eines Bildes  $I = [I(x,y)]$  wird als **additiver Überlagerungsprozess von** einem **Signal** und **Rauschen** modelliert:

- Signal  $I_u$  = regulär verhaltende Photonen
- Rauschen  $\eta$  = zufallsverteilte Störung

↗ Damit gilt für die resultierende Intensität  $I(x,y)$  an einer Bildkoordinate  $(x,y)$ :

$$I(x,y) = I_u(x,y) + \eta(x,y)$$

mit  $\eta(x,y)$  = Gaußsche Normalverteilung mit Erwartungswert Null

# Stochastische Einflüsse (4)

---

Die *Schätzung* des normalverteilten Rauscheinflusses erfolgt durch Bestimmung der *Varianz*  $\sigma^2$  und basiert auf einer hinreichend großen Zahl von Stichproben mit bekanntem Erwartungswert:

Man nehme einen *homogenen Bildbereich*  $B$  mit Pixeln  $p$ . Deren ungestörter Intensitätswert sei  $I_u(p) = I'$ . Die Varianz  $\sigma^2$  ist dann schätzbar durch die *korrigierte Stichprobenvarianz*

$$\sigma^2 = \frac{1}{|B|-1} \cdot \sum_{p \in B} (I(p) - I')^2$$

*Erwartungstreue  
Schätzung der Varianz*

Dieser Wert wird für das gesamte Bild benutzt unter der Annahme, dass das Rauschen dieselbe Charakteristik über das gesamte Bild zeigt.

# Stochastische Einflüsse (5)

---

Das Verhältnis zwischen Signalstärke und durchschnittl. Stärke des Rauschens wird als **Signal-Rausch-Verhältnis** (engl. **signal-to-noise ratio (SNR)**) bezeichnet.

Das SNR ist bei geg. Varianz  $\sigma^2$  wie folgt bestimmbar:

Ist der Bildinhalt unbekannt,

- wird das Hintergrundsignal mit Null und das maximale Signal durch den höchsten im Bild auftretenden Intensitätswert  $I_{\text{max\_given}}$  geschätzt:

$$\text{SNR}_{\text{max}}(I) = I_{\text{max\_given}}/\sigma$$

- Alternativ ist der mittlere Intensitätswert als Signalschätzung nutzbar:

$$\text{SNR}_{\text{avg}}(I) = (1/S \cdot Z) \sum_{x=0, \dots, S-1} \sum_{y=0, \dots, Z-1} I(x,y)/\sigma$$

# Stochastische Einflüsse (6)

---

Das Verhältnis zwischen Signalstärke und durchschnittl. Stärke des Rauschens wird als Signal-Rausch-Verhältnis (engl. signal-to-noise ratio (SNR)) bezeichnet.

Das SNR ist bei geg. Varianz  $\sigma^2$  wie folgt bestimmbar:

- Ist der Bildinhalt bekannt und damit das tatsächliche Signal (abgebildete relevante Objekte) vom tatsächl. Hintergrund unterscheidbar, kann für  $\text{SNR}_{\text{OB}}$  (**O**bject/**B**ackground) im Zähler der tatsächliche Abstand (Betrag) zw. gemittelten Signal und gemittelten Hintergrund stehen.

# Stochastische Einflüsse (7)

---

Die dargestellten SNR-Varianten machen deutlich:

Bei der Nennung eines SNR-Wertes muss  
dessen Herleitung genau spezifiziert werden.

Da die Objekt- bzw. Signalpixel nicht die hellsten Intensitäten zeigen müssen, kann z.B.  $\text{SNR}_{\text{OB}}$  erheblich von  $\text{SNR}_{\text{max}}$  oder  $\text{SNR}_{\text{avg}}$  abweichen.

