

# 1. EINFÜHRUNG

## 1.1. DEFINITIONEN UND ÜBERBLICK

- **Bildverarbeitung**

**Eingang (Bild) -> Operation(en) -> Ausgang (Bild)**

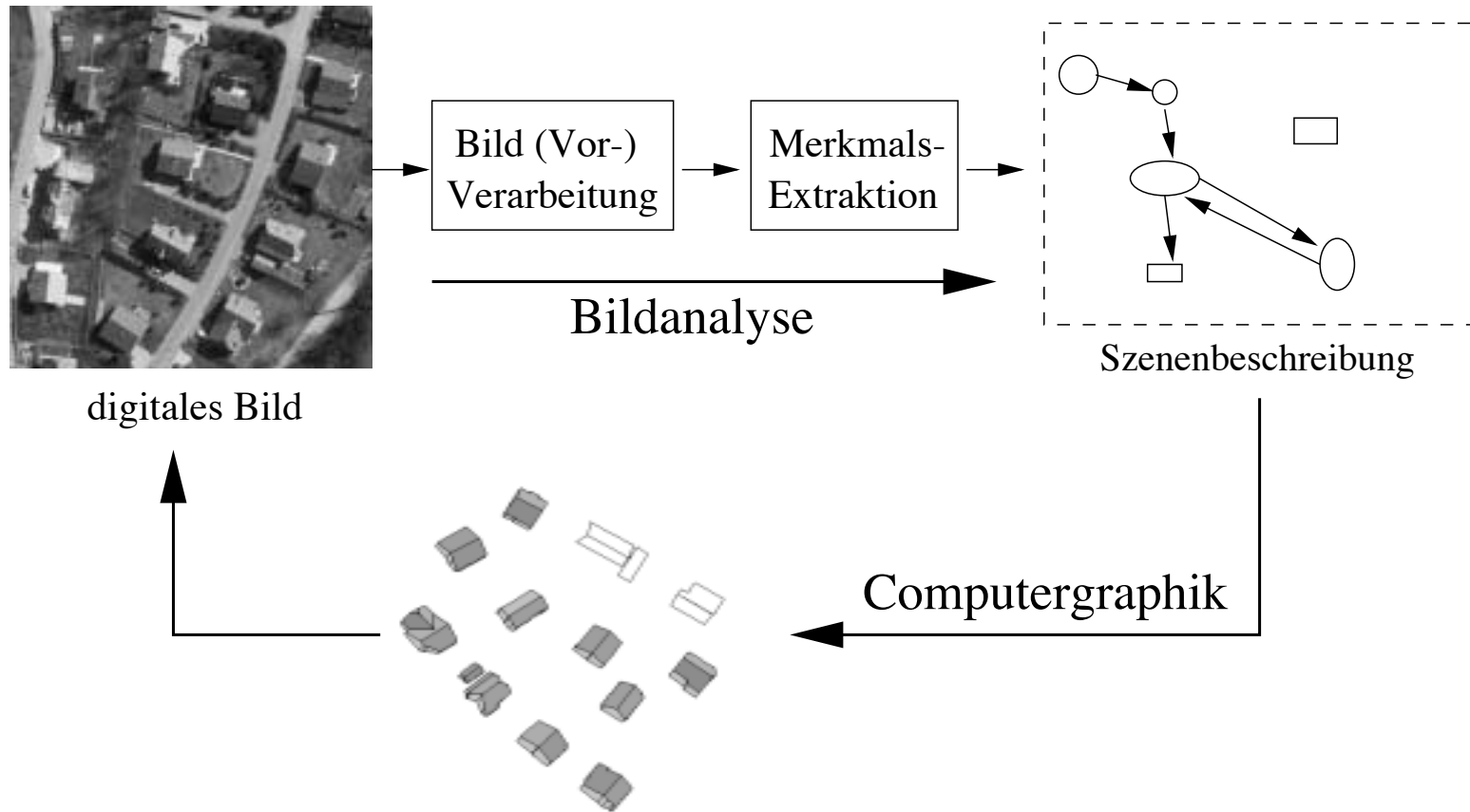
- **Bildanalyse**

**Eingang (Bild) -> Operation(en) -> Ausgang (Bild und/oder extrahierte geometrische und semantische Informationen)**

**Beinhaltet Bildverarbeitung**

**Nachbarbegriffe: Computer Vision, Machine Vision, Bildverstehen**

**Computergraphik (Bildsynthese): der inverse Prozess zur Bildanalyse**

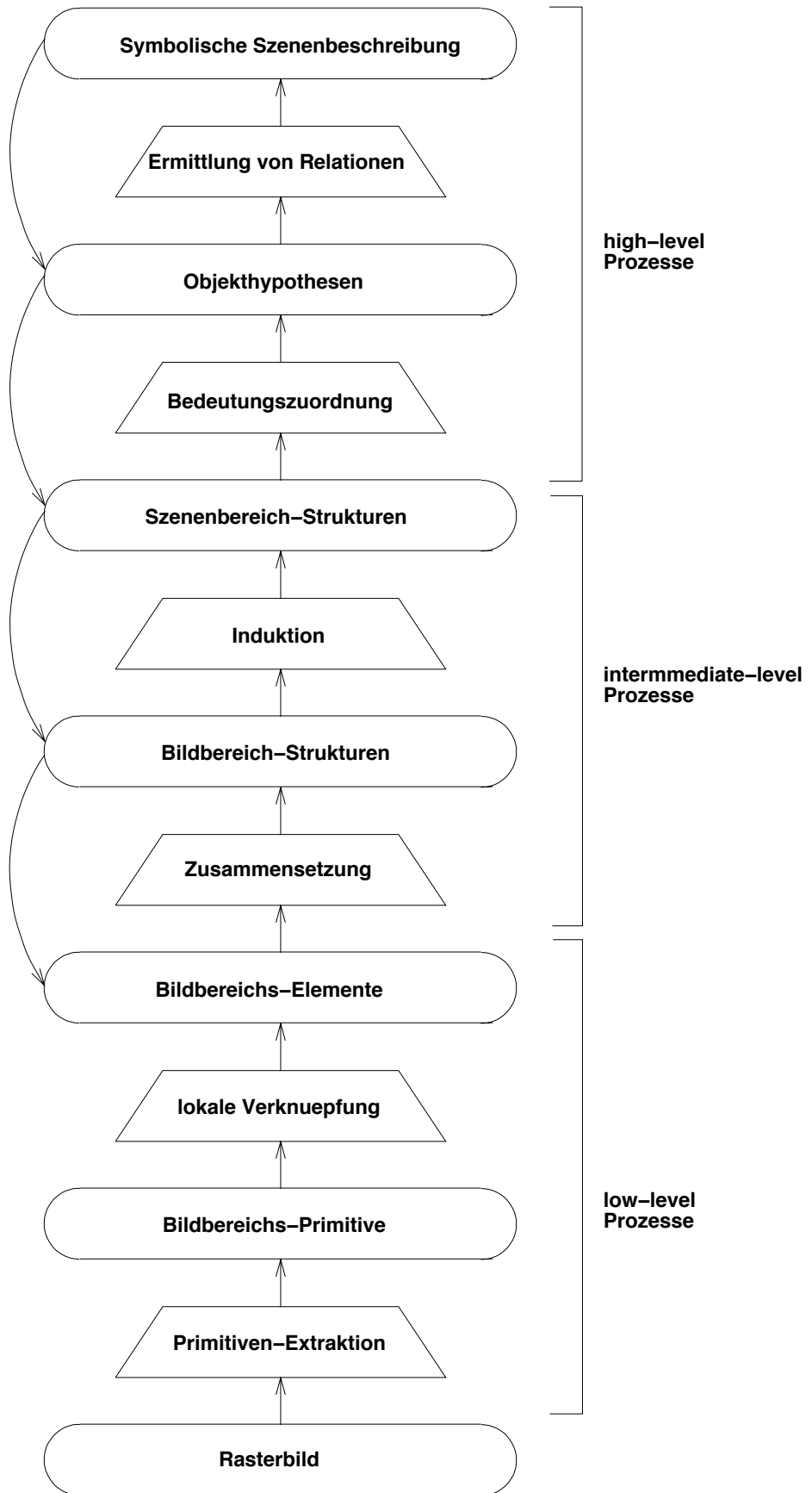


## Bildanalyse versus Computergraphik (Bildsynthese)

Top-down Verfahren  
(modell-, wissens-basierte)



Bottom-up Verfahren  
(daten-basierte)



## **ZIEL DER BILDANALYSE**

- **Geometrische, topologische und semantische Informationen über Objekte abzuleiten, sowie Relationen zwischen den Objekten**
- **Wichtige Faktoren:**  
**Genauigkeit, Zuverlässigkeit, Rechenaufwand, Flexibilität, Kosten, Komplexität, AUTOMATION**
- **Hauptproblem:**
  - **Intelligenz der Verfahren, vor allem zur Extraktion von semantischen Informationen, ist sehr gering**
  - **Szenen und Objekte sehr variabel und komplex**



### Wahrnehmung: Mensch versus Maschine

Wie lange wird es noch dauern bis der Rechner ein solches Bild (eine Matrix mit 0 und 1 Werten) interpretieren kann?

## **ZIEL DER VORLESUNG**

- **Nur ein kleiner Teil der Bildanalyse wird hier behandelt**
- **Mehr: Für ein bestimmtes Problem/Anwendung was kann ich einsetzen? Wie wähle ich einen geeigneten Algorithmus (Beurteilung ihrer Leistung)? Wie setze ich verschiedene Tools zusammen um kompliziertere Aufgaben zu lösen**

**Weniger: Theorie der Methoden. Wie kann ich neue Methoden entwickeln?**

- **Drei wesentliche Themen werden behandelt:**
  - **Bildverarbeitungsoperationen. Sie dienen vor allem für Vorverarbeitung (damit die nachfolgenden Methoden bessere/schnellere Ergebnisse liefern), Extraktion von Merkmalen, visuelle Repräsentation und Kombination von Daten**
  - **Extraktion von Merkmalen (Punkte, Linien, Flächen). Es wird prinzipiell nur die Geometrie behandelt, nicht die Semantik/Interpretation**
  - **Matching (Zuordnungs-) Verfahren (sehr nützliche Verfahren mit sehr breiten Anwendungen), automatische DTM Generierung**

## **1.2. ANWENDUNGEN**

- **Beziehungen zwischen Bildanalyse und folgenden Themenbereiche des NDK:**  
**Photogrammetrie, Fernerkundung, DTMs, manuelle und automatische Digitalisierung, Datenanalyse, Datenrepräsentation**
- **Verfahren der Bildanalyse einsetzbar in:**
  - **Datenakquisition**
  - **Datenspeicherung (z.B. Bildkompression), Datenabfrage (z.B. kontext-basierte Abfrage von Bildern)**
  - **Datendarstellung und Ausgabe (z.B. 3-D synthetische Ansichten, Animation etc.)**
  - **Datenanalyse (z.B. Rasterdatenanalyse-Funktionen in GIS)**
  - **Kombination, Fusion von Daten (z.B. Kombination von Bilddaten verschiedener Sensoren)**
  - **Datengeneralisierung**

## 1.3. ENTSTEHUNG UND GEOMETRIE VON DIGITALEN BILDERN

### DIGITALE BILDER

- **Bild: 2D - Lichtintensitätsfunktion  $f(x,y)$**

**Regelmässiges (normalerweise quadratisches) Gitter bestehend aus Bildelementen = Picture Elements = Pixels**

**Gitterwerte (= Grauwerte) proportional zur Helligkeit des Bildes**

- **Entsteht durch Diskretisierung in Bezug auf die Koordinaten  $(x,y)$  (Sampling) und auf die Helligkeit (Quantisierung)**
- **Relation zwischen Sampling und Pixelgrösse**
  - **Definition von Pixelgrösse:**
    - (a) physikalisch auf dem Sensor**
    - (b) im Objektraum (footprint) = Sensor-Pixelgrösse x Massstabsfaktor**
  - **Feines Sampling -> kleine Pixelgrösse**

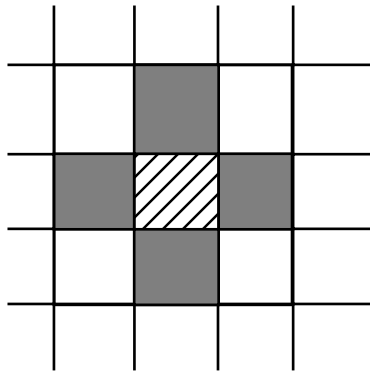


- **Relation zwischen Quantisierung und Anzahl Grauwerte**
  - **Feine Quantisierung -> grosser Grauwertbereich**
  - **Quantisierung wird mit Anzahl Bit pro Grauwert angegeben, z.B.**
    - n-bit Quantisierung ->  $2^n$  Grauwerte im Bild**
  - **Anzahl Bit**
    - 8-bit (256 Grauwerte): Normalfall**
    - 1-bit (2 Grauwerte): binäres Bild**
    - 24-bit (16,777,216 Grauwerte): Farbbild**
    - n-bit ( $1 < n < 8$ ): gescannte Karten, thematische Karten (Klassifikationsbilder)**
    - andere (10 - 16 bit): Medizin, Astronomie**

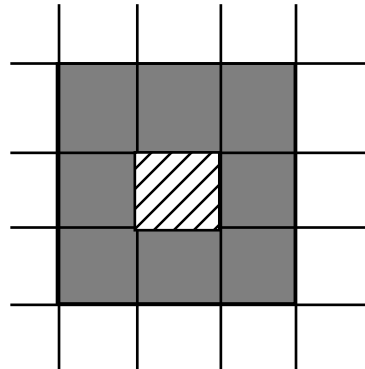
- **Andere Daten können als Bilder betrachtet (repräsentiert) und mit denselben Bildanalysealgorithmen verarbeitet werden, z.B.**
  - **andere Raster-Daten**  
**DTM, gescannte Karten und Pläne, Klassifikationsbilder etc.**
  - **Vektordaten**  
**Nach einer Vektor-zu-Raster Konvertierung**
  - **Sachdaten**  
**Nach einer Interpolation eines regelmässigen Gitters aus punktuellen Messungen von Eigenschaften (Lärmemission, Temperatur etc.)**
- **Erweiterung auf 3-D (Voxel statt Pixel), sowie “Multi-Valued” Rasterdaten möglich**
- **Bildanalyse gewinnt an Aktualität und Praxisreife**
  - **Anzahl Sensoren/Bildquellen und Anwendungen nimmt zu (satelliten- und flugzeug-gestützt, MultiMedia, Virtual Reality)**
  - **Tendenz in Richtung hybrider GIS**

