

---

# DIGITALE BILDVERARBEITUNG

VORLESUNGSSKRIPT

DR. S. GEHRIG

Version 2.19 (17.02.2016)

Latex-Urfassung Dirk Bolte (2002)  
Initiales Skriptum Dr. U. Franke (1996)

---

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung in die Digitale Bildverarbeitung</b>	<b>4</b>
1.1	Einsatzbereiche der Digitalen Bildverarbeitung . . . . .	4
1.2	Aufbau eines BV-Systems . . . . .	5
1.2.1	Bildaufnahme . . . . .	5
1.2.2	Erzeugung von Bildern . . . . .	5
1.2.3	Diskretisierung . . . . .	6
1.2.4	Bildvorverarbeitung . . . . .	6
1.2.5	Bildererkennung . . . . .	6
1.3	Literatur: Bücher, Web-Seiten . . . . .	6
<b>2</b>	<b>Grundlagen 2 - dimensionaler Signale und Systeme</b>	<b>7</b>
2.1	Die zweidimensionale Fouriertransformation . . . . .	7
2.1.1	Definition $S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft} dt$ . . . . .	7
2.1.2	Separierbarkeit der 2-dimensionalen Fouriertransformationen . . . . .	8
2.1.3	Theoreme . . . . .	10
2.1.4	Der Dirac-Stoß . . . . .	12
2.2	Abtastung zweidimensionaler Signale . . . . .	13
2.2.1	Das Abtasttheorem . . . . .	13
2.2.2	Bildwiedergabe . . . . .	14
2.3	Die diskrete Fouriertransformation . . . . .	14

---

2.3.1	Definition . . . . .	14
2.3.2	Fast Fourier Transformation - FFT . . . . .	16
2.3.3	Diskrete Faltung und diskrete $\delta$ -Funktion . . . . .	17
2.3.4	Darstellung von Spektren . . . . .	17
2.3.5	Interpretation des 2D-Spektrums . . . . .	18
2.3.6	Anwendungsbeispiel für Frequenzanalyse: jpeg-Codierung . . . . .	19
2.4	Quantisierung . . . . .	19
<b>3</b>	<b>Bildverbesserung</b>	<b>22</b>
3.1	Punktoperatoren . . . . .	22
3.1.1	Transformation der Grauwertkennlinie . . . . .	22
3.1.2	Histogrammausgleich . . . . .	25
3.1.3	Binarisierung . . . . .	26
3.1.4	Kompensationsverfahren . . . . .	29
3.2	Lokale Operatoren . . . . .	30
3.2.1	Lineare Glättung . . . . .	30
3.2.2	Bildverschärfung . . . . .	32
3.2.3	Nichtlineare Operatoren . . . . .	32
<b>4</b>	<b>Konturen</b>	<b>36</b>
4.1	Konturpunktdetektion - Kantenpunktdetektion . . . . .	36
4.2	Konturverdünnung . . . . .	41
4.2.1	Nonmaxima-Unterdrückung . . . . .	41
4.2.2	Übergang zur 8er Nachbarschaft . . . . .	44
4.3	Konturpunktverkettung . . . . .	44
4.4	Konturapproximation . . . . .	46
4.5	Hough-Transformation . . . . .	47
4.5.1	Extraktion von Geraden . . . . .	47

---

4.5.2	Hough-Transformation von Kreisen . . . . .	51
<b>5</b>	<b>Morphologische Bildverarbeitung</b>	<b>53</b>
5.1	Erosion und Dilatation . . . . .	54
5.2	Opening und Closing . . . . .	54
5.3	Detektion von Bildteilen bekannter Form . . . . .	56
<b>6</b>	<b>Schätzung von Verschiebungen</b>	<b>58</b>
6.1	Korrelation . . . . .	58
6.1.1	Differenz . . . . .	58
6.1.2	Mittelwertbefreite Differenz . . . . .	59
6.1.3	Korrelationsfunktion . . . . .	59
6.2	Messen mit Kameras . . . . .	60
6.2.1	Zentralprojektion . . . . .	60
<b>7</b>	<b>Klassifikation</b>	<b>62</b>

---

# Kapitel 1

## Einführung in die Digitale Bildverarbeitung

### 1.1 Einsatzbereiche der Digitalen Bildverarbeitung

a) Image Processing : Bild => Bild

b) Image Analysis/Computer Vision : Bild => Extrahierte Information

zu a)

- Bildgebende Verfahren der Medizin
- Reduktion von Rauschen, Motion Blur, Bildstörungen
- Bilddatenkompression (Übertragung, Speicherung)
- Filterung
- Fernerkundung (Auswertung multispektraler Bilder, digitale Geländemodelle)
- Kontrastverbesserung

zu b)

- Qualitätskontrolle
- Lesen handgeschriebener Texte (Briefsortieranlagen)
- Robotik, Automatisierung, Regelung
- Produktion (Bestückung, Sortierung)
- Überwachungsanlagen (Gefahrenbereiche, Flughafen)

- 
- automat. Auswertung medizinischer Bilder (Tumorerkennung)
  - Wettervorhersage
  - Kriminalistik (Fingerabdrücke)
  - Archäologie
  - Militärische Anwendungen
  - Zugangskontrolle
  - Fahrerassistenz (Innenraum, Objektdetektion)

In allen Bereichen des täglichen Lebens BV sinnvoll einsetzbar, insb. wo

- monotone Sehaufgaben anfallen
- der Mensch nicht präzise genug ist
- der Mensch zu teuer ist.

## 1.2 Aufbau eines BV-Systems

Bildaufnahme => Diskretisierung => Verbesserung => Segmentierung => Merkmalsextraktion  
=> Klassifikation/Interpretation => Ergebnis

### 1.2.1 Bildaufnahme

Standard bis Mitte 90er: CCD-Flächenkameras 768\*512 pixel, interlaced, 25 Bilder/s, analoger Ausgang.

Trend: digital, non-interlaced, CMOS.

Probleme: geringer Dynamikbereich, Empfindlichkeit

für schnelle Dinge: Zeilenkameras (Scanner), bis 6000 Pixel, Bildfrequenzen >> 25 Hz

### 1.2.2 Erzeugung von Bildern

Reflexion (Photographie, Luftbild), z.T. Aktive Beleuchtung, nicht notwendigerweise im sichtbaren Bereich

Absorption (Röntgenaufnahme, Durchlichtbilder)

Emission (Wärmebilder (FIR), Nuklearmedizin, NMR)

---

### 1.2.3 Diskretisierung

Umwandlung im sog. Framegrabber

Abtastung und Quantisierung

### 1.2.4 Bildvorverarbeitung

weitgehend problemspezifische Verarbeitungsschritte

### 1.2.5 Bilderkennung

basiert auf problemspezifischen Wissen

Hardware für Basisoperationen (Kantenextraktion, Modifikation der Grauwerte, Reduktion von Störungen) verfügbar.

Trend: General Purpose Prozessoren für Bildverarbeitung

## 1.3 Literatur: Bücher, Web-Seiten

B. Jähne: "Digitale Bildverarbeitung",  
vierte Auflage, Springer Verlag Berlin, Heidelberg, New York, 1997  
ISBN 3-540-61379-x

A. K. Jain: "Fundamentals of Digital Image Processing", Prentice Hall, 1989

<http://www.dai.ed.ac.uk/CVonline/>  
Topic collection of Computer Vision (for teaching)

<http://www.dai.ed.ac.uk/HIPR2>  
Tutorial to try image operators online

<http://www.cs.cmu.edu/afs/cs/project/cil/ftp/html/vision.html>

---

## Kapitel 2

# Grundlagen 2 - dimensionaler Signale und Systeme

Bilddbetrachtung basiert auf Ortssignalen.

Motivation:

- Filterung von Signalen
- Charakterisierung anhand des Spektrums, z.B. Klassifikation von Texturen, Oberflächen (Schraffuren)
- Analyse linearer Systeme

Filterung im Spektralbereich:

$$\frac{s(t)}{S(f)} \begin{array}{c} \boxed{h(t)} \\ H(f) \end{array} \frac{g(t)}{G(f)} = \frac{s(t) \times h(t)}{S(f) \cdot H(f)}$$

## 2.1 Die zweidimensionale Fouriertransformation

2.1.1 Definition  $S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft} dt$

$$S(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x, y) \cdot e^{-j2\pi(ux+vy)} dx dy = \int_{-\infty}^{\infty} \underbrace{\int_{-\infty}^{\infty} s(x, y) \cdot e^{-j2\pi(ux)} dx}_{S_x(v, y)} e^{-j2\pi(vy)} dy$$

---

$\mathbf{u}$  steht dabei für die Zeilenfrequenz,  $\mathbf{v}$  für die Spaltenfrequenz.

Bezeichnungen:

- $s(x, y)$  bezeichnet eine Ortsfunktion
- $S(u, v)$  bezeichnet eine Ortsfrequenzfunktion
- Bilder sind im Allgemeinen reellwertige Funktionen, Spektren komplex  
 $S(u, v) = \text{Re}\{S(u, v)\} + j \cdot \text{Im}\{S(u, v)\}$
- Betragsspektrum  $|S(u, v)| = \sqrt{\text{Re}^2\{S(u, v)\} + \text{Im}^2\{S(u, v)\}}$
- Phasenspektrum  $\phi(u, v) = \arctan\left(\frac{\text{Im}\{S(u, v)\}}{\text{Re}\{S(u, v)\}}\right)$

Für reellwertige Signale gilt:

- Betragsspektrum ist gerade:  $|S(u, v)| = |S(-u, -v)|$
- Phasenspektrum ist ungerade:  $\phi(u, v) = -\phi(-u, -v)$
- Realteil des Spektrums ist eine gerade Funktion:  $\text{Re}\{S(u, v)\} = \text{Re}\{S(-u, -v)\}$
- Imaginärteil des Spektrums ist eine ungerade Funktion:  $\text{Im}\{S(u, v)\} = -\text{Im}\{S(-u, -v)\}$

## 2.1.2 Separierbarkeit der 2-dimensionalen Fouriertransformationen

Separierbar bedeutet, dass eine Funktion in mehrere Multiplikationen aufgeteilt werden kann.

Die Fouriertransformierte des Rechteck-Impulses:

$$\begin{aligned} S(u, v) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x, y) \cdot e^{-j2\pi(ux+vy)} dx dy \\ &= \int_{-\infty}^{\infty} e^{-j2\pi vy} \left\{ \underbrace{\int_{-\infty}^{\infty} s(x, y) \cdot e^{-j2\pi ux} dx}_{S_x(u, y)} \right\} dy \end{aligned}$$

Rücktransformation über den Frequenzbereich:

$$s(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(u, v) \cdot e^{+j2\pi(ux+vy)} du dv$$

Die Berechnung erfolgt durch zwei 1-dimensionale Fouriertransformationen  
 $s(x, y) \circ \bullet S_x(u, y) \circ \bullet S(u, v)$

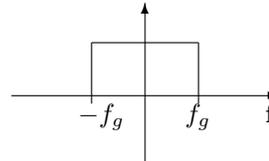
Bei der ersten Korrespondenz ist  $\mathbf{x}$  die Variable, in der zweiten  $\mathbf{y}$ .

---

**Exkurs : FT der Rechteckfunktion**

$$\begin{aligned}
 S(f) &= \int_{-\infty}^{\infty} \text{rect}(t) \cdot e^{-j2\pi ft} dt \\
 &= \int_{0,5}^{-0,5} e^{-j2\pi ft} dt = \frac{1}{-j2\pi f} e^{-j2\pi ft} \Big|_{0,5}^{-0,5} \\
 &= \frac{1}{-j2\pi f} (e^{j\pi f} - e^{-j\pi f}) \quad \text{mit } \sin x = \frac{e^{ix} - e^{-ix}}{2i} \quad \text{ergibt sich:} \\
 &\Rightarrow S(f) = \frac{\sin(\pi f)}{\pi f} = \text{si}(\pi f) \\
 \lim_{f \rightarrow 0} \text{si}(\pi f) &= \lim_{f \rightarrow 0} = \frac{\sin(\pi f)}{\pi f} \lim_{f \rightarrow 0} \frac{\pi \cos(\pi f)}{\pi} = \underline{\underline{1}}
 \end{aligned}$$

**Symmetrietheorem:**  $\text{si}(\pi t) \circ \bullet \text{rect}(-f) = \text{rect}(f)$   
 $2f_g \text{si}(2\pi f_g t) \circ \bullet \text{rect}\left(\frac{f}{2f_g}\right)$



**Separierbare Signale:**  $s(x, y) = s_1(x) \cdot s_2(y)$   
 $\Rightarrow S(u, v) = S_1(u) \cdot S_2(v)$

**Beispiel:**  $\text{rect}(x, y) = \begin{cases} 1 & \text{für } |x| \leq 0,5, |y| \leq 0,5 \\ 0 & \text{sonst} \end{cases}$   
 $\text{rect}(x, y) = \text{rect}(x) \cdot \text{rect}(y)$   
 $\circ$   
 $s(u, v) = \text{si}(\pi u) \cdot \text{si}(\pi v)$ , mit  $\sin(x)/x = \text{si}(x)$

### 2.1.3 Theoreme

#### Eindimensionale Faltung

$$s(t) \times h(t)$$



$$S(f) \cdot H(f)$$

#### Eindimensionale Multiplikation

$$s(t) \cdot h(t)$$



$$S(f) \times H(f)$$

#### Eindimensionale Superposition

$$a \cdot s(t) + b \cdot h(t)$$



$$a \cdot S(f) + b \cdot H(f)$$

#### Ähnlichkeitstheorem in 1-D

$$s(bt) \circlearrowleft \frac{1}{|b|} S\left(\frac{f}{b}\right)$$

#### Verschiebungstheorem in -D

$$s(t - t_0) \circlearrowleft S(f) \cdot e^{-j2\pi f t_0}$$

#### Differentiationstheorem in 1-D

$$\frac{d}{dt} s(t) \circlearrowleft j2\pi f S(f)$$

#### Zweidimensionale Faltung (ist von 2 Variablen abhängig)

$$s(x, y) \times \times h(x, y)$$



$$S(u, v) \cdot H(u, v)$$

#### Zweidimensionale Multiplikation

$$s(x, y) \cdot h(x, y)$$



$$S(u, v) \times \times H(u, v)$$

#### Zweidimensionale Superposition

$$a \cdot s(x, y) + b \cdot h(x, y)$$



$$a \cdot S(u, v) + b \cdot H(u, v)$$

#### Ähnlichkeitstheorem in 2-D

$$S(ax, by) \circlearrowleft \frac{1}{|ab|} S\left(\frac{u}{a}, \frac{v}{b}\right)$$

#### Verschiebungstheorem in 2-D

$$s(x - a, y - b) \circlearrowleft S(u, v) \cdot e^{-j2\pi(u \cdot a + v \cdot b)}$$

#### Differentiationstheorem in 2-D

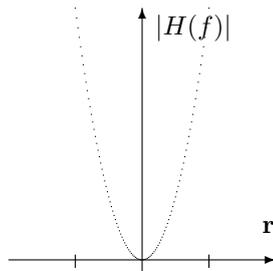
$$\frac{\delta}{\delta x} s(x, y) \circlearrowleft j2\pi u S(u, v) \text{ bzw. } \frac{\delta}{\delta y} s(x, y) \circlearrowleft j2\pi v S(u, v)$$