# Stochastische Einflüsse (7)

Beispiel: normalverteiltes Rauschen mit unterschiedlich hohem  $\text{SNR}_{\max}$ :



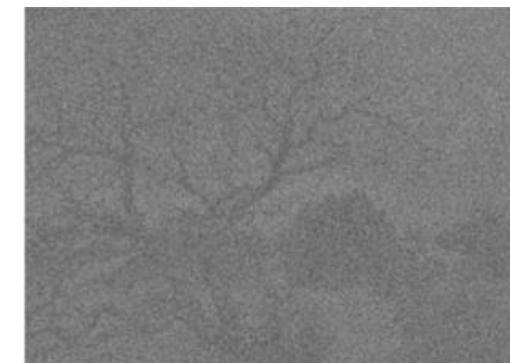
Original



$\text{SNR}_{\max} = 5$



$\text{SNR}_{\max} = 2$



$\text{SNR}_{\max} = 1$

Bilderquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Stochastische Einflüsse (8)

---

- Ggf. sind die zugrunde liegenden stochast. Störeinflüsse nicht durch das Gauß-verteilte Rauschmodell beschreibbar
- ↗ Ein Beispiel dafür ist das **Impulsrauschen**  
(engl. *salt-and-pepper noise*)



Bildquelle: [http://en.wikipedia.org/  
wiki/Image\\_noise](http://en.wikipedia.org/wiki/Image_noise) (26/10/2011)

# Stochastische Einflüsse (9)

---

Impulsrauschen zeigt drei Eigenschaften:

- 1) die Veränderung der Intensitätswerte ist entweder max. positiv (hellster Grauwert 255 = „Salz“) oder max. negativ (dunkelster Grauwert 0 = „Pfeffer“)
- 2) nur wenige Pixel (z.B. 5%) sind gestört
- 3) in der lokalen Umgebung eines gestörten Pixels sind meist keine anderen Pixel gestört



Bildquelle: [http://en.wikipedia.org/  
wiki/Image\\_noise](http://en.wikipedia.org/wiki/Image_noise) (26/10/2011)

# Stochastische Einflüsse (10)

---

Modellierung von *Impulsrauschen*:

- Die geringe Zahl gestörter Pixel und die Lokalität der Störung macht die räumliche Korrelation schwer modellierbar, da diese eine summarische Betrachtung der Intensitätswerte in Abhängigkeit von den Abständen zwischen den Pixeln verlangt
- Dennoch werden wir bald ein Filter zur Unterdrückung des Impulsrauschen lernen



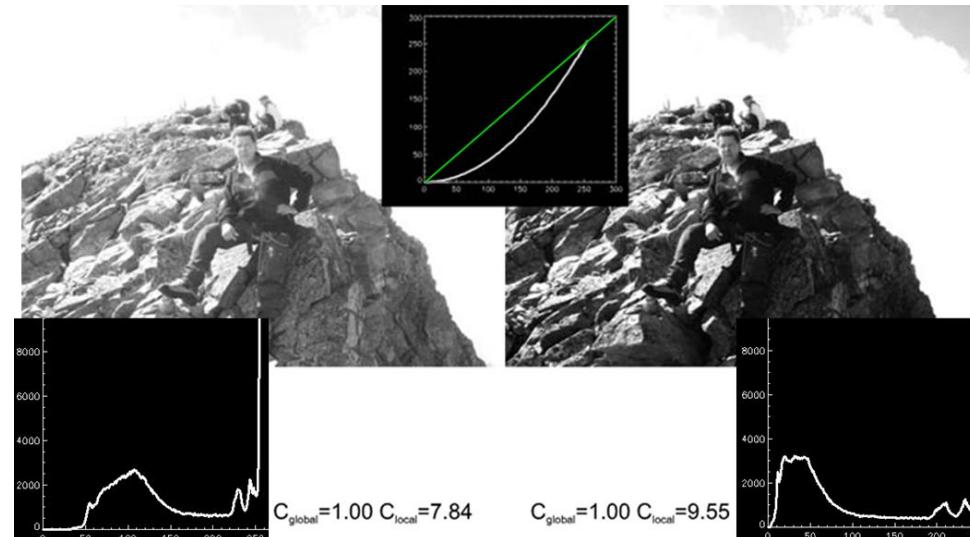
Bildquelle: [http://en.wikipedia.org/  
wiki/Image\\_noise](http://en.wikipedia.org/wiki/Image_noise) (26/10/2011)

# Histogrammbasierte Verfahren vs. Lineare Filter

Die erste Vorlesung zeigte histogrammbasierte Verfahren zur Bildverbesserung. Diese berücksichtigen nicht die räumliche Verteilung der Intensitätswerte im Bild.

