

Computergrafik 2: Kanten, Linien, Ecken

Prof. Dr. Michael Rohs, Dipl.-Inform. Sven Kratz

michael.rohs@ifi.lmu.de

MHCI Lab, LMU München

Folien teilweise von Andreas Butz, sowie von Klaus D. Tönnies
(Grundlagen der Bildverarbeitung. Pearson Studium, 2005)

Themen heute

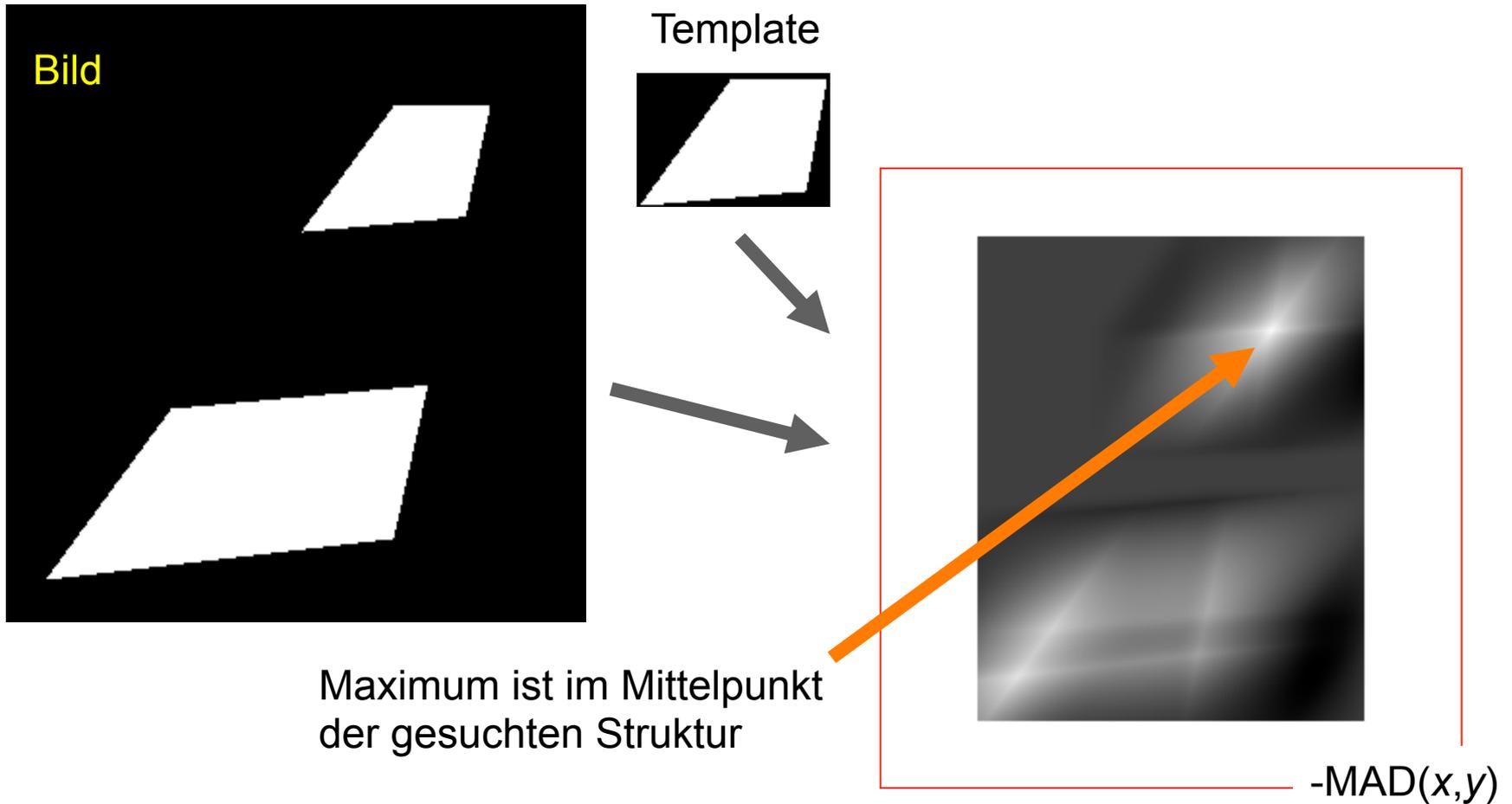
- Kanten
 - Kantenoperatoren
 - Kantenattribute
 - Kanten und Rauschen
- Canny-Kantenerkennung
- Harris-Eckenerkennung
- Hough-Transformation

NACHTRAG: TEMPLATE MATCHING DURCH KORRELATION

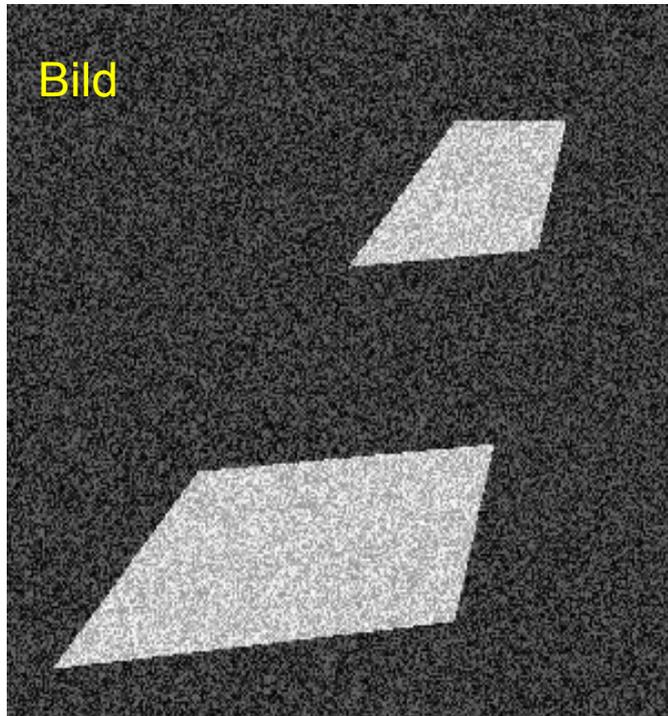
Template Matching

- Gezielte Suche nach einem vorgegebenen Objekt (Template / Modell)
- Für jeden Punkt im Bild wird ein Abstandsmaß zwischen Form und Bild berechnet
- Position des Objekts = Punkt, an dem Maß extrem ist
- Abstandsmaße zwischen Bild $f(x,y)$ und Template $t(x,y)$
 - Mean absolute difference:
$$\text{MAD}(x,y) = 1 / MN \cdot \sum_{xy} |f(x+m,y+n) - t(x,y)| \quad \leftarrow \text{Minimum}$$
 - Sum of squares:
$$\text{SSQ}(x,y) = \sum_{xy} (f(x+m,y+n) - t(x,y))^2 \quad \leftarrow \text{Minimum}$$
 - Korrelationskoeffizient: $cc(x,y) \quad \leftarrow \text{Maximum}$

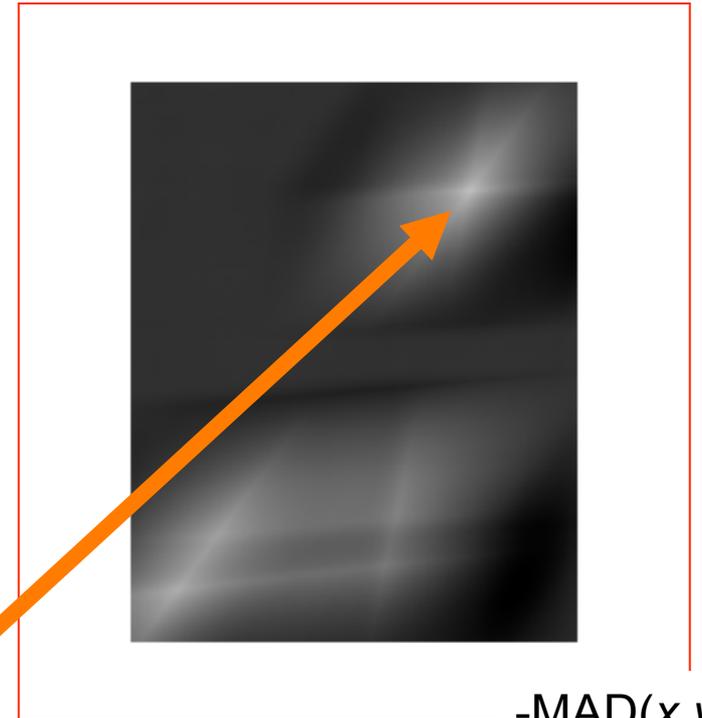
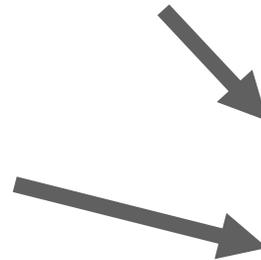
Template Matching



Template Matching – Rauschen



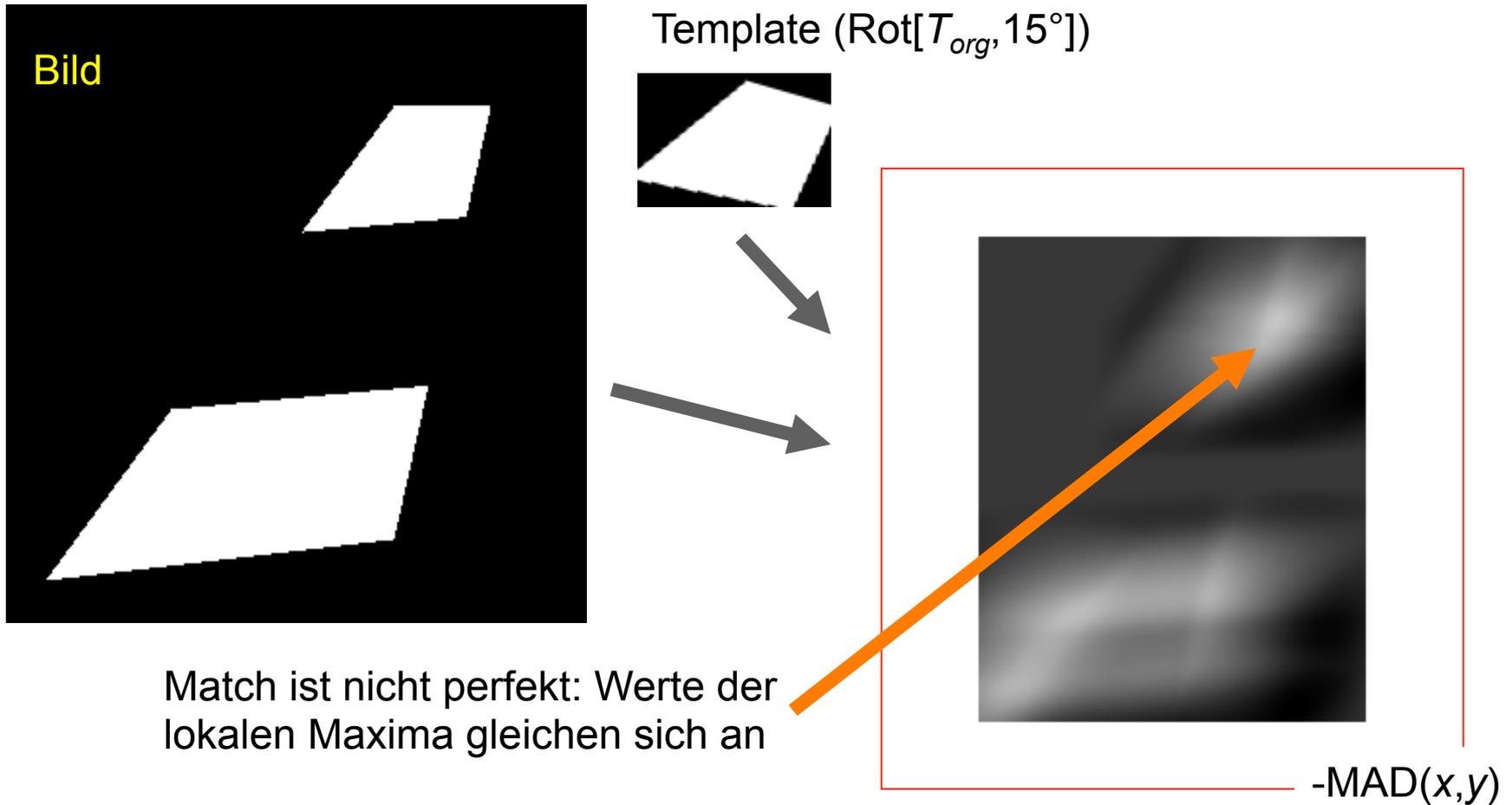
Template T_{org}



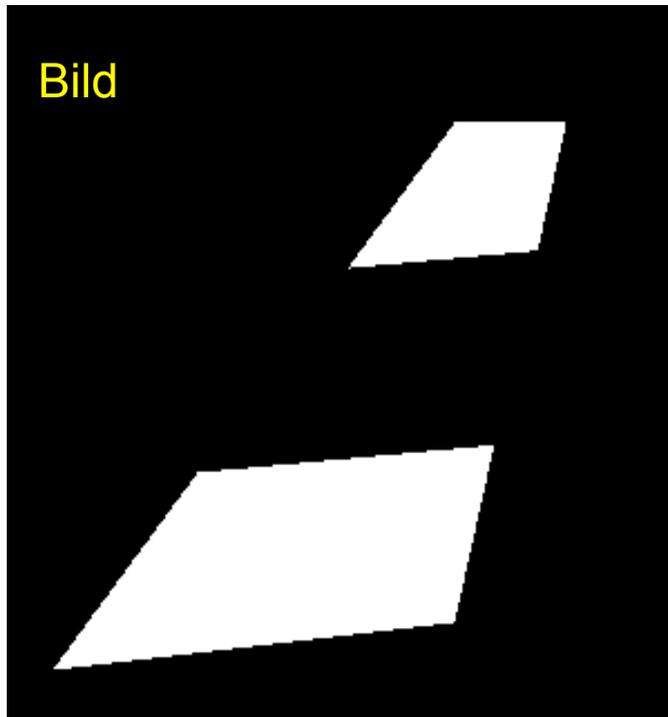
Unempfindlich gegenüber
Rauschen (Summation)

$-\text{MAD}(x,y)$

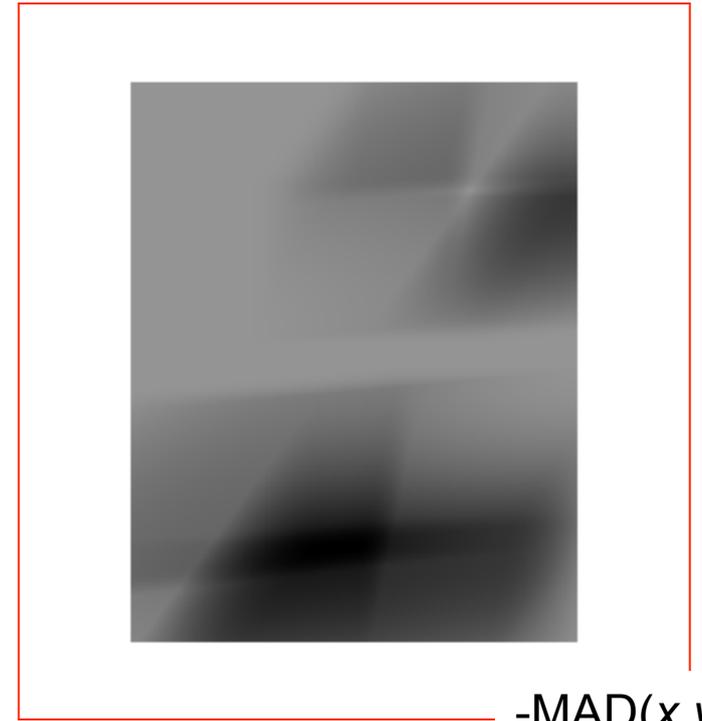
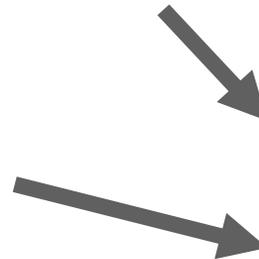
Template Matching – Probleme



Template Matching - Probleme

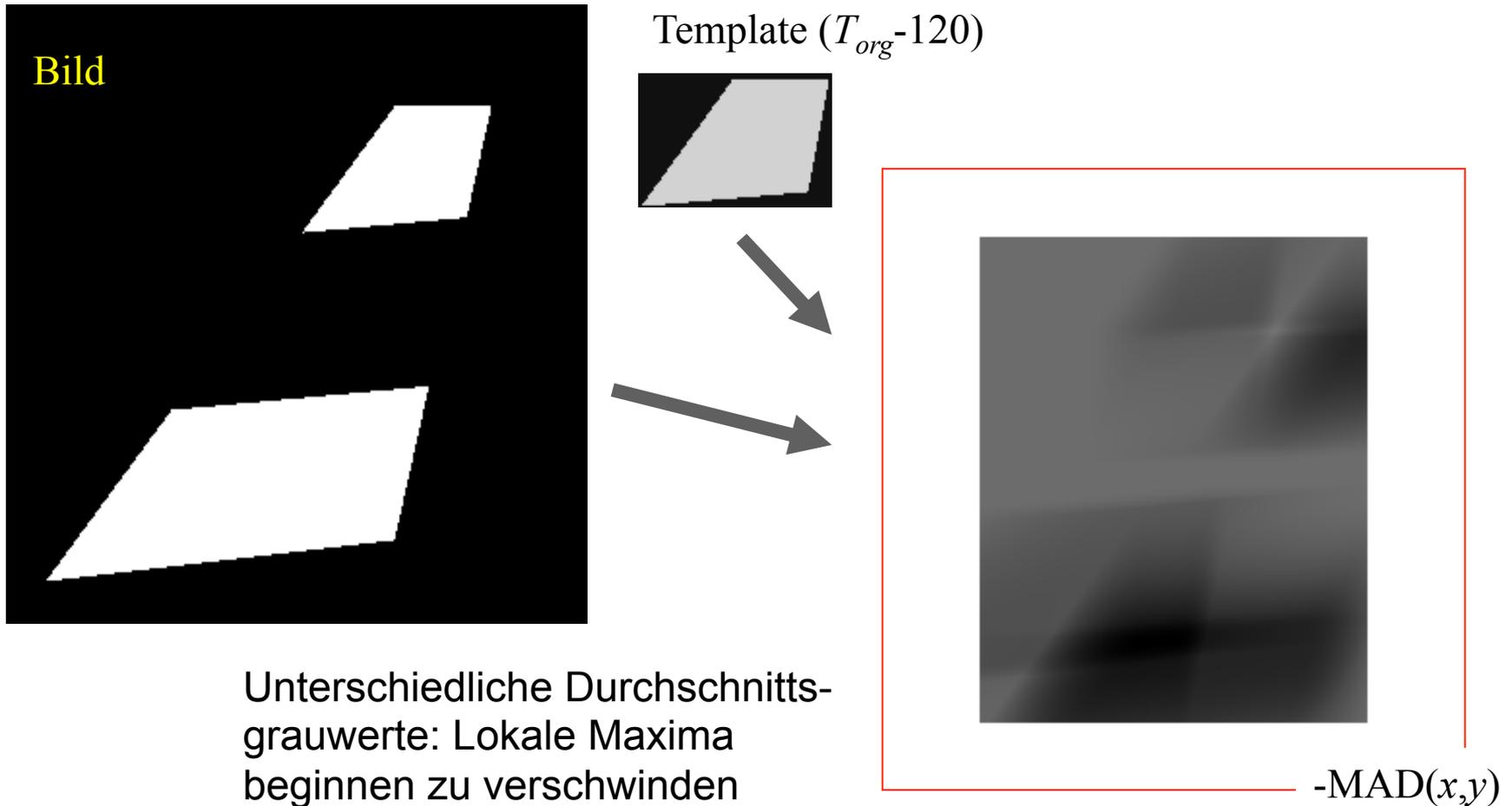


Template ($0.5 \cdot T_{org}$)



Unterschiedlich skalierte Grauwerte des Templates: Lokale Maxima beginnen zu verschwinden