**Symmetrie-Theorem:**  $s(x, y) \circlearrowleft S(u, v)$   
 $S(x, y) \circlearrowleft s(-u, -v)$

**Laplace-Operator:**  $\nabla^2 s(x, y) = \frac{\delta^2}{\delta x^2} s(x, y) + \frac{\delta^2}{\delta y^2} s(x, y)$

$$-4\pi^2 \underbrace{(u^2 + v^2)}_{r^2} S(u, v)$$

$\nabla$  steht für  $(\frac{\delta}{\delta x}, \frac{\delta}{\delta y})$



**Problem:** bei verrauschten Signalen werden hohe Signalanteile stark verstärkt

**Parsevalsches Theorem für den eindimensionalen Raum**

$$\int_{-\infty}^{\infty} s^2(t) dt = \int_{-\infty}^{\infty} |S(f)|^2 df$$

Tabelle 2.1. Theoreme der Fourier-Transformation			Tabelle 2.2. Signalfunktionen im Zeit- und Frequenzbereich			
Theorem	$s(t)$	$S(f)$	$s(t)$	$S(f)$	$ S(f) $	
$\mathcal{F}$ -Transformation	$s(t)$	$S(f) = \int_{-\infty}^{+\infty} s(t)e^{-j2\pi ft} dt$		$\frac{1}{1+j2\pi Tf}$		
inverse $\mathcal{F}$ -Transformation	$\int_{-\infty}^{+\infty} S(f)e^{j2\pi ft} df$	$S(f)$		$\frac{1}{1+(2\pi Tf)^2}$		
Zerlegung reeller Zeitfunktionen	$s(t) = s_g(t) + s_u(t)$	$S(f) = \text{Re}\{S(f)\} + j \text{Im}\{S(f)\}$		$-j \frac{2\pi Tf}{1+(2\pi Tf)^2}$		
mit	$s_g(t)$	$\text{Re}\{S(f)\} = \int_{-\infty}^{+\infty} s_g(t) \cos(2\pi ft) dt$		$\text{si}(\pi f)$		
	$s_u(t)$	$+j \text{Im}\{S(f)\} = -j \int_{-\infty}^{+\infty} s_u(t) \sin(2\pi ft) dt$		$\text{rect}(f)$		
Zeitspiegelung	$s(-t)$	$S(-f)$ , bei reellen Zeitfunktionen auch $S^*(f)$		1		
Konjugiert komplexe Zeitfunktionen	$s^*(t)$	$S^*(-f)$		$\delta(f)$		
Symmetrie	$S(f)$	$s(-f)$		$\text{III}(f)$		
Faltung	$s(t) * h(t)$	$S(f) \cdot H(f)$		$e^{-\pi f^2}$		
Multiplikation	$s(t) \cdot h(t)$	$S(f) * H(f)$		$\delta(f+F) + \delta(f-F)$		
Superposition	$a_1 s(t) + a_2 h(t)$	$a_1 S(f) + a_2 H(f)$		$\frac{1}{2} \delta(f) - j \frac{1}{2\pi f}$		
Ähnlichkeit	$s(bt)$	$\frac{1}{ b } S\left(\frac{f}{b}\right)$		$\delta(f+F) + \delta(f-F)$		
Verschiebung	$s(t-t_0)$	$S(f)e^{-j2\pi ft_0}$				
Differentiation	$\frac{d^n}{dt^n} s(t)$	$(j2\pi f)^n \cdot S(f)$				
Integration	$\int_{-\infty}^t s(\tau) d\tau$	$\frac{S(f)}{j2\pi f} + \frac{1}{2} S(0) \delta(f)$				
Frequenzverschiebung	$s(t)e^{j2\pi Ft}$	$S(f-F)$				

Abbildung 2.1: Fourier-Theoreme-Übersicht

**Parsevalsches Theorem für den zweidimensionalen Raum**

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s^2(x,y) dt = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |S(u,v)|^2 dudv$$

**Rotationsatz:**

Dreht eine Funktion  $s(x,y)$  im Ortsbereich um einen Winkel  $\alpha$ , bewirkt dies eine Drehung im Spektralbereich um den gleichen Winkel.

$$s_\alpha(x',y') \circ \bullet S_\alpha(u',v')$$

---

## 2.1.4 Der Dirac-Stoß

### Der Dirac-Stoß im eindimensionalen Raum

Es war  $\delta(t) = \lim_{b \rightarrow 0} \frac{1}{b} \text{rect}\left(\frac{t}{b}\right)$  mit  $\int_{-\infty}^{\infty} \delta(t) dt = 1$

**Eigenschaften:**

$$\begin{aligned} \delta(t) \circ \bullet 1 \\ s(t) \times \delta(t - t_0) &= s(t - t_0) \\ \int_{-\infty}^{\infty} s(t) \cdot \delta(t - t_0) dt &= s(t_0) \end{aligned}$$

### Der Dirac-Stoß im zweidimensionalen Raum

$$\delta(x, y) = \lim_{a, b \rightarrow 0} \frac{1}{ab} \text{rect}\left(\frac{x}{a}, \frac{y}{b}\right) \text{ mit } \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(x, y) dx dy = 1$$

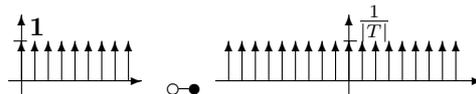
**Eigenschaften:**

$$\begin{aligned} \delta(x, y) \circ \bullet 1 \\ s(x, y) \times \delta(x - a, y - b) &= s(x - a, y - b) \end{aligned}$$

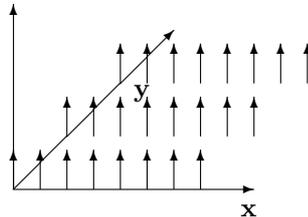
**Vorsicht:**  $\delta(x, y) = \delta(x) \cdot \delta(y)$  ist nicht gültig!

**Dirac-Stoßfolge:**

$$1 - D : \quad \sum_{n=-\infty}^{\infty} \delta(t - nT) \circ \bullet \frac{1}{|T|} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right)$$



$$2-D: \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y) \circ \bullet \frac{1}{\Delta x \Delta y} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(u - \frac{k}{\Delta x}, v - \frac{l}{\Delta y})$$



## 2.2 Abtastung zweidimensionaler Signale

Bei einer Abtastung wird über ein kontinuierliches Bild eine Matrix gelegt. Das Bild wird an den durch die Matrix definierten Punkten abgetastet.

Abtastung bedeutet, dass alle Informationen außerhalb der Gitterpunkte verloren gehen. Mathematisch entspricht dies einer Multiplikation des Bildes mit einer Funktion, die nur an den Gitterpunkten einen Wert  $\neq 0$  hat, einer sogenannten „Nagelbrettfunktion“ aus Diracstößen. Siehe [1], Seite 245,246.

### 2.2.1 Das Abtasttheorem

Das abgetastete Signal (a) berechnet sich aus dem kontinuierlichen Signal und der Dirac-Stoßfolge für die Abtastung.

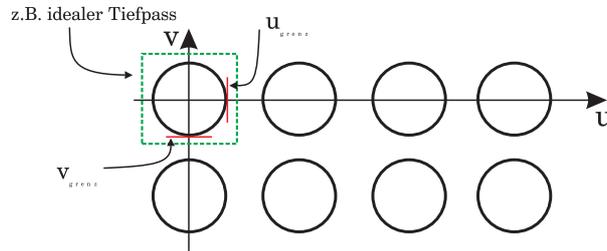
$$s_a(x, y) = s(x, y) \cdot \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - m\Delta x, y - n\Delta y)$$

$$S_a(u, v) = S(u, v) \times \times \frac{1}{\Delta x \Delta y} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \delta(u - \frac{k}{\Delta x}, v - \frac{l}{\Delta y})$$

**Satz:** Abgetastete Signale haben periodische Spektren.

Das Ausgangssignal kann fehlerfrei zurückgewonnen werden, wenn gilt:  $\frac{1}{\Delta x} \geq 2u_{\text{grenz}} \wedge \frac{1}{\Delta y} \geq 2v_{\text{grenz}}$

z.B. idealer Tiefpass



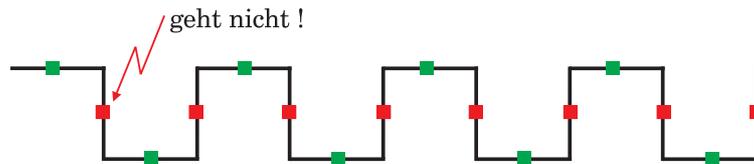
## 2.2.2 Bildwiedergabe

Üblicherweise wird jedem Abtastwert ein Quadrat konstanter Helligkeit zugeordnet  
 ⇒ Pixel

$$s_d(x, y) \cdot \text{rect}(x, y) \circ \bullet S(u, v) \times (\text{si}(\pi u) \cdot \text{si}(\pi v))$$

Beispiel:  $u_g = 10 \frac{\text{Linienpaare}}{\text{mm}}$   
 $\Rightarrow \Delta x \leq \frac{1}{20 \text{mm}} = 50 \mu\text{m}$

Vorsicht bei „=“ wegen Problemen bei der Abtastung: es besteht die Möglichkeit,



zufällig unsinnige Werte zu erwischen.

## 2.3 Die diskrete Fouriertransformation

### 2.3.1 Definition

Wiederholung: Diskrete periodische Signale haben diskrete periodische Spektren.

Eindimensionale diskrete Fouriertransformation:

$$S_d(k) = \sum_{n=0}^{N-1} s_d(n) e^{-j2\pi k \frac{n}{N}} \quad k = 0, \dots, N-1$$

$$s_d(n) = \frac{1}{N} \sum_{k=0}^{N-1} S_d(k) \cdot e^{+j2\pi k \frac{n}{N}} \quad n = 0, \dots, N-1$$

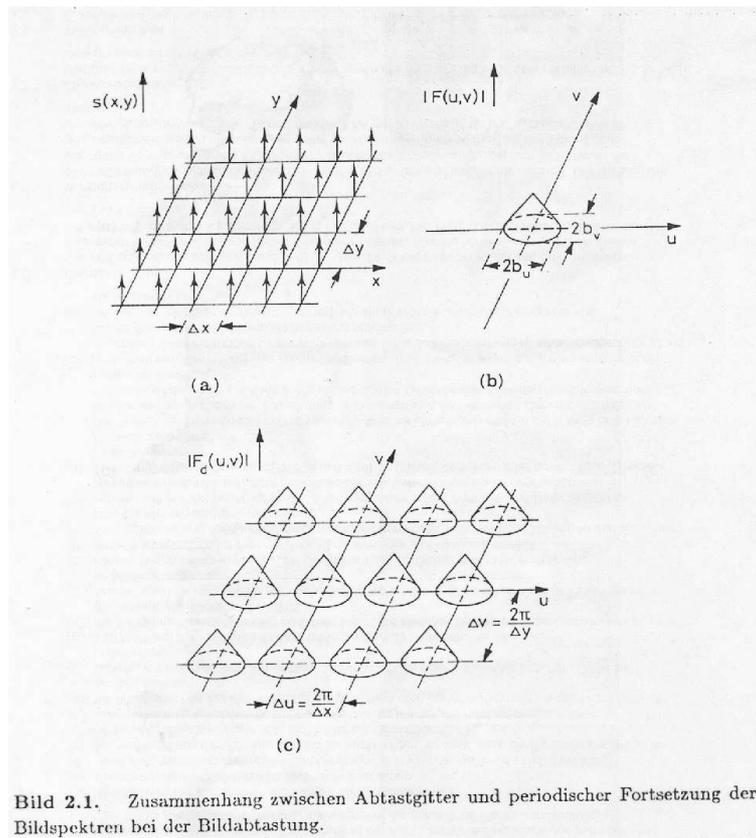
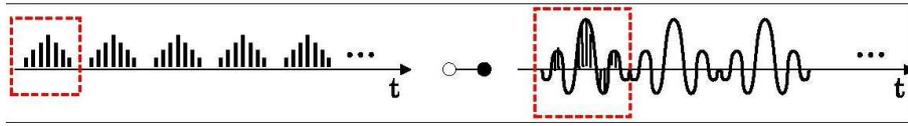


Abbildung 2.2: Periodizitaet



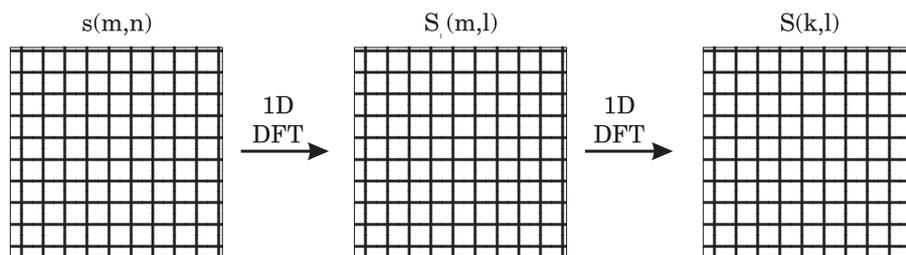
**Zweidimensionale diskrete Fouriertransformation:**

$$S(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} s(m,n) \cdot e^{-j2\pi(\frac{k \cdot m}{M} + \frac{l \cdot n}{N})}$$

$$s(m,n) = \frac{1}{M \cdot N} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} S(k,l) \cdot e^{+j2\pi(\frac{k \cdot m}{M} + \frac{l \cdot n}{N})}$$

Es gelten alle Theoreme sinngemäß:

**Separierbarkeit:** 
$$S(k,l) = \sum_{m=0}^{M-1} \underbrace{\left\{ \sum_{n=0}^{N-1} s(m,n) e^{-j2\pi \frac{l \cdot n}{N}} \right\}}_{S_1(m,l)} e^{-j2\pi \frac{k \cdot m}{M}}$$



mit  $s(m,n) \quad \circ \xrightarrow{n} \bullet \quad S_1(m,l) \quad \circ \xrightarrow{m} \bullet \quad S(k,l)$

**Aufwand:**  $(M \cdot N)^2$  direkt, bruteforce  $O(M^4)$   
 $(M \cdot N)(M + N)$  separiert  $O(2M^3)$

### 2.3.2 Fast Fourier Transformation - FFT

Die FFT ist ein schneller Algorithmus, der DFT für Signale einer Periode mit  $N = 2^n$  Abtastwerten anwendbar ist.

**Aufwand der FFT:**  $N \cdot \text{ld}(N)$  statt  $N^2$

**Aufwand für 2D-Signale der Größe  $M \cdot N$ :**  $M \cdot N \cdot \text{ld}(N) + N \cdot M \cdot \text{ld}(M) = M \cdot N (\text{ld}(N) + \text{ld}(M))$   
Für quadratische Bilder gilt:  $2M^2 \text{ld} M$

**Vergleich des Aufwands bei der Größe eines Bildes:**

---

$M$	$M^4$	$M^3$	$2M^2 \text{ld}(M)$
16	$65 \cdot 10^3$	$4 \cdot 10^3$	$1,5 \cdot 10^3$
128	$268 \cdot 10^6$	$2 \cdot 10^6$	$0,17 \cdot 10^6$
256	$4 \cdot 10^9$	$16 \cdot 10^6$	$0,78 \cdot 10^6$
512	$68 \cdot 10^9$	$134 \cdot 10^6$	$3,5 \cdot 10^6$

### 2.3.3 Diskrete Faltung und diskrete $\delta$ -Funktion

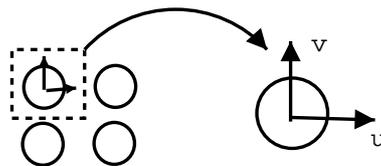
$$\frac{s(n)}{S(n)} \begin{array}{|c|} \hline h(k) \\ \hline \end{array} \frac{g(n)}{G(k)} = \frac{s(n) \times h(n)}{S(k) \cdot H(k)}$$

Definition:

$$g(n) = s(n) \times h(n) = \sum_{n'} s(n') \cdot h(n - n')$$

anschaulich: Übertragungsfunktion an y-Achse spiegeln und schrittweise untereinander summieren.

### 2.3.4 Darstellung von Spektren



$$S(0,0) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} s(m,n) \quad \text{ist der „Mittelwert“ des Signals}$$

Darstellung des Spektrums um  $S(0,0)$  zentriert.

Üblich: logarithmische Darstellung des Betragsspektrums:

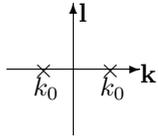
$$S'(k,l) = \log(1 + |S(k,l)|)$$

Praktisches Beispiel für die Verwendung von Spektralanalysen in der Bilderkennung: Erkennen von Zebrastrreifen aus einer Kamera in einem Auto. Die Schwierigkeit dabei ist die Erkennung, wenn das Bild „verrauscht“ ist, was bedeutet, dass beispielsweise Menschen darüber gehen. Wenn man nun das Bild mit Fouriertransformation bearbeitet, dann ist das sich ergebene Spektrum gleich dem Spektrum eines Zebrastrreifens plus das Spektrum der Menschen, so dass immer noch erkannt werden kann, dass ein Zebrastrreifen vorhanden.