So sind für eine Verbesserung des lokalen Kontrastes  $C_{local}$  eigentlich die lokalen Pixelnachbarschaften zu berücksichtigen.

Nur indirekt konnte  $C_{local}$  mit Hilfe der Gamma-Korrektur oder der adaptiven Histogrammlinearisation verbessert werden.



Bilderquelle: K. Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005

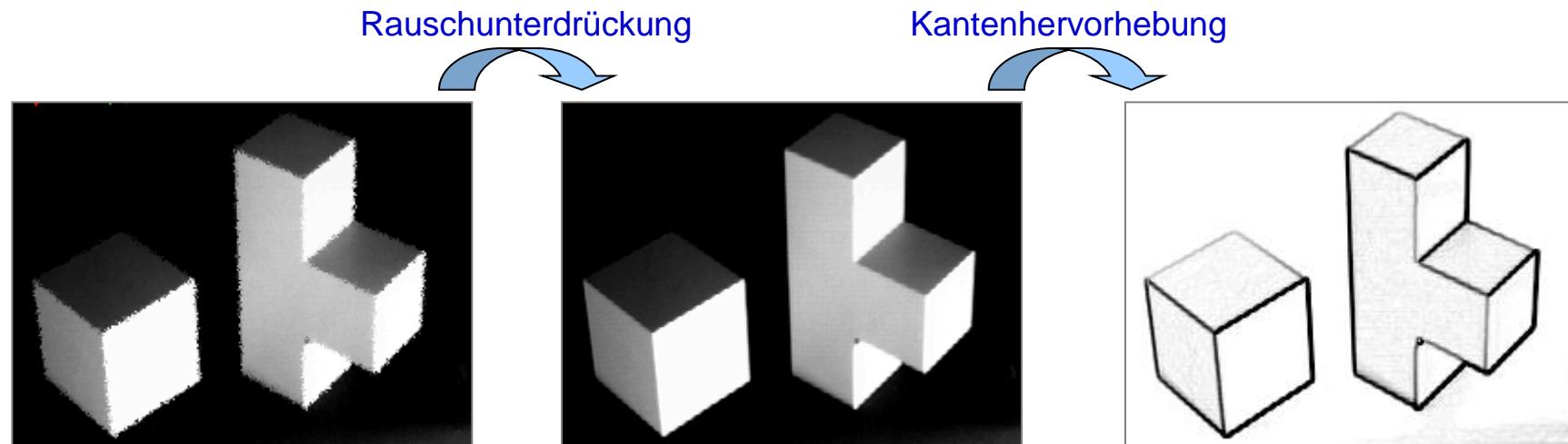
Die Konvolution kann hingegen Pixel in Abhängigkeit von ihrer lokalen Pixelnachbarschaft verändern. Damit sind lokale Bildverbesserungen möglich.

# Lineare Filter (1)

---

Zwei Arten der Bildverbesserung sind durch Konvolution durchführbar:

- 1) durch **Rauschunterdrückung** (auch als **Glättung** bezeichnet) kann das Signal gegenüber stochastischem Rauschen verstärkt werden,
- 2) durch **Hervorhebung von Kantenpixeln**, die Grenzen zwischen Objektoberflächen abbilden, werden abgebildete Objekte und ihre Abgrenzungen besser erkennbar und durch anschließende Prozesse leichter erkennbar.