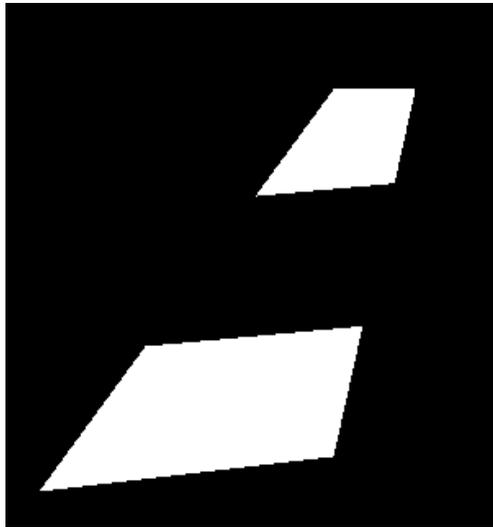
Template Matching - Probleme



Korrelationskoeffizient

- Korrelationskoeffizient: $cc = \frac{\sigma(f, g)}{\sigma(f) \cdot \sigma(g)}$
 - 1 \leq cc \leq 1
 - $\sigma(f)$, $\sigma(g)$: Standardabweichungen von f und g
 - $\sigma(f, g)$: Kovarianz zwischen f und g ist
- Werte des Korrelationskoeffizienten
 - 1, falls es einen positiven Skalar s gibt, so dass $f = s \cdot g$
 - 1, falls es einen negativen Skalar s gibt, so dass $f = s \cdot g$
 - 0, falls f und g voneinander linear unabhängig sind
- als Ähnlichkeitsmaß unempfindlich gegenüber Skalierungen und Mittelwertveränderungen des Templates

Template Matching mit Korrelationskoeffizient



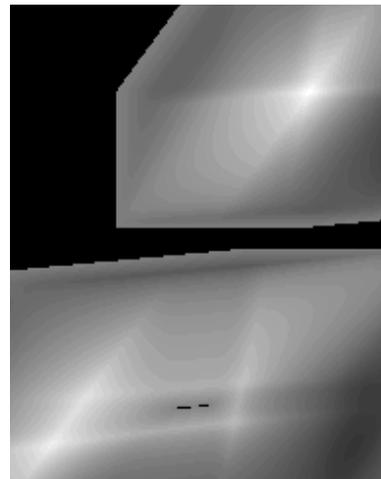
T_{org}



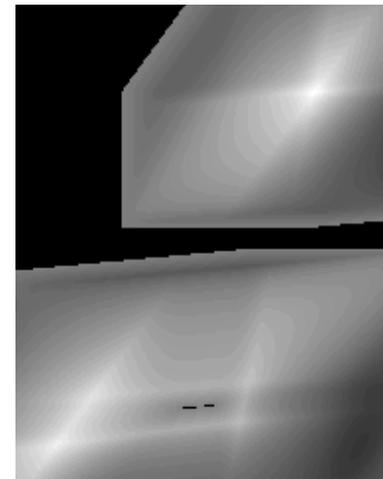
$0.5 T_{org}$



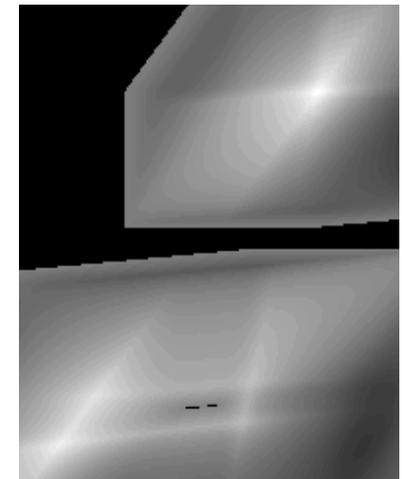
$T_{org} - 120$



(a)



(b)



(c)

Resultate der Korrelation mit (a) optimalen, (b) skalierten und (c) geminderten Templates

Erkennung von Merkmalen

- Merkmal (engl. „feature“): wichtiges/interessantes lokales Muster
- Erkennung von Merkmalen wichtiger Schritt beim Lokalisieren oder Erkennen von Objekten in Bildern
- Beispiele für Merkmale
 - Kanten
 - Linien
 - Punkte
 - Ecken
 - Spezifische Muster
 - Texturen

KANTEN, LINIEN, PUNKTE

Kanten, Linien, Punkte

- Starke lokale Intensitätsänderungen
 - „Ableitung“
- Stärke der Intensitätsänderung
 - Betrag / Gradient der Ableitungsfunktion
- Diskrete Bilder
 - Ableitung angenähert durch Differenzenquotient

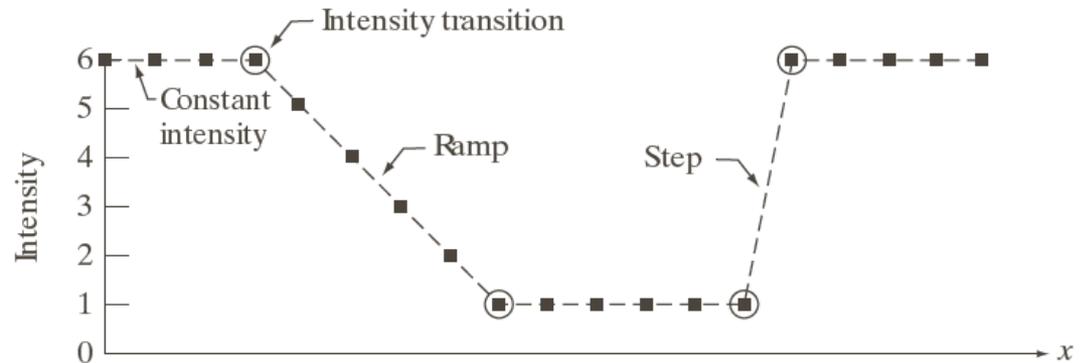
Erste und zweite Ableitung von Bildern

- Erste Ableitung:

$$\frac{\partial f}{\partial x}(x) = f(x+1) - f(x)$$

- Zweite Ableitung:

$$\begin{aligned} \frac{\partial^2 f}{\partial^2 x}(x) &= \frac{\partial f}{\partial x}(x+1) - \frac{\partial f}{\partial x}(x) \\ &= (f(x+1) - f(x)) \\ &\quad - (f(x) - f(x-1)) \\ &= f(x+1) + f(x-1) - 2f(x) \end{aligned}$$



Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6
1st derivative	0	0	-1	-1	-1	-1	-1	0	0	0	0	0	0	5	0	0	0	0	0
2nd derivative	0	0	-1	0	0	0	0	1	0	0	0	0	0	5	-5	0	0	0	0

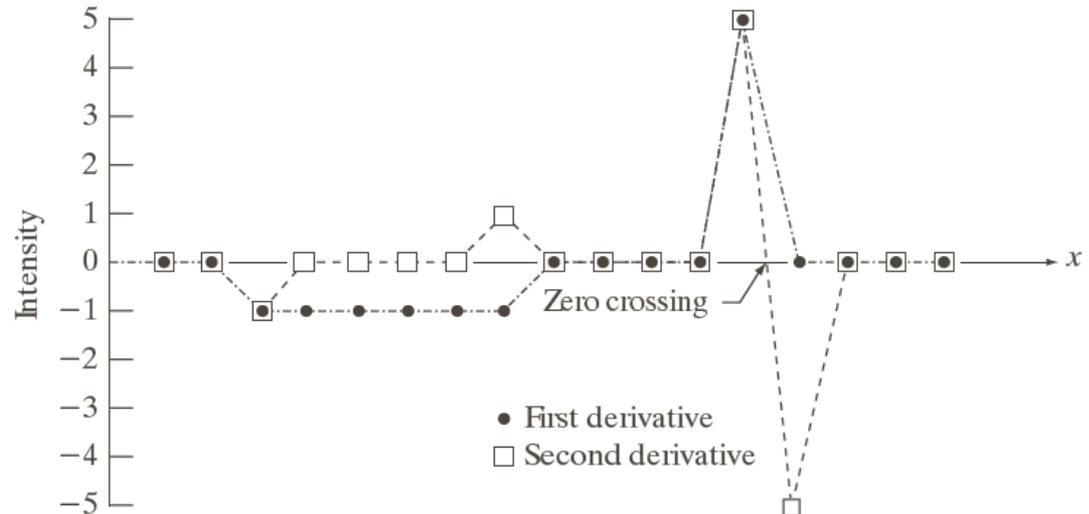
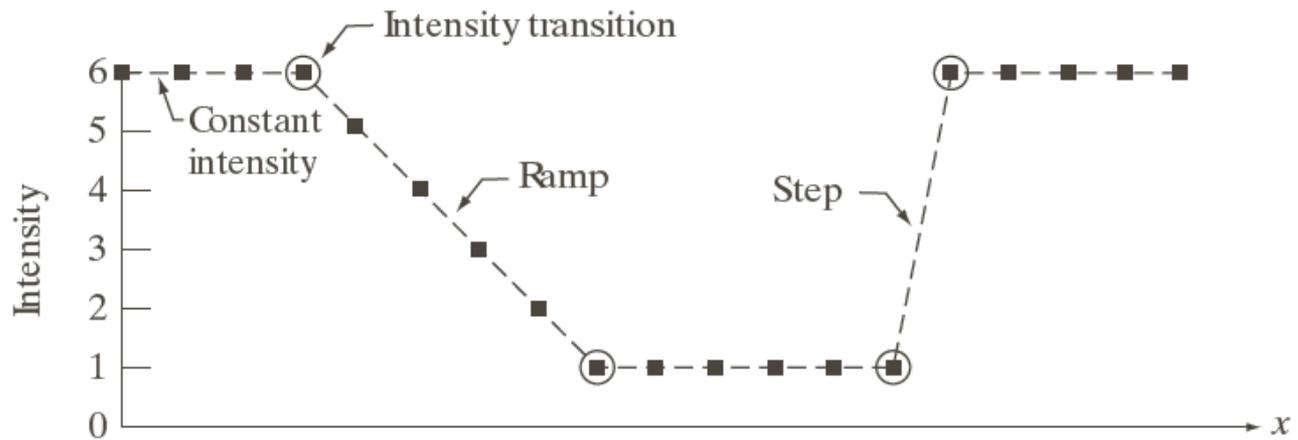


Abbildung: © R. C. Gonzalez & R. E. Woods, Digital Image Processing



Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6
1st derivative	0	0	-1	-1	-1	-1	-1	0	0	0	0	0	0	5	0	0	0	0	0
2nd derivative	0	0	-1	0	0	0	0	1	0	0	0	0	0	5	-5	0	0	0	0

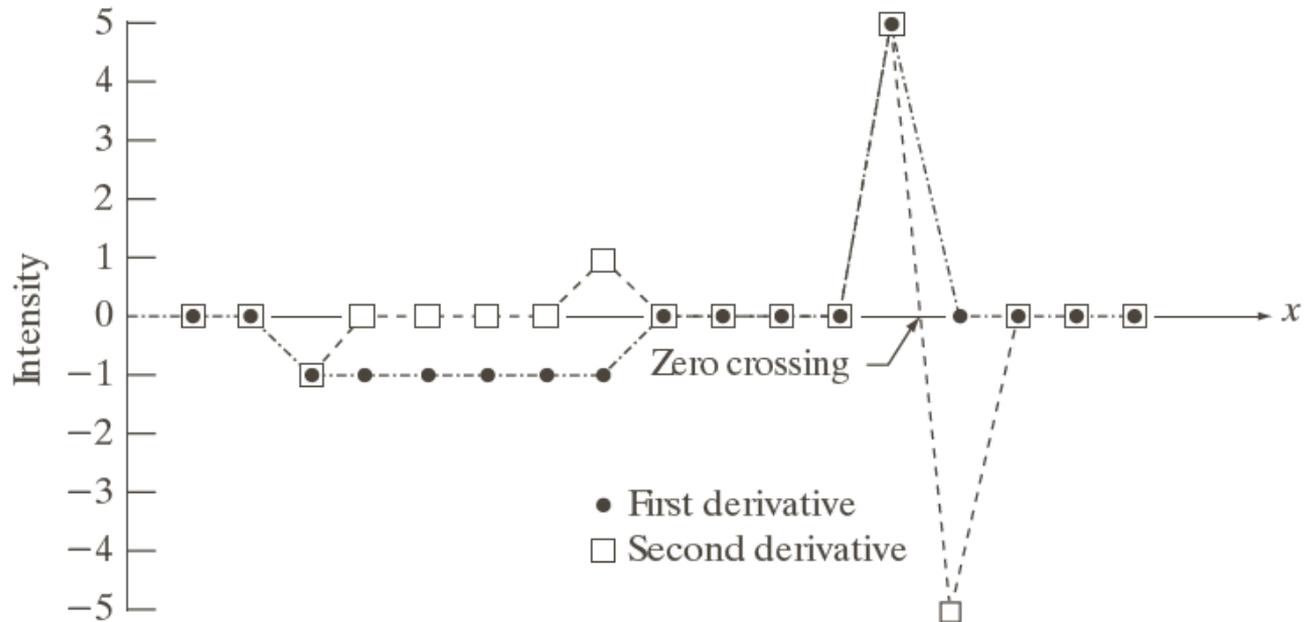


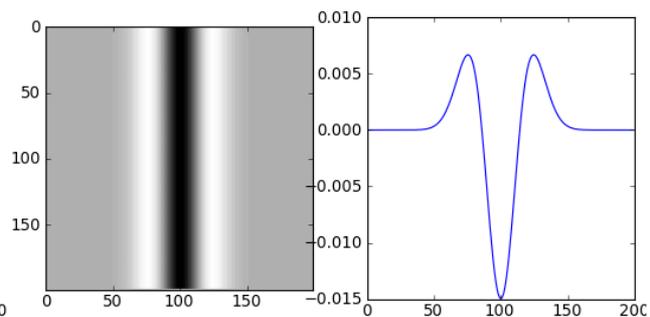
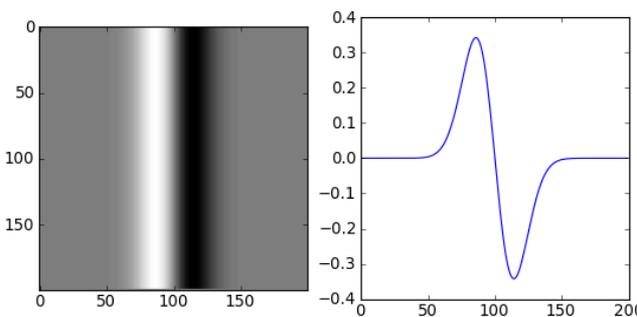
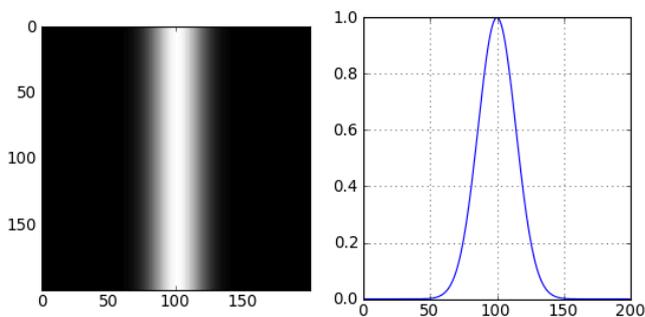
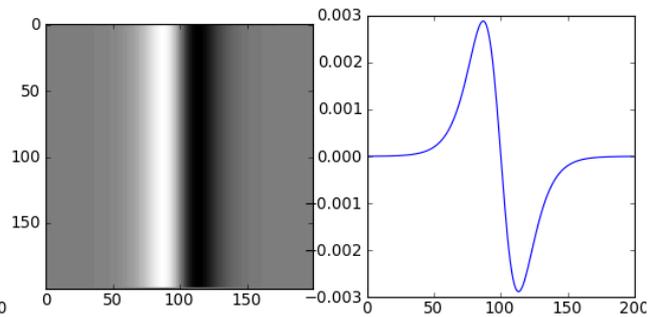
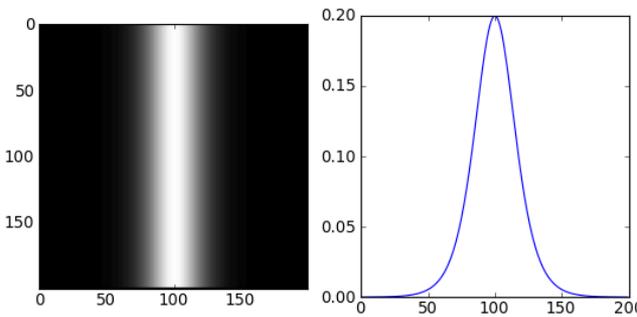
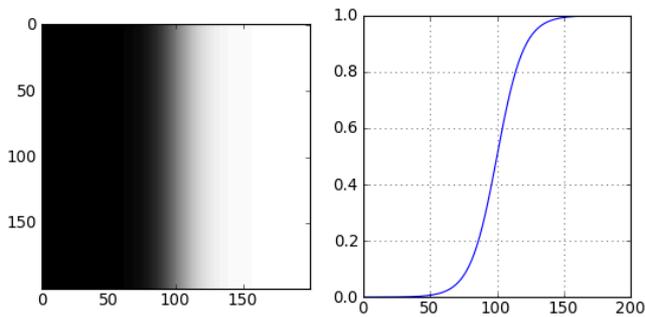
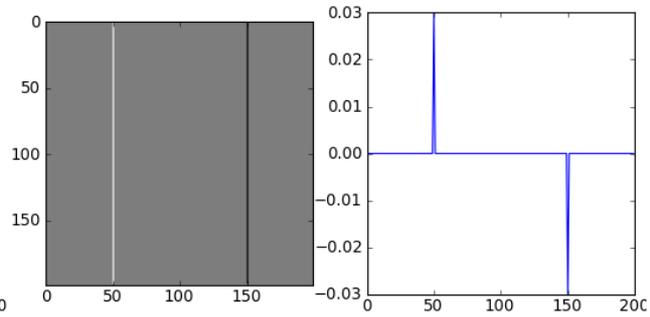
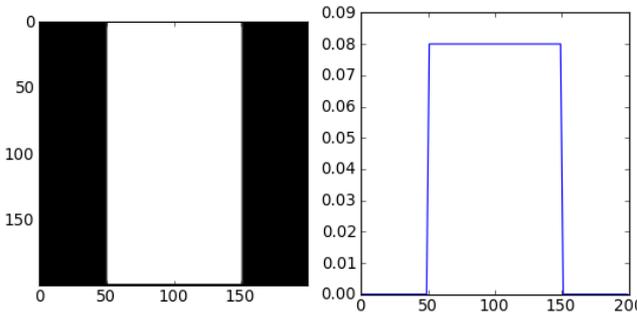
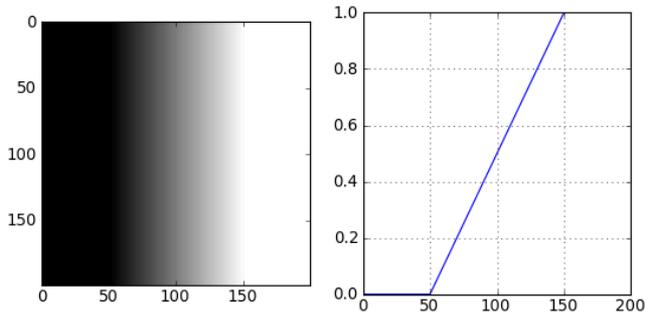
Abbildung: © R. C. Gonzalez & R. E. Woods, Digital Image Processing