### 2.3.5 Interpretation des 2D-Spektrums

$S(0,0)$  ist der „Mittelwert“

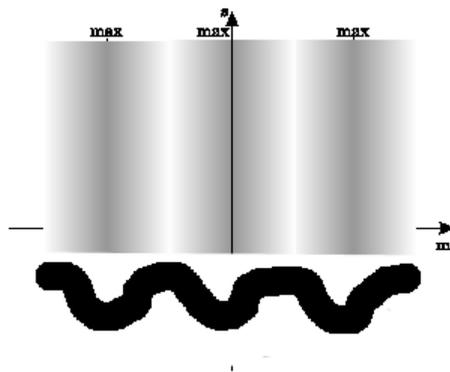
Betrachtung eines Spektrallinienpaares im 2-dimensionalen Raum (Die Pfeile zeigen aus der Tafel heraus):



Inverse DFT

$$S(m, n) = \frac{1}{M \cdot N} e^{2\pi j(k_0 \frac{m}{M})} + e^{-2\pi j(k_0 \frac{m}{M})}$$

Dies entspricht:  $S(m, n) = \frac{2}{M \cdot N} \cos\left(2\pi k_0 \frac{m}{M}\right)$



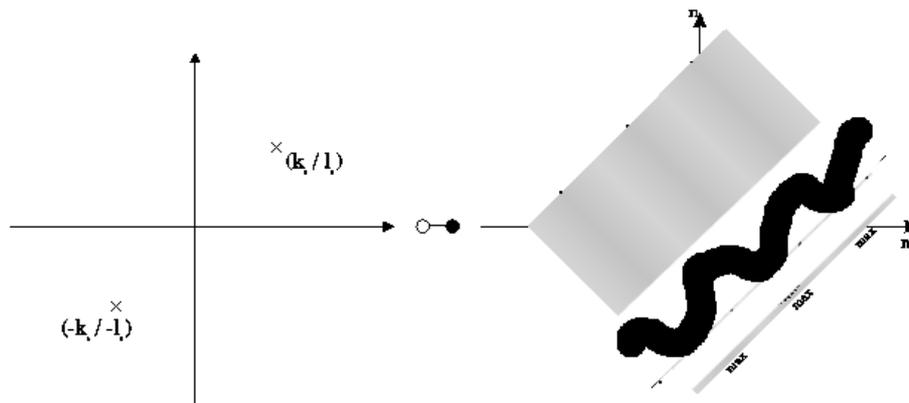
Wellenfront in x-Richtung

Ortsbereich	Ortsfrequenzbereich	
X Y	U V	kontinuierlich
m n	k l	diskret

Mit Rotation folgt:

Ein Spektrum ist eine Überlagerung von zum Ursprung symmetrischer  
Linienpaaren

Bild ist Überlagerung von  $\cos$ - und  $\sin$ -Wellen



### 2.3.6 Anwendungsbeispiel für Frequenzanalyse: jpeg-Codierung

Verlustbehafteter Codierungsalgorithmus für jpeg-Coding (gut für natürliche Bilder)

1. Input Farben rgb => yuv (Grauwert und Farbkanaldifferenzen), Anordnung des Bildes in  $8 \cdot 8$  Blöcke
2. DCT-Discrete Cosine Transformation auf  $8 \cdot 8$ -Bilder entspricht DFT, bei der gedanklich Signal gespiegelt wird, damit  $f(n) = f(-n)$ . Bisher noch kein Informationsverlust!
3. Quantisierung: Frequenzkoeffizienten werden quantisiert (entspricht "quality factor"), Gleichanteil bekommt viele Bits, hohe Frequenzen bekommen wenig Bits und werden damit oft 0 (quantisierte kleine Werte)
4. Codierung: Reihenfolge der Koeffizienten: Erst Niederfrequente Koeff., hinten höherfrequente, danach Lauflängenkodierung (run-length coding), was Datengröße reduziert ("ab jetzt 10 mal 0").

jpeg2000 ist eine Erweiterung von jpeg und benutzt Wavelets und variable Fenstergrößen.

## 2.4 Quantisierung

- Grauwertbilder

Weber-Fechnersches Gesetz:  $\frac{\Delta L}{L} = \text{const.}$

Hierbei steht L für die Beleuchtungsstärke  $\Rightarrow$  größte Empfindlichkeit in dunklen Bildbereichen.

CCD, CMOS haben lineare Kennlinien  $\Rightarrow$  typischerweise 8 bit (256 Graustufen). Um dies zu umgehen wird die Lichtmenge gesteuert, indem durch Shutter, Gain und Blende z.B. die Belichtungszeit geändert wird.

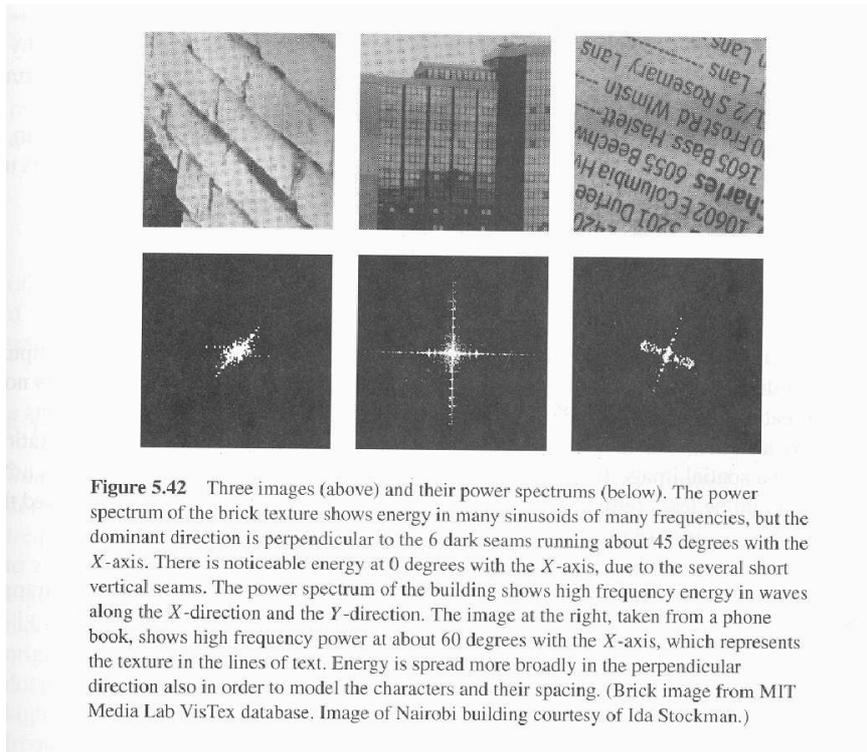
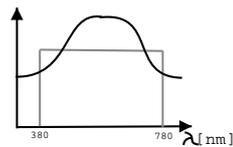


Abbildung 2.3: Beispiele für natürliche Fourierspektren

- **Farbbilder**

Das menschliche Auge hat folgendes Farbspektrum:



R-G-B-Darstellung, je 8  $\frac{\text{bit}}{\text{Farbkanal}}$   
 Farbwürfel:



---

## Kapitel 3

# Bildverbesserung

### 3.1 Punktoperatoren

Punktoperatoren modifizieren die Farbwerte, hier die Grauwerte, eines Bildes. Ein Punktoperator wird eingesetzt, um das Bild zu korrigieren und zu optimieren, z.B. in Hinsicht auf die Beleuchtung, zur Kontrastverstärkung und -dehnung, zur Bildmittelung, zur Korrektur inhomogener Beleuchtung, etc.

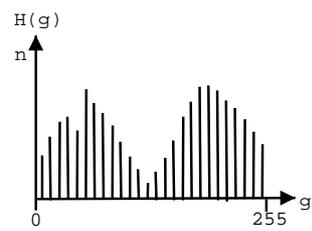
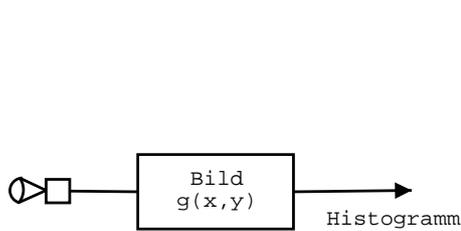
Punktoperatoren sind i.A. nicht umkehrbar, was bedeutet, dass die ursprüngliche Bildinformation verloren geht. Siehe [1], Seite 257.

#### 3.1.1 Transformation der Grauwertkennlinie

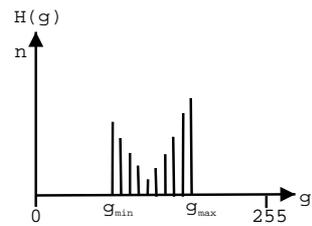
##### Lineare Transformation

Für eine Bildverbesserung wird hierbei erstmal ein Histogramm eines Bildes bestimmt. Ein Histogramm ist die Verteilung der Grauwerte im Bild - es listet auf, welcher Grauwert wie oft im Bild vorkommt. Ein Histogramm entspricht einer Statistik.

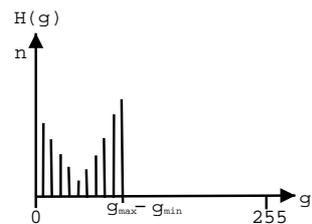
Eine Erhöhung des Kontrastes bedeutet, wenn man die Grauverteilung eines Histogramms nach links verschiebt und anschließend werden die Grauwerte verstärkt:  $g'_i = c(g_i - g_{min})$  mit  $c = \frac{255}{g_{max} - g_{min}}$ .  $g_{min}$  entspricht dem minimalen Wert unseres Histogramms, welcher dann auf 0 gesetzt wird. Die Verteilung wird dann gestreckt um den Faktor  $c$ , so dass  $g_{max}$  auf 255 endet. Dadurch bleiben manche Grauwerte unbesetzt.



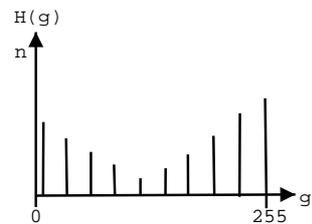
Beispiel: kontrastarmes Bild



Verbesserung  $\Downarrow$   
 1. Schritt: Verschiebung der Grauwerte im Histogramm, so dass das Histogramm beim Grauwert „0“ beginnt.



„Streckung“  $\Downarrow$   
 2. Schritt:  $g'_i = c(g_i - g_{min})$  mit  $c = \frac{255}{g_{max} - g_{min}}$



Achtung: manche Grauwerte bleiben unbesetzt!

---

## Kenngrößen

---

**Mittelwert eines Bildes**  $m = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M g(x, y)$   
 $= \frac{1}{MN} \sum_{i=0}^{255} i \cdot H(i)$  mit  $h(i)$  als Histogrammwert der Farbe

---

**Leistung**  $P = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M g^2(x, y)$   
 $\frac{1}{MN} \sum_{i=0}^{255} i^2 \cdot H(i)$

---

**Varianz  $\equiv$  Kontrast**  $\sigma^2 = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M (g(x, y) - m)^2$   
 $= \frac{1}{MN} \sum_{i=0}^{255} (i - m)^2 \cdot H(i)$   
 $= \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M (g^2 - 2mg + m^2)$   
 $P - 2m^2 + m^2 = P - m^2$

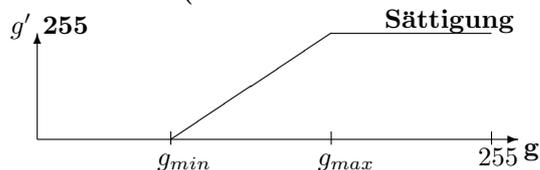
---

**Transformation:**  $m' = \bar{g}' = c(\bar{g} - \bar{g}_{min}) = c(m - g_{min})$  mit  $\bar{g} := \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M g(x, y)$

**Varianz:**  $\sigma'^2 = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M (cg - cg_{min} - m')^2 = \frac{1}{MN} c^2 \sum_{x=1}^N \sum_{y=1}^M (g - g_{min} - (m - g_{min}))^2$

**Daraus folgt:**  $\sigma'^2 = c^2 \sigma^2$

**T-Kennlinie (Transformation eines Histogramms):**



Realisierung durch eine Lookup-Table LUT. Eine LUT speichert die Ergebnisse von Bildoperationen für Grauwerte. Bei 8 Bit Farbtiefe werden die Ergebnisse für die 256 möglichen Farben in der LUT gespeichert. Soll nun ein Bild verändert werden, so wird für jeden Pixel in die LUT geschaut und der entsprechende Wert für den Originalwert des Bildes herausgeholt.

Da in einem Bild normalerweise ein Farbwert mehrfach vorkommt, wird durch einen Zugriff auf die LUT die Berechnung nur einmal durchgeführt, anstatt nacheinander für jeden Pixel. Siehe [1], Seite 259.

```
int LUT[256]
g'=LUT[g]
```

**Anwendung:**

- Spreizung des Histogramms zum Ausnutzen der vollen Dynamik bei flauen (kontrastarmen) Bildern

- Sichtbarmachen von Strukturen mit geringer Dynamik

Ein Spreizung des Histogramms wird verwendet, um Unterschiede zwischen einzelnen Grauwerten in einem Farbbereich mehr hervorzuheben. Dies wird vor allem zur Bildaufbereitung für Drucke benötigt.

Nichtlineare Kennlinien:  $g' = c \log(1 + g) = c\sqrt{g}$   
 Dies ergibt eine Spreizung im dunklen Bereich

$g' = cx^2 = ce^x$  ergibt eine Spreizung im hellen Bereich

Binarisierung stellt einen Sprung im Histogramm dar: bis zu einem bestimmten Wert sind alle Werte weiß, danach schwarz.

Markierung eines relevanten Grauwertbereichs schneidet bestimmte Werte aus dem Histogramm raus.

### 3.1.2 Histogrammausgleich

Man möchte keine Interaktion haben, um eine T-Kennlinie zu finden. Daher wird ein automatisches Verfahren gesucht.

Die interaktive Festlegung der T-Kennlinie ist oft unbefriedigend. Ein automatisches Verfahren ist dabei erstrebenswert. Die T-Kennlinie ist eine Transformationskennlinie - sie gibt an, welcher Grauwert des Originalbildes in welchen Grauwert des bearbeiteten Bildes umgewandelt wird.

Ansatz: Grauwert histogramm nach Transformation möglichst gleichverteilt - jeder Grauwert kommt gleich häufig vor.

Im Kontinuierlichen entspricht  $h(g)$  einer Verteilungsdichtefunktion (VDF).

$$\int_0^{255} h(\varphi) d\varphi = 1$$

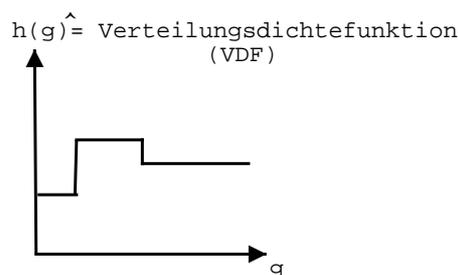


Abbildung 3.1: Verteilungsdichtefunktion

Normales Histogramm:  $\sum_{i=0}^{255} h(i) = 1$ ,  $h'(i) = \frac{H(i)}{MN}$

Verteilungsfunktion:  $VF(i) = \sum_{\varphi=0}^i h(\varphi)$ . Eine Verteilungsfunktion gibt an, wieviel Prozent der Pixel zwischen „0“ und „i“ liegen.

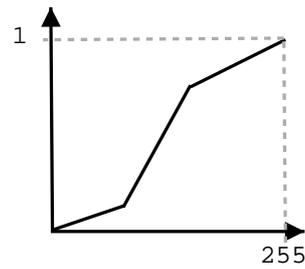


Abbildung 3.2: Verteilungsfunktion

Die Kennlinie kann aus der Verteilungsfunktion bestimmt werden:  $T(i) = 255 \cdot VF(i)$ .