# DISKRETE BILDGEOMETRIE

- **Definition der Nachbarschaft**



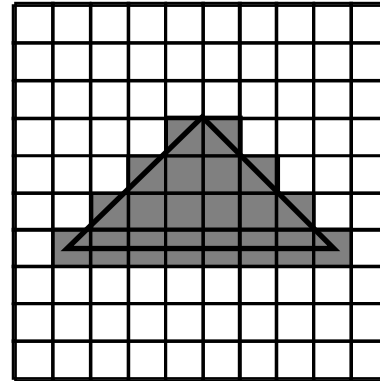
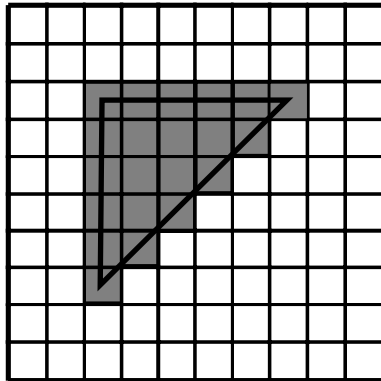
**4er-Nachbarschaft**



**8er-Nachbarschaft**

- **Probleme in der Definition von geometrischen Grössen (Abstand, Form etc.)**
  - Einfluss von Sampling
  - Einfluss der Drehlage des Objektes
  - Einfluss wegen der unterschiedlichen mathematischen Definition von geometrischen Grössen, z.B. Art der Nachbarschaft

## Beispiel



**Abbildung eines durch eine kontinuierliche Linie umschlossenen Objekts auf die diskrete Bildebene.**

**Form des digitalisierten Objekts hängt von der Drehlage des kontinuierlichen Objekts ab.**

- **Definition von Distanzen (zwischen Pixeln A und B)**

- $d_4(AB) = n_x + n_y$  , für die 4er-Nachbarschaft

- $d_8(AB) = \max(n_x, n_y)$  , für die 8er-Nachbarschaft

- $d(AB) = n_x + n_y + n_d \sqrt{2}$  , genauere Definition

mit  $n_x$  ,  $n_y$  ,  $n_d$  ...Anzahl horizontaler, vertikaler, diagonalen Schritte

**Beispiel für Längenmessungen mit verschiedenen Distanzmassen und Einfluss der Drehlage auf Umfang und Fläche digitalisierter Objekte (s. vorherige Figur mit den zwei Dreiecken). Die exakte Dreiecksfläche beträgt 12.5 Flächeneinheiten, der exakte Umfang 17.07 Längeneinheiten**

<b>Distanzmass</b>	<b>linkes Dreieck</b>	<b>rechtes Dreieck</b>
<b>Anzahl der Randpixel</b>	<b>15</b>	<b>14</b>
<b>Umfang (4er-Nachbarschaft)</b>	<b>20</b>	<b>20</b>
<b>Umfang (8er-Nachbarschaft)</b>	<b>15</b>	<b>14</b>
<b>Umfang (genauere Definition)</b>	<b><math>10 + 5 \sqrt{2} = 17.07</math></b>	<b><math>8 + 6 \sqrt{2} = 16.5</math></b>
<b>Fläche (Anzahl Pixel)</b>	<b>21</b>	<b>20</b>

- **Diskretisierungsprobleme bei der Winkelmessung**

**Kleinster auflösbarer Winkel  $\alpha = 90^\circ / (n-1)$**

**mit n ...Dimension der quadratischen Nachbarschaft zur Winkelberechnung**

## **2. BILDVERARBEITUNGSOPERATIONEN**

### **2.1. KLASSIFIKATION VON OPERATIONEN**

- **Punktoperationen, z.B. Bild  $f + 20$ , d.h. Aufhellung**
- **Nachbarschaftoperationen, z.B. 3 x 3 lokale Mittelwertbildung**  
**Nachbarschaft: quadratisch, rechteckig, kreuzförmig, kreisförmig etc.**
- **“Zonal”-Operationen**  
**z.B. Eingabe: (a) 1 Bild mit Teilung in Zonen (Klassen). Jede Zone kann aus mehreren separaten Regionen bestehen. (b) 1 Bild mit Wert für jedes Pixel. Ausgabe: der Wert jedes Pixel hängt von den Werten aller Pixel, die der gleichen Zone gehören, z.B. Wert des Pixels = Mittelwert aller Pixel der gleichen Zone.**
- **Global, d.h. Wert eines Pixels hängt von Werten der Pixel des ganzen Bildes (oder einem Teil davon)**  
**z.B. jedes Pixel kriegt als Wert die ID oder die Distanz oder die Richtung zum nächsten “Destination”-Pixel oder Linie (sind vorgegeben) oder**  
**Finde die n% höchsten/kleinsten Grauwerte**

### **Bemerkungen:**

- **In Nachbarschaft- oder Zonal-Operationen wird das Bild konzeptuell maskiert. Die Maske ist eine Division des Bildes in 2 oder mehr Klassen.**
- **Es gibt Operationen (Nachbarschaft-, Zonal-, Global-Operationen), welche die Grauwerte nicht verändern, z.B. finde den Mittelwert, Standardabweichung etc.**

### **Bemerkung:**

**Geometrische Operationen, Grauwertinterpolation, sowie Registrierung eines Bildes mit Bild/Karte/Vektoren und Orthobildgenerierung werden in der Vorlesung Photogrammetrie behandelt.**

## 2.2. ARITHMETISCHE OPERATIONEN

- **Addition, Subtraktion, Multiplikation, Division zwischen zwei Bildern oder einem Bild und einer Zahl**
- **$\exp(f)$ ,  $\log(f)$ ,  $\max(f,g)$ ,  $\min(f,g)$  etc. ( $f, g$  ...Grauwertbilder)**
- **In den meisten Fällen soll das Ergebnisbild im Grauwertbereich  $[0,255]$  transformiert werden (Bilder werden normalerweise mit 8-bit dargestellt). Sonst, ist die Darstellung falsch ("wrap-around"), z.B.**

**der Wert -2 wird als  $(-2+256) = 254$  dargestellt, also hell statt dunkel**

**der Wert 280 wird als  $(280-256) = 24$  dargestellt, also dunkel statt hell**

**Diese Transformation im Bereich  $[0,255]$  nennt man Normalisierung. Bei der Subtraktion können auch die absoluten Differenzen benutzt werden.**

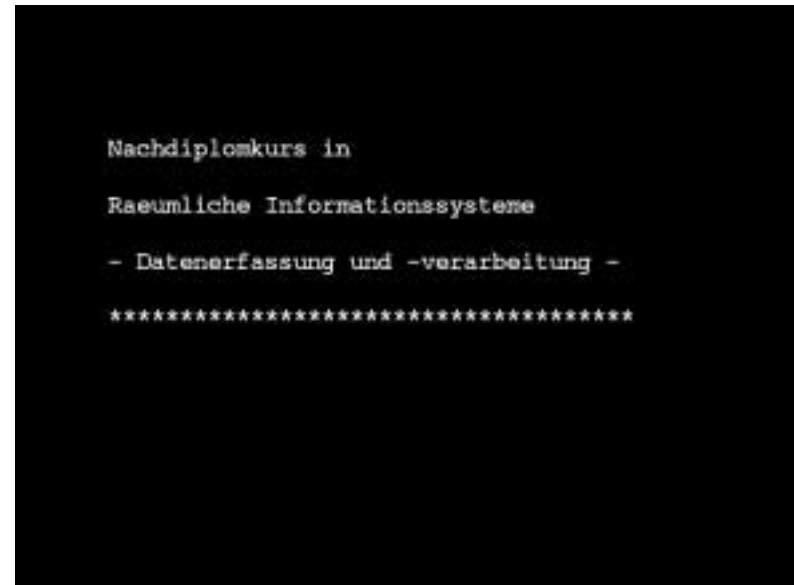
- **Andere singuläre Fälle (z.B. Division durch 0,  $\log(0)$  ) sollten auch abgefangen werden**



A



B



C



**Beispiel von Addition und Subtraktion:  
Eingabe Bilder**



**A + B (mit Normalisierung)**



**max (A , B)**



**A - B (mit Normalisierung)**



**A - B (absolute Werte)**

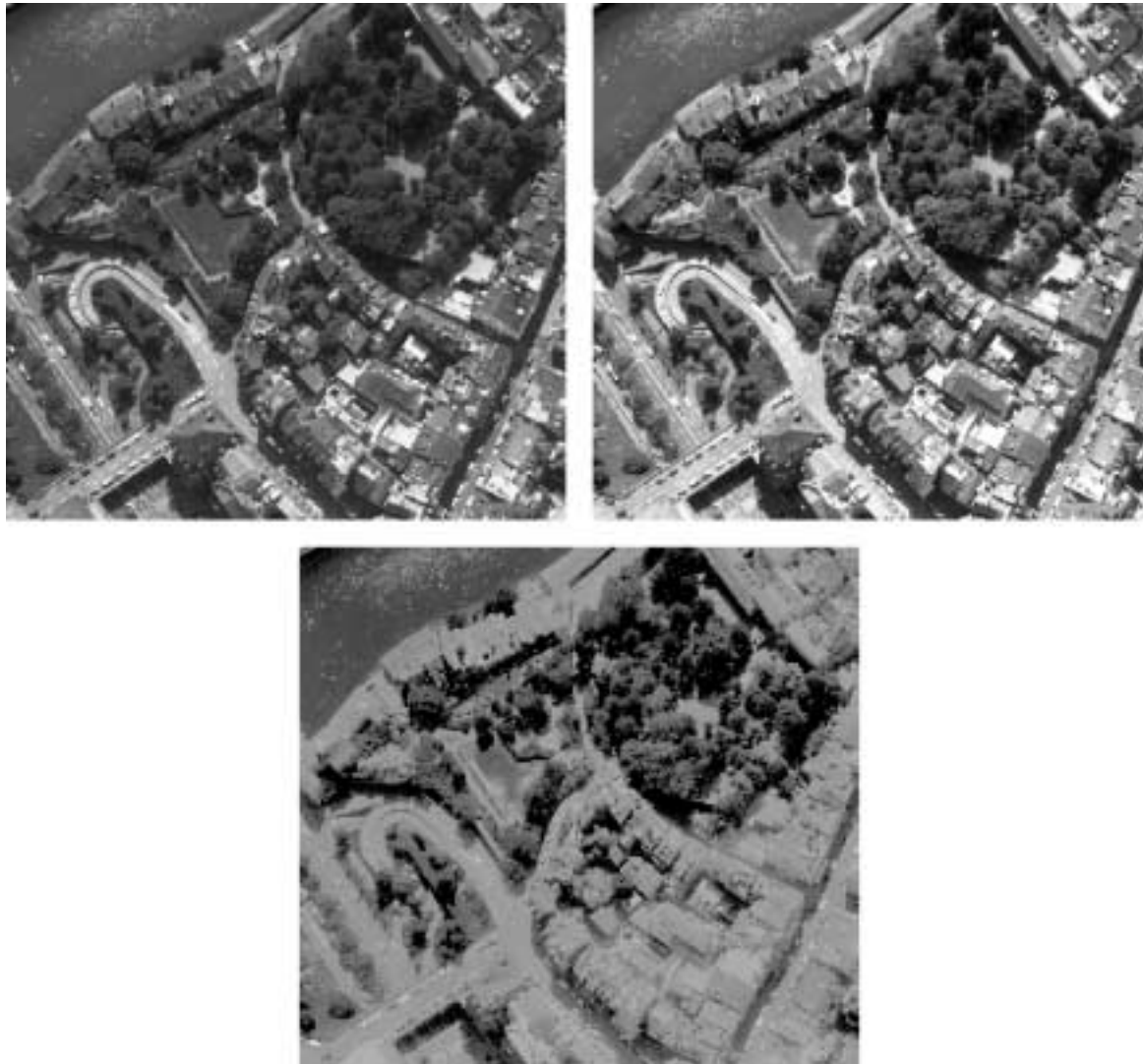
## ANWENDUNGEN VON ARITHMETISCHEN OPERATIONEN

- **Addition: Rauschreduktion (Mittelwertbildung)**
- **Subtraktion: Hintergrundsubtraktion (z.B. Aufhellung des Bildes durch atmosphärische Streuung), Bewegungsdetektion, Detektion von Änderungen zwischen multi-temporalen Bildern**
- **Multiplikation: Kreuzkorrelation zwischen zwei Bildern, Maskierung von Bildern (Ausblenden einzelner Teile), z.B. mit der Maske**

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

- **Division: Verhältnisbilder in der Analyse multispektraler Bilder -> bessere Klassifikation, Interpretation von gewissen Merkmalen**

## Beispiel Division

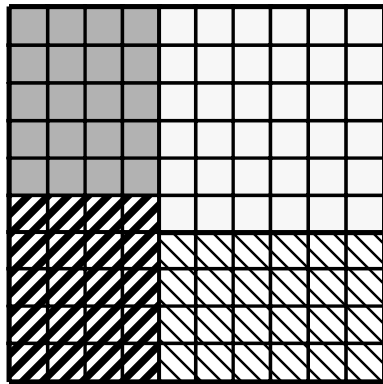


**Oben: zwei spektral Kanäle (Stadt Zürich). Unten: Division der 2 Kanäle ; anthropogene Objekte (Strassen, Gebäude) sind homogener und unterschiedlicher zu natürlichen Objekten als in den Originalbildern.**