Intensitätsprofil, erste und zweite Ableitung

Intensitätsprofil

erste Ableitung

zweite Ableitung

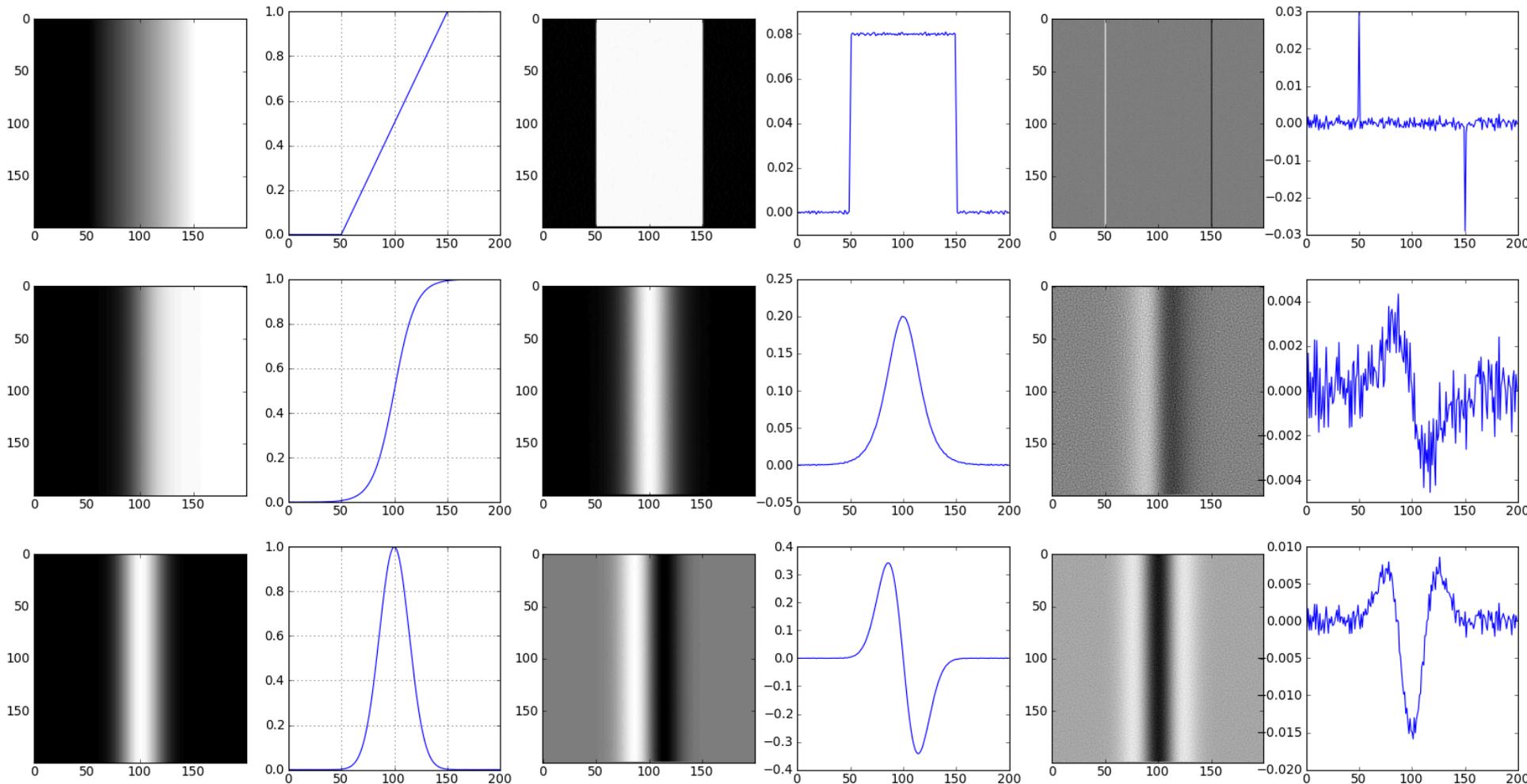


Gaußsches Rauschen, $\sigma = 0.0001$

Intensitätsprofil

erste Ableitung

zweite Ableitung

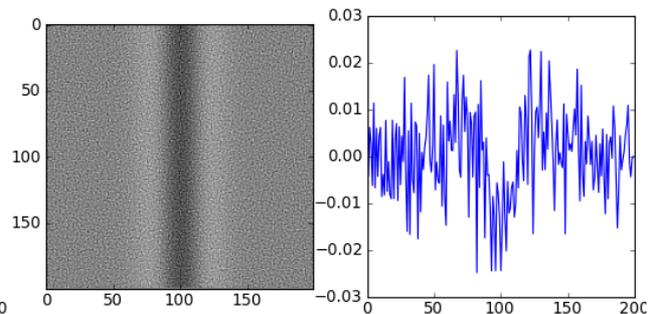
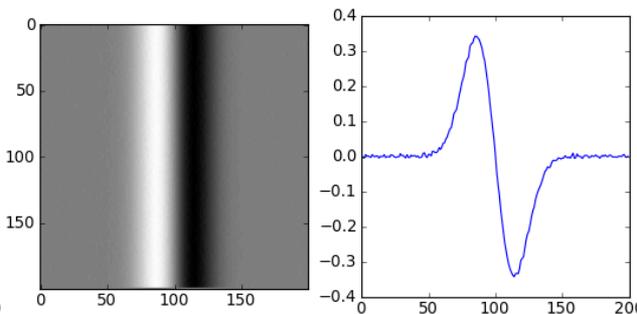
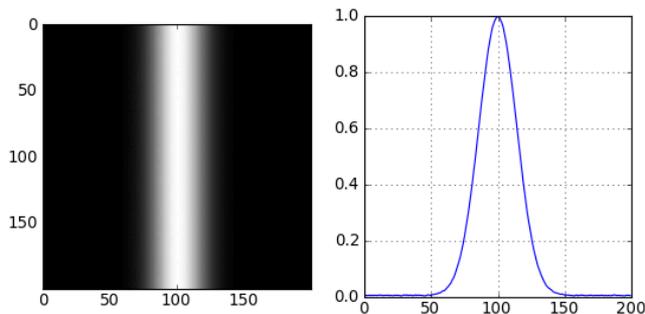
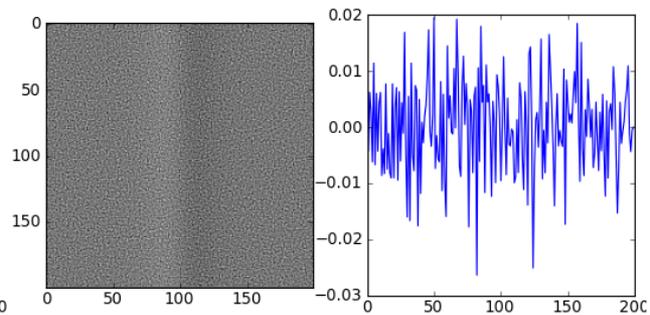
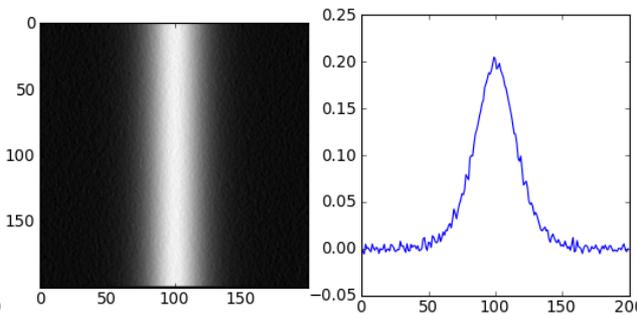
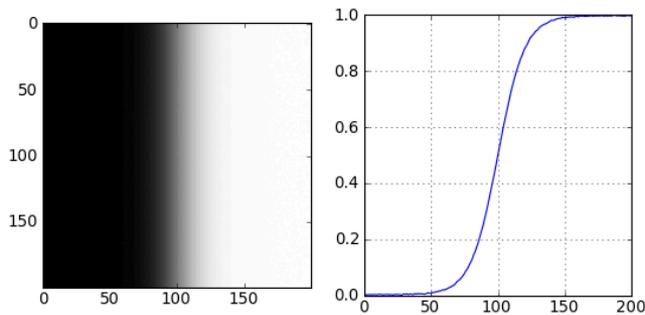
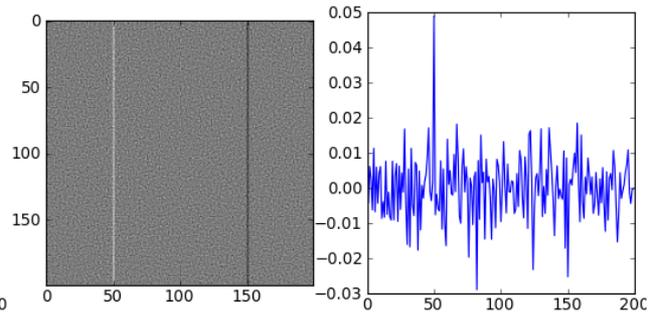
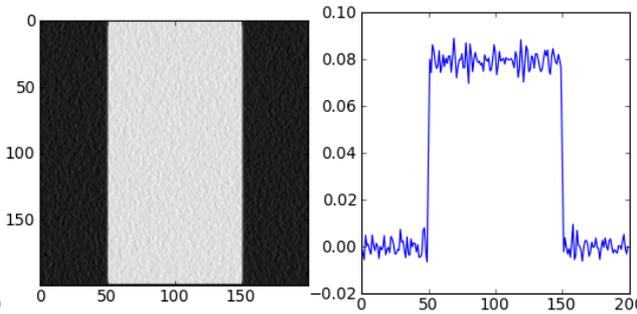
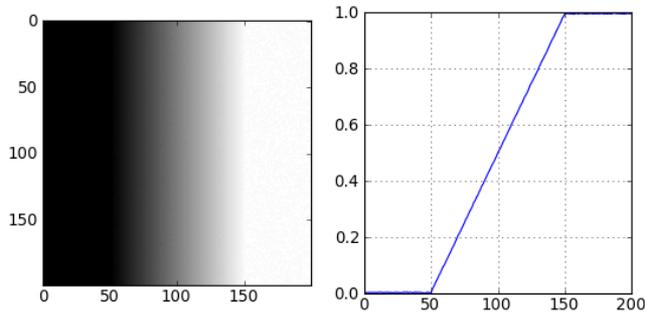


Gaußsches Rauschen, $\sigma = 0.001$

Intensitätsprofil

erste Ableitung

zweite Ableitung

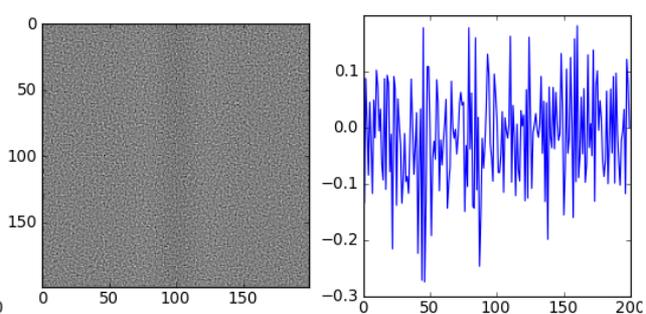
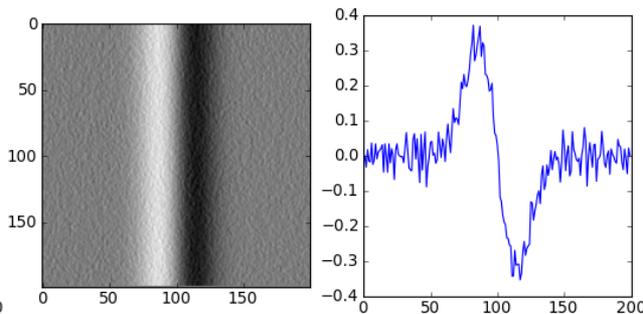
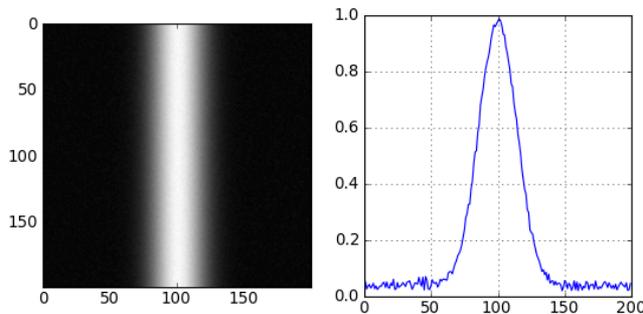
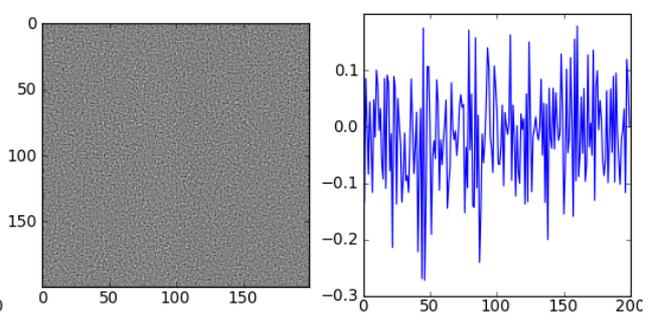
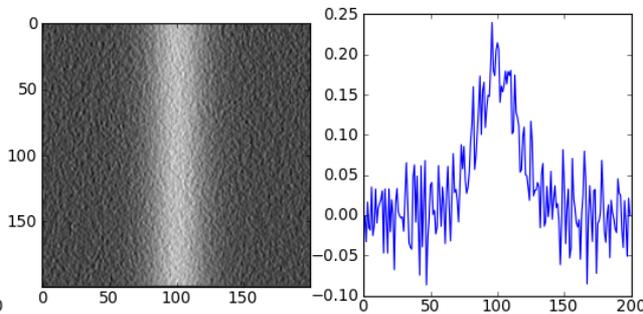
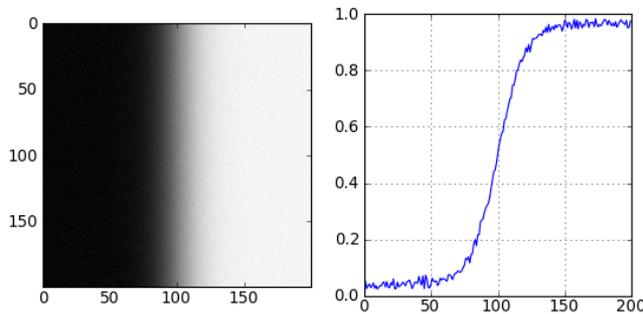
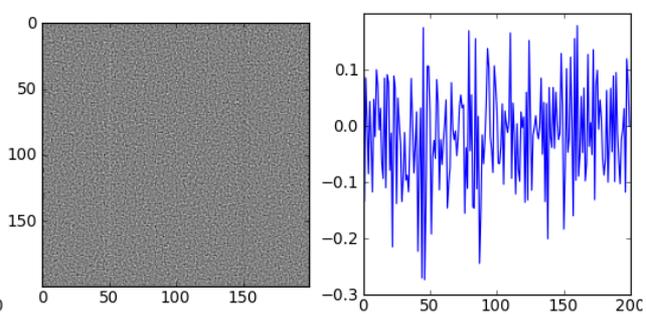
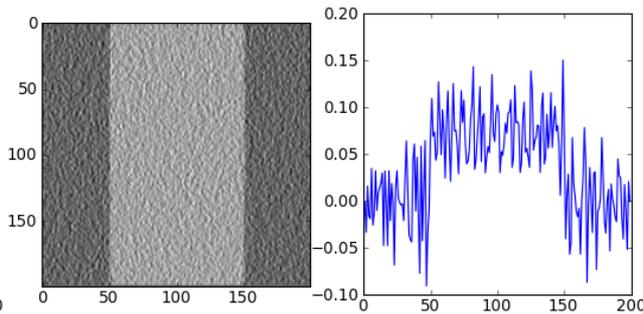
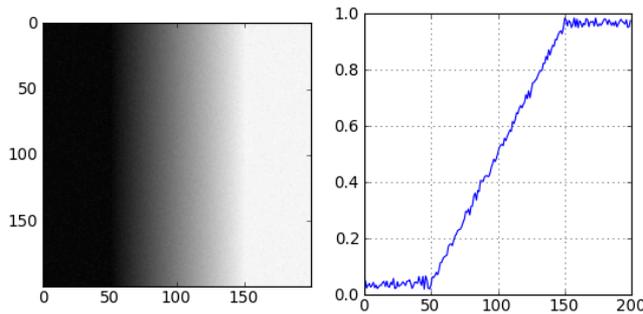


Gaußsches Rauschen, $\sigma = 0.01$

Intensitätsprofil

erste Ableitung

zweite Ableitung



Kantenhervorhebung durch Frequenzraumfilterung

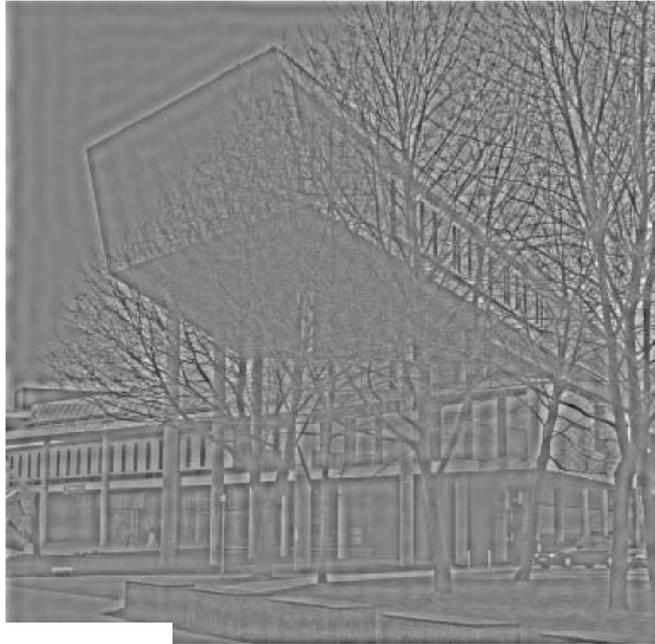
- Kanten weisen mehr hochfrequente Anteile auf als homogene Gebiete

► Hochpassfilterung

$$H_{F_{\max}}(u, v) = \begin{cases} 1 & , \text{ falls } u^2 + v^2 \geq F_{\max}^2 \\ 0 & , \text{ sonst.} \end{cases}$$



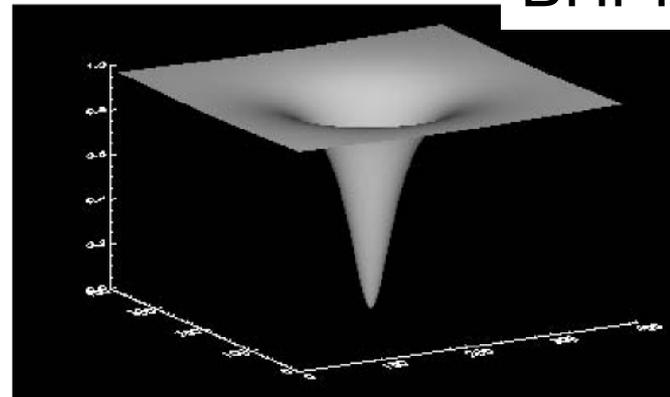
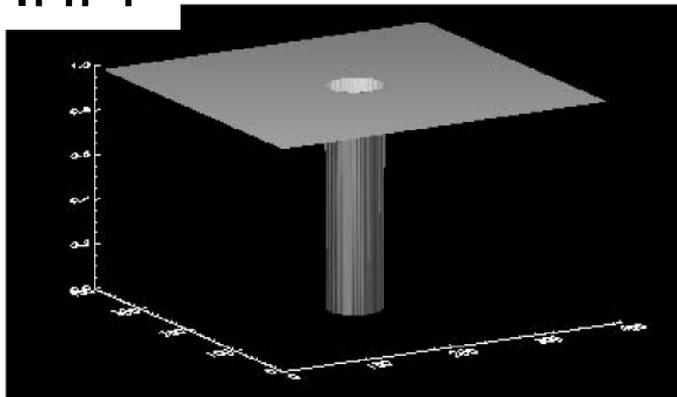
Filterung mit idealem Hochpass- und Butterworth-Hochpassfilter



IHPF



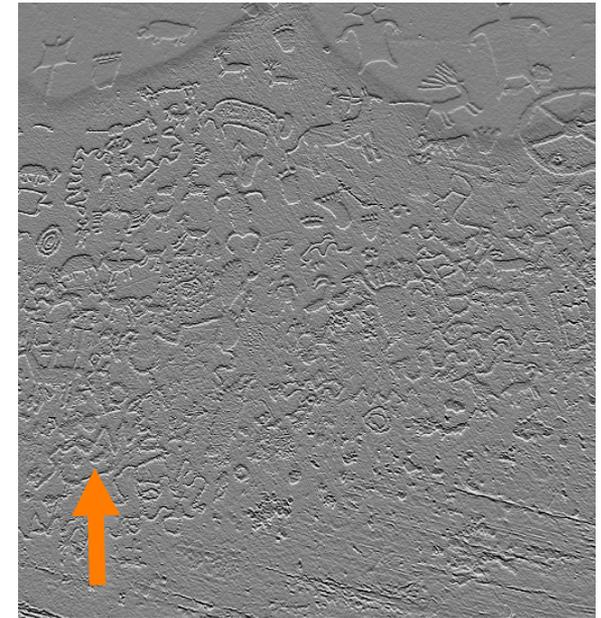
BHPF



Kanten im 2-D Raum

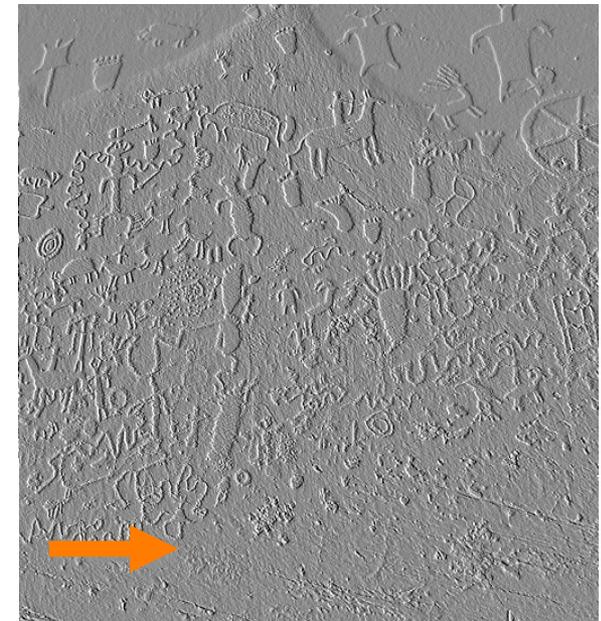
- Kanten können beliebig orientiert sein
- Differenzbildung in zwei orthogonalen Richtungen

Differenzbildung
in n -Richtung

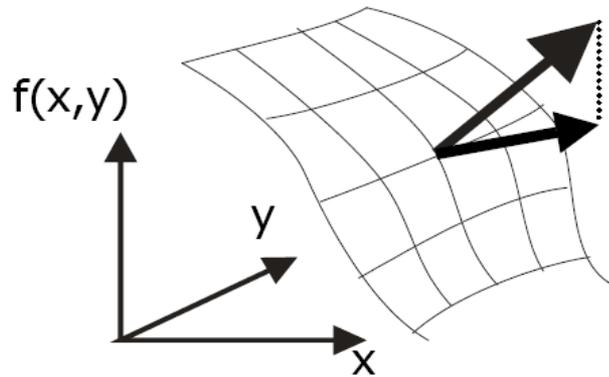


Pseudo-3D
Eindruck

Differenzbildung
in m -Richtung



Kanten im 2D-Raum: Gradienten



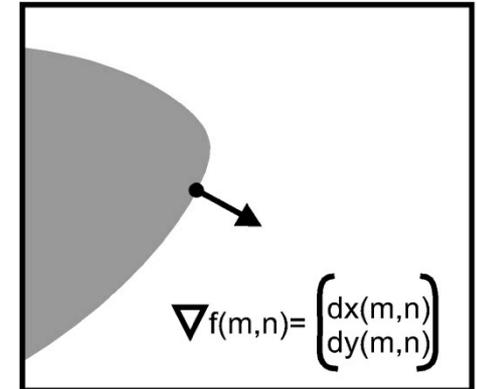
Richtung: Richtung der größten Steigung.