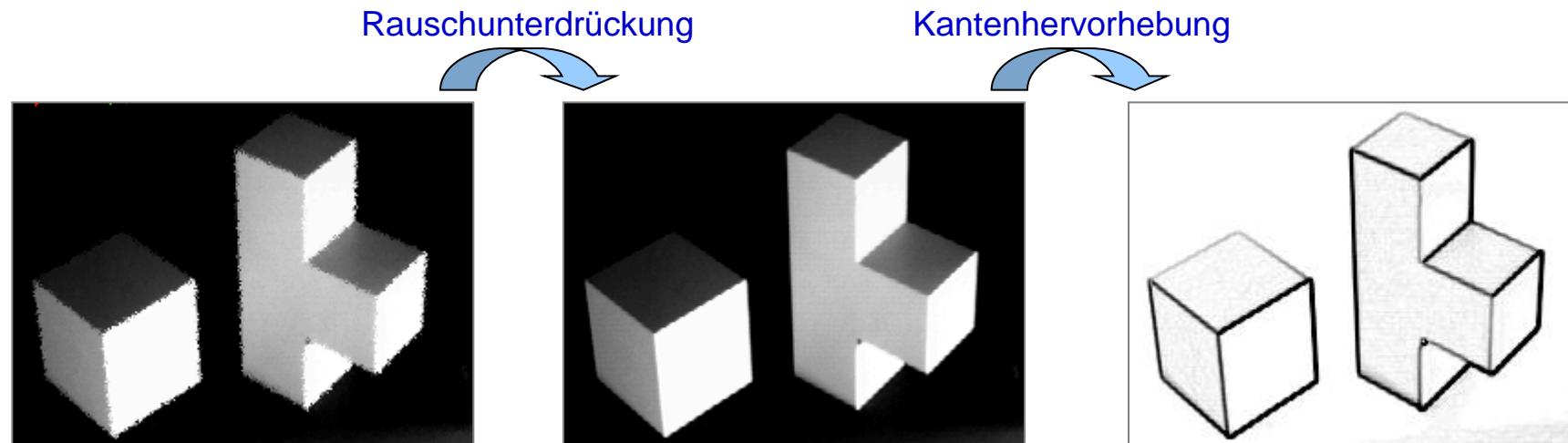
# Lineare Filter (2)

Methoden der **Rauschunterdrückung** bzw. **Glättung** müssen die lokalen Pixelnachbarschaften berücksichtigen, weil Rauschen nur als statistische Eigenschaft einer Menge von Pixel charakterisierbar ist.

diese  
Vorlesung

Methoden der **Kantenhervorhebung** benötigen die Information aus lokalen Pixelnachbarschaften, weil Kanten sich nur durch den Vergleich zwischen den Intensitäten benachbarter Pixel herausbilden.

nächste  
Vorlesung



# Lineare Filter zur Glättung (1)

---

Das erste einfache Glättungsfilter basiert auf dem additiven Überlagerungsmodell (vgl. Folie 20):

ein verrauschtes Bild  $\mathbf{I} = [I(x,y)]$  zeigt resultierende Intensitäten  $I(x,y)$  nach

$$I(x,y) = I_u(x,y) + \eta(x,y)$$

mit:

Signal  $I_u$  = ungestörtes Signal

Rauschen  $\eta$  = zufallsverteilte Störung

# Lineare Filter zur Glättung (2)

---

- Bei diesem additiven Überlagerungsmodell

$$I(x,y) = I_u(x,y) + \eta(x,y)$$

wurde die W'keit einer Abweichung des gemessenen Intensitätswertes vom ungestörten Intensitätswert  $I_u(x,y)$  durch eine **Gaußsche Normalverteilung mit Erwartungswert Null** modelliert:  $E(\eta(x,y)) = 0$

- ↗ Nun gilt: mit **steigender Stichprobenzahl** ergibt sich eine zunehmend bessere Schätzung des Erwartungswertes für das Rauschen  $\eta$

# Lineare Filter zur Glättung (3)

---

Ausgangspunkt also: Mit steigender Stichprobenzahl ergibt sich eine zunehmend bessere Schätzung des Erwartungswertes für das Rauschen  $\eta$

Dies ist zur Rauschunterdrückung wie folgt nutzbar:

- der Erwartungswert der ungestört. Intensitätsfunktion  $I_u(x,y)$  ist konstant:

$$E(I_u(x,y)) = I_u(x,y)$$

*Erwartungswert der  
Störung ist Null*

↗ damit gilt:

$$E(I(x,y)) = E(I_u(x,y)) + E(\eta(x,y)) = E(I_u(x,y)) = I_u(x,y)$$

↗ das Problem ist gelöst, sobald eine gute Schätzung für  $E(I_u(x,y))$  vorliegt!