## Beispiel

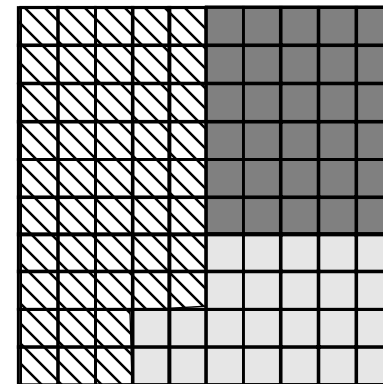
**Bild A**

**Steuerfuss in % (Werte 0. -1.)**



**Bild B**

**Preis des Bodens in Fr./m<sup>2</sup> in Rasterform**



**4 Gemeinden in Rasterform**

**(1 Pixel = 10 m<sup>2</sup>)**

$$c = 10$$

$$D = B * c = \text{Preis in Fr. jedes Pixels vom Bild B}$$

$$E = D * A = \text{Steuer für jedes Pixel vom Bild A}$$

- **Log(f):** dunkle/helle Regionen werden gestreckt/komprimiert

- **Exp(f):**  $f^a$

**$a < 1$ :** dunkle/helle Regionen werden gestreckt/komprimiert

**$a > 1$ :** helle/dunkle Regionen werden gestreckt/komprimiert



**Links: original Bild A ; Mitte In (A) ; Rechts: sqrt (A)**

## 2.3. LOGISCHE OPERATIONEN

- **Logisches UND (AND)**

**f = (1010 0010)**

**g = (0111 0110)**

**f AND g = (0010 0010)**

**Anwendung: bestimmte Bildbereiche ausmaskieren, bestimmte Bitebenen selektieren**

- **Logisches (inklusive) ODER (IOR oder OR)**

**f = (1010 0010)**

**g = (0111 0110)**

**f OR g = (1111 0110)**

**Anwendung: Mischung von zwei Bildern (helle Regionen setzen sich durch)**

- **Logisches exklusives ODER (XOR)**

$$f = (1010\ 0010)$$

$$g = (0111\ 0110)$$

$$f \text{ XOR } g = (1101\ 0100)$$

**Anwendung: Feststellung der Differenzen von zwei Bildern (Ergebnis > 0, nur dort wo die Bilder unterschiedlich sind)**

- **Logisches NICHT (NOT)**

$$f = (1010\ 0010)$$

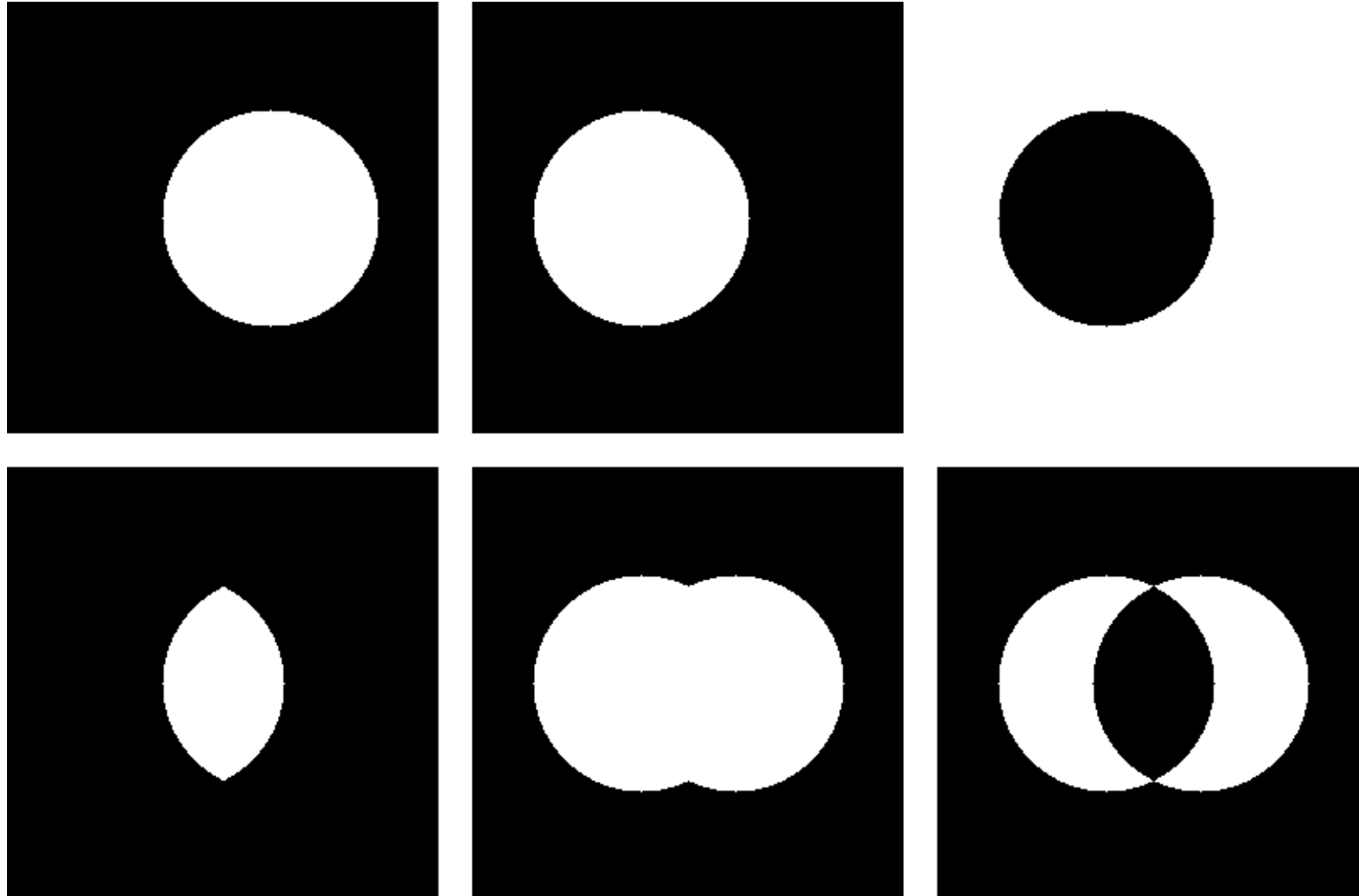
$$g = (0111\ 0110)$$

$$f \text{ NOT } g = (1000\ 0000)$$

**NOT g entspricht einer Inversion des Bildes g, d.h. der Grauwert g wird 255-g**



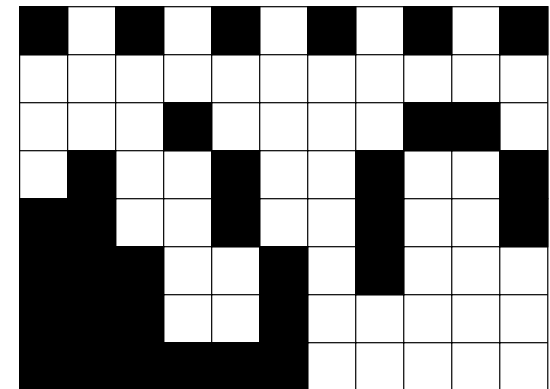
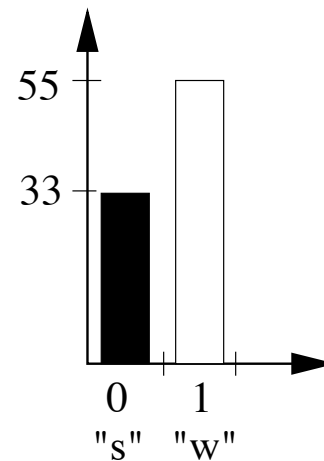
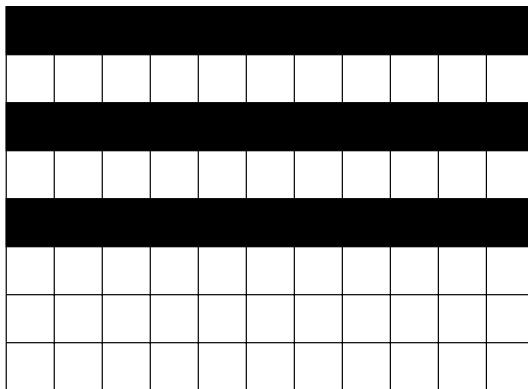
## Beispiel logischer Operationen



Zeilenweise von oben links: Eingabebilder A und B, NOT B (Inversion von B), A AND B, A OR B, A XOR B

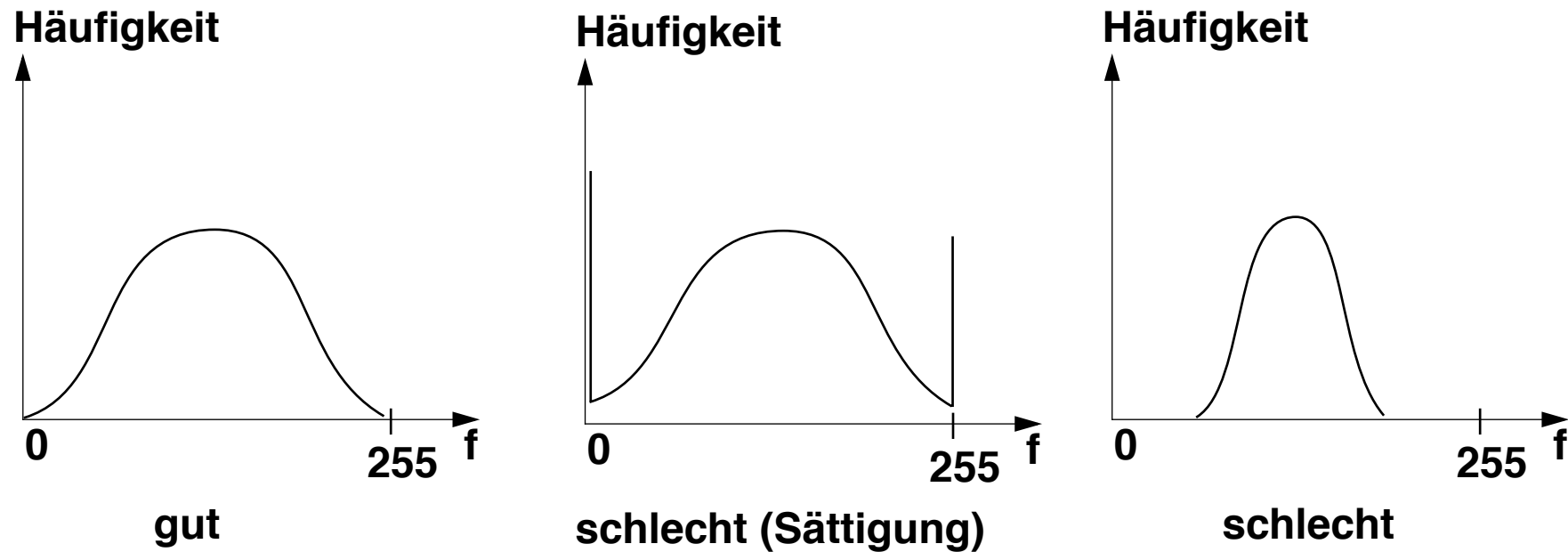
## 2.4. HISTOGRAMM-TRANSFORMATIONEN

- **Histogramm:** gibt Häufigkeitsverteilung der Grauwerte über das gesamte Grauwertintervall (z.B. 0, ...255), d.h. Anzahl der Pixel pro Grauwert.
- **Keine Lageinformation** für Pixel im Histogramm, d.h. eindeutiges Histogramm für ein Bild, aber kein eindeutiges Bild für ein Histogramm.



**Bilder sind unterschiedlich aber haben das gleiche Histogramm!**

- Histogramm als Indikator der Bildqualität:



- kumulatives Histogramm CH:

$$CH(j) = \sum_{i=0}^j H(i)$$

wo H ...Histogramm,  $0 \leq j \leq 255$

- **Histogrammoperationen**

**Oft interaktiv, via Look-Up Tabellen (LUTs) angebracht**

**- H neu = H alt + a**

**Aufhellung (wenn a > 0) oder Abdunklung (wenn a < 0)**

**- H neu = a \* H alt**

**Kontrastverstärkung (wenn a > 1) oder Komprimierung (wenn a < 1)**

**- Berechnung von Mittelwert und Varianz der Grauwerte (N = Anzahl Pixel):**

$$\bar{f} = \frac{\sum_{f=0}^{255} fH(f)}{N}$$

$$\sigma_f^2 = \frac{\sum_{f=0}^{255} (f - \bar{f})^2 H(f)}{N}$$

## 2.5. LOOK-UP TABELLE (LUT) OPERATIONEN

- Implementiert in Hardware -> sehr schnell
- Ursprüngliches Bild bleibt unverändert, nur Display wird geändert
- $f$  ...Grauwerte (Eingang) ,  $g$  ...Grauwerte (Ausgang)
- Clipping

$$g = \begin{cases} 0 & \text{wenn } f < z1 \\ 255 \times (f - z1) / (z2 - z1) & \text{wenn } z1 \leq f \leq z2 \\ 255 & \text{wenn } f > z2 \end{cases}$$

- Inversion
- Gamma-Korrektur

Oft wird  $\text{Gamma} > 1$  gewählt -> dunkle/helle Regionen werden gestreckt/komprimiert (auch logarithmische Transformation genannt)

- **Binarisierung (T = Schwellenwert)**

$$g = \begin{cases} 0 & \text{wenn } f < T \\ \text{sonst} & 255 \end{cases}$$

- **Äquidensitenttransformation**

$$g = u_i \quad \text{für} \quad z_i \leq f < z_{i+1} \quad \text{mit } i = 0, 1, \dots$$

Wird oft mit Pseudofarbedarstellung kombiniert, z.B.