Länge: Stärke der stärksten Steigung.

- Gradient im kontinuierlichen Raum (x, y) : Vektor der partiellen Ableitungen der Bildfunktion in x - und y -Richtung:
 $(f(x, y)) = (\partial f / \partial x \quad \partial f / \partial y)$
- Approximation des Gradienten: Differential wird durch Differenz approximiert:
 $\vec{G}(f)(m, n) \approx [G_x(m, n) \quad G_y(m, n)] = [f(m, n) - f(m-1, n) \quad f(m, n) - f(m, n-1)]$
- Die Länge des Gradienten ist sein Betrag $|G(f)|$ oder näherungsweise $|G_x| + |G_y|$.

Gradientenschätzung über Konvolution

- Gradient im N -dimensionalen Raum ist ein N -dimensionaler Vektor aus N partiellen Ableitungen
- Jede partielle Ableitung kann durch eine Differenz abgeschätzt werden, die durch Konvolution berechnet werden kann



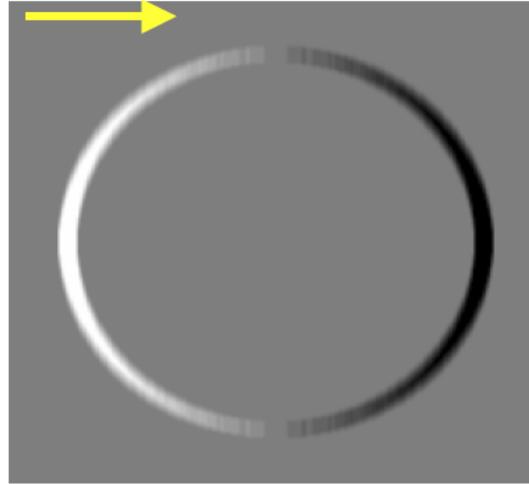
- Beispiele:

$$\vec{G}(m,n) = \begin{pmatrix} G_x(m,n) \\ G_y(m,n) \end{pmatrix}, \quad G_x(m,n) \approx [f * g_x](m,n), \quad G_y(m,n) \approx [f * g_y](m,n).$$

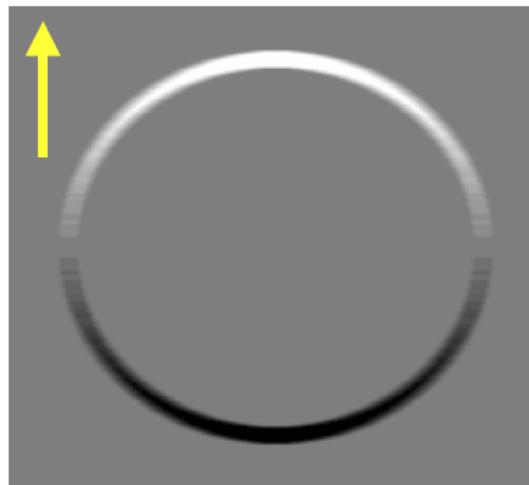
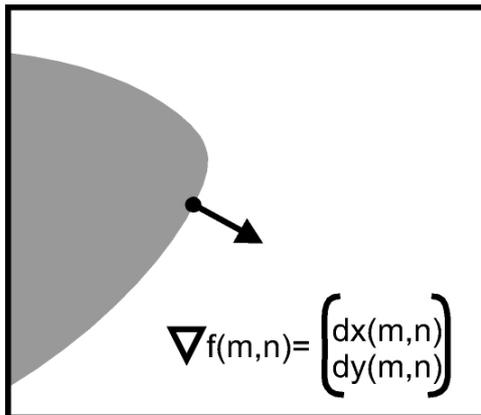
$$g_x = \begin{bmatrix} -1 & 1 \end{bmatrix} \quad g_y = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \quad g_{R1} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad g_{R2} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (g_{R1}, g_{R2}: \text{Robert's Gradient})$$

$$g_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \quad g_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

Elemente des Gradienten

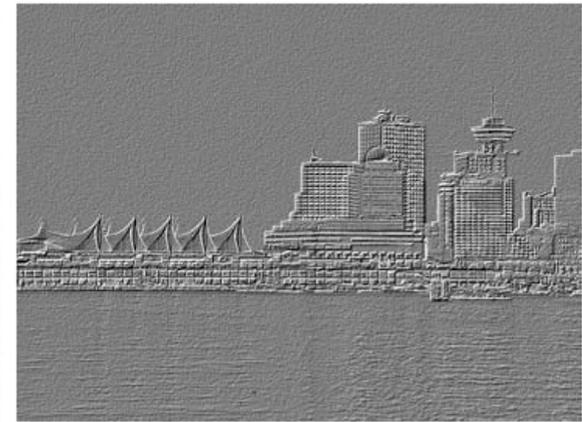
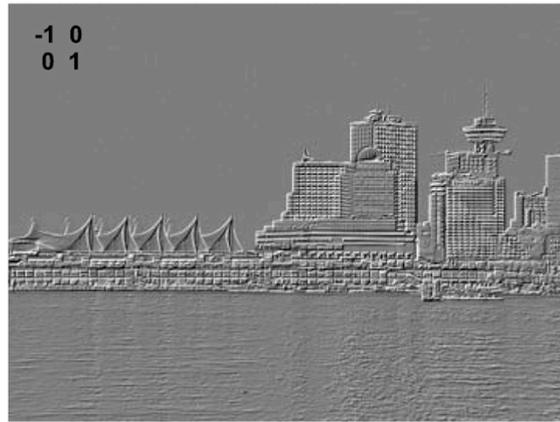


Betrag: $\sqrt{G_x^2 + G_y^2}$
Richtung: $\tan^{-1}(G_y / G_x)$



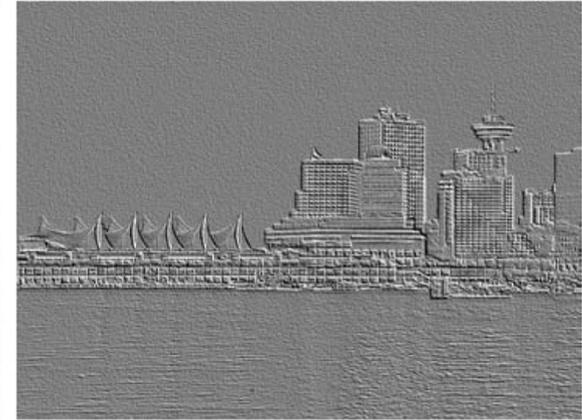
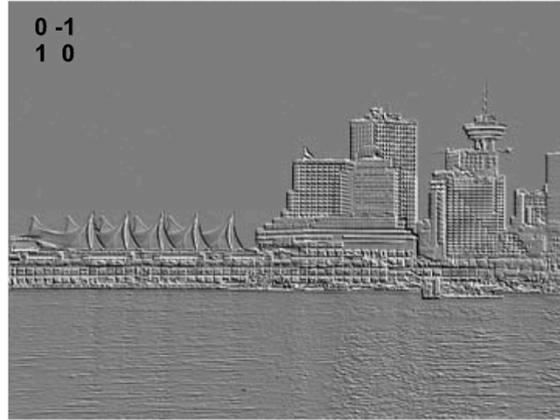
Roberts Gradient

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

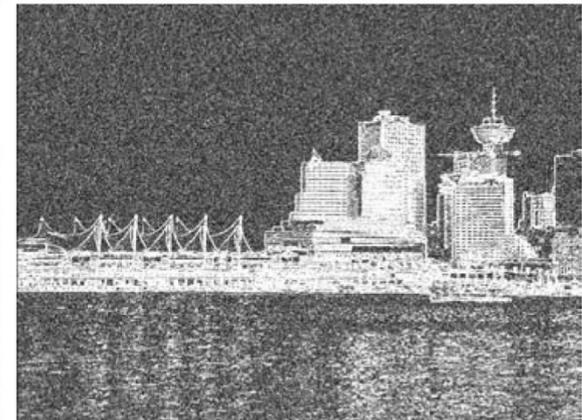


- rauscharmes Bild (links)
- verrauschtes Bild (rechts)

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



- Betrag des Gradienten (unten)
- Roberts-Operator rauschempfindlich



Rauschen und Kanten



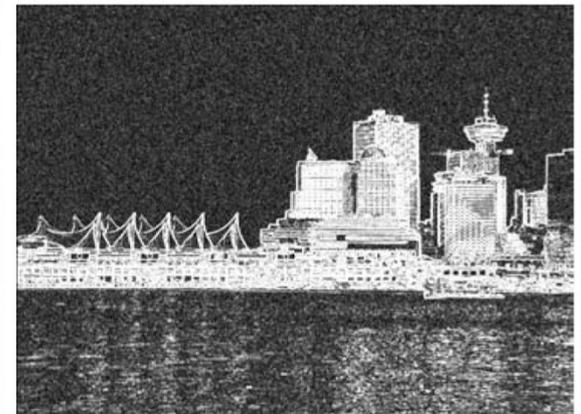
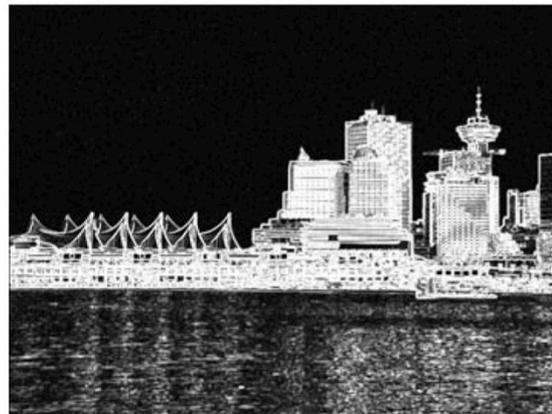
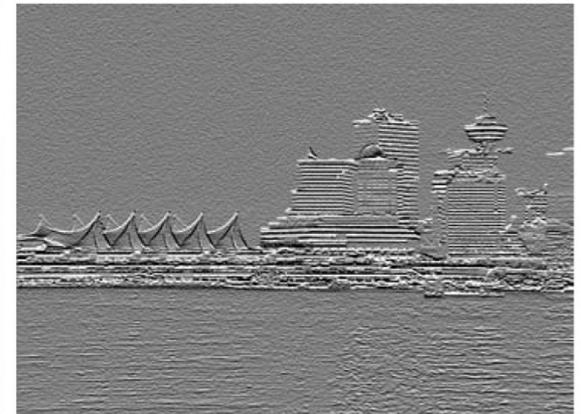
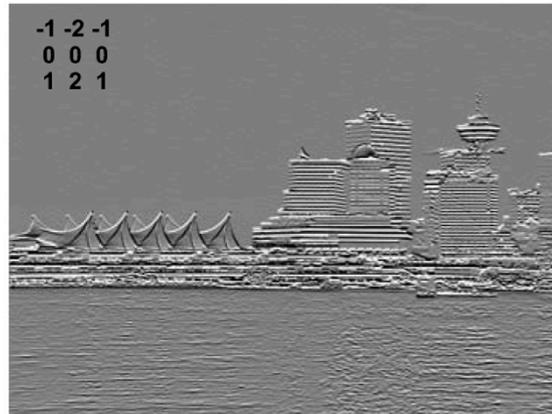
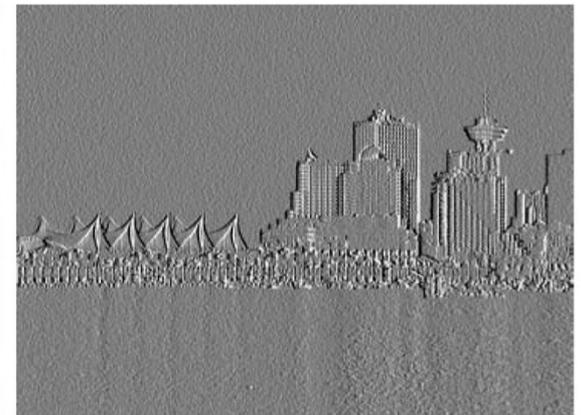
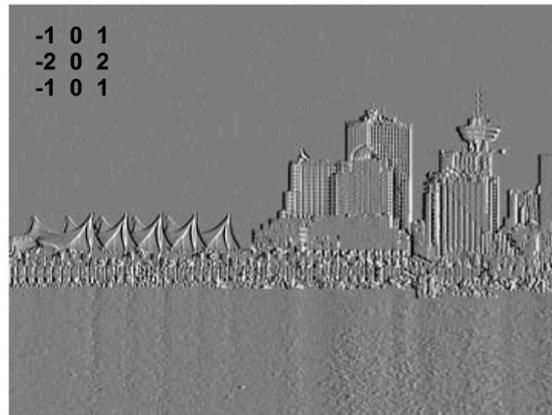
Hintergrundartefakte (Rauschen) werden verstärkt

Sobel Operator

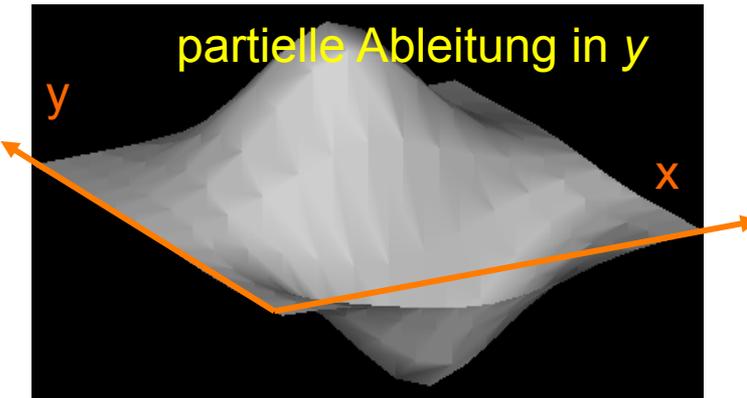
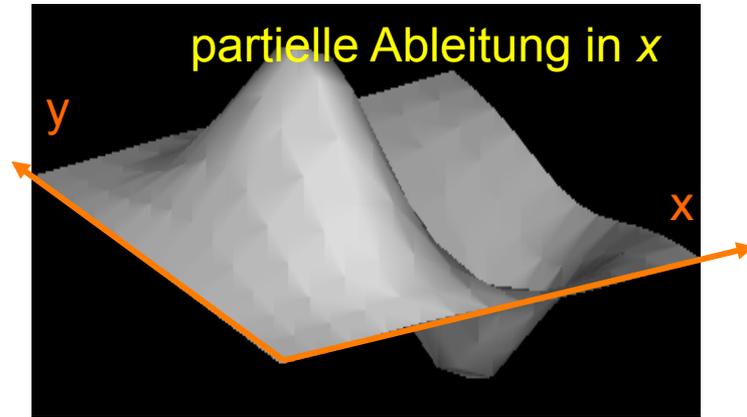
$$\begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \quad \text{und} \quad \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

Faltungskern
kombiniert
Glättung und
Differenzierung in
einem Operator

Nicht perfekt
rotationsinvariant



Ableitungen der Gaußfunktion



Ableitung der Gaußfunktion

- 2D-Gaußfunktion: $G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$

- kombiniert Glättung und Ableitung

- erste Ableitung der Gaußfunktion in x-Richtung

$$\frac{\partial G(x, y)}{\partial x} = \frac{1}{2\pi\sigma^2} \left[\frac{-2x}{2\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \right] = -\frac{x}{2\pi\sigma^4} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

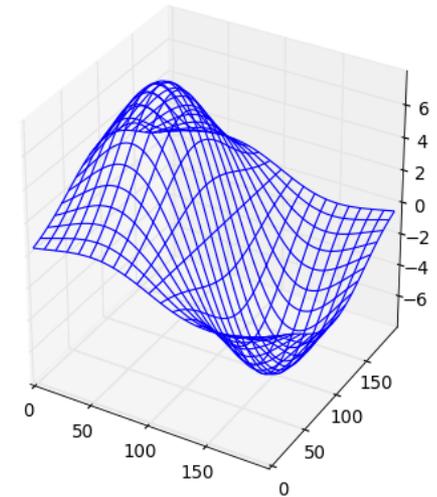
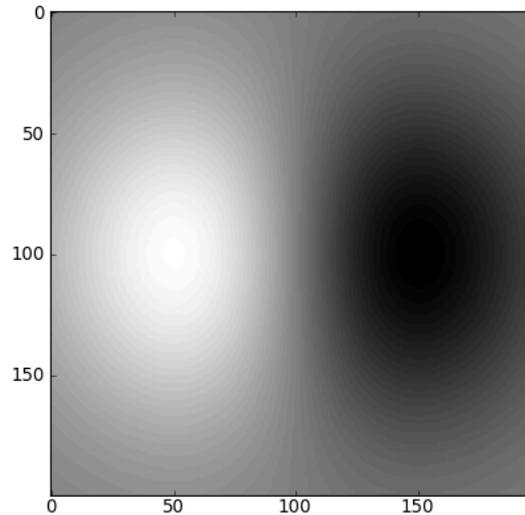
- erste Ableitung der Gaußfunktion in y-Richtung

$$\frac{\partial G(x, y)}{\partial y} = \frac{1}{2\pi\sigma^2} \left[\frac{-2y}{2\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \right] = -\frac{y}{2\pi\sigma^4} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

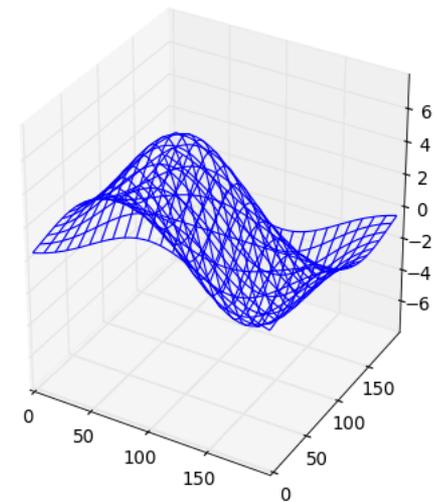
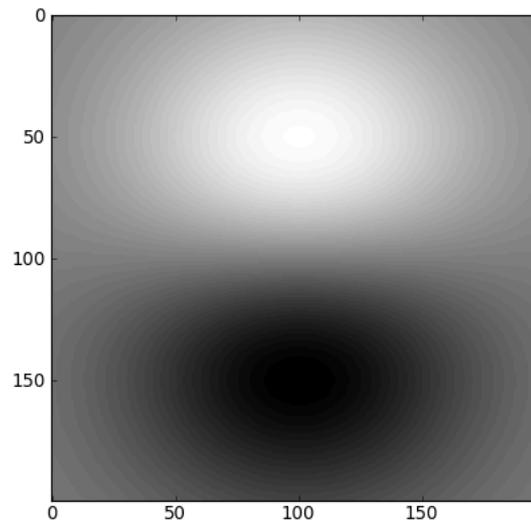
- separabel?