# Lineare Filter zur Glättung (4)

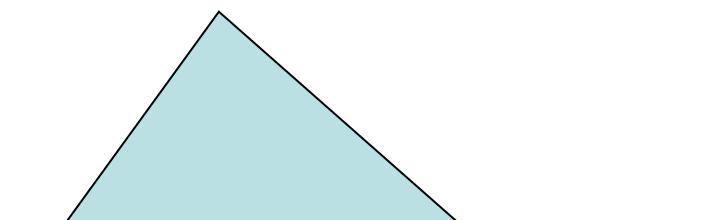
---

Eine gute Schätzung für  $E(I(x,y))$  wäre wie folgt ableitbar:

1) generiere eine Folge von  $k$  Bildern  $I_0, \dots, I_{k-1}$  derselben Szene

2) schätze  $E(I(x,y))$  über das aus der Folge gemittelte Bild

↪ Also wäre das ungestörte Bild ableitbar durch eine Konvolution entlang der Zeitachse mit dem Konvolutionskern  $f = (1/k, \dots, 1/k)$



Die Summe der Gewichtselemente des Konvolutionskerns ergibt 1. Diese Eigenschaft gilt für alle Konvolutionskerne zur Rauschunterdrückung!!!

Der Grund: es geht ja um die Schätzung von  $E(I(x,y))$ . Jede von 1 verschiedene Gewichtssumme würde eine Skalierung der Intensitäten der Bildfunktion ergeben.

# Lineare Filter zur Glättung (5)

---

Die Mittelung über eine Folge von Einzelbildern ist i.A. nicht praktikabel.

Daher wird von  $E(I(x,y))$  wie folgt geschätzt:

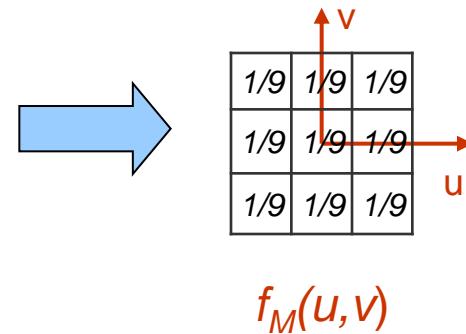
- 1) wir nehmen an, dass die Werte der ungestörten Bildfunktion  $I_u(x,y)$  in einer lokalen Nachbarschaft jedes Pixels konstant ist
- 2) statt über eine Bildfolge wird  $E(I(x,y))$  nun durch Mittelung über die Intensitätswerte innerhalb der lokalen Nachbarschaft geschätzt

# Mittelwertfilter (1)

Der einfachste Konvolutionskern hierfür definiert das sog. *Mittelwertfilter* (engl. *boxcar filter* oder *shifted mean filter*).

Für die  $3 \times 3$ -Nachbarschaft hat er folg. Form:

$$f_{M,3}(u,v) = \{(-1, 1) \rightarrow 1/9, (0, 1) \rightarrow 1/9, (1, 1) \rightarrow 1/9, \\ (-1, 0) \rightarrow 1/9, (0, 0) \rightarrow 1/9, (1, 0) \rightarrow 1/9, \\ (-1, -1) \rightarrow 1/9, (0, -1) \rightarrow 1/9, (1, -1) \rightarrow 1/9 \}$$

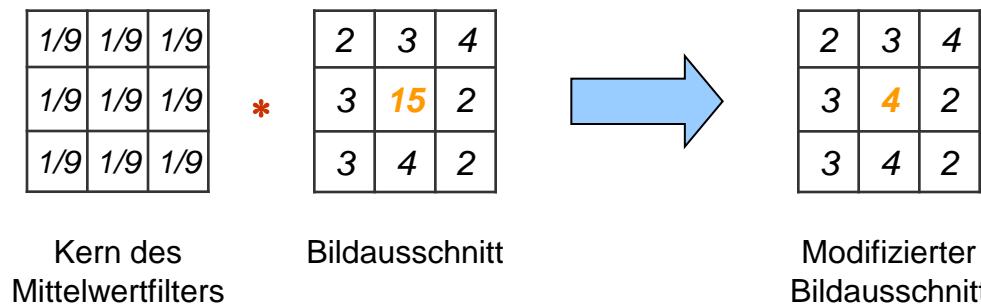


# Mittelwertfilter (2)

Die Anwendung des **3×3-Mittelwertfilters** bzw. des **Mittelwertfilters der Größe 3** auf ein Pixel und seine 8-Nachbarschaft über die Konvolution nach

$$(f_{M,3} * g)(x,y) = \sum_u \sum_v f_{M,3}(u,v) \cdot g(x-u, y-v) \text{ mit } u,v = -1, 0, 1$$

am Beispiel:



Bemerkung: aus Effizienzgründen erfolgt die Implementierung i.A. so, dass alle neun Gewichte auf Eins gesetzt werden und die Summe durch 9 dividiert werden:

$$f(u,v) = 1/9 \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

# Mittelwertfilter (3)

---

Die Zuverlässigkeit der Schätzung von  $E(I(x,y))$  steigt mit der Anzahl der Pixel, über die gemittelt wird.

Mit einem **Mittelwertfilter der Größe 5** wird z.B. fast über die dreifache Zahl von Pixel gegenüber dem Mittelwertfilter der Größe 3 gemittelt:

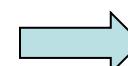
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9



0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04

Kern des  $3 \times 3$ -Mittelwertfilters

Kern des  $5 \times 5$ -Mittelwertfilters



...

# Mittelwertfilter (4)

---

## Grenzen des Mittelwertfilters:

Mit zunehmender Größe des Mittelwertfilters wird die Annahme unrealistischer, dass alle Pixel innerhalb der Nachbarschaft denselben ungestörten Intensitätswert  $I_u(x,y)$  haben.

Das ist insbes. bei Kanten der Fall. So werden Artefakte bei größer werdenden Filtern immer stärker und die resultierenden Bilder wirken immer unschärfer.

# Mittelwertfilter (5)

---

- Besonders kritisch wird die durch **Mittelwertfilter** resultierende Unschärfe, wenn **Impulsrauschen** bzw. ***Salt-and-Pepper-Noise*** vorliegt
- Der Erwartungswert des Impulsrauschen ist zwar ebenfalls 0, aber der überwiegende Anteil aller Pixel (z.B. 95%) ist vom Impulsrauschen nicht betroffen
  - ~ Für die Filterung müsste die Umgebungsnachbarschaft sehr groß sein, um genügend positive und negative Impulse zu erfassen, damit ihr Einfluss sich in der Summation aufhebt
  - ~ Damit ist das **Rauschen nicht zu entfernen**, ohne dass Kanten im Bild **unscharf werden**. Insofern kann praktikabel nur eine Abschwächung, nicht aber eine Unterdrückung des Impulsrauschens erzielt werden (s. Abb.)

# Mittelwertfilter (6)

---

Beispiel für Anwendung von Mittelwertfiltern auf Impulsrauschen:



Bilderquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Gauß-Filter (1)

Die Anwendung von Mittelwertfiltern führt auch bei Photonenauschen zu einer Glättung von Bildkanten.

Das Photonenauschen wurde durch eine Gauß-Verteilung modelliert. Eine optimale Unterdrückung des Photonenauschens ist demnach durch einen nach einer Gauß-Funktion gewichteten Filter möglich.

Für eine Dimension:

$$f_{G,\sigma}(u) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(u^2)/2\sigma^2}.$$

Mit Standardabweichung  $\sigma$  der Gauß-Funktion. Je größer  $\sigma$ , desto stärker die Glättung des Gauß-Filters.

Die zweidimensionale Gauß-Funktion ergibt sich als Produkt zweier eindimensionaler Gauß-Funktionen\*:

$$f_{G,\sigma}(u, v) = \frac{1}{2\pi\sigma^2} e^{-(u^2+v^2)/2\sigma^2}.$$

\* siehe D. A. Forsyth, J. Ponce: *Computer Vision – A Modern Approach*. Pearson, 2003, Seite 173.