Themalkartierung: Temperatur  $\rightarrow f \rightarrow$  Farbe

- **Pseudofarbedarstellung**

Grauwert  $f$  ist Eingang für 3 LUTs für die R-, G-, B-Ausgabe

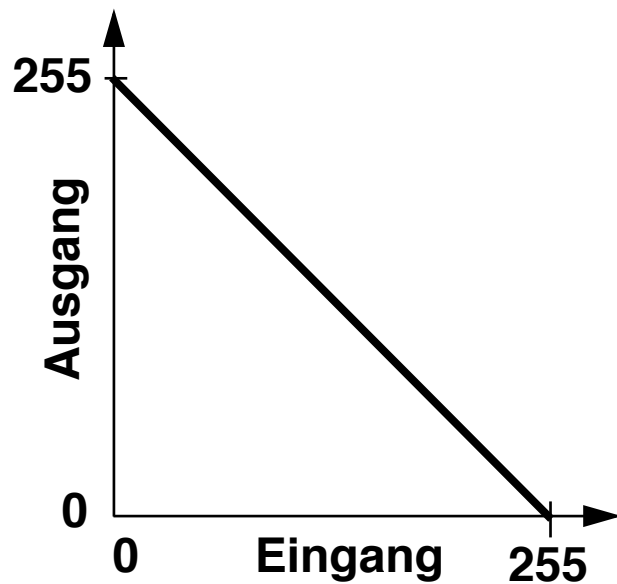
Anwendung: farbige Hervorhebung wichtiger Details, farbige Markierung von bestimmten Grauwerten

- **Falschfarbedarstellung**

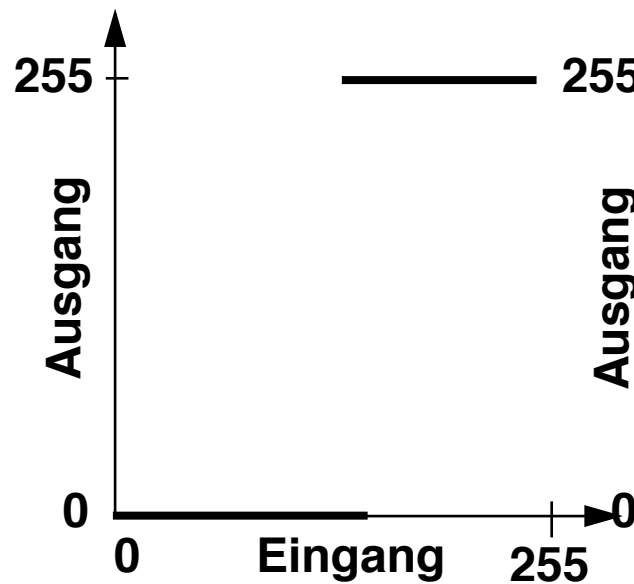
**Drei Bilder (Kanäle) werden als R-, G-, B-Bilder dargestellt, d.h. es werden drei separate LUTs benutzt**

**Anwendung: Darstellung von nicht sichtbaren Kanälen in der Fernerkundung (z.B. Infrarot, Mikrowellen), kombinierte Darstellung von Bildern und anderen Informationen (z.B. DTM)**

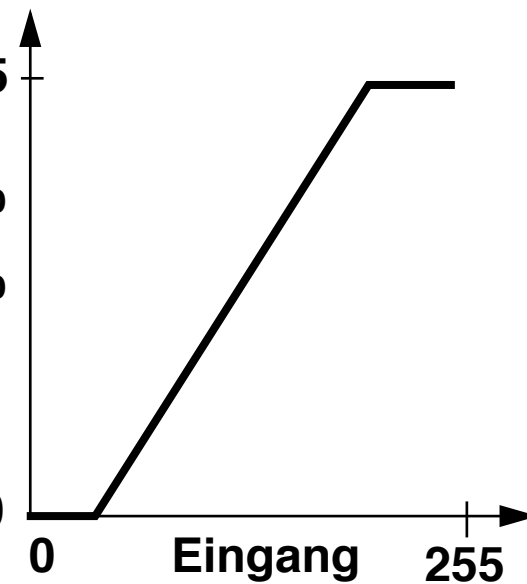
**Beispiele von LUT-Operationen**



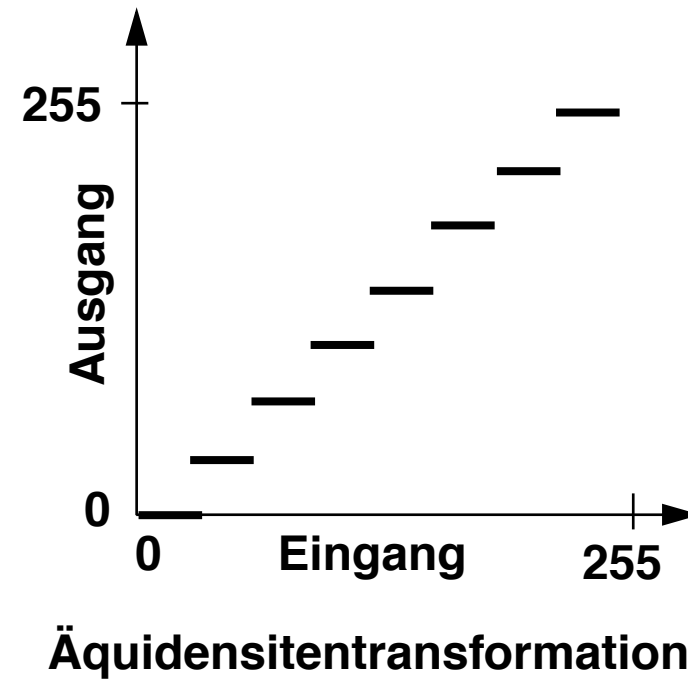
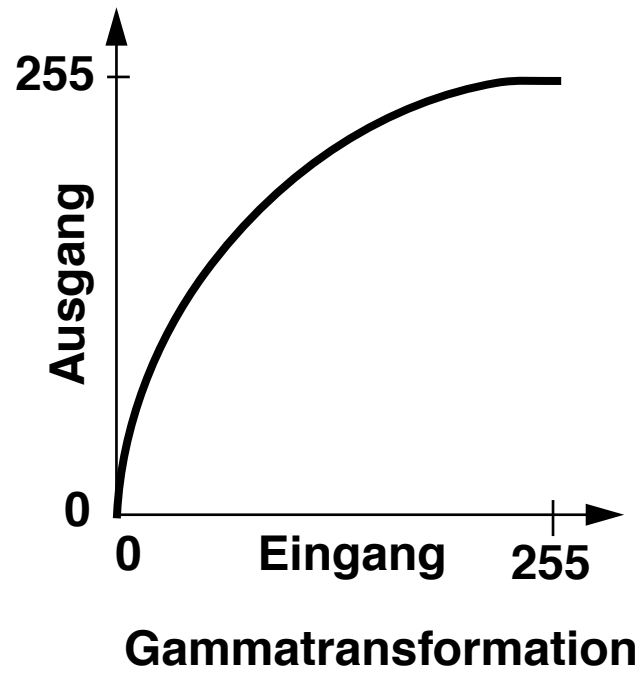
**Inversion**



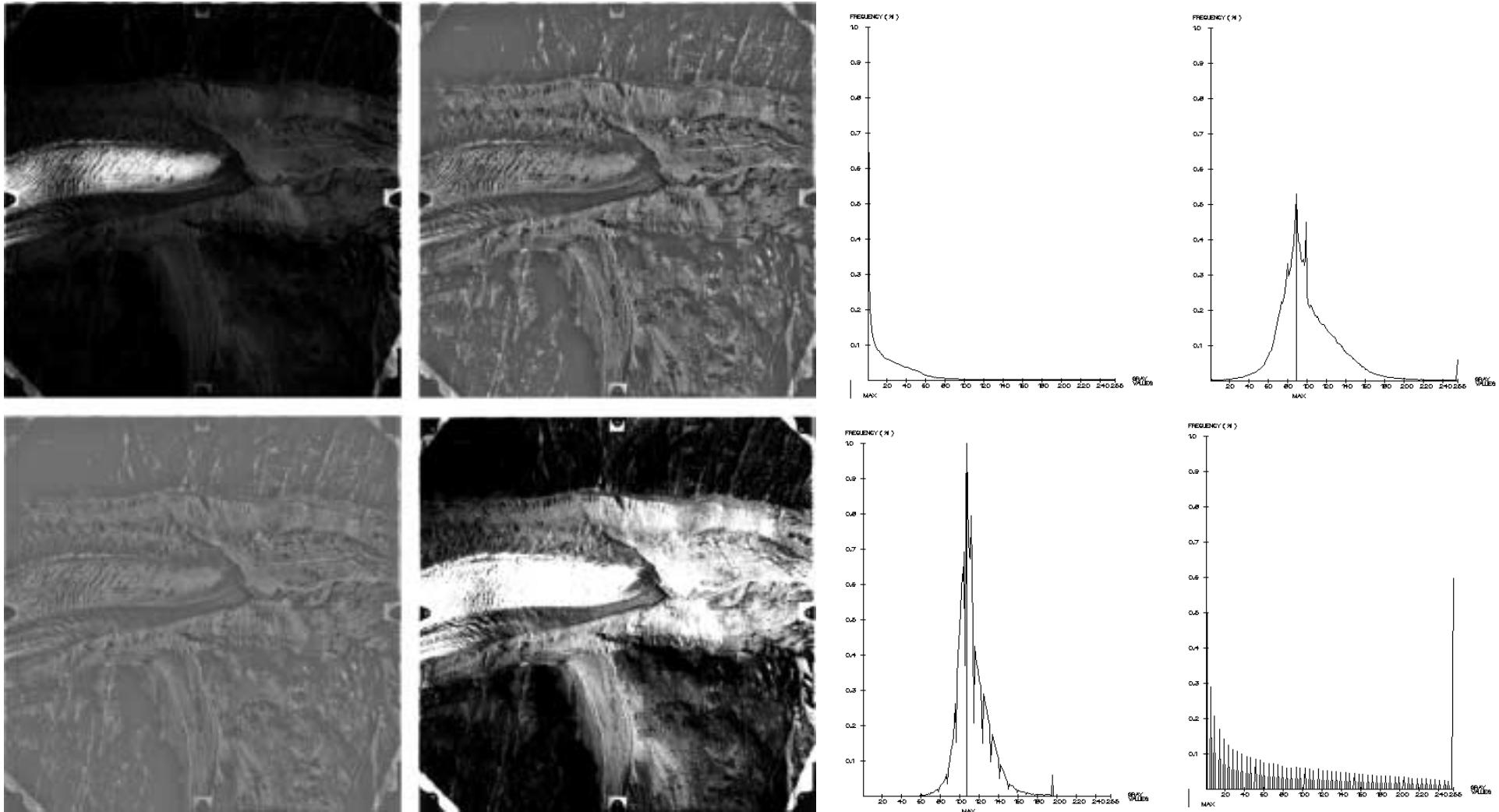
**Binarisierung**



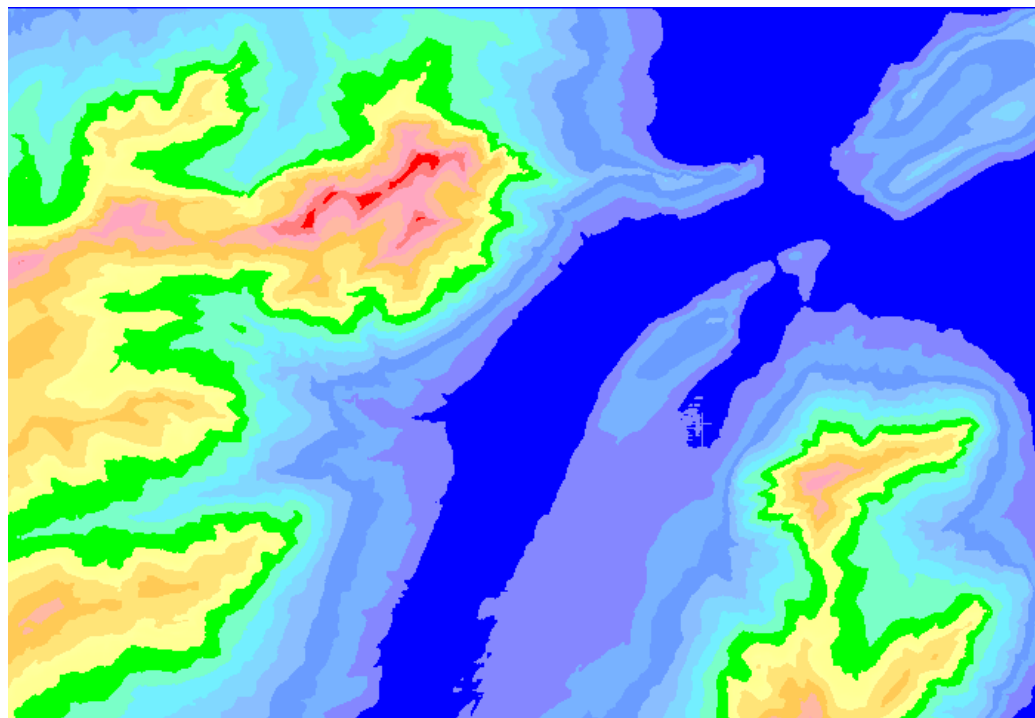
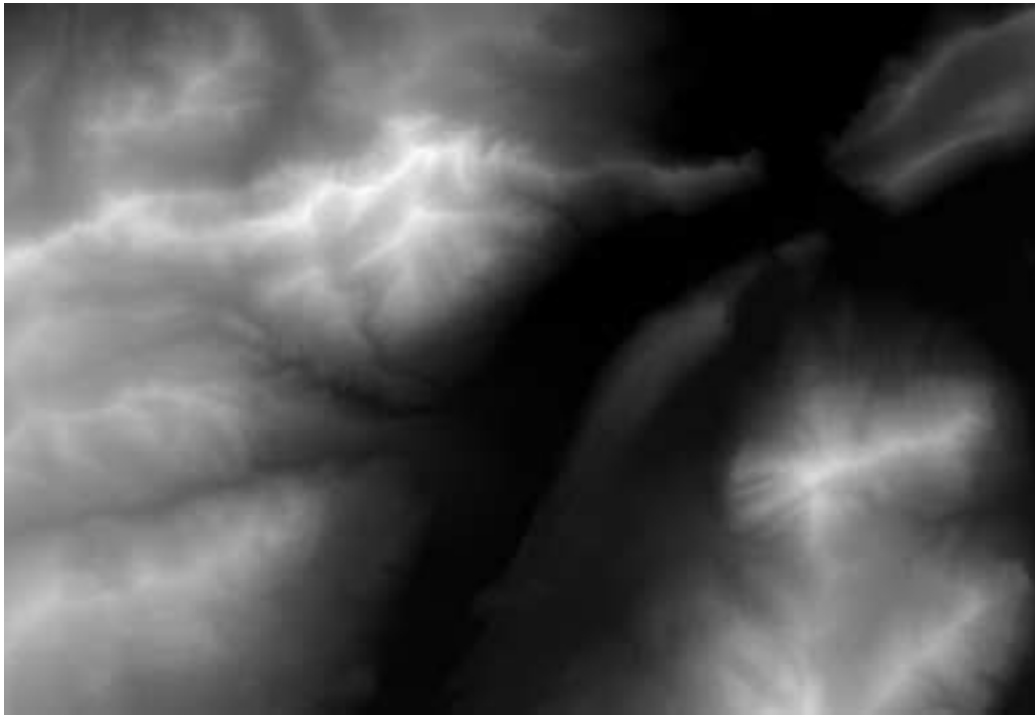
**Clipping**







**Bilder und ihre Histogramme. Von oben links zeilenweise: (a) Originalbild, (b) Kontrastverstärkung mit Wallis Filter (s. entsprechendes Kapitel), (c) wie (b) aber Histogramm komprimiert im Bereich 60 - 195, (d) Werte im Bereich 0 - 55 des Originalbildes werden im Bereich 0 - 255 transformiert (Clipping)**



**Oben: DHM25 (Kartenblatt 1170 mit Pilatus) als Grauwertbild dargestellt. Unten: Darstellung in 16 Äquidensiten kodiert in Pseudofarbe.**

## 2.6. FILTERUNG

**Definition:** Transformation der Grauwerte, um bestimmte Bildmerkmale hervorzuheben und/oder zu unterdrücken

**Ergebnis:** => neues digitales Bild

**Anwendungen:** Bildrestauration, Bildhervorhebung, Kantendetektion, Objekterkennung und Detektion, Rauschunterdrückung, Datenreduktion.