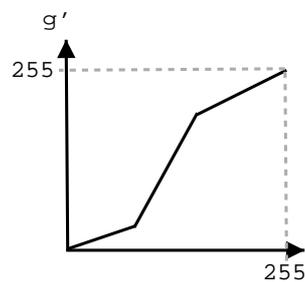


Abbildung 3.3: T-Kennlinie

Anmerkungen:

- Wegen des diskreten Charakters sind nicht alle Ausgangsgrauwerte besetzt, d.h. nur näherungsweise gleichverteiltes Ausgangssignal
- Die wiederholte Anwendung ist ohne Effekt.

### 3.1.3 Binarisierung

Motivation für eine Binarisierung:

- oft entscheidender Schritt zur Objektextraktion
- Trennung von Hintergrund

Ein Beispiel hierfür ist die Erkennung von Schrauben auf einem Fließband.

konstante Schwelle  $g' = \begin{cases} 255 & \text{für } g \geq s \\ 0 & \text{sonst} \end{cases}$

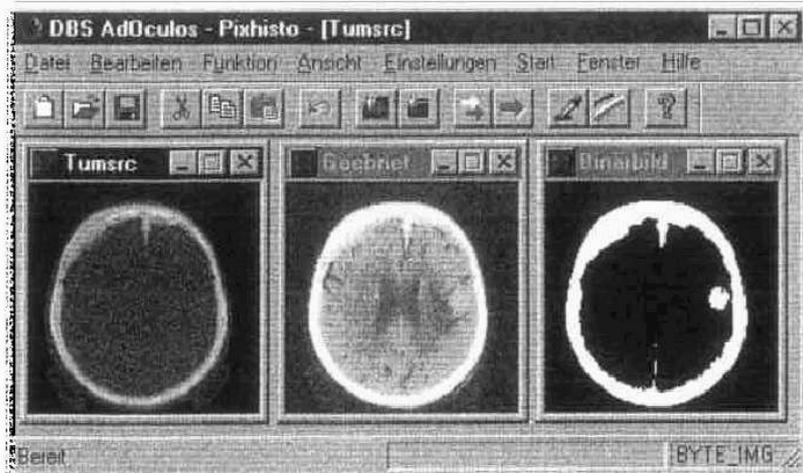


Abb. 4.25. Das Eingabebild ist das bereits bekannte Tomografiebild TUMSRC.IV. GEEBNET ist das Ergebnis der Funktion *Histogramm ebenen* mit den in Abb.4.23 gegebenen Parametern. BINÄRBILD ist das Ergebnis von *Binarisieren* mit den in Abb. 4.24 gegebenen Parametern.

Abbildung 3.4: Histogrammausgleich - ein Beispiel aus der Medizin

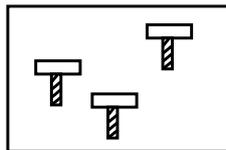


Abbildung 3.5: Schrauben auf dem Fließband

### Positionierung der Schwelle

Das Schwierige bei der Binarisierung ist die Positionierung der Schwelle. Der Idealfall eines abzutastenden Bildes besitzt eine bimodale Verteilung.

### Automatische Wahl der Schwelle

1. Statistischer Ansatz  $p(i) = \frac{h(i)}{MN}$

Klassenwahrscheinlichkeit  $P_0 = \sum_{i=0}^s p(i)$ ,  $P_1 = \sum_{i=s+1}^{255} p(i) = 1 - P_0$

Klassenmittelwerte  $m_0, m_1$

Mittelwert  $m = P_0 m_0 + P_1 m_1$

Varianzen  $\sigma_0^2 = \sum_{i=0}^s (i - m_0)^2 p(i)$ ,  $\sigma_1^2 = \sum_{i=s+1}^{255} (i - m_1)^2 p(i)$

Wähle  $s$  so, dass

$$\sigma_{inter}^2 = P_0 (m_0 - m)^2 + P_1 (m_1 - m)^2 \rightarrow \max$$

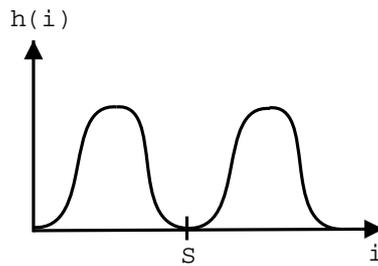
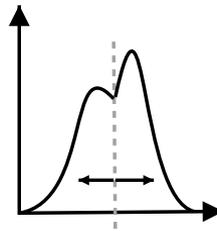


Abbildung 3.6: Idealfall: bimodale Verteilung



Es soll ein möglichst großer Abstand zwischen den Klassen sein. Die aufsummierten Wahrscheinlichkeiten der Klassen multipliziert mit dem Quadrat der Differenz des Klassenmittelwertes vom allgemeinen Mittelwert ungefähr gleich groß sind  $\Rightarrow$  Die beiden Werte aufsummiert sollten einen möglichst großen Wert ergeben

$$\sigma_{intra}^2 = P_0 \delta_0^2 + P_1 \delta_1^2 \leftarrow \min$$

$$\frac{\sigma_{inter}^2}{\sigma_{intra}^2} \leftarrow \max$$

Die Intra-Klassen-Varianz sollte möglichst klein sein, damit Unterschiede innerhalb der Klasse moderat ausfallen. Der Quotient  $quo = \sigma_{inter} / \sigma_{intra}$  soll möglichst groß werden. Bei Implementierung einfach alle Schwellwerte ausprobieren und sich die Schwelle  $s$  zum Maximum des Quotienten  $quo$  merken.

## 2. Berücksichtigung von Größeninformation

Der Vordergrund (Objekt) bedecke  $p\%$  der Bildfläche (Beispiel: Auf einem Fließband befinden sich  $7\%$  Schrauben).  $M$  und  $N$  entsprechen der Breite und Höhe des Bildes.

$$\sum_{i=0}^s p(i) \leq 1 - \frac{p}{100} \quad p(i) = \frac{1}{MN} h(i)$$

Die Wahl einer festen Schwelle ist im allgemeinen problematisch, wenn folgende Kriterien auftauchen:

- Helligkeitsabfall an den Rändern

- schräge Beleuchtung

Abhilfen dafür:

- Lokale Histogramme
- Subtraktion des Hintergrunds
- Korrektur von Beleuchtungsfehlern (bei bekannter Umgebung)

### 3.1.4 Kompensationsverfahren

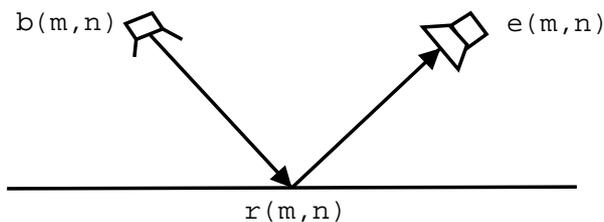


Abbildung 3.7: Schema einer Beleuchtung

**b:** Beleuchtungsfunktion  
**r:** Reflexionsfunktion  
**e:** Sensorempfindlichkeitsfunktion

**b** ist inhomogen wegen Winkel und inhomogene Abstrahlung, **e** ist inhomogen wegen der Optik und dem Sensorelement

Es resultiert:  $f(m, n) = r(m, n) \cdot \underbrace{b(m, n) \cdot e(m, n)}_{\text{eliminieren!}}$

Gesucht:  $r(m, n)$

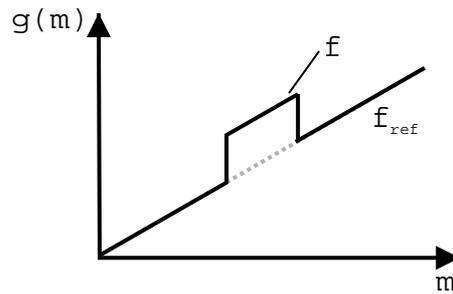
Ansatz: homogenes Testbild  $r(m, n) = K_1$ , liefert  $f_{ref} = K_1 \cdot b(m, n) \cdot e(m, n)$

damit ergibt sich:  $f'(m, n) = \frac{K_2 f(m, n)}{f_{ref}(m, n)} = \frac{K_2}{K_1} r(m, n)$

$K_1, K_2$  muss dabei geeignet gewählt werden ( $K_1$  grau (z.B. 128),  $K_2$  so, dass Dynamikumfang genutzt wird)!

Hintergrundsubtraktion (change detection)

Binarisierung unter Umständen einfach durch Subtraktion des bekannten Hintergrunds.



Wenn der Hintergrund bekannt ist, ist eine Verteilung im Bild bekannt. Eine Änderung erscheint im Bild als Zusatzsignal: die Grauwerte im Bild werden nach oben oder unten verändert. Um die Vordergrundinformation zu erhalten, wird vom neuen Bild das alte Bild (Referenzbild) abgezogen:  $f'(m, n) = f(m, n) - f_{ref}(m, n)$

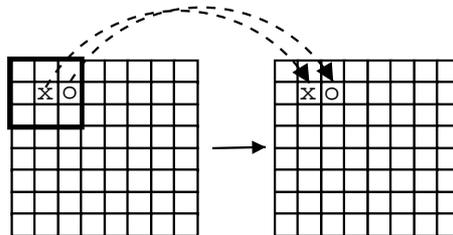
Probleme hierbei ergeben sich bei der Änderung der Beleuchtung. Dies kann man in den Griff kriegen, indem man  $f_{ref}$  adaptiert.

## 3.2 Lokale Operatoren

### 3.2.1 Lineare Glättung

Motivation: Glättung des Bildsignals

Bei der Glättung des Bildsignals soll der Mittelwert des Bildes erhalten bleiben. Die lineare Glättung entspricht grob der Anwendung eines Tiefpassfilters.



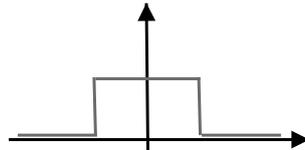
Dafür ergibt sich folgende Formel für die Mittelwertbildung:  $g_0 = \frac{1}{9} \sum_{i=0}^2 \sum_{j=0}^2 f(i, j)$ . Dies entspricht einem Rechteckfilter. Es werden de facto die Werte aller Bildpunkt in der Maske addiert und durch die Anzahl der Punkte geteilt. Dies ergibt den neuen Wert des mittleren Bildpunkts. Siehe [1], Seite 302.

Ansatz: Lokale Mittelwertbildung in einem Fenster, dass sukzessive vor das Bild geschoben wird.

Aus beispielsweise einem Quadrat von 3 Pixeln Seitenlänge wird der neue Wert des mittleren Pixels bestimmt.

- Ränder werden meist auf 0 gesetzt
- die Werte  $g(i, j)$  dürfen nicht direkt in  $f$  zurückgeschrieben werden
- Fenstergrößen im allgemeinen ungerade  $3 \times 3, 5 \times 5, 7 \times 7, \dots$
- Glättungswirkung steigt mit Filtergröße

### Signaltheoretische Betrachtung



$$g(i, j) = f(i, j) \times \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = f(i, j) \times h(i, j)$$

⇒ alle linearen, ortsunabhängigen Abbildungen lassen sich als Faltung formulieren

$$\begin{array}{c}
 \circ \\
 \bullet \\
 G(k, l) = F(k, l) \cdot H(k, l) \\
 \text{rect}\left(\frac{x}{3}, \frac{y}{3}\right) = \text{rect}\left(\frac{x}{3}\right) \cdot \text{rect}\left(\frac{y}{3}\right) \\
 \circ \\
 \bullet \\
 \text{si}(3 \cdot \pi u) \cdot \text{si}(3 \cdot \pi v)
 \end{array}$$

**Ähnlichkeitstheorem:** Dehnung der Ortsfunktion bewirkt Stauchung des Spektrums

**Rechenaufwand:**

**Faltungskern  $M \cdot M$  für einen einzelnen Bildpunkt:**  $M^2 \frac{\text{Additionen}}{\text{Bildpunkt}}$

**Nutzung der Separierbarkeit:** Der 2-dimensionale Filter wird in eine vertikale und

eine horizontale 1-dimensionale Komponente aufgeteilt.  $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$

Dieser Trick gelingt bei allen Faltungskernen, die sich auf 2 eindimensionale Faltungen zurückführen lassen.

Spezialfall: alle Koeffizienten gleich

Dies bezeichnet man als „sliding averaging“.

⇒ Aufwand der 2-dimensionalen Mittelung beträgt unabhängig von der Filtergröße  $4 \frac{\text{Additionen}}{\text{pixel}}$ . Randeffekte werden dabei vernachlässigt!

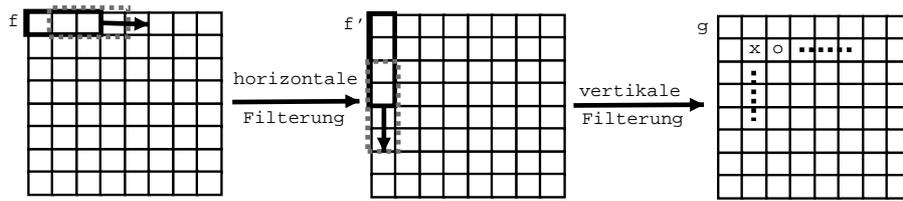
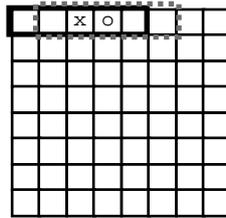
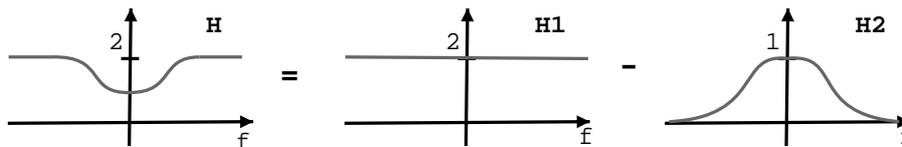


Abbildung 3.8: Grafische Darstellung der Separation



### 3.2.2 Bildverschärfung

Die Glättung eines Bildes wird durch einen Tiefpass (TP) realisiert. Um ein Bild zu schärfen, sprich die Kanten hervorzuheben, wird ein Hochpass (HP) verwendet.



$$h_{HP}(x) = 2\delta(x) - h_{TP}(x)$$

⇒ Eine Hochpassfilterung kann auf eine TP-Filterung zurückgeführt werden.  
Blockdiagramm:

Beispiel: 
$$\underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{6s(x,y)} - \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

### 3.2.3 Nichtlineare Operatoren

Lineare Operatoren verwischen Kanten. Darüber hinaus mischen sie in einem Bildpunkt Informationen aus Objekt und Hintergrund. Gleiches ergibt sich für Störungen im Bild.

Es wird nun ein Verfahren gesucht, welches das Bild glättet, dabei aber nicht über Objektkanten hinweggeht, so dass nur die Bildinformationen für ein Objekt zum Glätten des Objektes verwendet werden. Siehe [1], Seite 323.

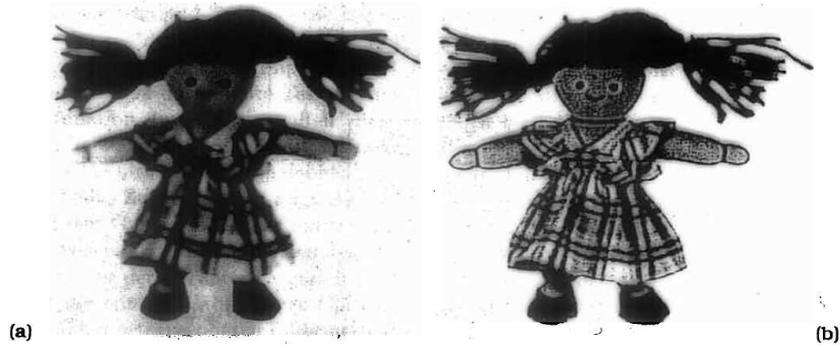
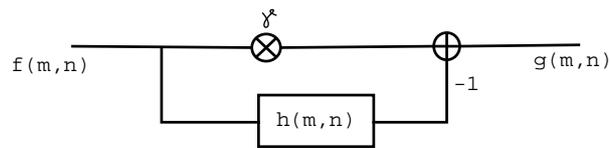


Bild: 3.11. Bildverschärfungsbeispiel mit lokalem Hochpaßfilter.

Abbildung 3.9: Beispiel für eine Hochpassoperation: links Originalbild, rechts mit Hochpass kantenverstärktes Resultatbild.