# Gauß-Filter (2)

---

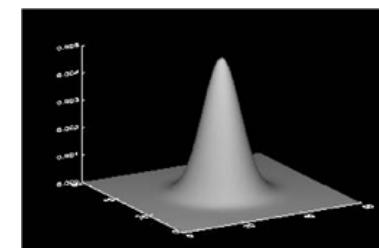
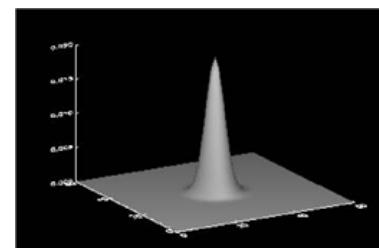
Separierbarkeit des zweidimensionalen Gauß-Filters:

- Die zweidimensionale Gauß-Funktion wurde als Produkt zweier eindimensionaler Gauß-Funktionen erzeugt.
- Analog ist die Konvolution mit dem zweidimensionalen Gauß-Filter durch zwei aufeinander folgende Konvolutionen mit eindimensionalen Gauß-Filtern in Zeilen- und Spaltenrichtung durchführbar.

# Gauß-Filter (3)

---

Beispiel für Anwendung eines Gauß-Filters mit unterschiedl.  $\sigma$ :



Bilderquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

# Gauß-Filter (4)

---

## Die Größe des Konvolutionskerns eines Gauß-Filters

- ist so zu wählen, dass die Werte der Gauß-Funktion hinreichend gut approximiert werden\*
- ist von der Standardabweichung  $\sigma$  der Gauß-Funktion abhängig

Eine angemessene Filtergröße  $m$  ist z.B.  $m = 2 \cdot \lceil 3\sigma \rceil + 1$ :

- der Funktionswert am Rand des Filterkerns beträgt dann noch 1% des Maximums der Gauß-Funktion
- das Maximum in  $f_{G,\sigma}(0,0)$  ist  $(2\pi\sigma^2)^{-1}$

---

\* Durch die Diskretisierung wird die Summe i.A. nicht mehr 1 ergeben. Entspr. muss die Gauß-Funktion durch die Summe aller Gewichte normiert werden.

# Binomialfilter (1)

---

Binomialfilter sind sehr gute Approximationen der Gauß-Filter, die effizient mit ganzzahligen Operationen berechnet werden können. Sie heißen so, weil sich ihre Werte aus den Binomialkoeffizienten ergeben.

Binomialfilter werden zunächst als 1-dim. Filter entwickelt und dann auf 2-dim. Filter erweitert. Die *unnormierten* Koeffizienten der *eindimensionalen Binomialfilter  $B^k$  der Ordnung  $k$  und der Größe  $k+1$*  werden nach folgender Formel entwickelt:

$$b_i^k = \binom{k}{i}.$$

Also normiert:

$$B^1 = \frac{1}{2} (1 \ 1), \quad B^2 = \frac{1}{4} (1 \ 2 \ 1),$$

$$B^3 = \frac{1}{8} (1 \ 3 \ 3 \ 1), \quad B^4 = \frac{1}{16} (1 \ 4 \ 6 \ 4 \ 1),$$

...

Für die Filterung sind nur Binomialfilter von gerader Ordnung  $k$  brauchbar, da diese eine ungerade Anzahl  $k+1$  von Filterkoeffizienten zeigen.

## Binomialfilter (2)

---

Ein quadratisches, 2-dim. Binomialfilter  $B^{k,k}$  ergibt sich durch Matrixmultiplikation der zwei 1-dim. Binomialfilter  $B^k$  gleicher Ordnung:

$$B^{k,k} = B^k \times (B^k)^T$$

Für die zweidimensionalen Binomialfilter der Ordnungen 2 und 4:

$$B^{2,2} = \frac{1}{4} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} \times \frac{1}{4} \begin{pmatrix} 1 & 2 & 1 \end{pmatrix} = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix},$$

$$B^{4,4} = \frac{1}{16} \begin{pmatrix} 1 \\ 4 \\ 6 \\ 4 \\ 1 \end{pmatrix} \times \frac{1}{16} \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \end{pmatrix} = \frac{1}{256} \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix}.$$