**Filterung entweder im**

- Ortsbereich (originale Bilddaten) oder
- Frequenzbereich (transformierte Bildfrequenzdaten mit einer Frequenztransformation, z.B. Fourier, Hadamard)

**Meistens Filterung im Ortsbereich (nur das wird hier behandelt)**

## **PRAKTISCHE ANWENDUNG**

- **An jedem Pixel wird eine Maske (Fenster, Kernel) zentriert. Die Grösse der Maske definiert die Nachbarschaft des Pixels, die für die Filterung benutzt wird (Filterung = Nachbarschaftsoperation (NO)). Die Koeffizienten der Maske bestimmen den Beitrag jedes Pixels innerhalb der Maske zu dem neuen modifizierten Grauwert des zentralen Pixels.**
- **Je grösser die Maske und/oder je kleiner die zentralen Koeffizienten, desto stärker die Effekte der Filterung. Iterative Filterung führt ebenfalls zu stärkeren Effekten.**
- **Die NO kann linear oder nicht linear sein.**

- Nicht-lineare NO:**
- mathematisch schwierig zu analysieren
  - algorithmisch aufwendig
  - können sehr leistungsfähig sein
  - anwendbar für spezielle Probleme (Bildverbesserung, Restauration)
  - Beispiele: Median, Wallis.

- Die lineare Filterung kann als eine Konvolution (Faltung) des Bildes mit der Maske betrachtet werden.

Mit  $\bullet$  = Konvolutionsoperator

$g(x,y)$  = Bildfunktion

$f(x,y)$  = Filterfunktion (Maske)

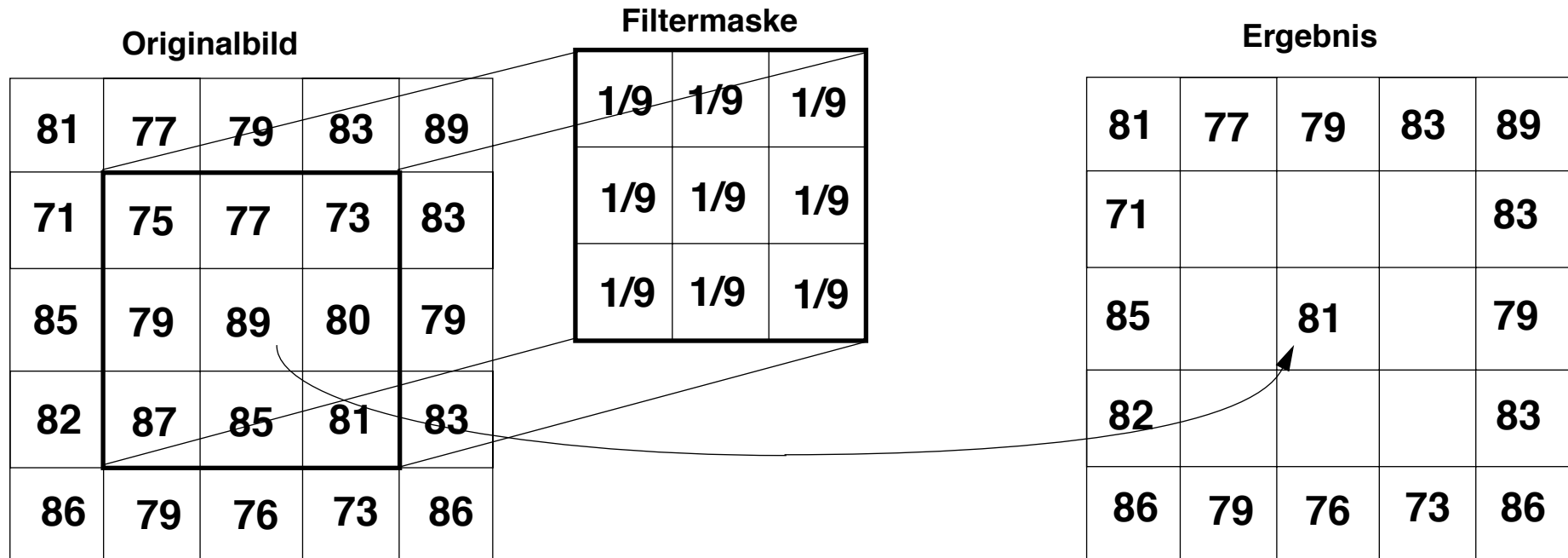
$h(x,y)$  = gefiltertes Bild

$2M+1, 2N+1$  = x , y Dimensionen der Maske

Digitale (diskrete) Bilder:

$$h(x, y) = f \bullet g = \sum_{i=-M}^M \sum_{j=-N}^N f(i, j) g(x-i, y-j)$$

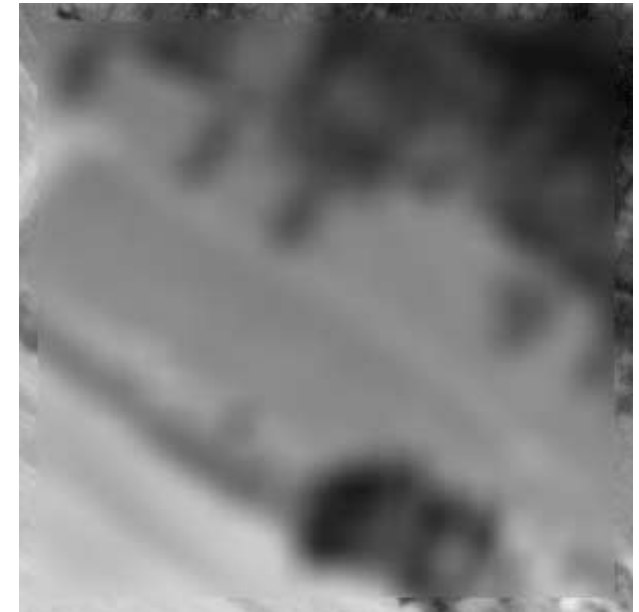
## Beispiel einer lokalen Mittelwertbildung (3 x 3 Nachbarschaft)



Ergebnis für das zentrale Pixel:  $1/9 * (75 + 77 + 73 + 79 + 89 + 80 + 87 + 85 + 81) = 80.666$

$\text{nint}(80.666) = 81$

Bemerkung: die Randpixel bleiben ungefiltert



**Je grösser die Filtermaske, desto stärker die Filterung. Von links: Originalbild, Mittelwertbildung mit 9 x 9 Maske, Mittelwertbildung mit 19 x 19 Maske. Merke die Unterschiede bei feinen Strukturen.**

- **Filterung der Bildränder. Problem: Maske teilweise ausserhalb des digitalen Bildes**
  - **Randpixel erhalten ihre alten Grauwerte (bleiben ungefiltert)**
  - **Randpixel erhalten zur Markierung einen bestimmten Grauwert**
  - **Randpixel werden gefiltert. Grauwerte ausserhalb des Bildes werden bestimmt durch: (a) Spiegelung der Grauwerte durch Randlinien, (b) Kopieren der zwei Rand-Zeilen und Kolonnen, (c) durch Extrapolation, (d) wrap-around ...Annahme einer periodischen Wiederholung der Grauwerte**
  - **Randpixel werden gefiltert nur mit dem Teil der Maske, der im Bild enthalten ist, und entsprechend modifizierte Koeffizienten**
- **Schnelle Ausführung der Filterung durch Update-Philosophie, d.h.**

**Teilergebnis aus einem vorherigen Schritt in folgenden Schritten wieder zu verwenden**



- **Filterklassifizierung**

- **Tiefpassfilter**

**Die tiefen (kleinen) Frequenzen, d.h. lange Wellenlänge (Merkmale grösser als die Maske), werden hervorgehoben, die hohen Frequenzen unterdrückt.**

**Ergebnis => weiches, weniger scharfes Bild**

**Rauschen, Kanten, kleine Merkmale geglättet**

**Charakteristiken der meisten Tiefpassfilter:**

- **Summe der Maskenkoeffizienten = 1 (Normalisierung)**
- **Positive Koeffizienten**
- **Symmetrische Koeffizienten in x,y oder beide Richtungen**

## **- Hochpassfilter**

**Das Gegenteil vom Tiefpassfilter**

**Ergebnis => Schärferes Bild**

**Kanten und Rauschen hervorgehoben**

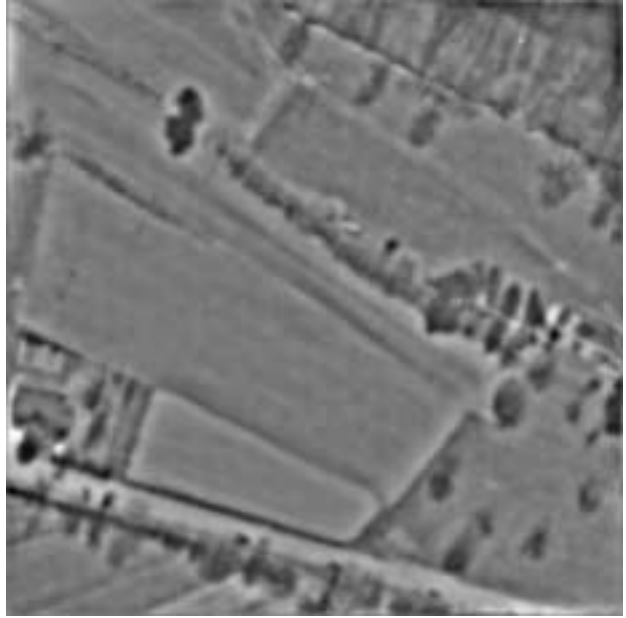
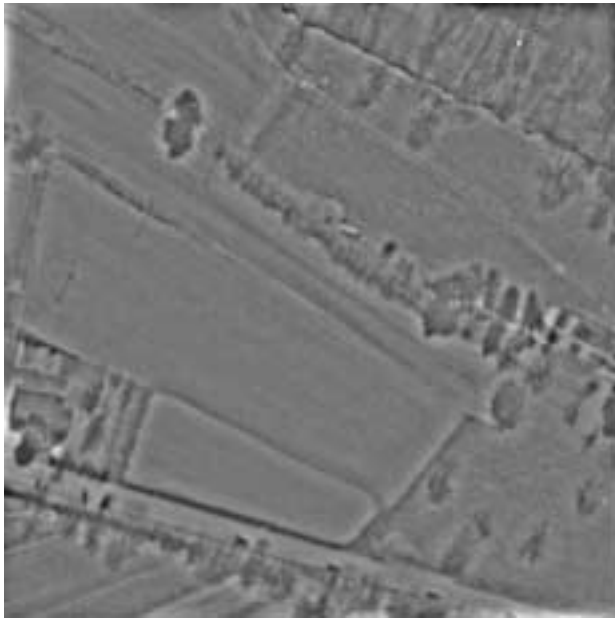
**Gefahr: Rauschhervorhebung ; deshalb Anwendung, wenn Rauschanteil gering, d.h. Rauschwerte wesentlich kleiner als die Bildwerte.**

**Charakteristiken der meisten Hochpassfilter:**

- Summe der Maskenkoeffizienten = 0**
- Positive und negative Koeffizienten (Differenzoperation)**
- Symmetrie**

## **- Bandpassfilter**

**Vorgewählte Frequenzbereiche (Bänder) werden hervorgehoben, die übrigen unterdrückt. Bei Bandstopfilter passiert das Gegenteil.**



**Zeilenweise von oben links: Originalbild, starke Tiefpassfilterung (nur grosse Objekte bleiben erhalten), Hochpassfilterung (Kanten und kleine Objekte werden hervorgehoben, grosse Objekte bekommen einen ähnlichen Grauwert), Bandpassfilterung (nur Objekte einer mittleren Grösse bleiben erhalten)**

## **- Texturfilter**

**Jedes Pixel erhält einen Wert, der abhängig von einem Texturmass (z.B. Standardabweichung, oder Bereich der Grauwerte) ist.**