Annahme: durch Störung entstehen isolierte weiße Punkte. Was kann getan werden?

$$\text{MIN-Operator} \begin{bmatrix} 51 & 53 & 52 \\ 49 & 255 & 49 \\ 50 & 48 & 50 \end{bmatrix} \xrightarrow{\text{MIN}} \begin{bmatrix} \\ \\ 48 \end{bmatrix}$$

Der Pixel, welcher weiß ist, wird durch den minimalen Farbwert in der Matrix ersetzt - in diesem Beispiel wird die 255 durch die 48 ersetzt.

Der MIN-Operator weist dem Zentralpixel das Grauwert-Minimum innerhalb des Fensters zu.

Selbiges gilt bei schwarzen Bildpunkten in umgekehrter Reihenfolge:

$$\text{MAX-Operator} \begin{bmatrix} 51 & 53 & 52 \\ 49 & 0 & 49 \\ 50 & 48 & 50 \end{bmatrix} \xrightarrow{\text{MAX}} \begin{bmatrix} 53 \\ \\ 53 \end{bmatrix}$$

Der Pixel, welcher schwarz ist, wird durch den maximalen Farbwert in der Matrix ersetzt - in diesem Beispiel wird die 0 durch die 53 ersetzt.

Der MAX-Operator weist dem Zentralpixel das Grauwert-Maximum innerhalb des Fensters zu.

Wirkung: Eliminierung kleiner {heller / dunkler} Bereiche

Beide Operatoren machen nur in Ausnahmefällen Sinn!

---

## Medianfilterung

Medianfilterung ist eine Mischung aus dem MAX-Operator und dem MIN-Operator.

- Grauwerte innerhalb des Fensters sortieren

$$48,49,49,50, \underbrace{50}_{\text{Median}}, 51,52,53,255$$

255 wird hierbei durch den Median ersetzt

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 50 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Wirkung: Reduziert Rauschen, erhält Kanten.

Der Medianfilter eliminiert einzelne Impulse aus einer konstanten Nachbarschaft. Anhand definierter Schemata erkennt es Kanten, Rampen, etc. als Fixpunkte. Abweichungen von diesen Fixpunkten, die Impulse, werden ersetzt. Siehe [1], Seiten 324 und 325.

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Anmerkung: Das Verhalten nichtlinearer Operatoren lässt sich **NICHT** mit den Methoden der Systemtheorie beschreiben ( $\rightarrow$  statistische Methoden)

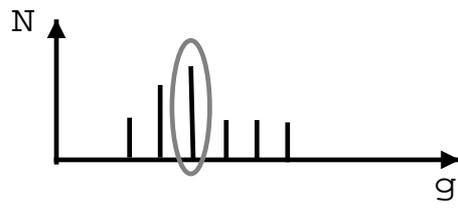
Regel: Gaußverteiltes Rauschen durch Mittelung (Tiefpassfilterung), Salz- und Pfefferrauschen (vereinzelte weiße und schwarze Punkte) (auch Binärrauschen genannt) durch Medianfilterung.

Bei der Anwendung einer Mittelung, sprich bei einem gaußverteilten Rauschen, gehen wir davon aus, dass jeder Pixel brauchbare Informationen enthält. Bei einem Salz- und Pfefferrauschen fehlt einigen Bildpunkten ihre ursprüngliche Information. Eine Mittelung hätte nun zur Folge, dass diese fehlerhaften Informationen auf die benachbarten Bildpunkte übertragen werden. Um eine solche Störung zu beseitigen müssen diese richtig detektiert und anschließend eliminiert werden. Dies versucht die Medianfilterung. Siehe [1], Seite 324.

### Schnelle Medianberechnung

#### 1. Diskreten Signalcharakter nutzen:

- Histogramm aufbauen
- $\frac{N+1}{2}$  Elemente herausgreifen



**2. Update: Sliding histogramming**

---

# Kapitel 4

## Konturen

- wichtig für das menschliche und maschinelle Sehen!

### 4.1 Konturpunktdetektion - Kantenpunktdetektion

Kantendetektion erkennt Veränderungen in einem Bildpunkt gegenüber seiner Nachbarschaft. Ein Problem bei der Kantendetektion ist dabei, dass Kanten nicht in alle Richtungen gleich gut erkannt werden.

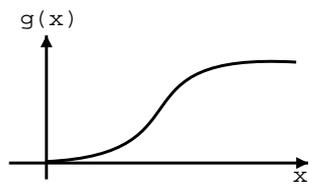
Bei der Kantendetektion wird ein sogenanntes „Merkmalsbild“ erzeugt, in welchem die Veränderungen hell, die anderen Bereiche entsprechend dunkel angezeigt werden.

Kantendetektoren sind Filter, welche Veränderungen verstärken und konstante Bereiche unterdrücken. Hierfür sind Ableitungsfunktionen geeignet. Siehe [1], Seiten 333 und 334.

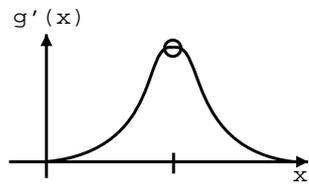
1D-Betrachtung

Auftretende Fragen:

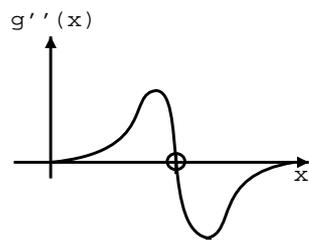
- wie detektiere ich den Sprung?
- wo lokalisiere ich den Sprung?



Ausgangssignal



Detektion/Lokalisation des Sprunges Var.1:  $\text{MAX}(g'(x))$



Detektion/Lokalisation des Sprunges Var.2: Nullstelle der 2. Ableitung.

Sprünge kann man detektieren, indem man lokale Sprünge in der Ableitung sucht:  
 $\text{MAX}(g'(x))$

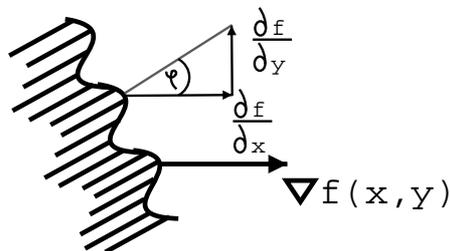
Um die Sprünge zu lokalisieren, kann man die Nullstelle der 2. Ableitung nehmen. Die 2. Ableitung wird für die genaue Detektion der Kante verwendet, wenn sich diese über mehrere Pixel erstreckt. In diesem Fall kennzeichnet die 2. Ableitung den Punkt der stärksten Steigung bzw. des stärksten Gefälles in der Kante, welcher dann zur Bestimmung der Kante auf einen Pixel genau genutzt wird. Siehe [1], Seite, 334.

## 2D-Betrachtung

1. Ableitung:  $\nabla f(x, y) = \frac{\partial f}{\partial x} \vec{e}_x + \frac{\partial f}{\partial y} \vec{e}_y = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$

Betrag:  $|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$

Richtung:  $\varphi(x, y) = \arctan\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right)$



2. Ableitung: Laplace-Operator  $\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$

---

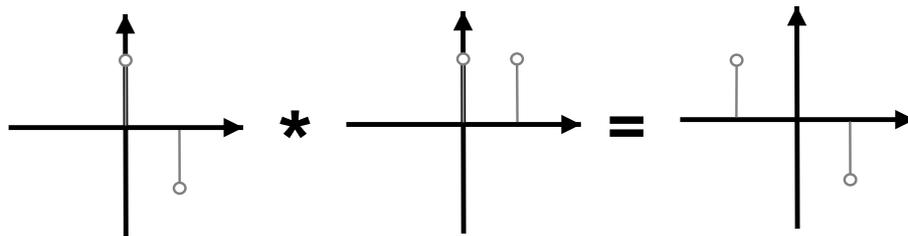
Konturpunkte:  $\nabla^2 f = 0$

Richtung durch Betrachtung von  $\varphi(x, y)$

Konturpunktdetektion der digitalen Bildverarbeitung beruht auf diskreter Approximation des Differentialquotienten.

$$\frac{\partial f}{\partial x} \approx \frac{f(x, y) - f(x - 1, y)}{1} \approx \frac{f(x + 1, y) - f(x - 1, y)}{2}$$

$\frac{\partial f}{\partial y}$  wird analog gerechnet:



Der Filter wird über die Matrix des Bildes bewegt. Für jeden Pixel wird die Umgebung gemäß der Matrix zusammengerechnet und das Ergebnis in den Zentralpixel geschrieben. man multipliziert die Werte der Filtermatrix mit den Werten des Bildes und schreibt das Ergebnis in den Zentralpixel.

Beispiel:

$$\text{Filter: } \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{Pixel und Umgebung: } \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} \Rightarrow \begin{matrix} 0 & 0 & 0 \\ 0 & -4 & 0 \\ 0 & 0 & 0 \end{matrix}$$

Als Rechnung:  $0 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 1 \cdot 0 + (-4) \cdot 1 + 1 \cdot 0 + 0 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 = -4$  im mittleren Pixel

$$\text{Filtermasken: } [-1 \ 1], \begin{bmatrix} -1 \\ 1 \end{bmatrix}; [-0,5 \ 0 \ 0,5], \begin{bmatrix} -0,5 \\ 0 \\ 0,5 \end{bmatrix}$$

$$2 \times 2 \text{ Filtermasken: } \frac{1}{2} \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}, \frac{1}{2} \begin{bmatrix} -1 & -1 \\ +1 & +1 \end{bmatrix} \text{ Prewitt-Operator}$$

$$\frac{1}{\sqrt{2}} \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \text{ Roberts-Operator}$$

$2 \times 2$  Filtermasken sind

- extrem lokal und
- haben keinen Zentralpixel

$3 \times 3$  Filtermasken:

---

Prewitt-Filtermaske zur Detektion von Gradientensteigungen (Kanten) in x-Richtung (von links nach rechts)	$\frac{1}{3}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$
Prewitt-Filtermaske zur Detektion von Gradientensteigungen (Kanten) in y-Richtung (von oben nach unten)	$\frac{1}{3}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Sobel-Operator für die x-Richtung (von links nach rechts)	$\frac{1}{4}$	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$
Sobel-Operator für die y-Richtung (von oben nach unten)	$\frac{1}{4}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

**3×3 Filtermasken sind**

- immer noch sehr lokal rauschempfindlich, daher problematisch in natürlicher Umgebung und
- die Winkelauflösung ist gering

Für die 3×3 Filtermasken spricht, dass sie extrem schnell sind.

Bei verrauschten Bildern müssen größere Filtermasken verwendet werden, damit stärkere TP-Filterung durch Berücksichtigung eines größeren Einflussbereiches.

$$5 \times 5 \text{ nach Prewitt: } \begin{bmatrix} -1 & -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 & 1 \\ -1 & -1 & 0 & 1 & 1 \end{bmatrix} \quad \text{nach Sobel: } \begin{bmatrix} -10 & -10 & 0 & 10 & 10 \\ -17 & -17 & 0 & 17 & 17 \\ -20 & -20 & 0 & 20 & 20 \\ -17 & -17 & 0 & 17 & 17 \\ -10 & -10 & 0 & 10 & 10 \end{bmatrix}$$

Sobel bildet in seiner Matrix eine Gaußkurve nach.

Effiziente Realisierung des Prewitt-Operators:



- ⇒ Schritt 1: gleitende Mittelwertbildung  
Schritt 2: Differenzbildung

**2D:** Aufwand  $4+1 \frac{\text{Additionen}}{\text{pixel}}$  unabhängig von Operatorgröße

Generierung des Betragsbildes:

Für die Berechnung des Betragsbildes wird für jeden Pixel der Gradientenbetrag bestimmt:  $\sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$ . Die Werte in den Klammern entsprechen dem Ergebnis der Filteroperationen.

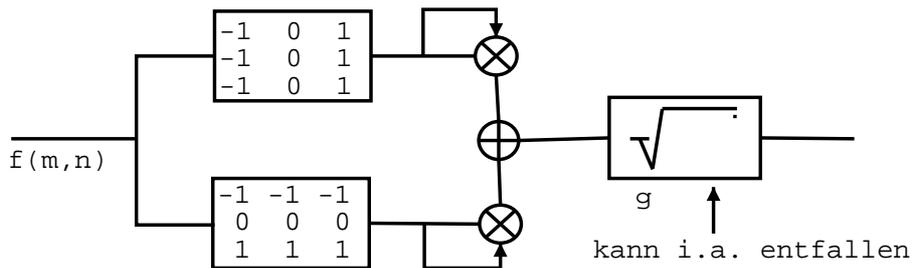


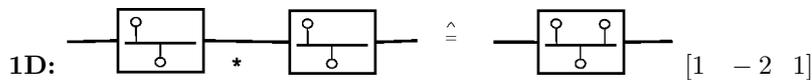
Abbildung 4.1:  $|\nabla f(m, n)| = g(m, n)$

Alternativen: statt den Gradientenbetrag zu errechnen ( $\sqrt{(\cdot)^2 + (\cdot)^2}$ ) kann man

- das Maximum der beiden Gradienten verwenden:  $g(m, n) = \{\text{MAX}\{|\frac{\partial f}{\partial x}|, |\frac{\partial f}{\partial y}|\}\}$
- Die Beträge der Gradienten addieren:  $g(m, n) = |\frac{\partial f}{\partial x}| + |\frac{\partial f}{\partial y}|$

Gradientenrichtung (die Richtung der Steigung):  $\varphi = \arctan\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right) \Rightarrow$  Tabellen

### Diskrete Ableitung 2. Ordnung



2D:  $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$  (Laplace-Filter)

$$\begin{bmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 \\ -2 & -2 & -2 \\ 1 & 1 & 1 \end{bmatrix}$$

- rotationsinvariant, kein  $\varphi$  (keine Winkelinformationen)
- 2. Ableitung verstärkt Rauschen  $\left(\frac{\partial^2}{\partial x^2}\right) g \circ \bullet (j2\pi f)^2 G$

$\Rightarrow$  Rauschen kann in kontrastarmen Bereichen eine Vielzahl von Nulldurchgängen erzeugen

Filterantworten bei idealer senkrechten Kante:

000	111
000	111
000	111

---

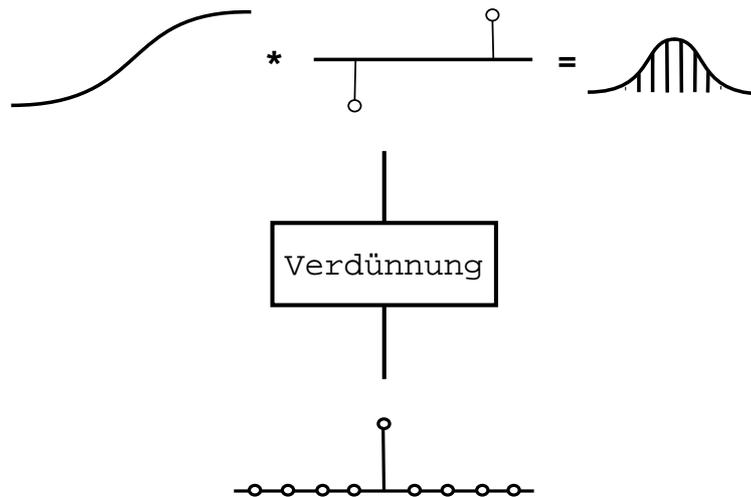

$$\begin{array}{l}
 \mathbf{2 \times 2 \text{ Prewitt:}} \quad \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{3 \times 3 \text{ Prewitt:}} \quad \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \\
 \\
 \mathbf{5 \times 5 \text{ Prewitt:}} \quad \begin{bmatrix} 0 & 1 & 2 & 2 & 1 & 0 \\ 0 & 1 & 2 & 2 & 1 & 0 \\ 0 & 1 & 2 & 2 & 1 & 0 \end{bmatrix}
 \end{array}$$

- Maxima erscheinen durch Tiefpassfilterung verbreitert
- Nulldurchgänge müssen gesucht und auf Signifikanz getestet werden
- eine ideale Kante befindet sich zwischen 2 Pixeln auf einem Gitter

In Abbildung 4.2 werden Betrags- und Richtungsbild in Unterabbildung 4.2. gezeigt.