Ableitung der Gaußfunktion

- erste Ableitung der Gaußfunktion in x-Richtung



- erste Ableitung der Gaußfunktion in y-Richtung



Sobel vs. Gauß



3x3 Sobel

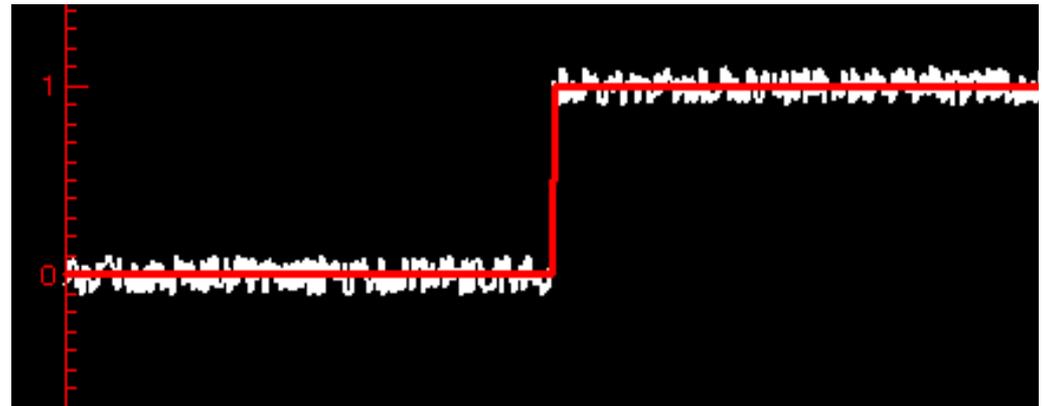
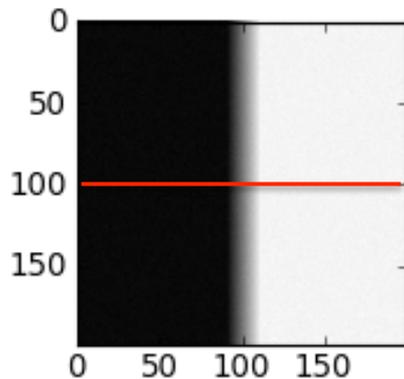


13x13 Gauß

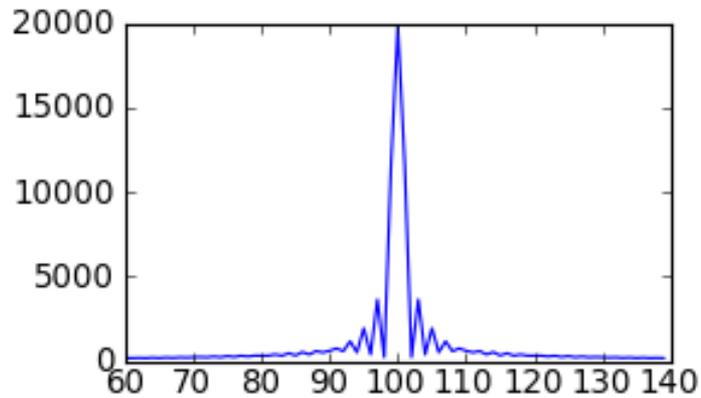


Optimale Kantenfilterung

- Annahme: Bild ist homogen mit Ausnahme von Funktionswertänderungen an idealen Kanten
- Repräsentation im Frequenzraum
 - homogener Bereich mit Grauwert g :
 $F(0,0) = g$ und $F(u,v) = 0$ für $u,v \neq 0$
 - Kante:
 $F(0,0) = 0$ und $F(u,v) \neq 0$ für $u,v \neq 0$
 - Rauschen:
 $F(u,v) \neq 0$



Kanten und Rauschen

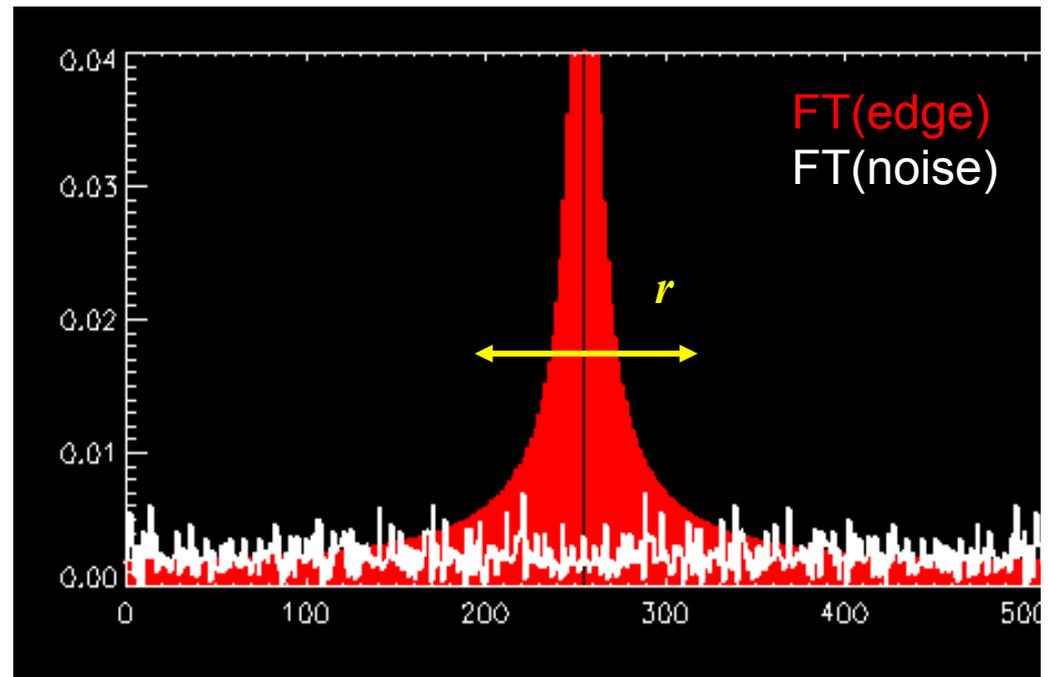


Kante im Frequenzraum

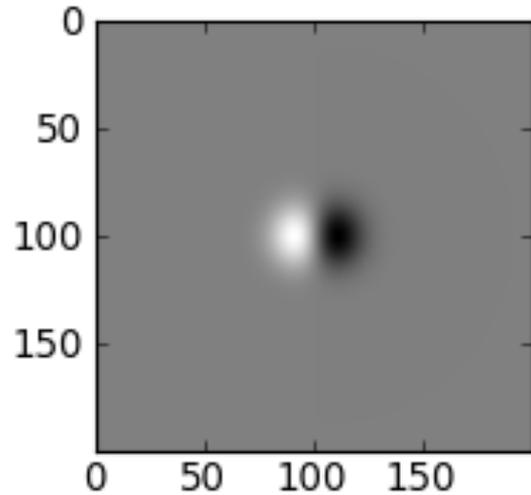
Filter suchen mit:

$$F(0,0) = 0$$

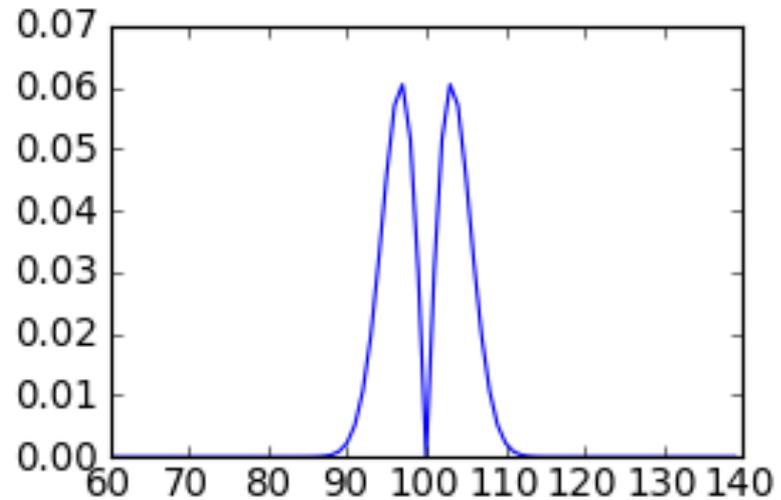
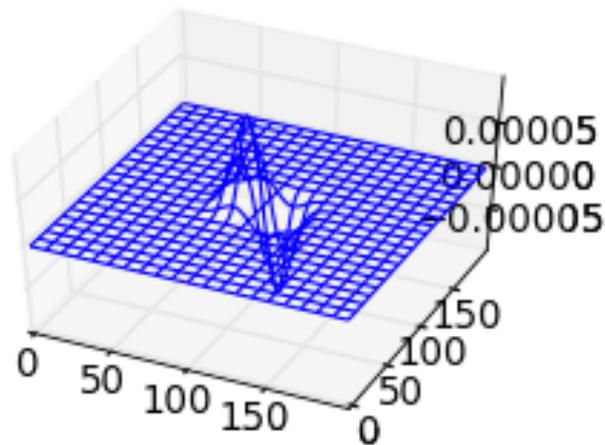
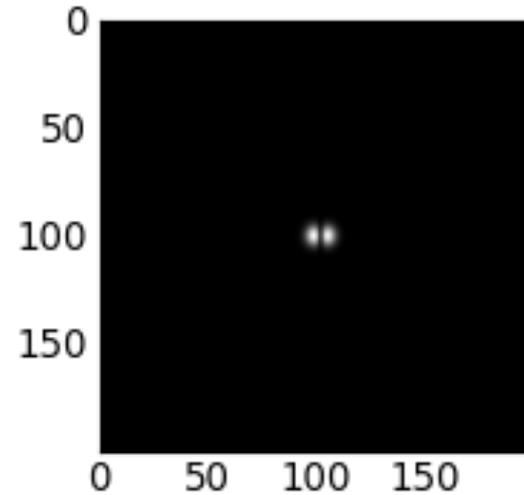
$$F(u,v) = w, \text{ falls } (u^2+v^2) < r^2$$
$$0, \text{ sonst}$$



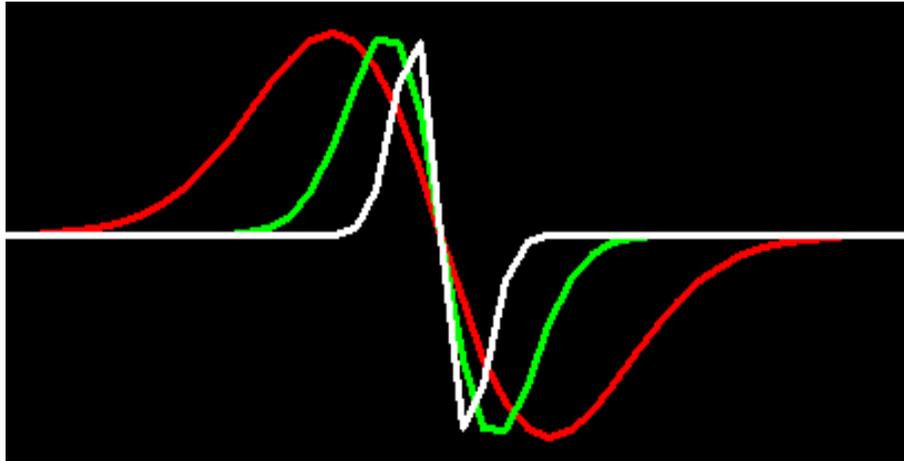
Fourier-Transformation der Ableitung des Gaußfilters



FFT

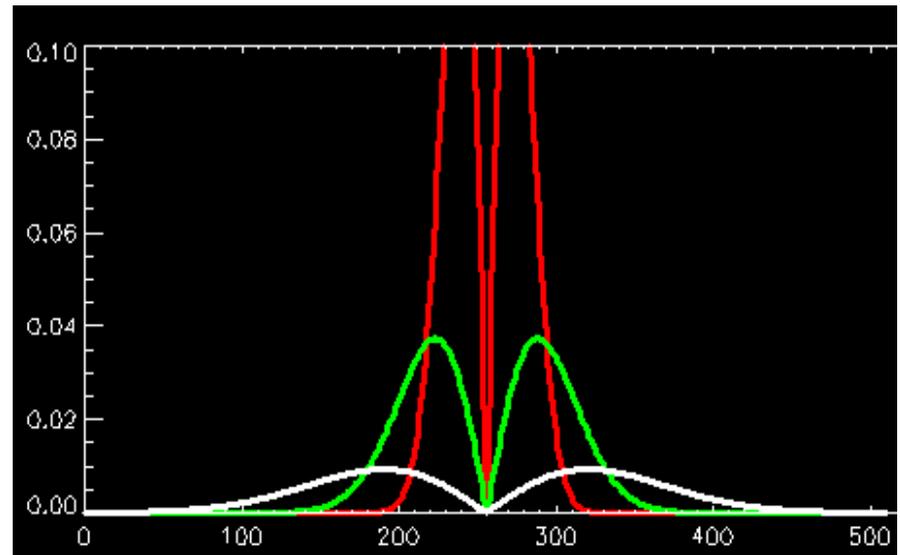


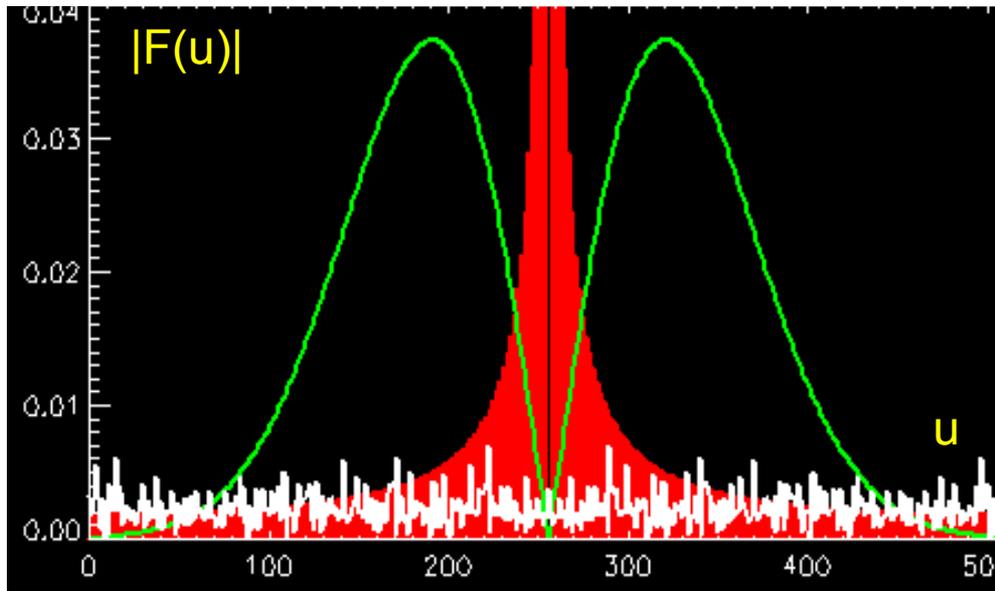
Fourier-Transformation der Ableitung des Gaußfilters



Ableitungen des Gaußfilters
unterschiedlicher Varianz
(im Ortsraum)