- **Andere Filterklassifizierung**

- **Isotrope Filter (Effekt des Filters ist gleich in allen Richtungen)**

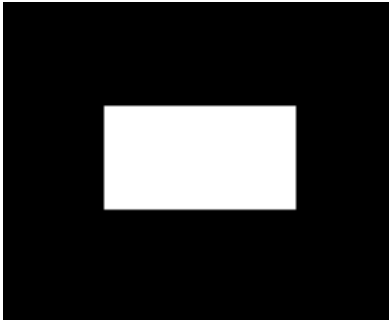
- **Richtungsabhängige Filter**

**Bei allen 2-D Filtern gewünscht:**

- **Zerlegung des Filters in zwei gleiche 1-D Masken. Vorteil: schnelle Computerimplementation mit unabhängiger Verarbeitung in x und y Richtung**
- **Normierung der Koeffizienten der Maske, d.h. in der Regel**

**Summe der Koeffizienten = 1 (Tiefpassfilter) oder 0 (Hochpassfilter)**

**Falls negative Koeffizienten, Ergebnis muss im Grauwertbereich [0,255] transformiert werden**


 $E(x,y)$ 

0	0	0
1	0	-1
0	0	0

 $K1(x,y)$ 

0	1	0
0	0	0
0	-1	0

 $K2(x,y)$ 

1	0	0
0	0	0
0	0	-1

 $K3(x,y)$ 

0	0	1
0	0	0
-1	0	0

 $K4(x,y)$ 

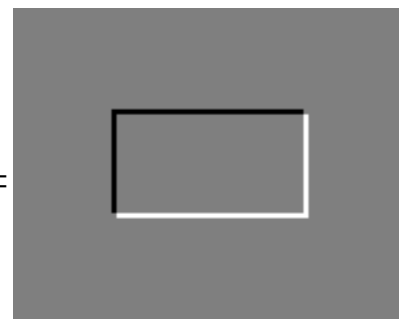
$$E * K1(x,y) =$$



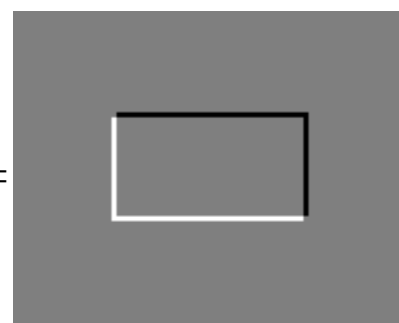
$$E * K2(x,y) =$$



$$E * K3(x,y) =$$



$$E * K4(x,y) =$$



**Beispiele von richtungsabhängigen Filtern  
(Kantendetektoren).**

## EINIGE OFT GEBRAUCHTE FILTER

### Tiefpassfilter

- Gleitender Mittelwert (oder Box-Filter) (3 x 3). Maske =  $\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
- Median (3 x 3)

Jedes Pixel wird mit dem Median seiner Nachbarschaft ersetzt. Glättung der Kanten weniger als beim Mittelwert. Ideal für "Salz und Pfeffer" Rauschen.

- Gaussisches (oder binomial Filter)

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

Je grösser  $\sigma$  und/oder je grösser die Maske, desto stärkere Filterung

## Hochpassfilter

- Subtraktion vom Hintergrund

$$h(x,y) = g(x,y) - g_H(x,y)$$

$$\text{wo } g_H(x,y) = c \quad \text{oder}$$

$$g_H(x,y) = m f + c \quad \text{oder}$$

$g_H(x) =$  Tiefpassfilterung von  $g(x,y)$  (unscharfe Maskierung)

- Differentialoperatoren (s. Kantenoperatoren)
- High-Emphasis Filter

$$h(x, y) = g(x, y) - \nabla^2 g(x, y)$$

$$\text{mit } \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} = \text{Laplace Operator}$$

**Weniger empfindlich auf Rauschen als Laplace**

- **Difference of Gaussians (DOG)**

$$DOG(\sigma_1, \sigma_2) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2 + y^2}{2\sigma_1^2}} - \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2 + y^2}{2\sigma_2^2}}$$

**DOG sieht wie Mexikaner Hut aus**

**Wenn  $\sigma_1 / \sigma_2 = 1 : 1.75$  ;  $\sigma_1 = \sigma$**

$$DOG(\sigma, 1.75\sigma) \approx \nabla^2 G(\sigma)$$

**d.h. DOG wird mit Laplace der 2-D gaussischen Funktion approximiert**

$$h(x, y) = DOG(\sigma, 1.75\sigma) \times g(x, y) \approx \nabla^2 G(\sigma) \times g(x, y) = \nabla^2 [G(\sigma) \times g(x, y)]$$

**Also, erst Tiefpassfilterung (Gaussische), dann Hochpassfilterung (Laplace)**

**Ergebnis => Hervorhebung der Kanten (Nulldurchgänge der zweiten Ableitung der Grauwerte), Kantendetektion**



- **Logarithmischer Filter**

Wenn  $g(x,y) = E(x,y) \cdot r(x,y)$

dann  $h(x,y) = \log[g(x,y)] = \log [E(x,y)] + \log[r(x,y)]$

Damit kann man die zwei Faktoren E und r separieren und durch entsprechende Filterung den einen der beiden unterdrücken. Das multiplikative Rauschen wird in additives Rauschen umgewandelt.

**Beispiel:**

$E(x,y)$  = Bestrahlungsstärke (abhängig von Beleuchtung)

$r(x,y)$  = Rückstrahlung (abhängig von Objektmerkmalen)

$E(x,y)$  variiert langsam und durch Hochpassfilterung des logarithmisch gefilterten Bildes, kann unterdrückt werden ("Shading"-Effekte eliminieren).

## Bandpassfilter

- Matched Filter (Template Matching)

Detektion eines vorgegebenen Bildes (=Template) in ein anderes Bild. Objekt wird detektiert, wenn eine Ähnlichkeitsfunktion zwischen Template und Bild maximiert wird. Beispiel:

### Kreuzkorrelation

$$R_{gf}(x, y) = \sum_{i=-M}^M \sum_{j=-N}^N f(i, j)g(x-i, y-j)$$

mit  $f(i,j)$  ...Template

### Normalisierte und zentrierte Kreuzkorrelation

$$\bar{R}_{gf} = \frac{\sum \sum (g - \bar{g})(f - \bar{f})}{\sqrt{\left[ \sum \sum (g - \bar{g})^2 \sum \sum (f - \bar{f})^2 \right]}}$$

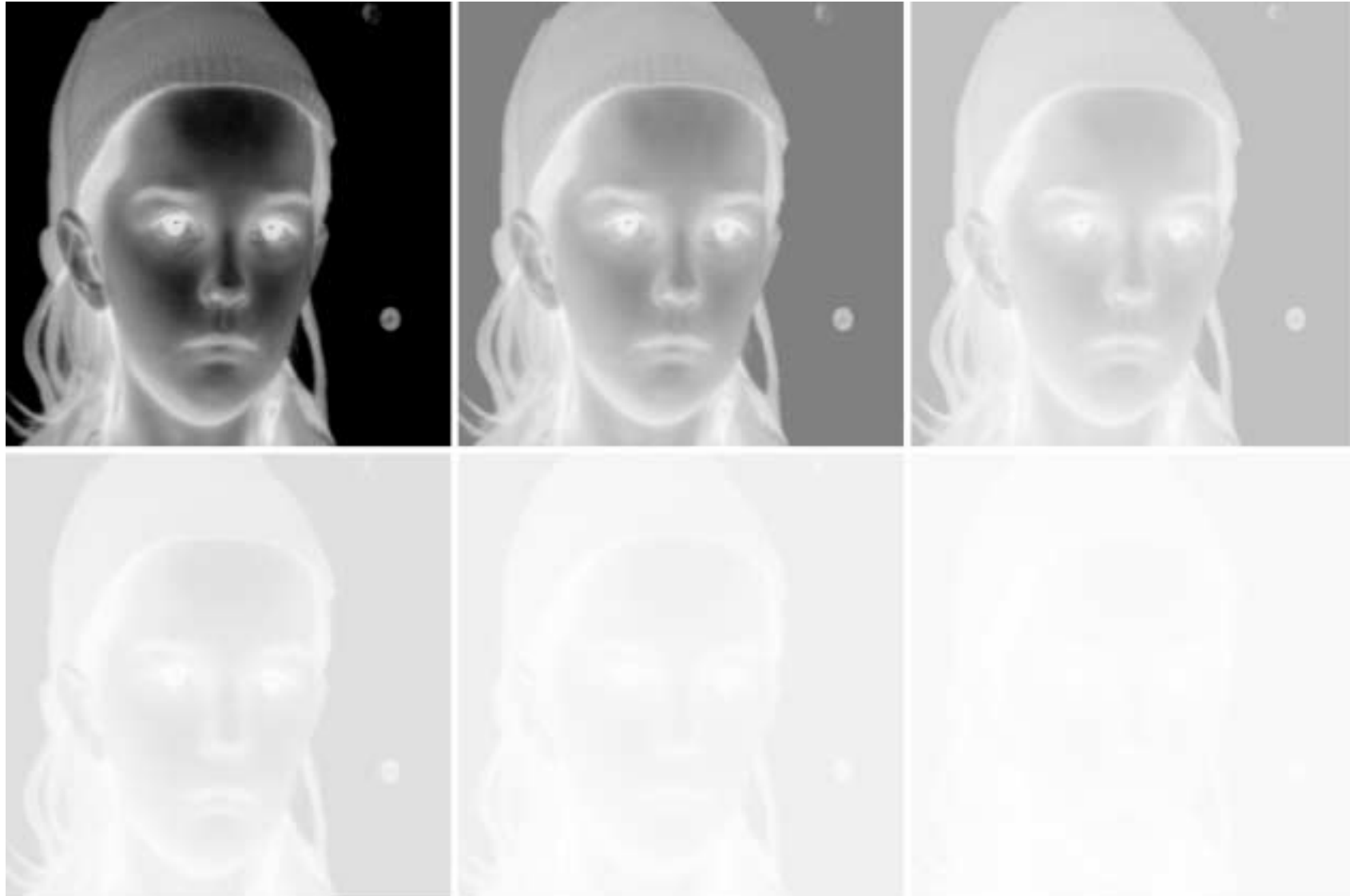
## **2.7. HIERARCHISCHE REPRÄSENTATIONEN / BILDPYRAMIDEN**

### **KOMPONENTEN EINER HIERARCHISCHEN REPRÄSENTATION (HR)**

- **Originaldaten (z.B. Grauwertbild, Modell eines Objektes)**
- **Wahl einer variablen Parameter (z.B. geometrische, radiometrische Auflösung)**
- **Regel zur Reduktion der Information (z.B. numerische, heuristische)**
- **Stufen (Ebenen) der HR**
  - **falls Originaldaten ...unterste Stufe, dann je höher die Stufe**
    - \* **desto weniger die Information (Vollständigkeit der Beschreibung)**
    - \* **desto weniger die zur Repräsentation der Information notwendigen Daten Sub-sampling)**
  - **Relationen zwischen verschiedenen Stufen definiert durch Reduktionsregel und Subsampling**
  - **Auch wichtig: die Möglichkeit das Original aus einer subsampled- Beschreibung zu rekonstruieren**



**Beispiel einer HR (Bildpyramide). Von links: Originalbild und Bilder, die jeweils bei Faktor 2 kleiner sind. Wichtige Merkmale bleiben auch in den kleinen Bildern erhalten.**



**Radiometrische Pyramide. Von oben links die Grauwerte sind im Bereich 0-255, 0-127, ..., 0-7. Pyramide wurde durch Kompression der Grauwerte erzeugt.**

## **ZIELE DER HR**

- **Sortieren der Eigenschaftswerte von Merkmalen der Beschreibung gemäss ihrer Wichtigkeit (Grösse, Länge, Kontrast)**
- **Repräsentation und Kodierung einer Beschreibung**
- **Wichtige Merkmale in die obersten HR-Stufen**  
**=> reduzierte Merkmale, reduzierte Daten => schnelle Entscheidungen**
- **Reduktion des Rechenaufwandes**
- **Merkmale auf oberste Stufen: mehr informativ, zuverlässig, robust, stabil, distinkt, geeignet um Suche in unteren Stufen zu steuern**
- **Visualisierung (Übersichtsbilder)**
- **Für Image Matching: schnellere Konvergenz, grösserer Konvergenzradius**

**HR sind eine Art Generalisierung!**

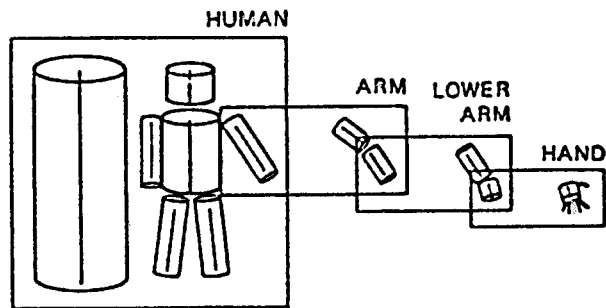
## **ANWENDUNGEN**

- **Extraktion von Merkmalen**
- **Matching**
- **Bildkodierung und Transmission**
- **Texturanalyse und Klassifikation, Mustererkennung**
- **Bewegungs-Analyse und -Detektion, Objektverfolgung**
- **Form-Analyse und -Beschreibung**
- **Segmentierung**
- **Objekt-Oberflächenanalyse**
- **Generalisierung**

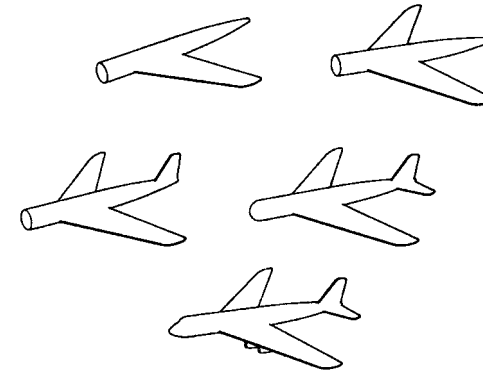
**HR haben auch Beziehung zur GIS Theorie:**