## 4.2 Konturverdünnung

Motivation: nach Faltung mit  $N \times N$ -Operator erscheinen Konturen verbreitert, Ziel ist ein 1 Pixel breiter Strich.



### 4.2.1 Nonmaxima-Unterdrückung

Ziel ist es, alle Punkte, die nicht das Maximum sind, zu unterdrücken.

Im Eindimensionalen geht man über die Kante, merkt sich das Maximum, und setzt alle anderen Werte auf 0.

2D-Prinzip: Quer zur Kante die Nachbarn betrachten.

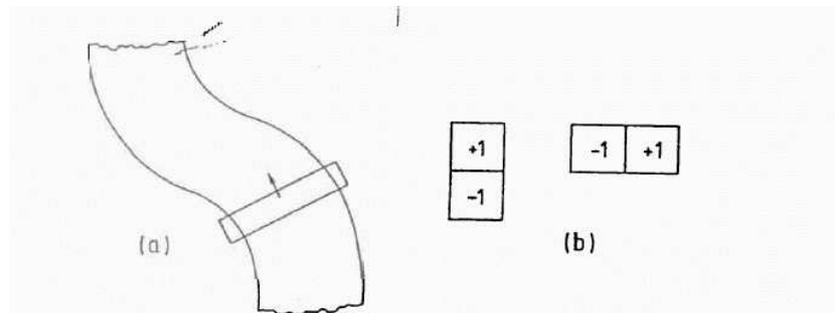


Bild 4.1. Zur Grundidee der Nonmaxima-Unterdrückung, (a) Quer zur Richtung der Kante wird das lokale Maximum gesucht, (b) Im Beispiel verwendete Masken

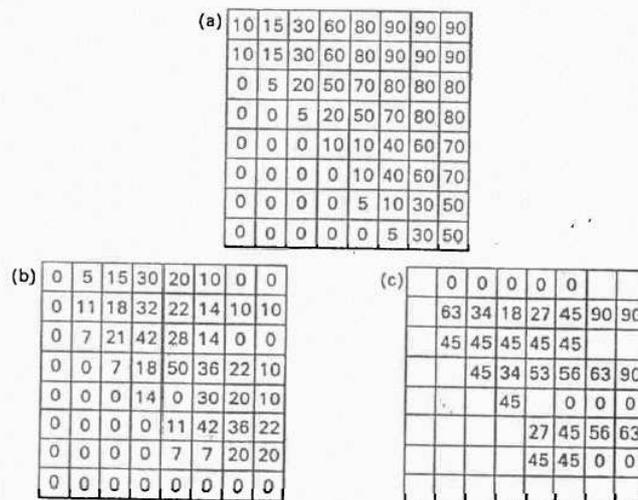


Bild 4.2. Beispiel einer Konturpunktdetektion, (a) Ursprungsbild, (b) Gradientenbetrag, (c) Gradientenrichtung

Abbildung 4.2: Nonmaximaunterdrückung

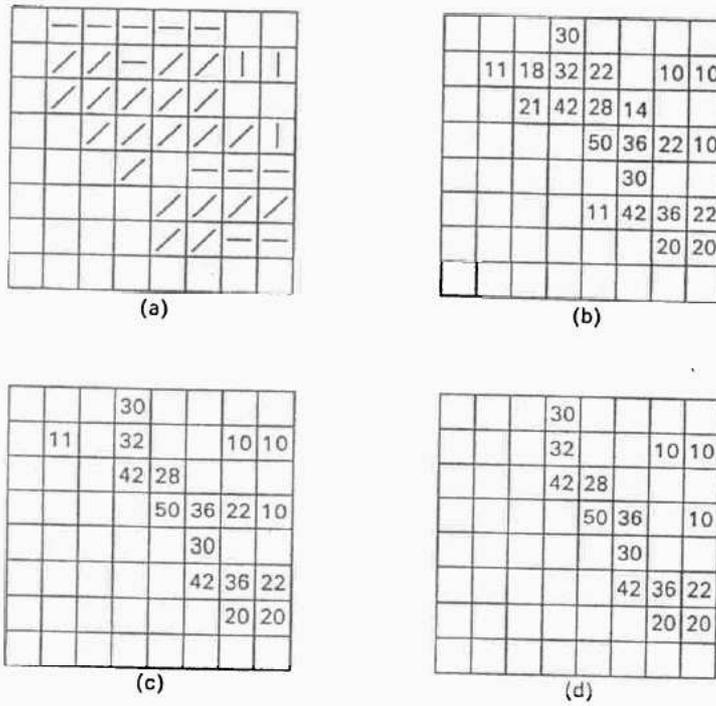


Bild 4.3. Verdünnung des Gradientenbetrages aus Bild 4.2 mit dem Basisalgorithmus, (a) Auswahl der Masken (vgl. Bild 4.4); (b) Toleranz  $\pm 15^\circ$ , (c) Toleranz  $\pm 45^\circ$ , (d) Toleranz  $\pm 90^\circ$

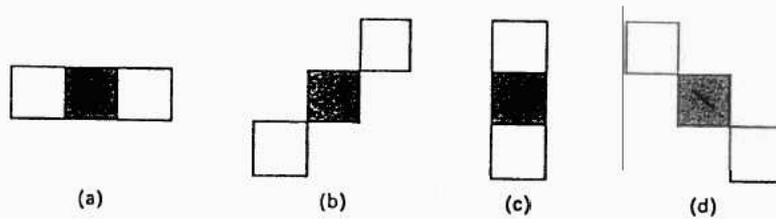
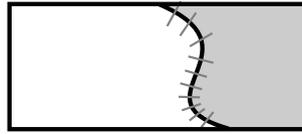


Bild 4.4. Die vier Masken des Basisalgorithmus

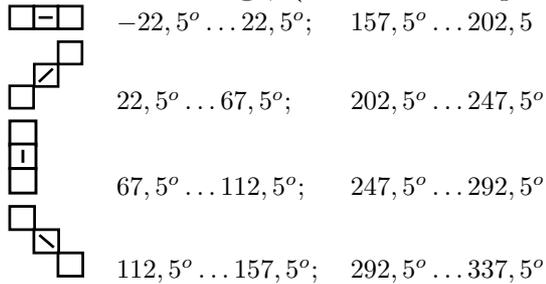
Abbildung 4.3: Konturverdünnung



Für alle Punkte:

1. Ermitteln der Nachbarn quer zur Kante:

Gradientenrichtung  $\phi$  (mathematisch positiv,  $0^\circ$  = waagrecht nach rechts))



2. Prüfe die Gradientenrichtungen der Pixel auf Kompatibilität.

Übliche Toleranzen:  $\underbrace{\pm 15^\circ, \pm 45^\circ}_{\text{glatt}}, \underbrace{\pm 90^\circ}_{\text{gekrümmt}}$

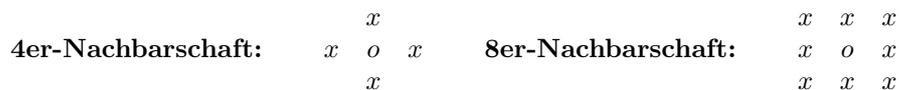
Bei Überschreitung der Toleranzen setze den Punkt im Zielbild.

3. Setze den Punkt im Zielbild, falls er betragsmäßig  $\geq$  als beide Nachbarn ist

4. Optional: setze den Bildpunkt nur, falls sein Gradientenbetrag über einer Schwelle  $S$  liegt (Rauschunterdrückung)

### 4.2.2 Übergang zur 8er Nachbarschaft

Motivation: Nonmaxima-Unterdrückung liefert Konturen mit 4er Nachbarschaft



In-Place-Algorithmus: Man geht über das Bild und löscht die Zentralpixel für bestimmte Konstellationen: Ecken. Es reicht uns dabei, anstatt einen Pixel in der 4er Nachbarschaft einen in der 8er Nachbarschaft zu finden.

### 4.3 Konturpunktverkettung

Ziel: Überführung des verdünnten Konturbildes in Konturlisten.

Extreme Datenreduktion, wesentlicher Schritt in der Bildverarbeitung

---

Typ: Ergebnis

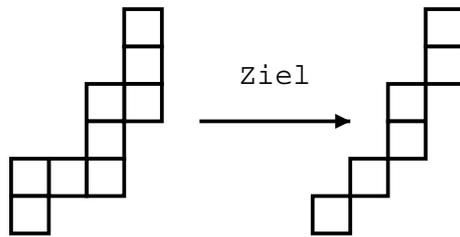


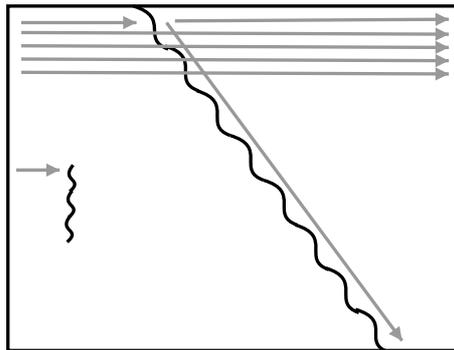
Abbildung 4.4: Anwendung des In-Place-Algorithmus auf eine Kante



Abbildung 4.5: x ist der zu löschende Zentralpixel

Prinzip:

- Scanne das Konturbild
- Wenn ein Konturpunkt gefunden wird, verfolge die Kontur bis zum Ende und lösche sie dabei



- Scanne weiter

Es existieren viele Abwandlungen bzgl. Schwellen, Binär- oder Betragsbild, Berücksichtigung der Winkelinformation etc. möglich.

Probleme breiten vor allem Berge  $\frown \rightarrow \smile$ , zerfallen in 2 Konturen  
Abhilfe: Laufe zunächst zum Konturende, starte dort

Übliche Variante:

1. Schwellentest mit Hysterese  
Akzeptiere Startpunkt mit  $|\text{Grad}| > S$

---

Akzeptiere Nachbarn als Konturpunkt, falls  $|\text{Grad}| > S_H$  ( $S_H < S$ ) -  $S_H$  ist die Hystereseschwelle  
Resultat: Chain-Code

## 4.4 Konturapproximation

Motivation: Weitere Informationsreduktion durch Approximation der Konturen durch einfache Funktionen (Geraden, Kreisbogen).

Man arbeitet hier nicht mehr auf den Bildern, sondern auf den zuvor gewonnenen Konturlisten.

Vorgehen:

Zuerst wird eine Gerade vom Anfang der Konturkette zu deren Ende gezogen. Dann wird der Punkt mit dem größten Abstand zur originalen Konturkette ausgemacht. Wenn dieser Abstand größer ist als eine vorgegebene maximal erlaubte Abweichung, wird die Gerade an diesem Punkt so geknickt, dass sie über diesen ausschlaggebenden Punkt umgeleitet wird. Es entsteht somit eine Gerade vom Anfangspunkt der Konturkette zu diesem Punkt. Nun wird der am weitesten von dieser entstandenen Gerade (bzw. ihrer virtuellen Verlängerung) entfernte Punkt gesucht. Dieser Vorgang wird so oft wiederholt, bis der größte Abstand der Approximation zur ursprünglichen Konturkette unter die maximale Abweichung fällt.

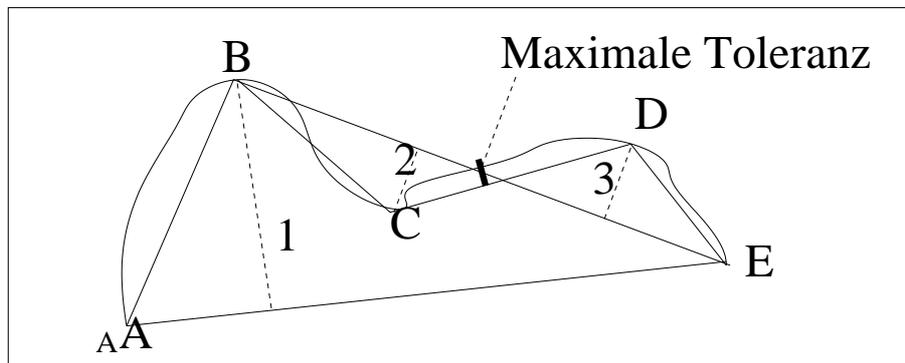


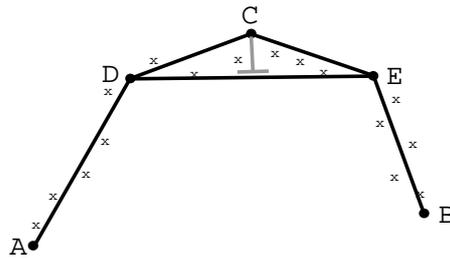
Abbildung 4.6: Ramer-Douglas-Peucker-Algorithmus

Sukzessives Splitten an dem Konturpunkt mit einem erlaubten maximalen Fehler, bis der Fehler tolerierbar ist.

Problem: Es können „unnötige“ Punkte entstehen.

Verbesserung: Split & Merge in einem zweiten Durchlauf

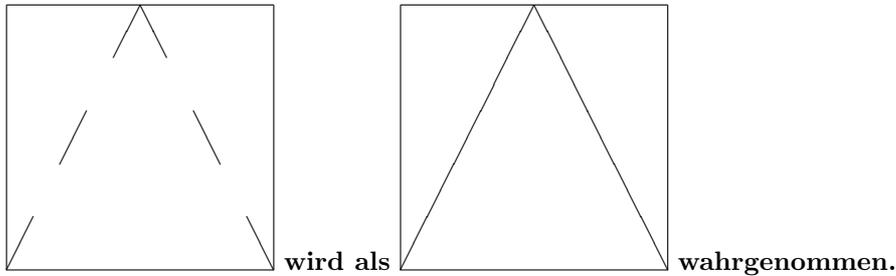
Man löscht Punkte, die nicht nötig sind, durch eine nochmalige Überprüfung der Fehlertoleranz.



## 4.5 Hough-Transformation

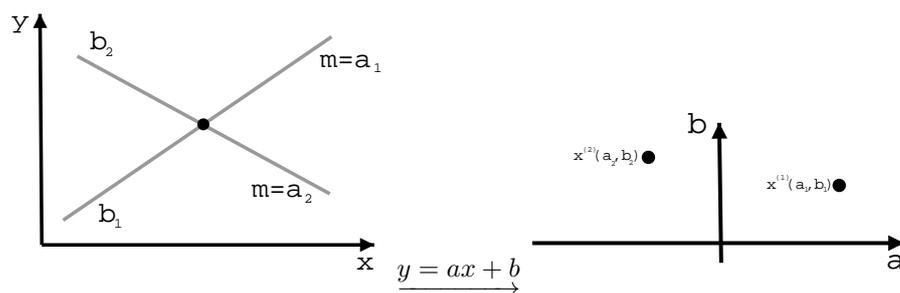
Die Hough-Transformation gehört zur modellbasierten Segmentierung. Segmentierung wird angewendet, um festzustellen, ob ein Bildpunkt zu einem bestimmten Objekt auf dem Bild gehört. Die modellbasierte Segmentierung sucht dabei nach bekannten Formen im Bild. Siehe [1], Seite 459.

Motivation: Kanten zerreißen in der Praxis



Lokale Verkettung nach Kapitel 4.3 kann die globalen Zusammenhänge nicht erfassen  
 $\Rightarrow$  HT erkennt globale Zusammenhänge.