Repräsentation
im Frequenzraum

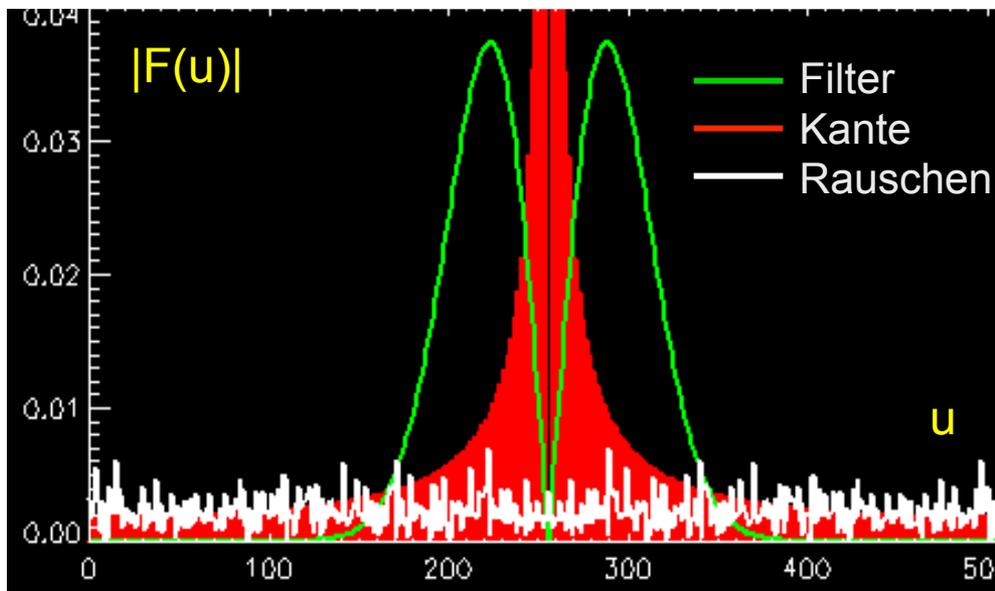




$s=1.0$

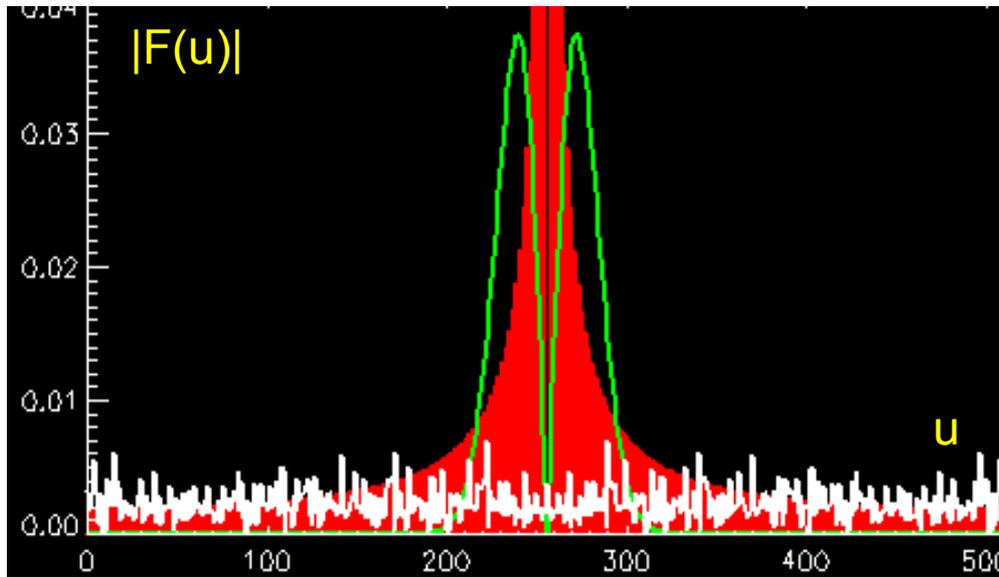
Filter unterdrückt
Rauschen ungenügend

Filter



$s=2.0$

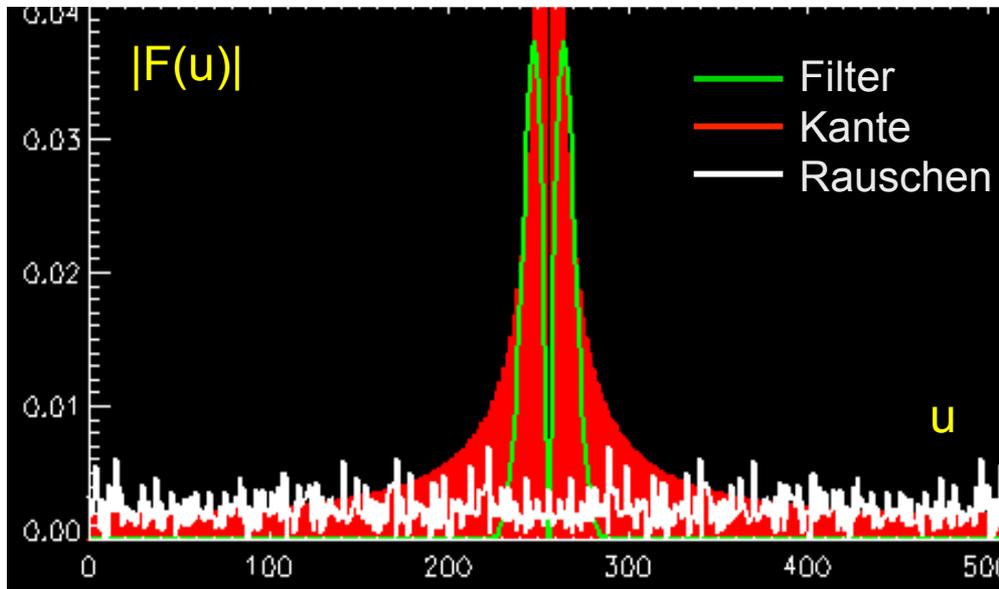
Filter ist nahezu optimal



$s=4.0$

Filter ist nahezu optimal

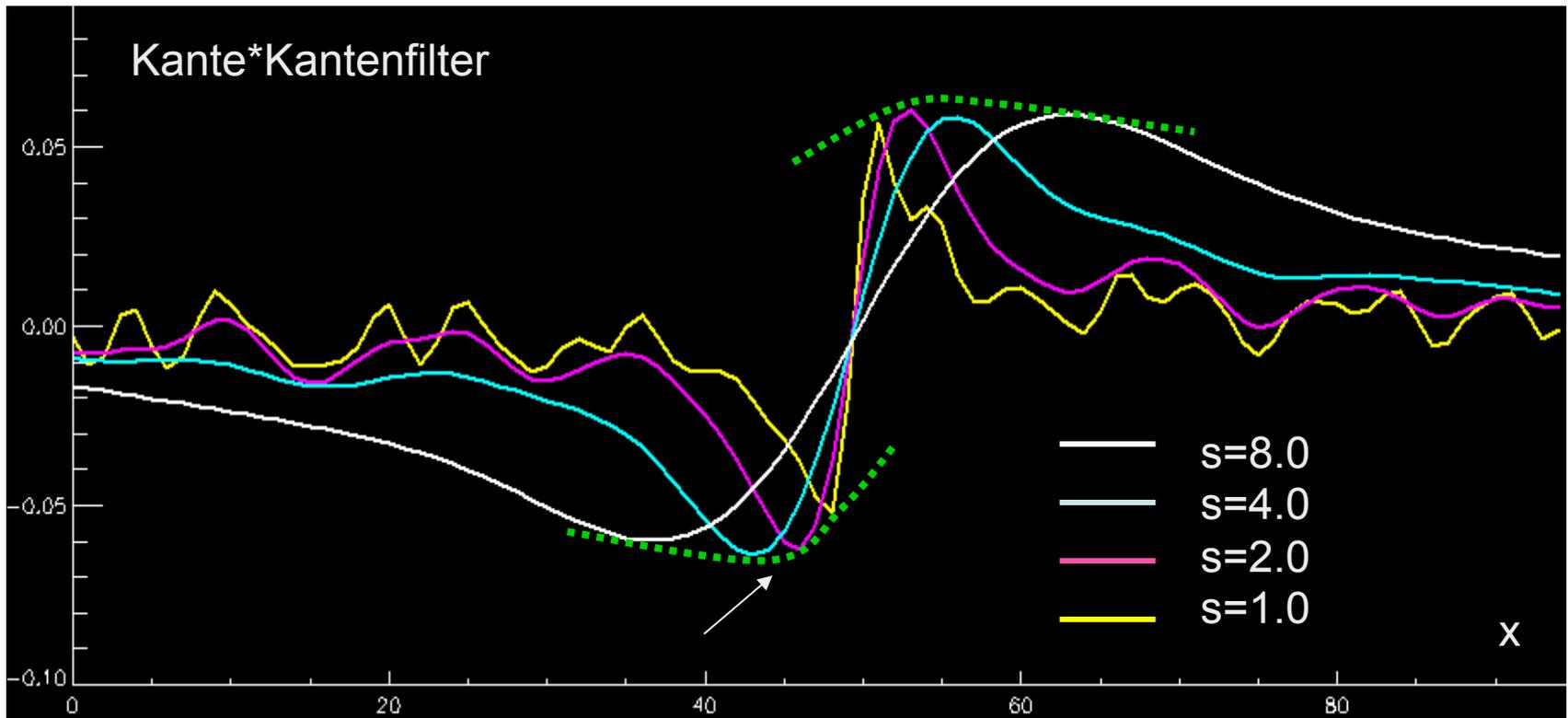
Filter



$s=8.0$

Filter unterdrückt Kante

Filterresultat



Zweidimensionales optimales Filter

- Rauschunterdrückung und Kantenhervorhebung gemäß Frequenzraumcharakteristik
- Filterung nur entlang größter Steigung

Kompassfilter

- Dienen der Hervorhebung von Kanten in einer bestimmten Richtung
- Prewitt-Operator

$$h_1 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}, \quad h_2 = \begin{pmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix}, \quad h_3 = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \dots \quad h_8 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{pmatrix}$$

- Kirsch-Operator (nicht-linear, sucht maximale Kantenstärke)

$$h_1 = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{pmatrix}, \quad h_2 = \begin{pmatrix} 3 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & -5 & 3 \end{pmatrix}, \quad h_3 = \begin{pmatrix} -5 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & 3 & 3 \end{pmatrix} \dots \quad h_8 = \begin{pmatrix} 3 & 3 & 3 \\ 3 & 0 & -5 \\ 3 & -5 & -5 \end{pmatrix}$$

$$h(n, m) = \max_{z=1, \dots, 8} \sum_{i=-1}^1 \sum_{j=-1}^1 h_z(i, j) \cdot f(n+i, m+j)$$

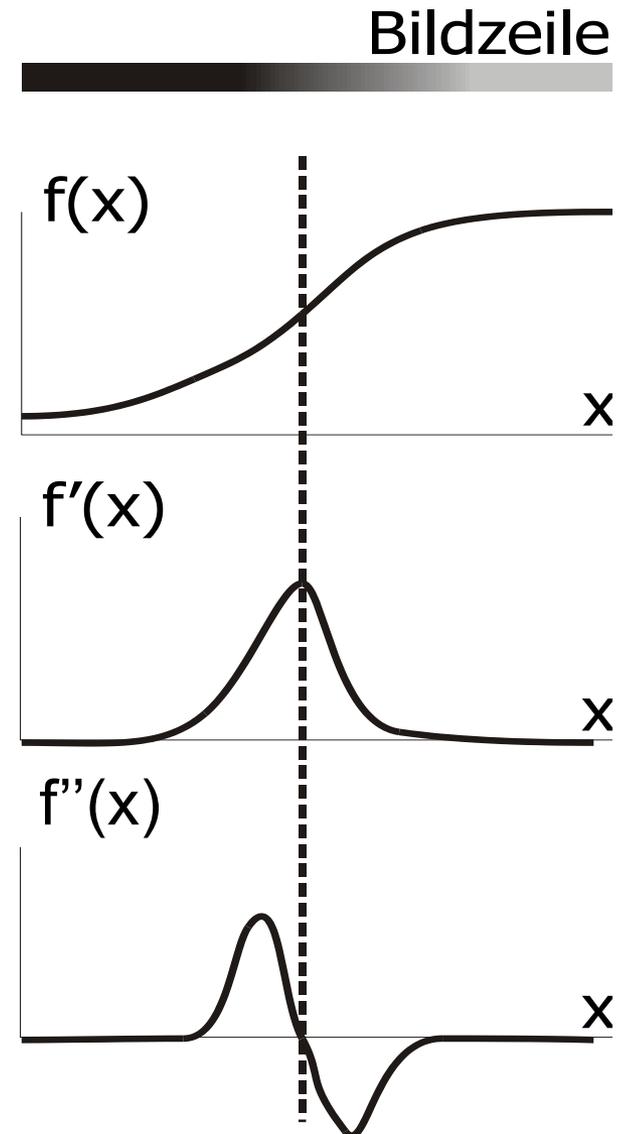
Kirschoperator?



ZWEITE ABLEITUNG

Zweite Ableitung

- Vorzeichenwechsel ist leichter zu erkennen, als ein Minimum oder Maximum
 - Gradient (Länge) als Maß für die Wichtigkeit einer Kante
 - zweite Ableitung für den Ort der Kante (Nulldurchgang)
- Operatoren zur Berechnung der zweiten Ableitung
 - Laplace Filter
 - Marr-Hildreth Filter (LoG Filter, Mexican Hat)
 - DoG (Difference of Gaussians)



Laplace-Funktion

- Summe der partiellen zweiten Ableitungen

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

- Nulldurchgänge der Laplace-Funktion produzieren zusammenhängende Kurven entlang von Bildkanten
- Approximation durch Kombination einer doppelten Differenzbildung in x- und y-Richtung

$$\begin{aligned}\frac{\partial^2 f}{\partial^2 x}(x) &= \frac{\partial f}{\partial x}(x+1) - \frac{\partial f}{\partial x}(x) \\ &= (f(x+1) - f(x)) - (f(x) - f(x-1)) \\ &= f(x-1) - 2f(x) + f(x+1)\end{aligned}$$

$$\begin{aligned}\frac{\partial^2 f}{\partial^2 y}(y) &= \frac{\partial f}{\partial y}(y+1) - \frac{\partial f}{\partial y}(y) \\ &= (f(y+1) - f(y)) - (f(y) - f(y-1)) \\ &= f(y-1) - 2f(y) + f(y+1)\end{aligned}$$

Laplace-Funktion

- zweite Ableitung in x-Richtung

$$\begin{aligned}\frac{\partial^2 f}{\partial^2 x}(x) &= \frac{\partial f}{\partial x}(x+1) - \frac{\partial f}{\partial x}(x) \\ &= (f(x+1) - f(x)) - (f(x) - f(x-1)) \\ &= f(x-1) - 2f(x) + f(x+1) \quad \Longrightarrow \quad \begin{pmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{pmatrix}\end{aligned}$$

- zweite Ableitung in y-Richtung

$$\begin{aligned}\frac{\partial^2 f}{\partial^2 y}(y) &= \frac{\partial f}{\partial y}(y+1) - \frac{\partial f}{\partial y}(y) \\ &= (f(y+1) - f(y)) - (f(y) - f(y-1)) \\ &= f(y-1) - 2f(y) + f(y+1) \quad \Longrightarrow \quad \begin{pmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{pmatrix}\end{aligned}$$

Laplace-Funktion

- Summe der partiellen zweiten Ableitungen

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

- Addition von um diagonale Richtungen, um rotationsinvarianten Operator zu erhalten

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 1 \\ 0 & -2 & 0 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Laplace-Operator



- Nulldurchgänge des Laplace-Operators
- mittleres grau = 0, dunkle Pixel < 0 , helle Pixel > 0

Schärfen eines Bildes mit Laplace-Operator

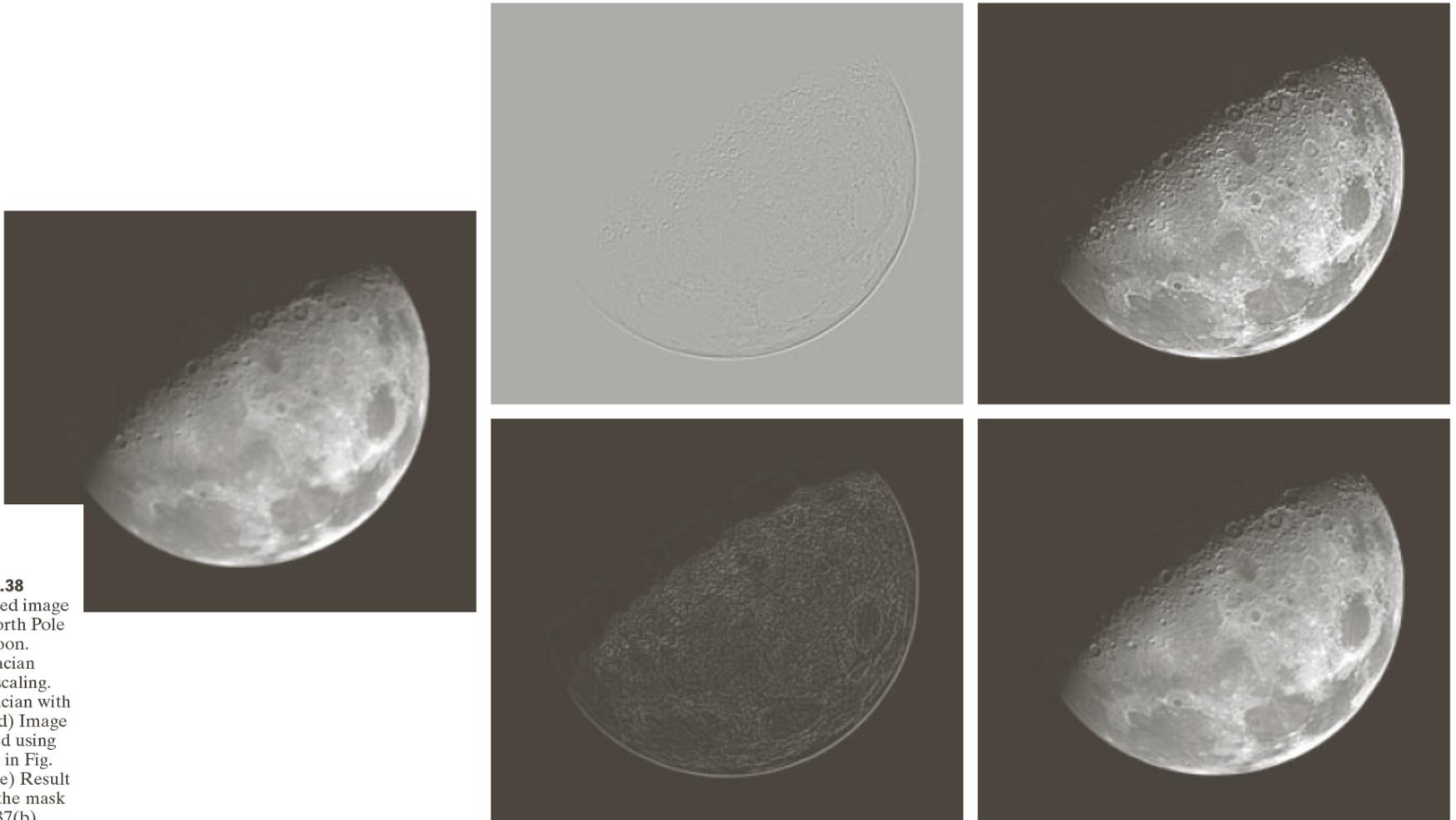
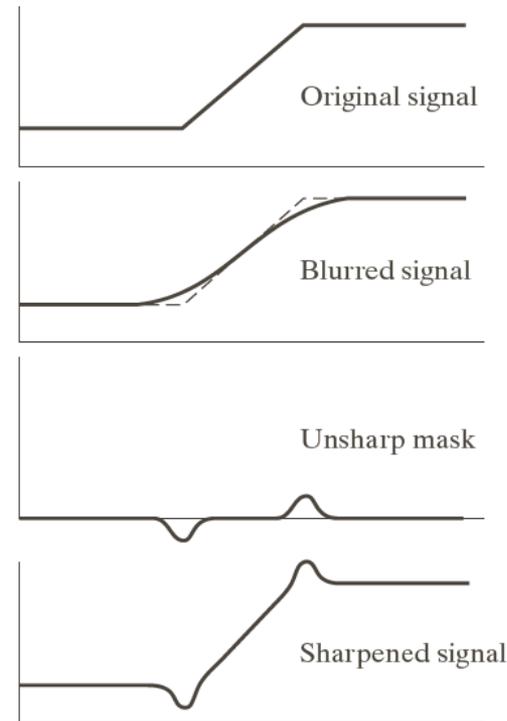


FIGURE 3.38
(a) Blurred image of the North Pole of the moon.
(b) Laplacian without scaling.
(c) Laplacian with scaling.
(d) Image sharpened using the mask in Fig. 3.37(a).
(e) Result of using the mask in Fig. 3.37(b).
(Original image courtesy of NASA.)

Abbildung: © R. C. Gonzalez & R. E. Woods, Digital Image Processing

Unscharfes Maskieren

- Originalbild unschärfer machen (Tiefpass filtern)
- Unscharfes Bild vom Original subtrahieren (ergibt die unscharfe Maske)
- Maske (nach Skalierung) zum Original addieren

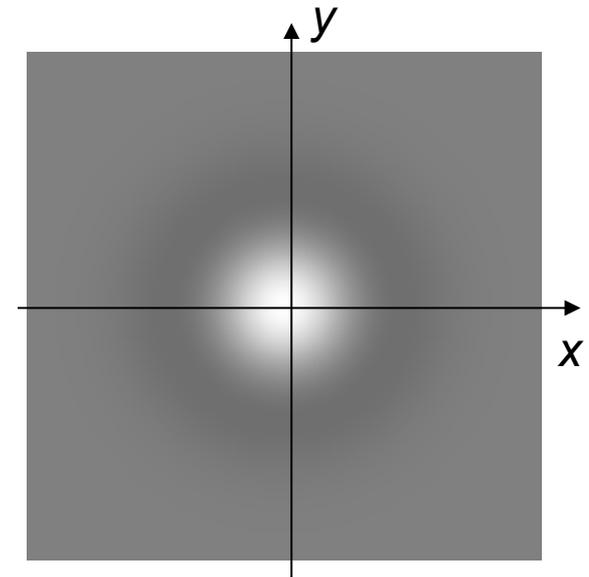
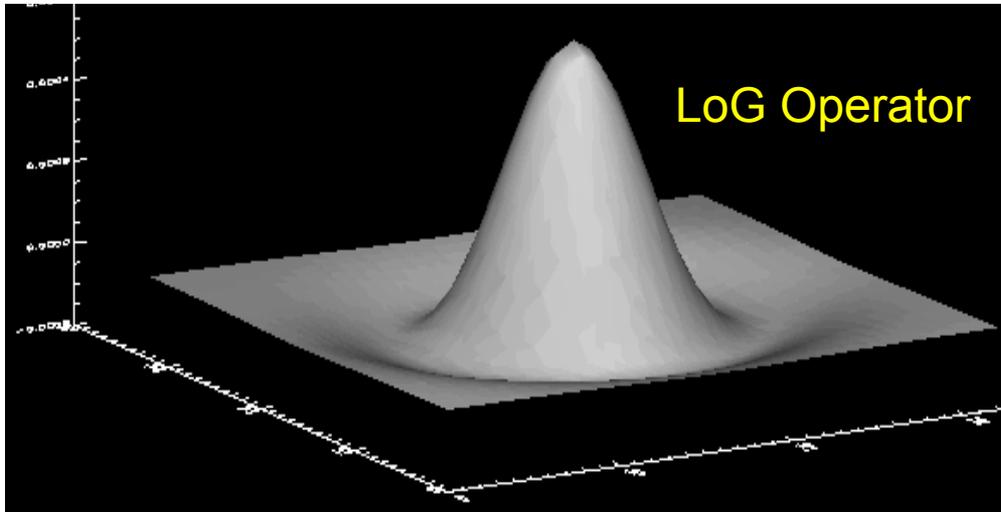


Highboost filtering

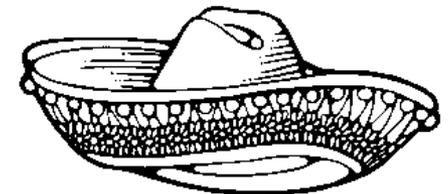


Marr-Hildreth-Filter = LoG-Filter

LoG-Filter: Laplacian-of-Gaussian,
d.h. der **Faltung** mit dem Laplacefilter geht eine
Glättung mit einer **Gaußfunktion** voraus



$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2 + y^2}{2\sigma^2} \right) \exp\left(-\frac{x^2 + y^2}{2\sigma^2} \right)$$



Auch genannt: „Mexican hat“ filter

Laplace-Funktion der Gaußfunktion

- 2D-Gaußfunktion: $G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$
- Laplace-Funktion der Gaußfunktion (zweite Ableitung)