# Binomialfilter (3)

---

Anmerkungen zu den Binomialfiltern

- Per Definition sind auch die zweidimensionalen Binomialfilter  $B^{k,k}$  separabel
- Wenn die Ordnung  $k$  der Binomialfilter gegen Unendlich geht, ergibt sich das Gauß-Filter

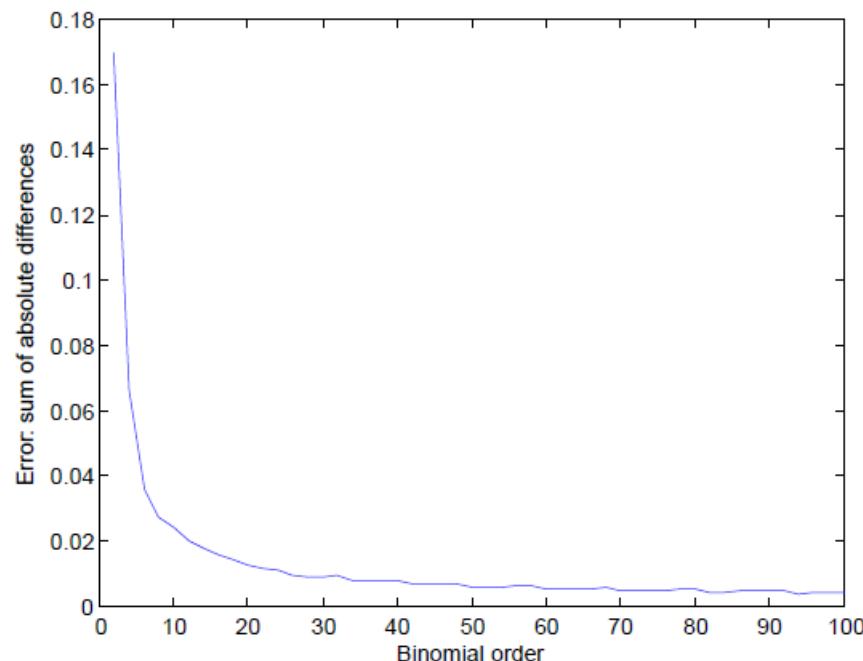
# Binomialfilter (4)

---

Zur approximativen Güte der Binomialfilter:

- Ein Binomialfilter der Ordnung  $k$  führt näherungsweise dieselbe Glättung durch wie ein Gauß-Filter mit  $\sigma = k^{1/2}/2$ .
- Die Approximation der Gauß-Funktion  $G_{x,\sigma}$  (bis inkl.  $3\sigma$ ) durch die Binomialfilter  $B^k$  zeigt die Fehlerkurve  $\sum |G_{x,\sigma} - B^k|$  mit  $\sigma = k^{1/2}/2$ .

Quelle: K.G. Derpanis: Overview of Binomial Filters. Lecture notes, March 2005, York Univ. (CA).



# Binomialfilter (5)

---

Der Vergleich zwischen Mittelwertfilter und Binomialfilter zeigt, dass die Glättung beim gleichgroßen Binomialfilter nicht so stark ausfällt.

Im Ggs. zum Mittelwertfilter nimmt beim Binomialfilter – wie auch beim Gauß-Filter – der Glättungseffekt mit zunehm. Distanz zum Zentrum des Filterkerns ab.



Mittelwertfilter 5x5



Binomialfilter 5x5

# Zusammenfassung

---

- Deterministische Einflüsse basieren auf wiederholbaren Einwirkungen bei der Bildaufnahme.
- Viele deterministische Einflüsse lassen sich durch lineare Operatoren und insbes. verschiebungsinvariante, lineare Operatoren beschreiben.
- Die Konvolution einer Signalfunktion mit einer Konvolutionsfunktion ist eine verschiebungsinvariante, lineare Operation, die in der Bildverarbeitung zur Rauschunterdrückung und Kantenhervorhebung benutzt wird.
- Bildstörungen durch Rauschen werden als additive Überlagerungen des Signals durch ein stochastisch modelliertes Rausches modelliert.
- Lineare Filterung zur Rauschunterdrückung lässt sich durch verschiedene Filter umsetzen wie Mittelwertfilter, Gauß-Filter und Binomialfilter.