### 4.5.1 Extraktion von Geraden



Alle Punkte einer Gerade  $([x_n, y_n])$  müssen die Bedingung  $y_n = a_0 + a_1 x_n$  erfüllen. Dies entspricht einer normalen Geradengleichung.  $a_0$  und  $a_1$  sind die Parameter der Kurve, also der Achsenabschnitt und die Steigung. Aus der Geradengleichung kann

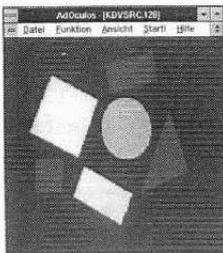


Abb. 6.8. Ursprungsbild für die Beispiele zur Kontursegmentierung

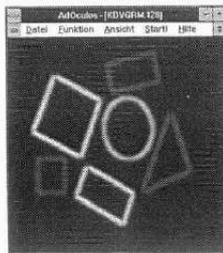


Abb. 6.9. Ergebnis der Gradientenoperation: Gradientenbetrag

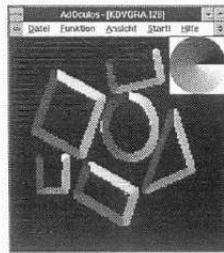


Abb. 6.10. Ergebnis der Gradientenoperation: Gradientenrichtung

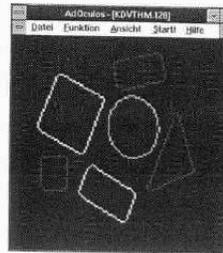


Abb. 6.11. Ergebnis der Verdünnung des Gradientenbetragsbildes



Abb. 6.12. Ergebnis der Verdünnung des Gradientenrichtungsbildes

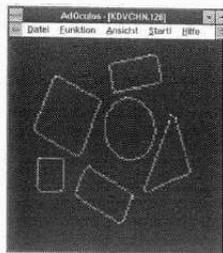


Abb. 6.13. Ergebnis der Verkettung

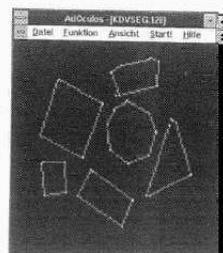


Abb. 6.14. Ergebnis der Approximation

Abbildung 4.7: Beispiel für eine Konturextraktion

eine Formel für die Parameter hergeleitet werden:

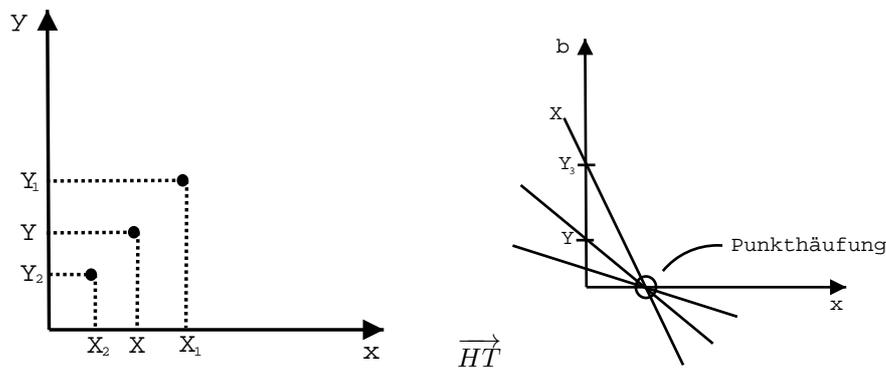
$$a_1 = \frac{y_n}{x_n} - \frac{1}{x_n} a_0$$

Mit Hilfe dieser Gleichung können alle Punkte des Datenraums im Modell- oder Parameter-Raum abgebildet werden. Der Parameterraum ist ein Koordinatensystem, in dem  $a_1$  in Abhängigkeit von  $a_0$  dargestellt wird. Siehe [1], Seiten 459 und 460.

Abbildung der parametrischen Kurve  $y = ax + b$  auf den Punkt  $(a/b)$  des Parameter-Raums.

Jedes Objekt im Datenraum wird auf ein anderes Objekt im Parameterraum abgebildet:

- Punkt  $\rightarrow$  Gerade
- Gerade  $\rightarrow$  Geradenbüschel + Punkthäufung



$$Y = a_i X + b_i$$

$$b_i = -a_i X + Y$$

$$b_i = -a_i X_2 + Y_2$$

Durch einen Kantenpunkt  $(X, Y)$  können Geraden mit  $Y = a_i X + b_i \leftrightarrow b_i = -a_i X + Y$  gehen

$\rightarrow$  die zu kollinearen Kantenpunkten gehörenden Parameter-Ortskurven schneiden sich in einem Punkt des Parameterraums. An diesem Punkt sind  $a_0$  und  $a_1$  für alle Punkte gleich.

Prinzip der Hough-Transformation

- Diskretisierung des Parameterraums  $\rightarrow$  Akkumulator

- $\forall$  Bildpunkte

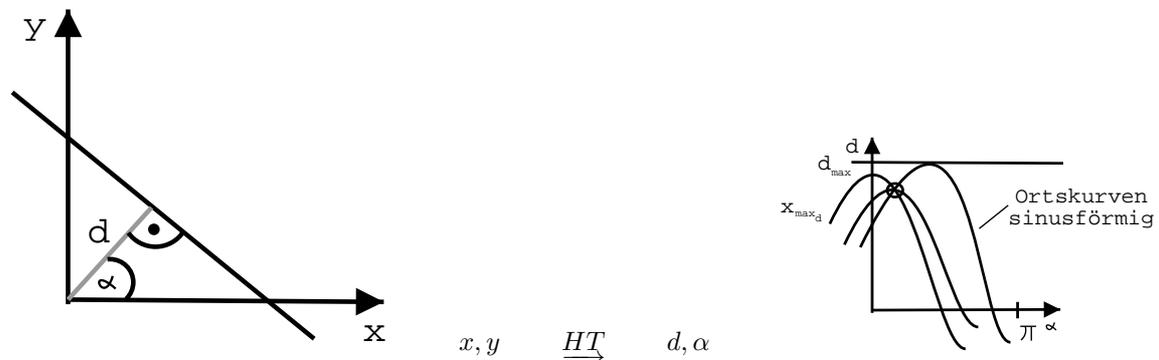
Für jeden Konturpunkt inkrementiere Akku-Zellen, durch die die Gerade  $b_i = Y - a_i X$  verläuft

- jedes lokale Maxima im Akku entspricht erkannten Geraden

Standard-Geradendarstellung  $y = ax + b$  bereitet Probleme wegen  $a \rightarrow \text{inf}$ . In diesem Fall kann die Gerade nicht in einen Parameterraum übertragen werden.

Besser: Hesse'sche Normalform  $d = x \cos(\alpha) + y \sin(\alpha)$   $x, y \xrightarrow{HT} d, \alpha$   
 $d, \alpha$  waren vorher  $[a, b]$

Hierbei werden anstatt einer Geradengleichung mit den Parametern „Achsenabschnitt“ und „Steigung“ die Parameter „Steigungswinkel“ und „Abstand vom Ursprung“ verwendet. Siehe [1], Seiten 460 und 461.



- Quantisierung des Parameterraums problemabhängig
- Eventuell nur ausgewählte Winkelbereiche berechnen

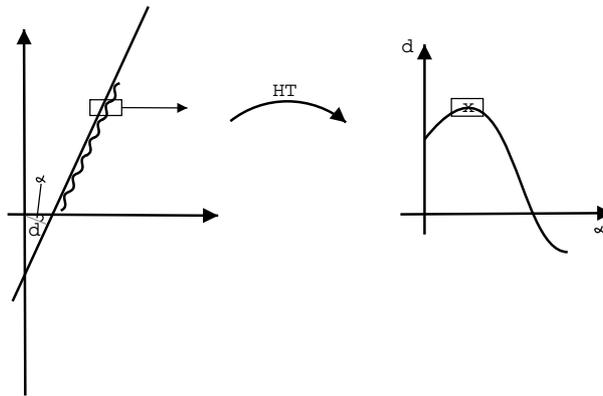
### Beschleunigte Hough-Transformation

Bei Berücksichtigung der Richtungsinfo kann gezielt eine einzige ( $\pm 2$ ) Akku-Zelle inkrementiert werden.

Beispiel:

### Bemerkungen

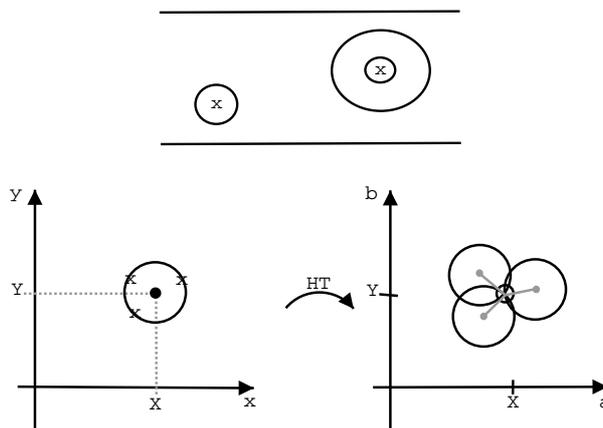
- Extrahiert werden Geraden, keine Strecken!  
Anfangs- und Endpunkte müssen in einem weiteren Schritt extrahiert werden.
- Die Stärke der Kante bleibt unberücksichtigt, zu kurze Kanten fallen unter Umständen unter eine Schwelle.  
Abhilfe: Betrag des Gradienten akkumulieren.



### 4.5.2 Hough-Transformation von Kreisen

**Kreis:**  $(x - a)^2 + (y - b)^2 = c^2$      **3 Parameter**  $\Rightarrow$  **3D**

**Vereinfachung hier:**  $c^2 = \text{konst.} = \text{bekannt}$



**Konturpunkt**  $(X_1, Y_1) \Rightarrow (X_1 - a)^2 + (Y_1 - b)^2 = c^2$  Ortskurve ist ein Kreis.

**Jeder Konturpunkt im Bild generiert einen Kreis im Parameterraum, die sich alle im Punkt  $(X, Y)$  schneiden.**

#### Bemerkungen

Nur praktikabel bei

1. einfachen Formen wie Geraden und Kreisen oder anderen einfach parametrisierbaren Formen  $y = f(x, \vec{p})$ . Mehr als 3 Parameter sind nicht sinnvoll.
2. einer nicht zu großen Anzahl der Konturen im Bild
3. geeigneter Quantisierung

- 
4. Nach Extraktion einer langen Geraden HT evtl. neu berechnen (Verdeckung im Parameterraum).

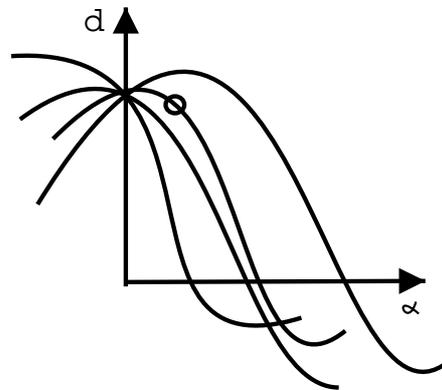


Abbildung 4.8: eine kleine Gerade wird u. U. übersehen

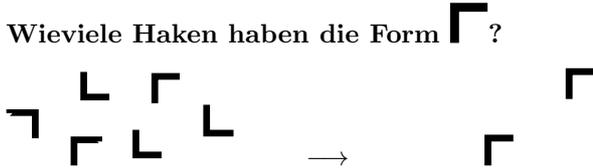
---

# Kapitel 5

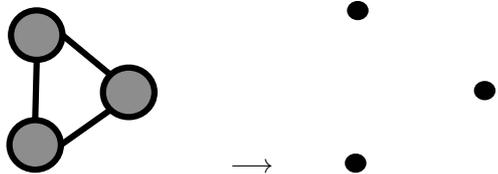
## Morphologische Bildverarbeitung

Motivation:

1. Wieviele Haken haben die Form  $\Gamma$  ?



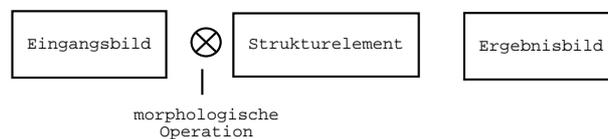
2. Wieviele Lötunkte im Bild?



Morphologie wird normalerweise auf binarisierte Bilder angewendet.

Idee: schiebe eine Maske über ein Bild und schaue, wo sie passt ( $\rightarrow$  Erosion).

Vorgehen bei der Morphologie

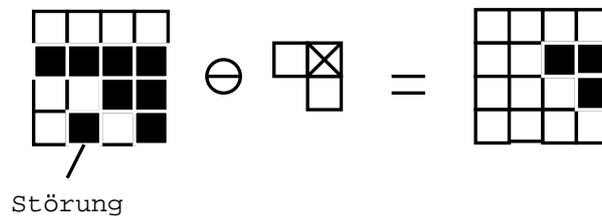


- Morphologische Operationen sind nichtlinear
- Strukturelemente werden auf das Problem angepasst.

---

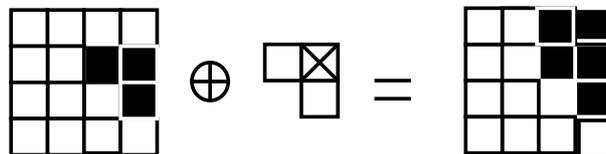
## 5.1 Erosion und Dilatation

### 1. Erosion: Schnittmenge von Bild + Strukturelement



Verschiebung des Referenzpunktes  $\Rightarrow$  Verschiebung des Resultats. Erosion verkleinert durch Abtragung.

### 2. Dilatation: Vereinigung von Bild und Strukturelement



Für alle Pixelpositionen: Wenn die Vereinigungsmenge von Bild und Strukturelement an Pixelposition nicht leer ist, setze Pixelposition im Ergebnisbild. Die Dilatation vergrößert das Bild.

### Übliche Strukturelemente

 für horizontale Linien

 für Flächen

 für Blobs ...

Beispiel:

## 5.2 Opening und Closing

### 1. Opening = Erosion mit nachfolgender Dilatation

Opening wird normalerweise mit dem gleichen Strukturelement ausgeführt.

Gerade Kanten kommen dabei zurück auf den Originalwert, Spitzen und Einbuchtungen werden abgerundet und gehen dadurch verloren, Lücken bleiben dabei erhalten.

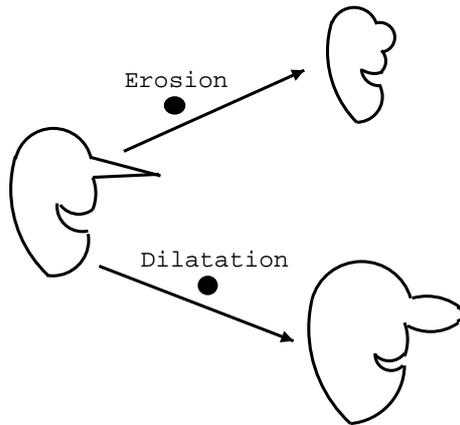
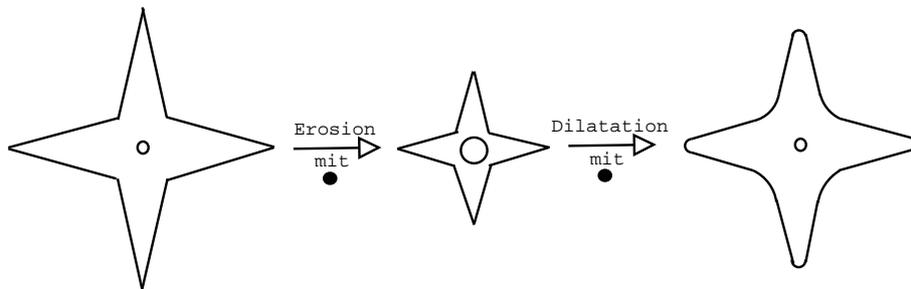


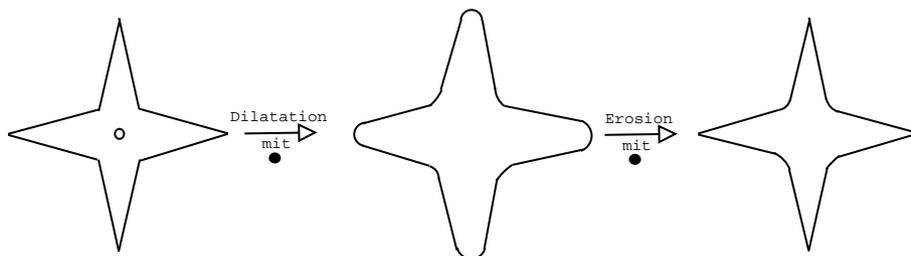
Abbildung 5.1: Modifikation eines Gesichtes



- Opening entfernt die Spitzen und ausgefranste Ränder
- Opening entfernt kleine Bereiche

2. Closing = Dilatation mit nachfolgender Erosion

Gerade Kanten und Spitzen kommen dabei auch zurück auf den Originalwert,



Einbuchtungen werden abgerundet, Lücken geschlossen. Es wird nur etwas geschlossen, wenn ein Bereich des Originalbildes übermalt wurde.