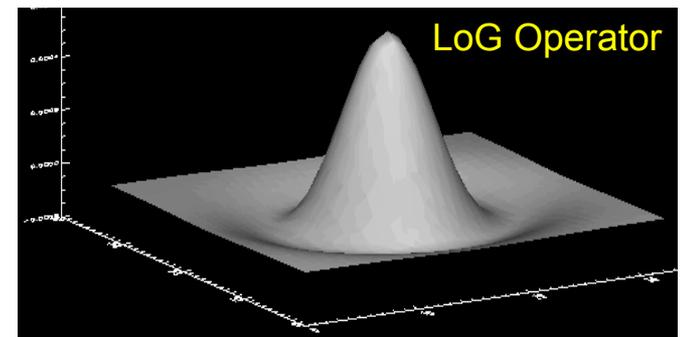
$$\begin{aligned}\nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} \\ &= \frac{1}{2\pi\sigma^2} \frac{\partial}{\partial x} \left[\frac{-2x}{2\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \right] + \frac{1}{2\pi\sigma^2} \frac{\partial}{\partial y} \left[\frac{-2y}{2\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \right] \\ &= \frac{1}{2\pi\sigma^2} \left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right] \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) + \frac{1}{2\pi\sigma^2} \left[\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right] \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \\ &= \frac{1}{\pi\sigma^4} \left[\frac{x^2 + y^2}{2\sigma^2} - 1 \right] \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)\end{aligned}$$

Marr-Hildreth-Filter = LoG-Filter

- Vorteile der Laplace-Funktion der Gaußfunktion
 - Glättung und Kanten hervorhebung in einem Operator
 - wegen Glättung weniger empfindlich bei Rauschen
 - Gaußfunktion erzeugt keine „Ringing-Artefakte“
 - rotationsinvariant
 - separabel
- Marr-Hildreth-Algorithmus zur Kantenerkennung

$$g(x, y) = \left[\nabla^2 G(x, y) \right] * f(x, y) = \nabla^2 \left[G(x, y) * f(x, y) \right]$$

1. nxn Gaußfilter anwenden
2. 3x3 Laplace-Filter anwenden
(-8 umrahmt von 1en)
3. Zero-crossings finden
(threshold verwenden)

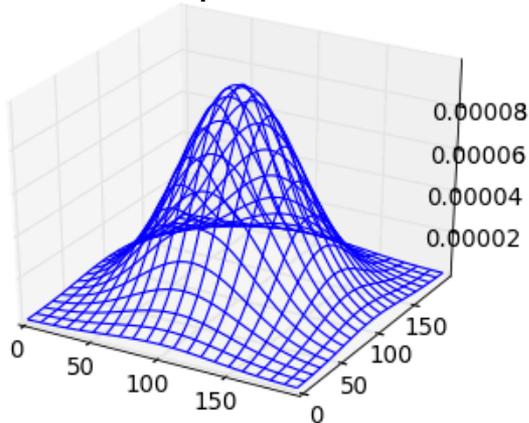


DoG (Difference of Gaussians)

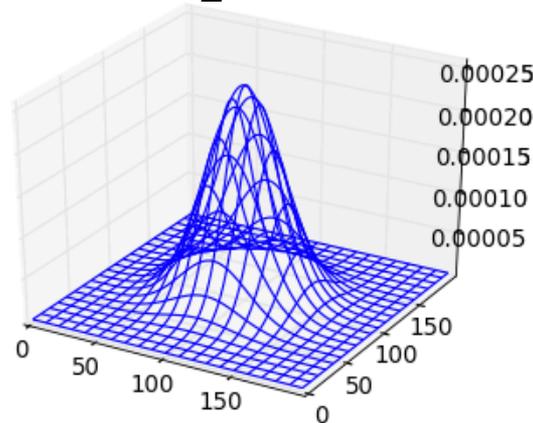
- Approximation des LoG durch DoG
- Subtraktion zweier Gaußfilter unterschiedlicher Varianz

$$DoG(x, y) = \frac{1}{2\pi\sigma_1^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_1^2}\right) - \frac{1}{2\pi\sigma_2^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_2^2}\right)$$

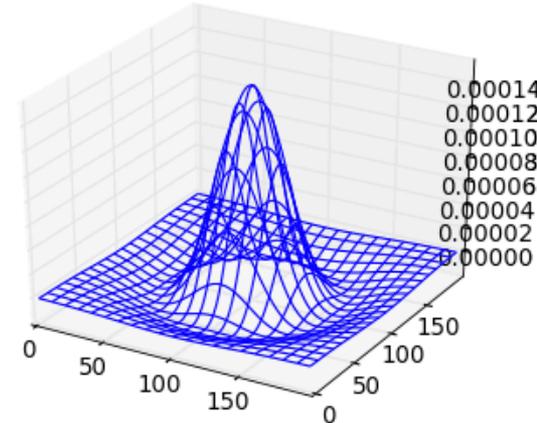
$\sigma_1=40.0$



$\sigma_2=25.0$

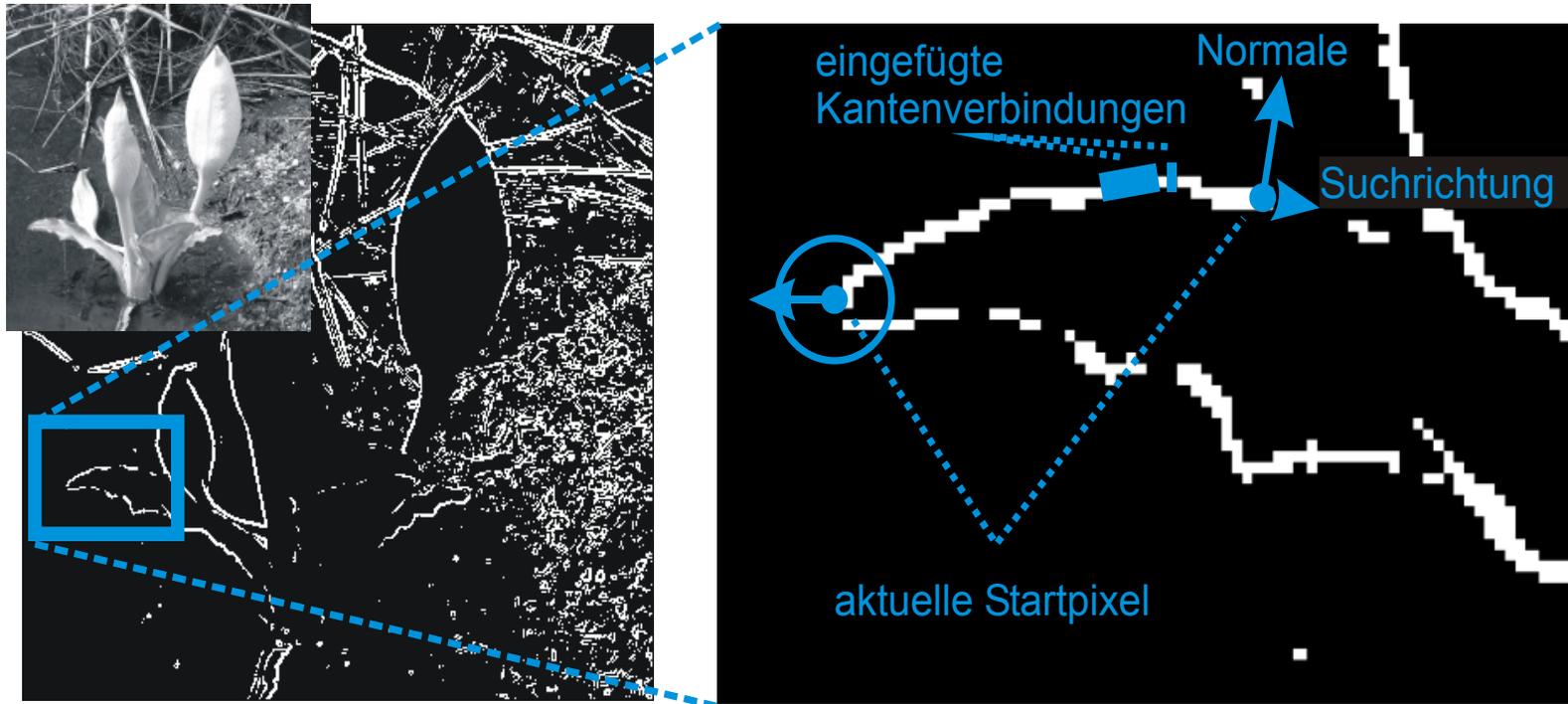


DoG



CANNY EDGE DETECTION

Edge Linking zur Unterscheidung von Kanten und Rauschen



Anfangs sind alle Kantenpixel frei und nicht untersucht
Edge Linking sucht sich das nächste nicht untersuchte
und freie Kantenpixel und versucht es mit anderen
Pixeln zu einem Kantenzug zu verknüpfen

Edge Linking verbindet Kantenpixel zu Kantenzügen

1. Suche nächstes Kantenpixel, das noch nicht als „untersucht“ markiert wurde, erkläre es zum Startpixel eines Kantenzugs
2. Falls sich in Umgebung des Kantenpixels orthogonal zur Kantenrichtung unmarkierte Kantenpixel mit ähnlicher Gradientenrichtung und –stärke befinden
 - a. Markiere die Pixel als zum selben Kantenzug gehörend
 - b. Erkläre diese Pixel zu neuen Startpixeln
 - c. Gehe zu Schritt 2
3. Falls sich in Umgebung markierte Pixel befinden, die den obigen Bedingungen genügen, dann wurde eine Verzweigung gefunden
4. Falls ein Kantenpixel gefunden, gehe zu Schritt 1

Canny Edge Operator

- Ziele
 - möglichst viele Kanten fehlerfrei vom Hintergrund unterscheiden (niedrige Rate von Fehldetektionen)
 - Kanten genau lokalisieren
 - für jede Kante genau eine Detektorantwort
- Canny Operator besteht aus Kantenhervorhebung und Erzeugung von Kantenzügen
 - Startpixel können nur Pixel sein, deren Gradientenlänge oberhalb einer Signifikanzschwelle T_1 liegt
 - Neue Kantenpixel werden in den Kantenzug eingefügt, wenn ihre Gradientenlänge größer als T_2 ist ($T_2 < T_1$)

Canny Edge Detection – Algorithmus

1. Bild mit Gaußfilter glätten
 2. Gradientenoperator (Sobel, Prewit, etc.) anwenden
 - ergibt Gradientenlänge und –richtung
 3. Non-maxima Unterdrückung anwenden
 - a. Unterdrücke Punkt, falls ein Nachbar in Gradientenrichtung größere Gradientenlänge hat
 4. Doppeltes Tresholding anwenden ($T_1 > T_2$)
 - a. Wähle Punkt als Startpunkt eines Kantenzugs, falls Gradientenlänge $\geq T_1$ (starke Kante)
 - b. Nimm nächsten Punkt entlang der Kante in Kantenzug auf, falls dessen Gradientenlänge $\geq T_2$ (schwache Kante, falls $< T_1$)
- Verhalten hängt von T_1 , T_2 , sowie Gaußfilter ab

Non-Maxima Unterdrückung

- Quantisieren der Kantenorientierung in eine von vier Richtungen: horizontal, vertikal, $+45^\circ$, -45°
- Falls Gradientenlänge kleiner als einer seiner Nachbarn in Kantenorientierung, dann unterdrücken, sonst behalten

Doppeltes Thresholding

- starkes Kanten: $L[x,y] \geq T_1$
- schwaches Kante: $T_1 > L[x,y] \geq T_2$

- typischerweise $T_1 / T_2 = 2..3$

- schwache Kanten überleben nur, wenn sie starke Kanten fortsetzen

Nulldurchgänge

Die Orte der Nulldurchgänge der zweiten Ableitung sind Ränder zusammenhängender Gebiete

Methode:

- Laplace-Operator
- Nulldurchgänge bestimmen:

$$\nabla^2(f(i,j)) \cdot \nabla^2(\text{shift}(f(i,j))) \leq 0$$

(shift: Verschiebung des Bilds um ein Pixel in jede Richtung)

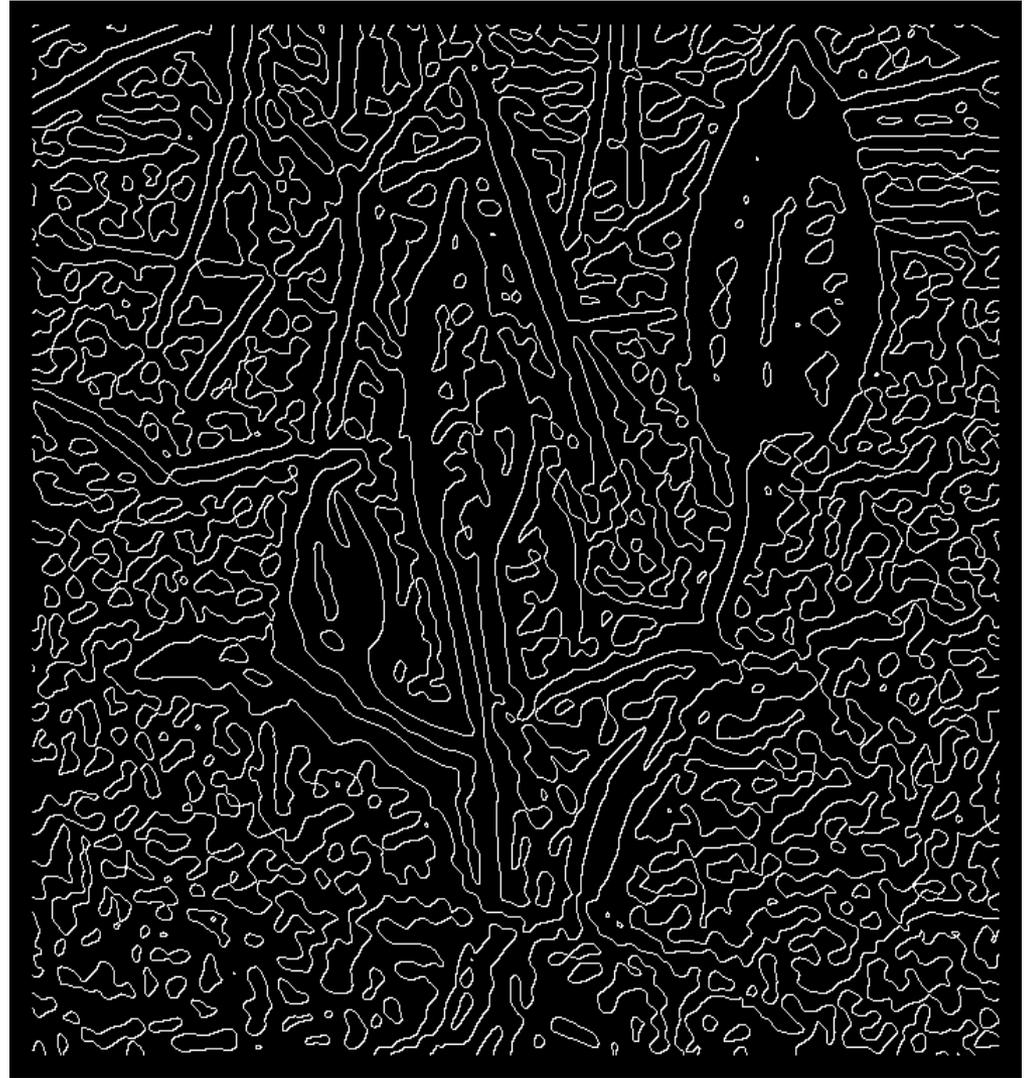
Laplace-Operator:
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



Nulldurchgänge

Kombination des Laplace-Operators mit Glättungsoperator (z.B. als LoG-Operator) reduziert die Anzahl der Nulldurchgänge

Threshold auf Gradientenlänge reduziert Anzahl der Nulldurchgänge ebenfalls



HOUGH TRANSFORMATION

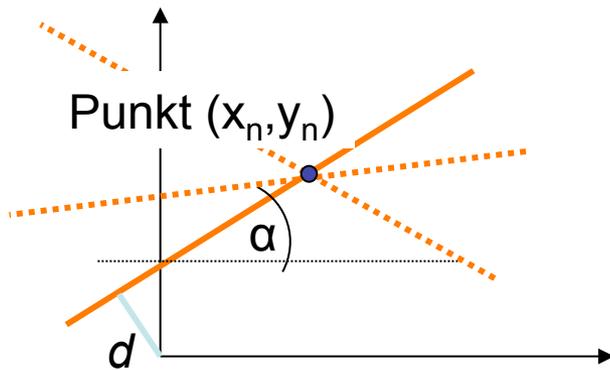
Hough Transformation

- Modellbasierte Suche (Template Matching)
 - Modell einer Regionengrenze ist vorgegeben
 - Information über mögliche Kantenorte ist gegeben (z.B. durch Gradientenlänge)
 - Zielsetzung: Orte finden, an denen Modell und Bild optimal übereinstimmt
- Hough-Transformation
 - Voting(Abstimmungs)-Mechanismus, bei dem jeder Ort in Abhängigkeit der lokalen Information für das Modell stimmt
 - entwickelt für Geraden, erweiterbar für beliebige Formen

Hough Transformation (HT)

Suche von Geraden in einem Binärbild

Geradenrepräsentation: $x \cos(\alpha) + y \sin(\alpha) - d = 0$

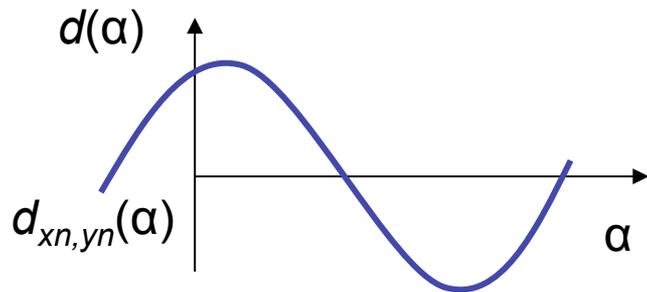


Hough-Transformation:

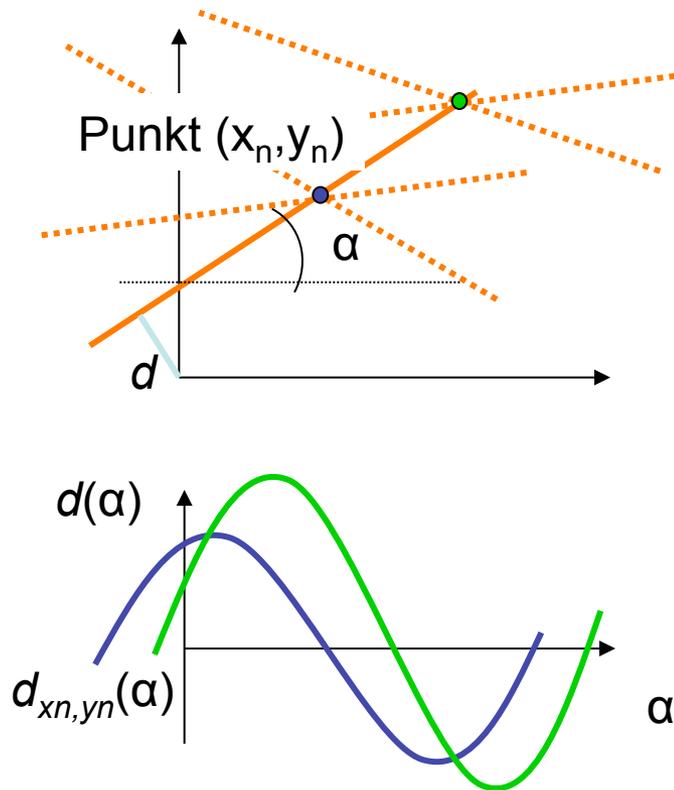
Suche alle Parameter (α, d) für Geraden, die durch einen Punkt (x_n, y_n) gehen

$$d(\alpha) = x_n \cos(\alpha) + y_n \sin(\alpha)$$

Der Raum, der durch (α, d) aufgespannt wird, heißt Hough-Raum

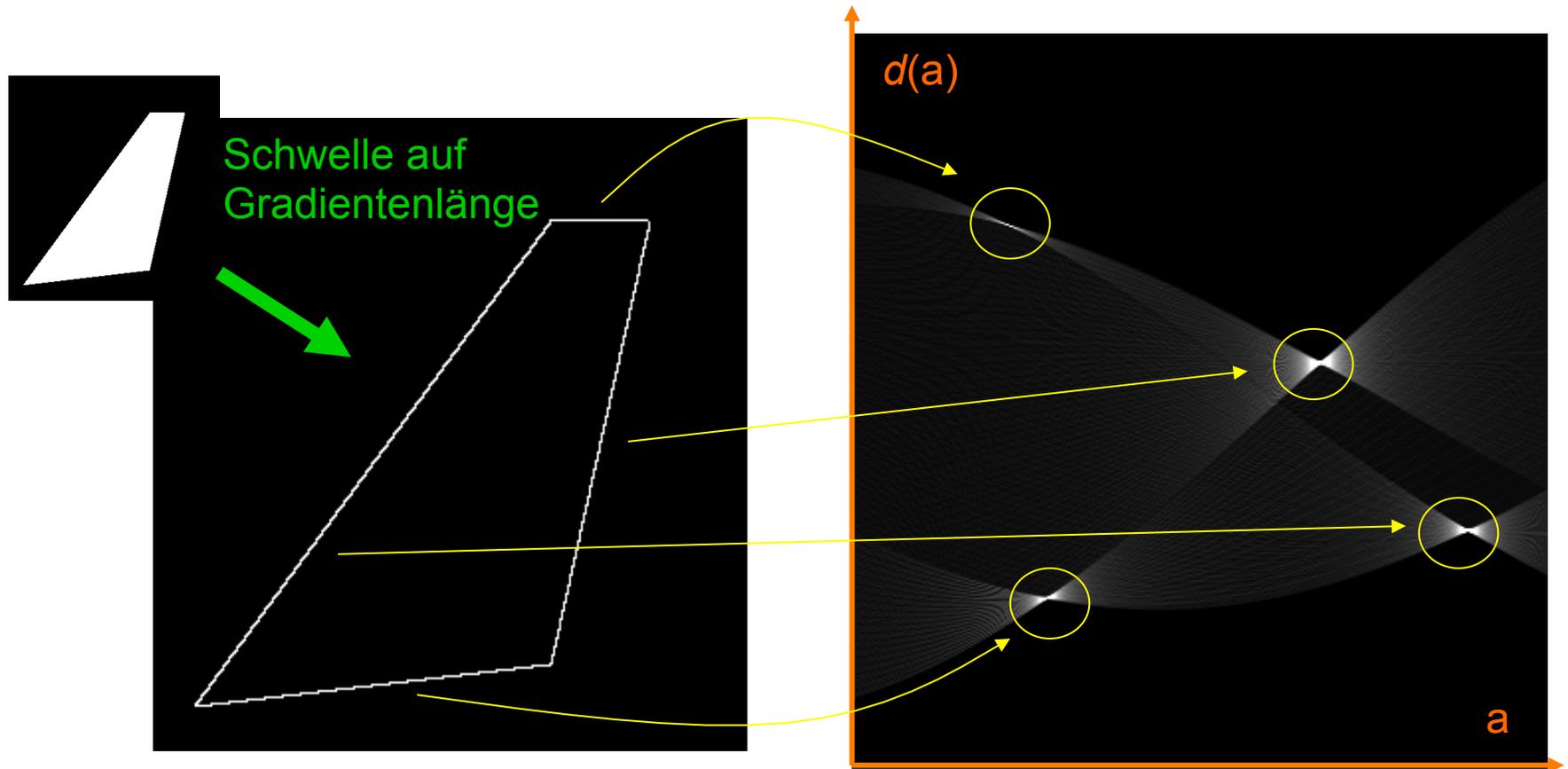


Berechnung der HT



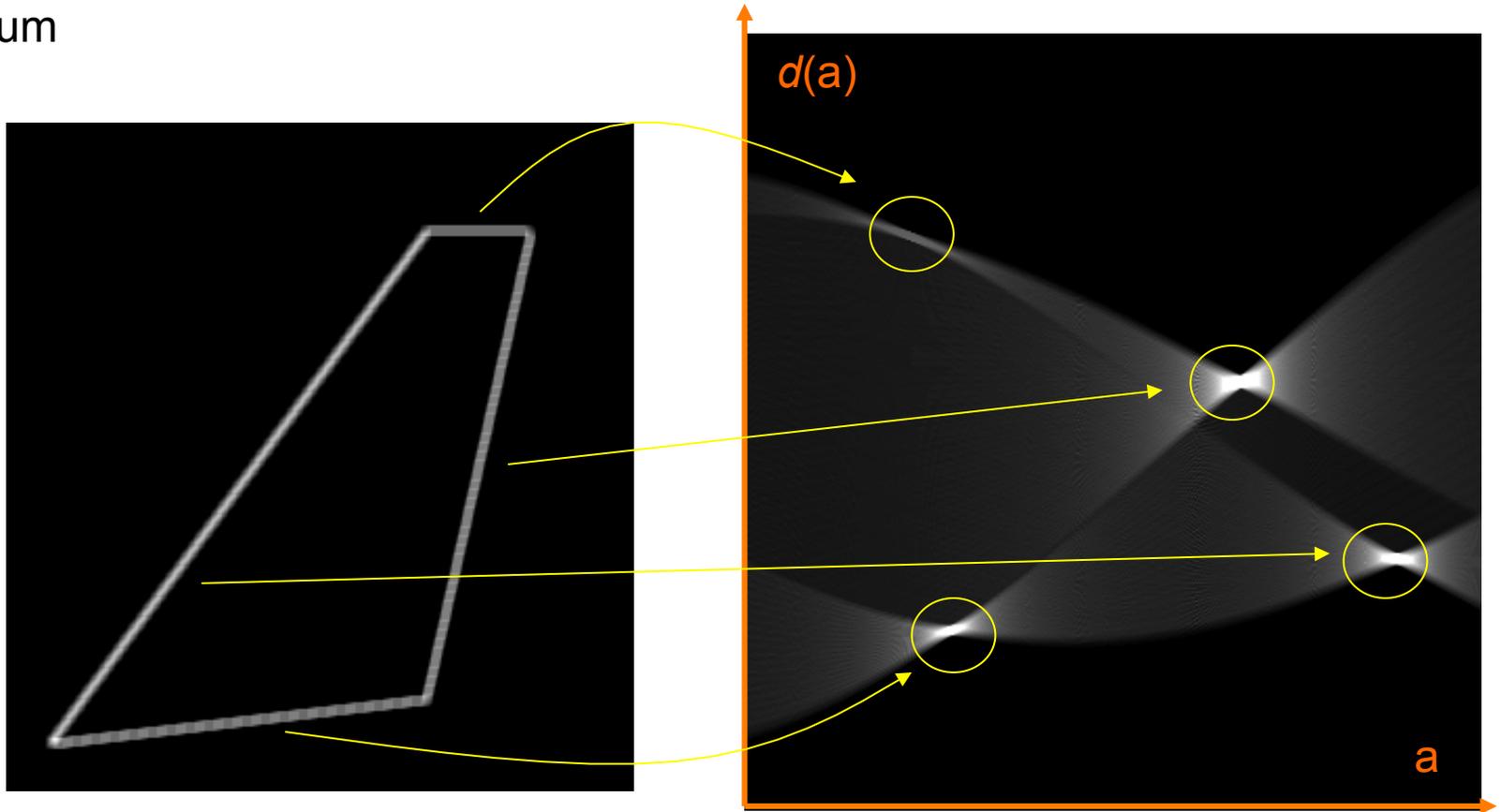
- Erzeugung eines Kantenbilds durch Schwellenwertsetzung auf Gradientenlängen
- Diskretisierung des (α, d) -Raums (Zerlegung in Akkumulatoren)
- Für jeden Punkt x_n, y_n wird eine Kurve im (α, d) -Raum diskretisiert
- Jeder Akkumulator wird inkrementiert, sobald eine Kurve durch ihn verläuft
- Parameter von Linien im Ortsraum sind durch (α, d) -Kombinationen gegeben, deren Wert (Stimmenanzahl, votes) nach Ausführung der Transformation am höchsten sind

Hough Transformation



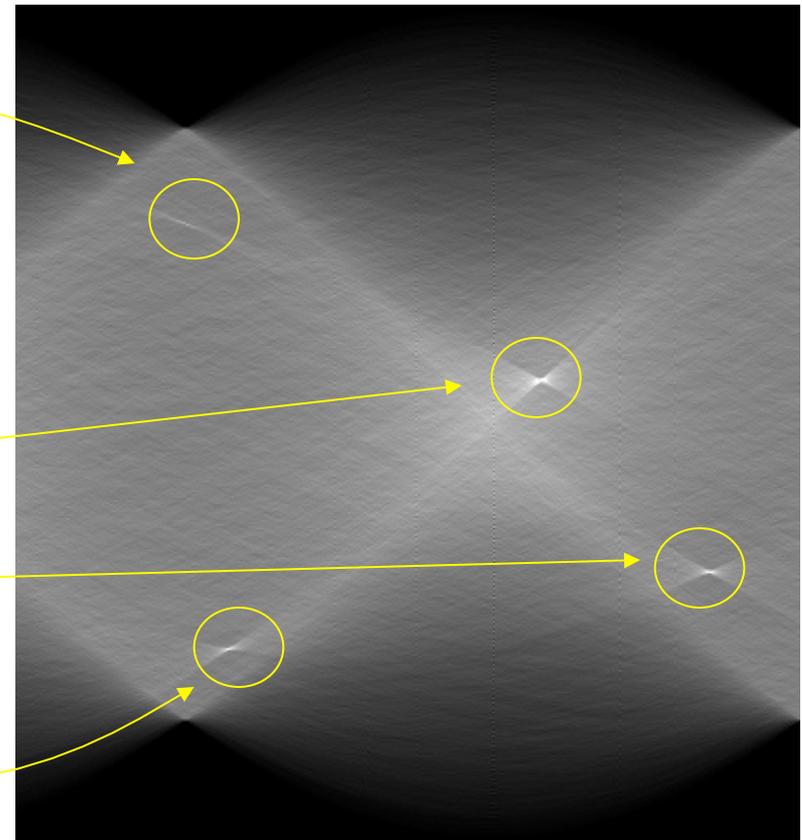
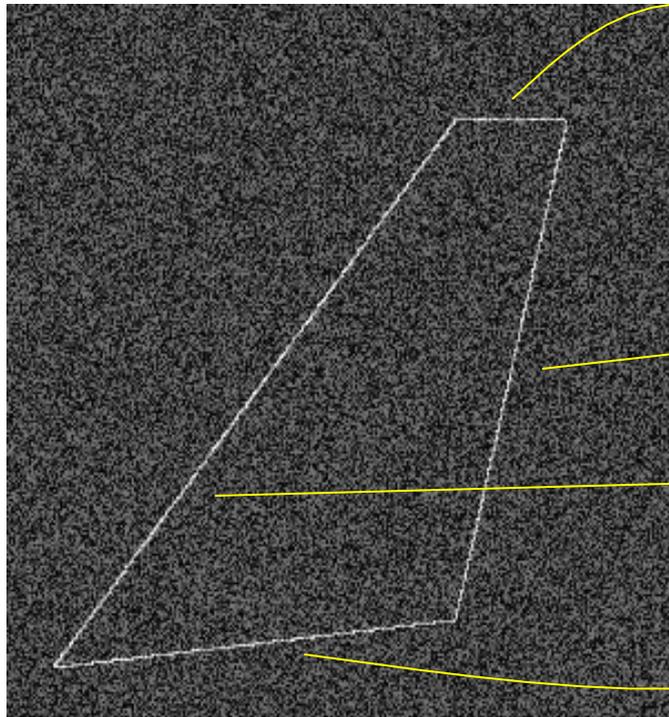
Hough Transformation

Breite Kanten führen zu flacheren Maxima im Hough-Raum



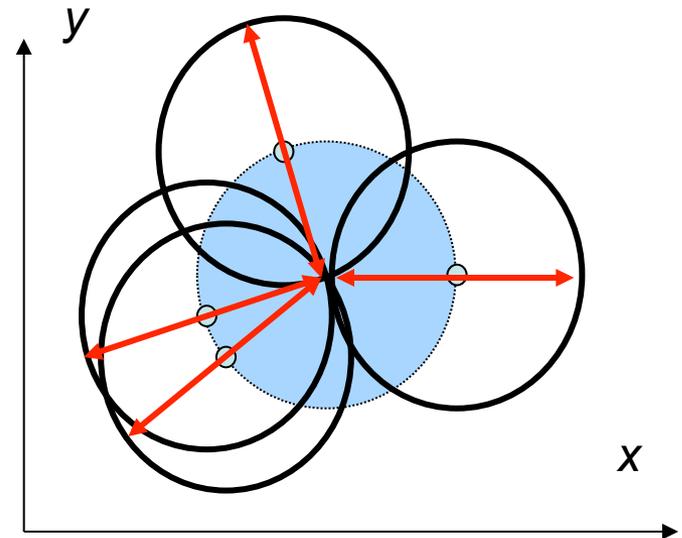
Hough Transformation in nicht-binären Bildern

Bei Gewichtung der Stimmanzahl
durch die Kantenstärke lassen sich
Kanten von Rauschen unterscheiden

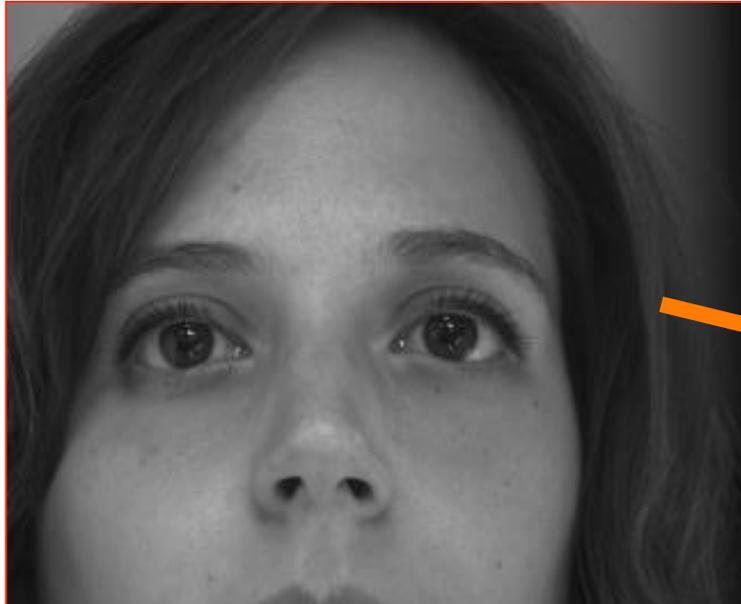


Hough Transformation für Kreise

- Kreisgleichung für Kreis mit Mittelpunkt (x_c, y_c) und Radius r : $(x-x_c)^2 + (y-y_c)^2 - r^2 = 0$
- Falls der Radius bekannt ist, ist nur der Verschiebevektor (x_c, y_c) gesucht
 - Hough-Raum = Ortsraum
 - Um jeden Kantenpunkt wird ein Kreis mit Radius r diskretisiert
- Beschleunigung: Akkumulator wird nur in Distanz r in und entgegen der Gradientenrichtung inkrementiert



Hough Transformation für Kreise



selektierte Kreise mit $r = 4.5\text{mm}$

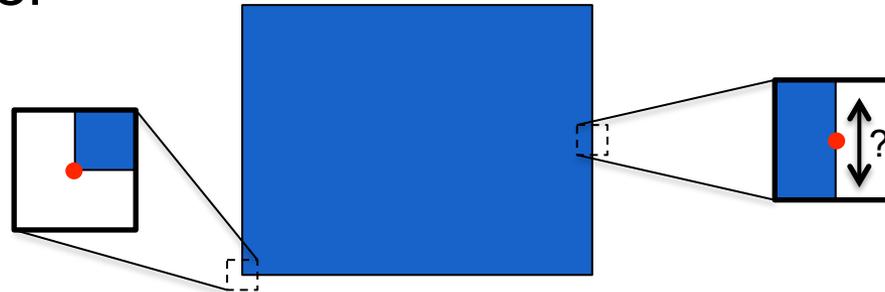
Akkumulatorzellen mit mehr als 33%
der maximalen Stimmenanzahl



ECKEN

Erkennung von Ecken

- Viele Bilderkennungsalgorithmen benötigen Merkmale, die eine stabile Position in (x,y) haben
- Kanten sind nur in einer Richtung lokalisiert
→ Ecken in zwei



- Gewünschte Eigenschaften von Merkmalen (in verschiedenen Bildern vom gleichem Objekt / der gleichen Szene)
 - Genaue Lokalisierbarkeit
 - Invarianz gegenüber Rotation, Skalierung, Helligkeitsänderung
 - Robust gegenüber Rauschen

Slide and illustration adapted from Bern Girod, Digital Image Processing

Verschiebung

- Effekt einer Verschiebung um (kleine) Δx , Δy
 - Flache Region: keine Änderung im Erscheinungsbild
 - Kante: keine Änderung bei Verschiebung entlang der Kante
 - Ecke: große Änderung in jeder Richtung

- Harris-Corner-Detektor beschreibt diese Fälle

- Intensitätsänderung bei Verschiebung um Δx , Δy

$$s(\Delta x, \Delta y) = \sum_{(x,y) \in \text{window}} [f(x, y) - f(x + \Delta x, y + \Delta y)]^2$$

- Lineare Approximation für (kleine) Δx , Δy

$$f(x + \Delta x, y + \Delta y) \approx f(x, y) + f_x(x, y)\Delta x + f_y(x, y)\Delta y$$

- f_x , f_y sind Gradienten in x- bzw. y-Richtung

Verschiebung

- Lineare Approximation für (kleine) Δx , Δy

$$f(x + \Delta x, y + \Delta y) \approx f(x, y) + f_x(x, y)\Delta x + f_y(x, y)\Delta y$$

$$\begin{aligned} s(\Delta x, \Delta y) &\approx \sum_{(x,y) \in \text{window}} [f_x(x, y)\Delta x + f_y(x, y)\Delta y]^2 \\ &= \sum_{(x,y) \in \text{window}} \left[\begin{pmatrix} f_x(x, y) & f_y(x, y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right]^2 \\ &= \sum_{(x,y) \in \text{window}} \left[\begin{pmatrix} f_x(x, y) & f_y(x, y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right] \left[\begin{pmatrix} f_x(x, y) & f_y(x, y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right] \\ &= \sum_{(x,y) \in \text{window}} \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \begin{pmatrix} f_x(x, y) & f_y(x, y) \end{pmatrix} \begin{pmatrix} f_x(x, y) & f_y(x, y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \end{aligned}$$

Verschiebung

$$\begin{aligned}
 &= \sum_{(x,y) \in \text{window}} \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \begin{pmatrix} f_x(x,y) & f_y(x,y) \end{pmatrix} \begin{pmatrix} f_x(x,y) & f_y(x,y) \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\
 &= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \left[\sum_{(x,y) \in \text{window}} \begin{pmatrix} f_x(x,y) & f_y(x,y) \end{pmatrix} \begin{pmatrix} f_x(x,y) & f_y(x,y) \end{pmatrix} \right] \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\
 &= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \left[\sum_{(x,y) \in \text{window}} \begin{pmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{pmatrix} \right] \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\
 &= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \begin{pmatrix} \sum_{(x,y) \in \text{window}} f_x^2 & \sum_{(x,y) \in \text{window}} f_x f_y \\ \sum_{(x,y) \in \text{window}} f_x f_y & \sum_{(x,y) \in \text{window}} f_y^2 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} M \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}
 \end{aligned}$$

Harris Corner Detector (Harris, Stephens, 1988)

- “Second-moment matrix”

$$M = \sum_{(x,y) \in \text{window}} w(x,y) * \begin{pmatrix} f_x^2(x,y) & f_x(x,y)f_y(x,y) \\ f_x(x,y)f_y(x,y) & f_y^2(x,y) \end{pmatrix}$$

$w(x,y)$: Gewichtungsfunktion, z.B. Gauß-Funktion

- Maß für die “Stärke” der Ecke

$$C(x,y) = \det(M) - k(\text{trace}(M))^2 = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2$$

$$k = 0.04..0.06$$

Harris Corner Detection – Algorithmus

- Berechne die Ableitungen f_x und f_y in x- und y-Richtung
- Berechne für jedes Pixel die Produkte der Ableitungen
$$f_x^2 = f_x f_x, \quad f_y^2 = f_y f_y, \quad f_{xy} = f_x f_y$$
- Berechne für jedes Pixel (x,y) die gewichtete Summe im Fenster um (x,y)
- Definiere für jedes Pixel (x,y) die Matrix $M(x,y)$
- Berechne für jedes Pixel das Maß der Eckenstärke
- Wende eine untere Schranke und non-maximum Unterdrückung an