**=> “Multi-Scale” Modellierung/Generalisierung/Aggregation von Objekten**

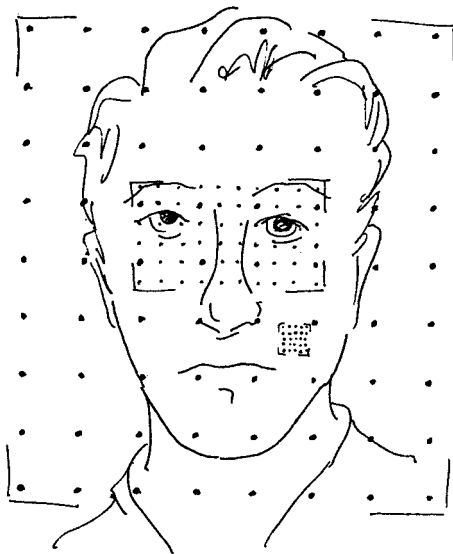
## BEISPIELE VON HR



**Hierarchische Objektrepräsentation (Witkin und Tenenbaum, 1983)**

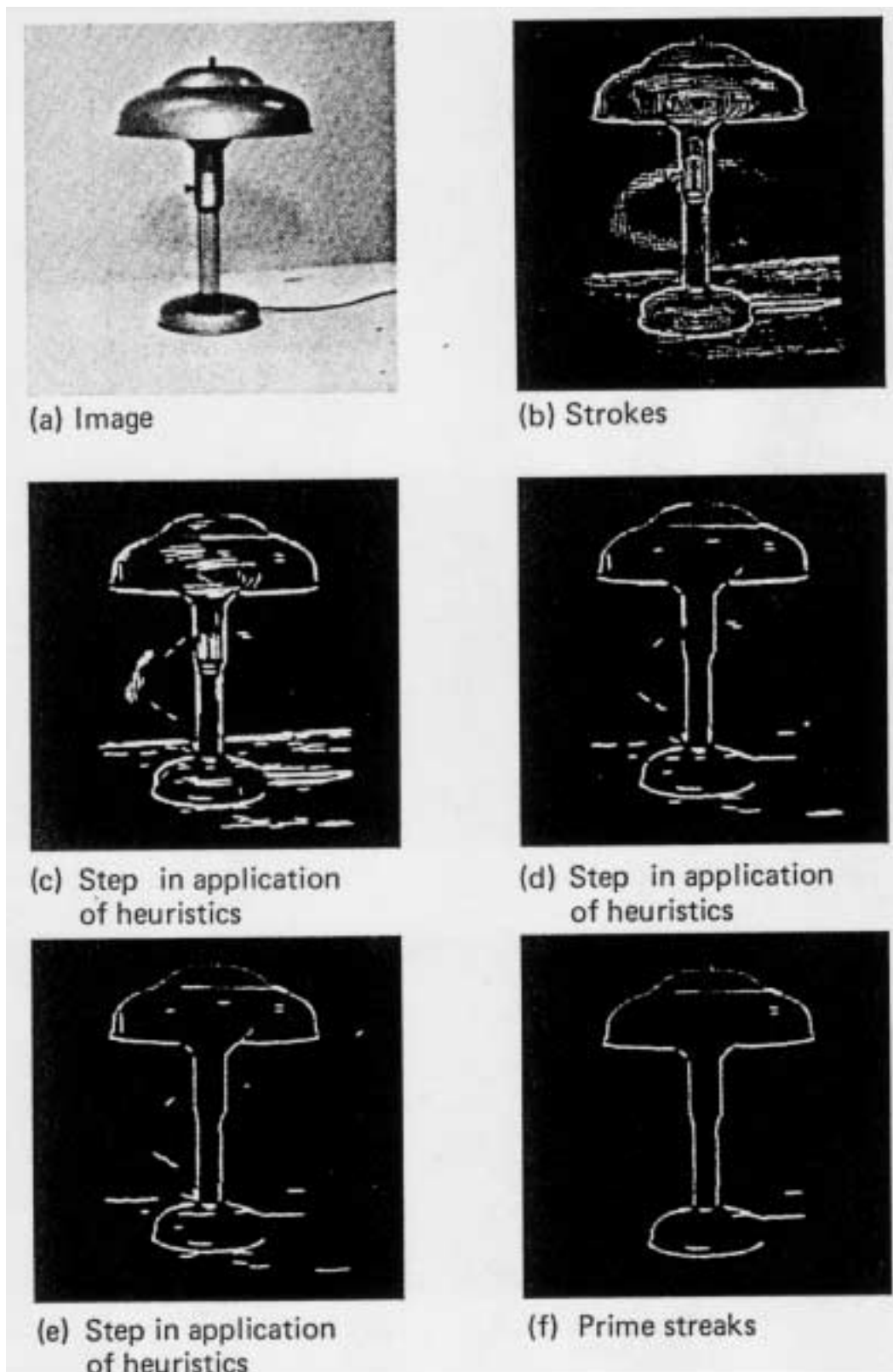


**Objekterkennung durch Komponenten. Partielle und komplette Versionen eines Flugzeuges mit 9 Komponenten (Biederman, 1986)**



**Erkennung eines Musters (Gesicht) durch eine hierarchische Matching Methode basierend auf Teilung des Targets in Submuster (Fenster) mit variabler Grösse und Auflösung (zoom-in) (Burt, 1988)**





**Hierarchische Kantenextraktion: b) Extraktion von "Strokes" (Kantenpixel), c) -e) Benutzung von heuristischen Regeln (z.B. beginne von den markantesten Kanten, verwerfe kleine Kantensegmenten (streaks)) um Kantenpixel in Kantensegmenten zu gruppieren, f) Ergebnis (Ballard und Brown, 1982)**

## **BILDPYRAMIDEN (BP)**

- **Typen von BP (je nach Filterung): tiefpass, hochpass, bandpass**
- **Generierung einer Tiefpass-BP**
  - 1. Tiefpass-Filterung mit einer  $M \times M$  Maske**
  - 2. Subsampling (Reduktion) des gefilterten Bildes mit Faktor  $S$** 
    - Schritte 1. und 2. werden kombiniert, d.h. nur jedes  $S$ te Pixel wird gefiltert
    - Wenn  $M < S$ , Information von Stufe zur Stufe systematisch verloren
    - Wenn  $M > S$ , weniger Aliasing, mehr Operationen
    - $S$  normalerweise Integer, oft 2
    - Filtermaske und Reduktionsfaktor normalerweise konstant innerhalb einer BP

- **Von einer Pyramidenstufe zur höheren**
  - **Anzahl Pixel kleiner bei  $S^2$**
  - **Pixelgrösse (footprint) grösser bei  $S$**
- **Summe der Filegrösse aller Stufen (ausser Originaldaten):  $1/(S^2-1)$  der Originaldatengrösse.**

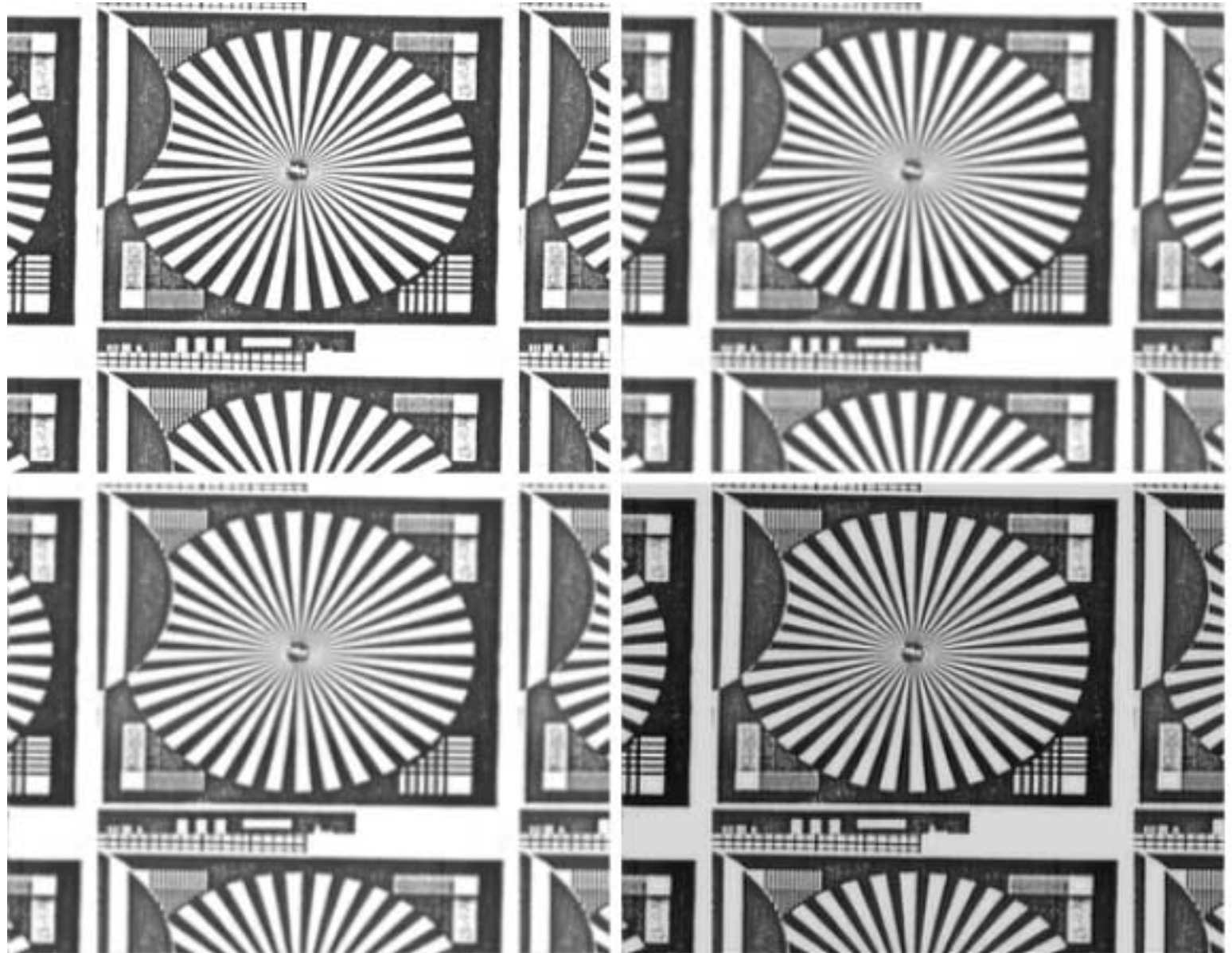
**Einige für BP oft gebrauchte Tiefpassfilter (alle symmetrisch, also nur die Hälfte der 1-D Koeffizienten, beginnend vom zentralen, werden hier gegeben):**

- **$N \times N$  lokaler Mittelwert**
- **$N \times N$  Gaussisches Filter,  $\sigma \geq 0$ , Vorschlag  $\sigma \in (0.5, 1)$**   
**je kleiner  $\sigma$ , desto grösser das Gewicht des zentralen Koeffizienten**
- **11 x 11 Approximation des idealen Filters (sinc-Funktion): 0.5, 0.288, 0, -0.096, 0, 0.058**

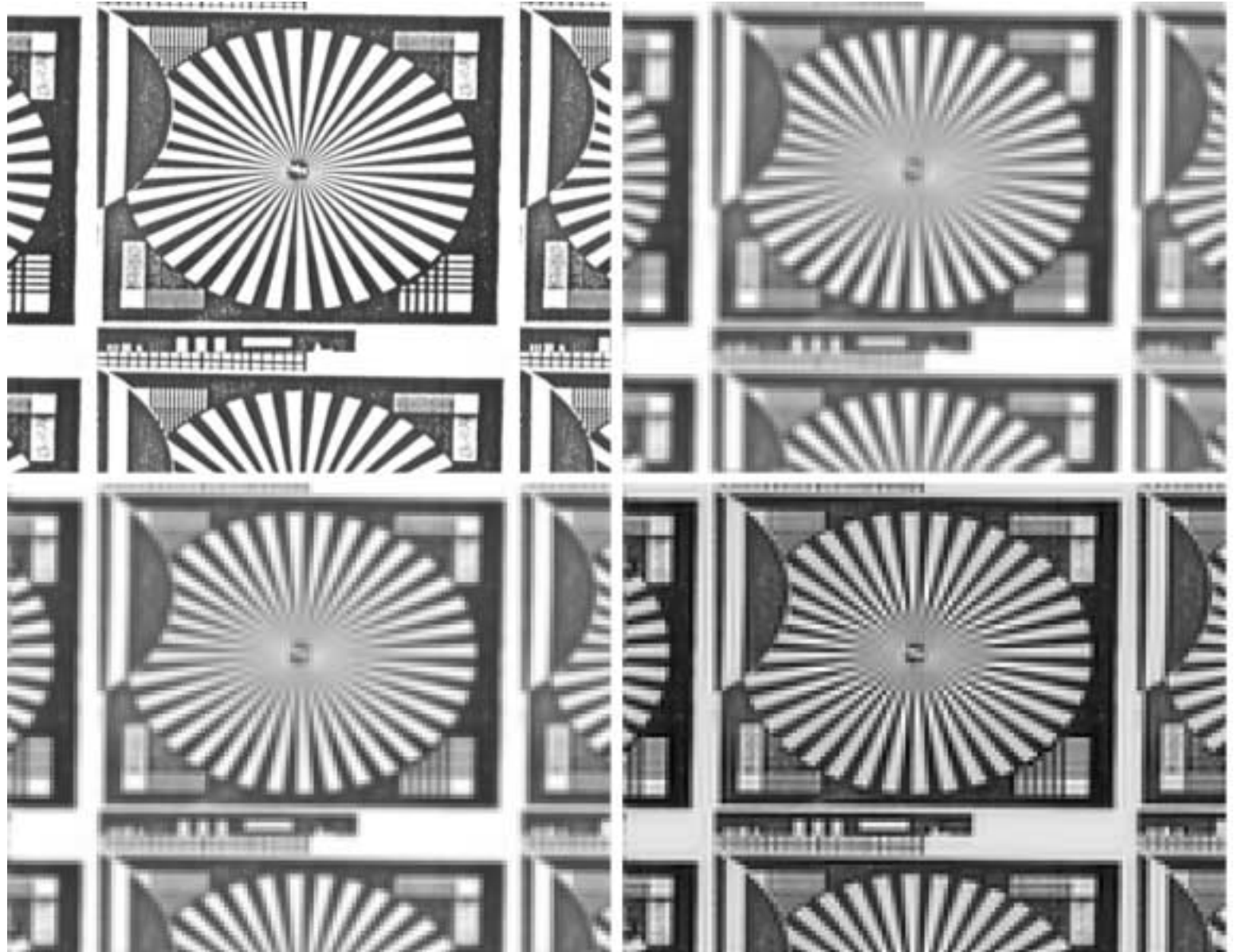
## BEISPIELE VON BP

- **Original Bild 512 x 512 Pixel (oben links). Alle Bilder sind auf die Originalbildgrösse gezoomt. Reduktionsfaktor 2.**
- **Drei Filter werden verglichen: 4 x 4 lokaler Mittelwert (oben rechts), 7 x 7 Gaussisches Filter mit  $\sigma = 1$  (unten links), 11 x 11 Approximation des idealen Filters (unten rechts)**
- **Anzahl Operationen/Pixel sind 16, 49 und 49**