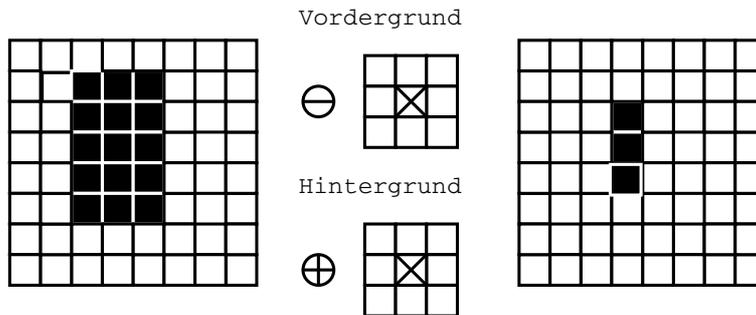
- Closing schließt Lücken

**Satz:** Opening mit Closing sind idempotente Operationen, d.h. eine wiederholte Anwendung ändert das Resultat nicht.

- Was tun, wenn das relevante Objekt schwarz ist?  
 Variante 1: Invertieren vor Morphologie  
 Variante 2: Dualismus nutzen

Satz: Erosion des Vordergrund  $\triangleq$  Dilatation des Hintergrund

Beispiel:



Satz: Opening auf dem Vordergrund  $\triangleq$  Closing auf Hintergrund

### 5.3 Detektion von Bildteilen bekannter Form

Schritt 1: Opening mit Strukturelement, dass das gesuchte Objekt sicher löscht  $\triangleq$  Hintergrundschätzung.

Dadurch erhalten wir nur den Hintergrund des Bildes.

Schritt 2: XOR von Ursprungsbild und Hintergrundbild

- liefert das Objekt
- Ränder
- Störungen

Schritt 3: Opening mit Strukturelement, dass das gesuchte Objekt nicht löscht.

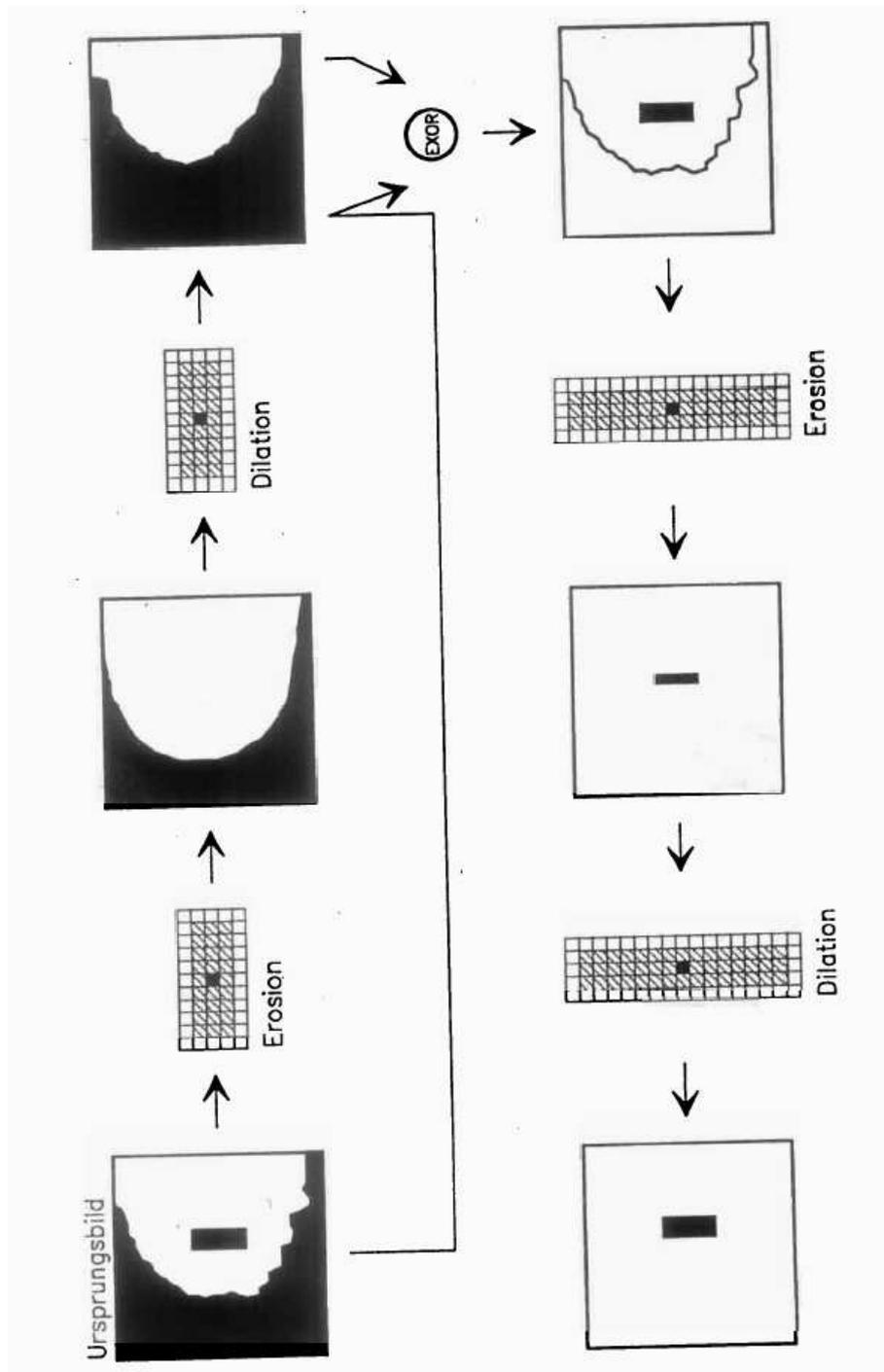
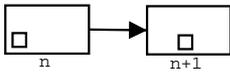


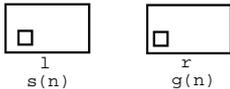
Abbildung 5.2: Detektion von Bildteilen bekannter Form

---

# Kapitel 6

## Schätzung von Verschiebungen

Motivation: Bewegungsschätzung  ...

Stereosehen 

Bilddatenkompression

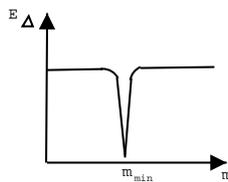
### 6.1 Korrelation

#### 6.1.1 Differenz

Einfaches Ähnlichkeitsmaß: Differenz, „Sum of squared differences“.

$$1D: E_{\Delta}(m) = \sum_{n=m-\frac{N}{2}}^{m+\frac{N}{2}} [s(n) - g(n-m)]^2 \rightarrow \min$$

N entspricht dabei der Maskengröße



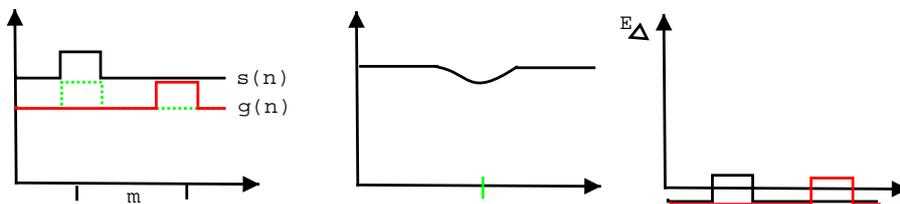


Verschieben der Referenzmaske und Berechnung der Ähnlichkeit  $\forall$  Verschiebungen  $(n,m)$ .

- Sicherheit steigt mit Maskengröße
- nur strukturierte Masken liefern verlässliche Ergebnisse  
Die Maske muss eine Struktur oder ein Muster enthalten, welches man in einem anderen Bild wiederfinden kann. Masken mit einem gleichmäßigen Hintergrund sind ungeeignet.
- Es sollten horizontale und vertikale Strukturen vorhanden sein  
Eine einzelne horizontale Struktur würde verhindern, dass man die Struktur in einem anderen Bild wiedererkennt.  
→Besonders geeignet für das Verfahren sind Punkte und Ecken.
- Randbedingungen (durch Vorwissen):
  - maximale Verschiebung in einer Bildfolge
  - maximale Verschiebung in einem Stereo-Bildpaar

### 6.1.2 Mittelwertbefreite Differenz

Problem: Unterschiedliche Mittelwerte



$$E_{\Delta}(m) = \sum_{n=m-\frac{N}{2}}^{n=m+\frac{N}{2}} [(s(n) - \bar{s}) - (g(n-m) - \bar{g})]^2$$

$\bar{s}$  und  $\bar{g}$  sind der Mittelwert im Fenster

### 6.1.3 Korrelationsfunktion

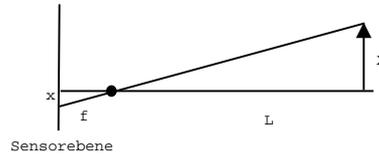
$$\bar{E}_{\Delta}(m) = \frac{\sum s(n) \cdot g(n-m)}{\sqrt{E_s} \sqrt{E_g}} \quad \epsilon[-1; 1]$$

$$E_s = \sum s^2(n)$$

---

## 6.2 Messen mit Kameras

### 6.2.1 Zentralprojektion



$$\begin{aligned} \frac{x}{f} &= \frac{X}{L} && \text{Strahlensatz} \\ s &= \frac{f}{L} X && [\text{in } \mu\text{m}] \\ 1 \text{ pixel} &= 10 \mu\text{m} = s_x \\ x[\text{pixel}] &= \frac{f}{L} X = \frac{f_x}{L} X \end{aligned}$$

$x$  und  $f$  entweder in pixel oder in metrischen Koordinaten (Entfernungen auf dem Imager),  $L$  ist Entfernung entlang der optischen Achse.

### Standard-Stereogeometrie

Für folgende Gleichungen siehe untenstehende Skizze:

$$\begin{aligned} \frac{f}{z} &= \frac{u_l}{x+0.5B} = \frac{u_r}{x-0.5B} \\ \frac{f}{z} &= \frac{u_l - u_r}{B} \\ \Leftrightarrow z &= \frac{f \cdot B}{u_l - u_r} \rightarrow \text{Disparität } D = u_l - u_r \\ y &= \frac{v_r \cdot z}{f} = \frac{v_l \cdot z}{f} \\ x &= \frac{u_l \cdot z}{f} - \frac{B}{2} = \frac{u_r \cdot z}{f} + \frac{B}{2} \end{aligned}$$

Interpretation: Wie weit ist das Objekt bezüglich meinem Kamerasystem versetzt ( $x, y, z$ )?  $B$ : Basisbreite, Entfernung der Kameras voneinander

$$(U, V) \rightarrow (X, Y, Z)$$

- Korrespondenzsuche nur in gleicher Zeile  $V$  (bei Standardstereogeometrie), allgemein: Suche entlang einer durch Kameraparameter festgelegten im Bild liegenden Linie (Epipolargeometrie)!

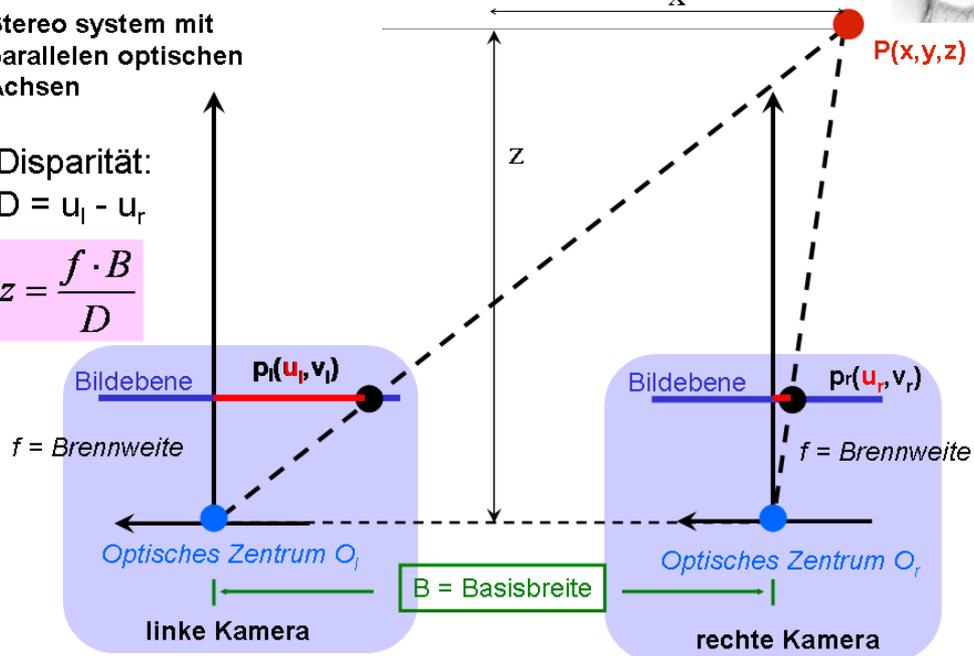
# Einfaches Stereo - Disparitätsgleichung



Stereo system mit  
parallelen optischen  
Achsen

Disparität:  
 $D = u_l - u_r$

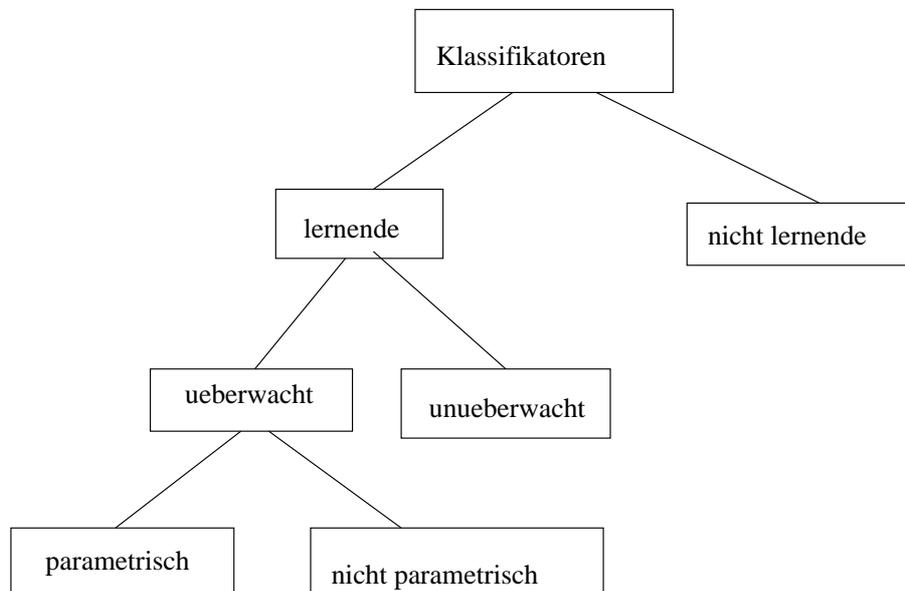
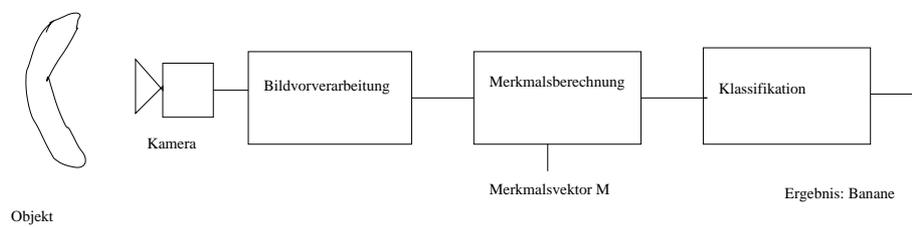
$$z = \frac{f \cdot B}{D}$$



---

## Kapitel 7

# Klassifikation



- Merkmalsvektoren spannen einen N-dimensionalen Merkmalsraum auf
- Lernen anhand einer Lernstichprobe
- Verifikation anhand einer Teststichprobe

---

# Literaturverzeichnis

- [1] Bernd Jähne, Digitale Bildverarbeitung 5. Auflage, ISBN 3-540-41260-3, Springer Verlag