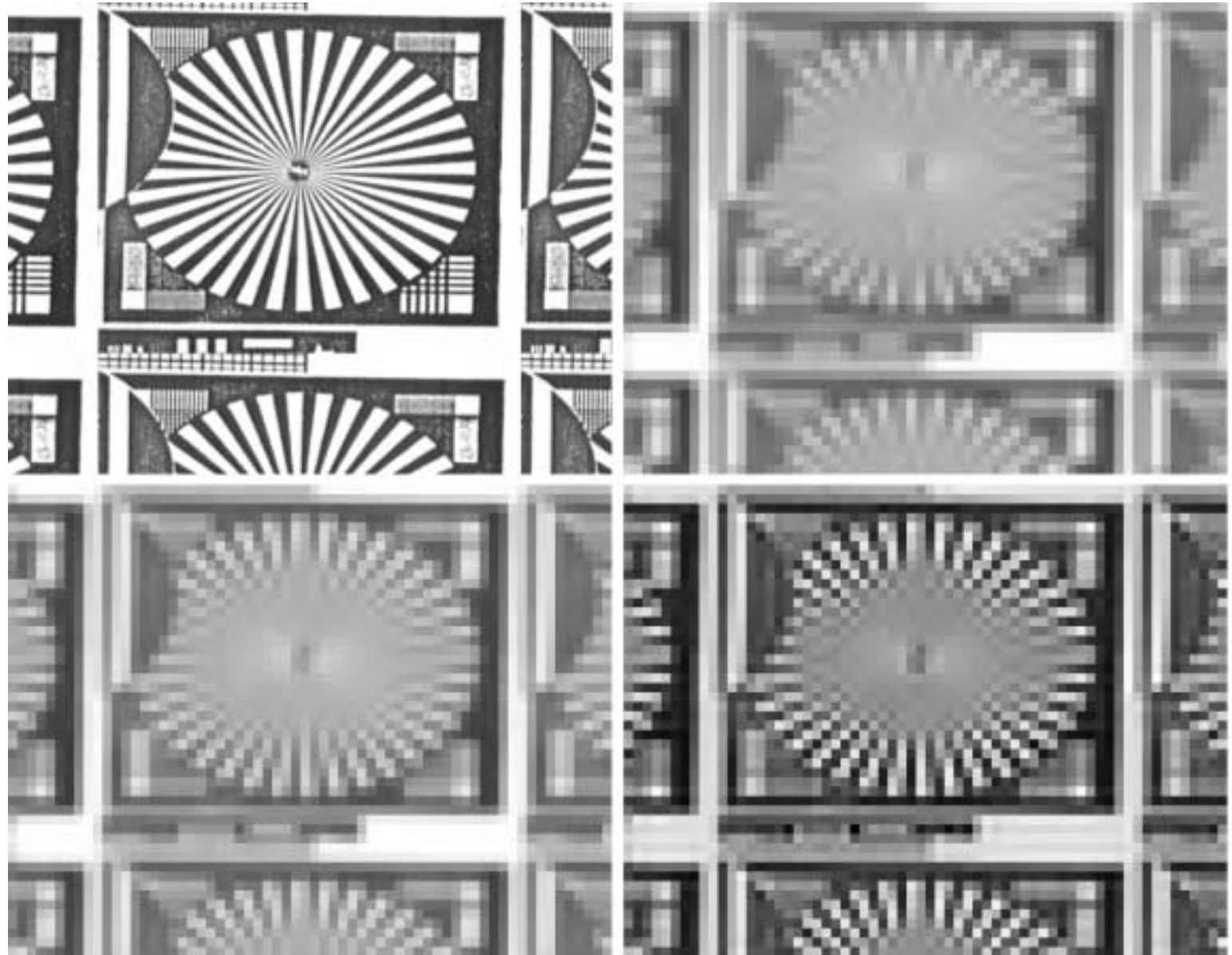
**1te Stufe**  
**(256 x 256 Pixel)**



**2te Stufe**  
**(128 x 128 Pixel)**



**3te Stufe**  
**(64 x 64 Pixel)**





**Beispiel einer Bildpyramide. Von oben links: Originalbild und tiefpass-gefilterte Bilder, mit Reduktionsfaktor 2 - 6.**





**Oben: Stufen 1- 4 einer Tiefpass-Bildpyramide (vergrössert auf  $240^2$  Pixel).**

**Unten: 4 Stufen einer Bandpass-Bildpyramide (Laplace einer Tiefpass-Pyramide). Stufe 0 (links) verkleinert auf  $240^2$  Pixel, Stufen 2 - 3 vergrössert auf  $240^2$  Pixel.**

## **2.8. BILD-RESTAURATION UND -HERVORHEBUNG**

**Differenzen zwischen Restauration und Hervorhebung sind vage. Verschiedene Bildverarbeitungsalgorithmen können für beides benutzt werden. Generell;**

- **Restauration**

**Ziel ist die Entfernung oder Reduktion von Degradationen, die während der Bildakquisition erfolgt sind => schätze das ideale (“wahre”) Bild aus dem beobachteten**

**Degradationen verursacht durch verschiedene Faktoren:**

- Bewegungsunschärfe**
- Defokussierung**
- Linsenverzeichnung**
- Punktübertragungsfunktion des abbildenden Systems**
- Rauschen (Photonen, elektronisch, “line jitter”, Staub) etc.**

- **Hervorhebung**

**Ziel ist die subjektive Verbesserung eines Bildes => der Benutzer entscheidet was zu verbessern ist, z.B.**

- **Reduktion einer spezifischen Art von Rauschen**
- **Grauwertkanten-Verstärkung**
- **Kontrastverstärkung**

## **BILDRESTAURATION**

### **Klassifikation von Degradationen**

- **Physikalisch: geometrisch, radiometrisch**
- **Statistisch: systematisch, zufällig**

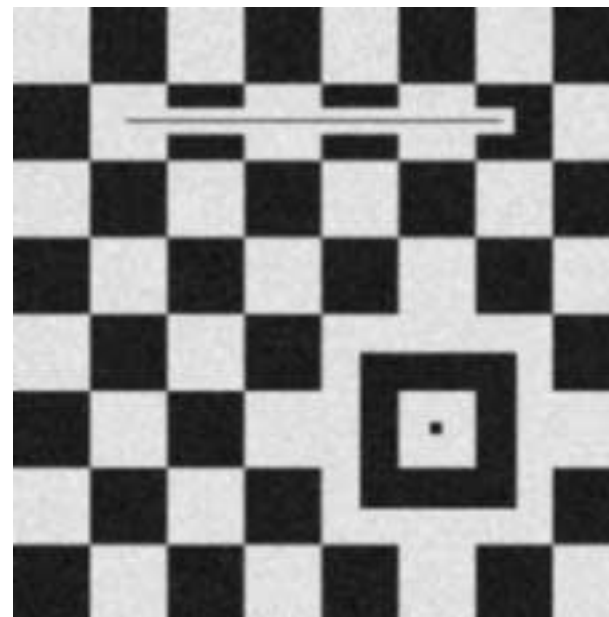
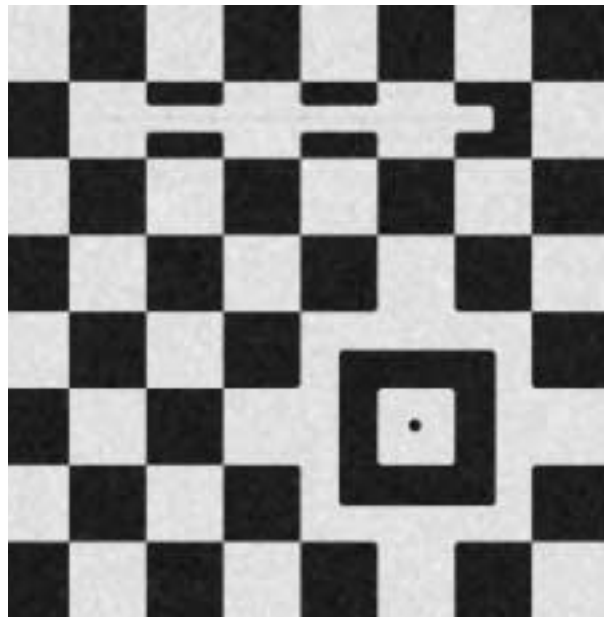
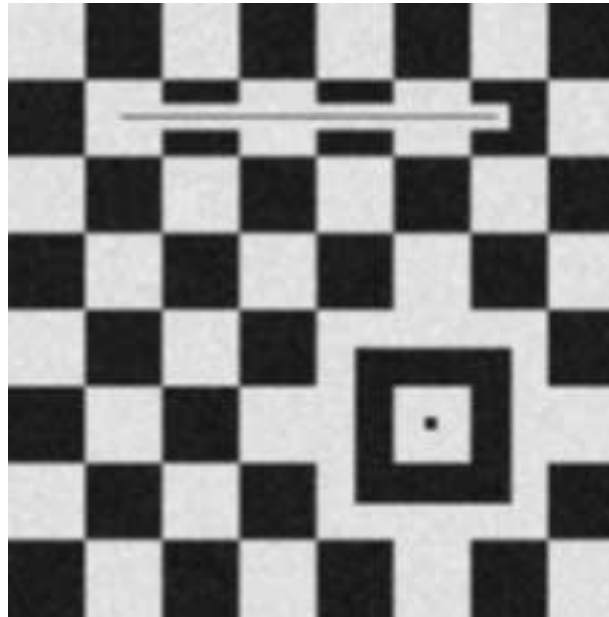
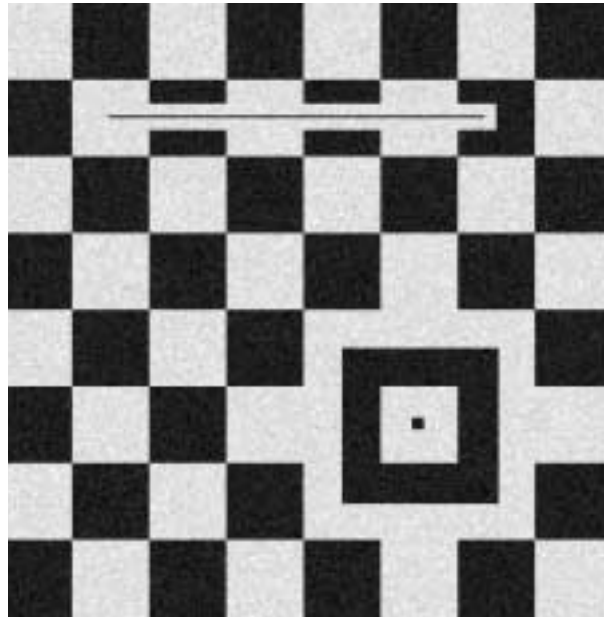
- **Normalerweise a priori Information existiert**  
**Wir restaurieren nur wenn wir entscheiden, dass das Bild degradiert ist => wir haben eine Vorstellung wie das ideale Bild aussehen sollte**
- **Bestimmte Arten von Degradationen korrigiert basierend auf spezifische Modelle, z.B. Linsenverzeichnung**
- **Also in der Bildanalyse Restauration benutzt Bild- und Rauschen-Modelle und zielt der**
  - **Reduktion des Rauschens (Glättung) innerhalb homogenen Regionen**
  - **Erhaltung (oder Hervorhebung) der Information (besonders Kanten und Ecken)****Hauptproblem der Restauration: ein Gleichgewicht zwischen diesen zwei Zielen zu finden**
- **Quantitative Information über Bild und Rauschen existiert generell nicht. Sie wird geschätzt**
  - **basierend auf gewisse Annahmen, z.B. Gaussisches Rauschen mit Mittelwert 0**
  - **aus dem beobachteten Bild, z.B. Rauschenvarianz geschätzt aus der Standardabweichung der Grauwerte in homogenen Regionen**

## **EINIGE ALGORITHMEN**

- **Lokaler Mittelwert (Box-Filter)**
- **Gaussisches (binomial) Filter**
- **Median-Filterung**
- **Kantenerhaltende Glättung (Nagao und Matsuyama, 1979)**
- **Sigma-Filter (Lee, 1983)**
- **Glättung gewichtet durch die inversen Gradienten (Wang und Vagnucci, 1981)**
- **Anisotropische Diffusion (Perona und Malik, 1990, Nordstrom, 1990)**
- **Informationerhaltendes Filter (Förstner, 1991)**
- **K-nächster Nachbar (Davis und Rosenfeld, 1978)**
- **Adaptive Rauschenreduktion (Schowengerdt, 1983)**
- **Spezifische Algorithmen für spezifische Probleme, z.B. defekte Linien, besonders mit Satellitendaten**

**Es gibt keine Methode, die für Bildrestauration generell optimal ist. Basierend auf die Anforderungen kann eine geeignete Methode gewählt werden.**

**Generell: Algorithmen, die lokal, adaptiv, ortsvariant und nichtlinear sind, haben eine bessere Leistung.**



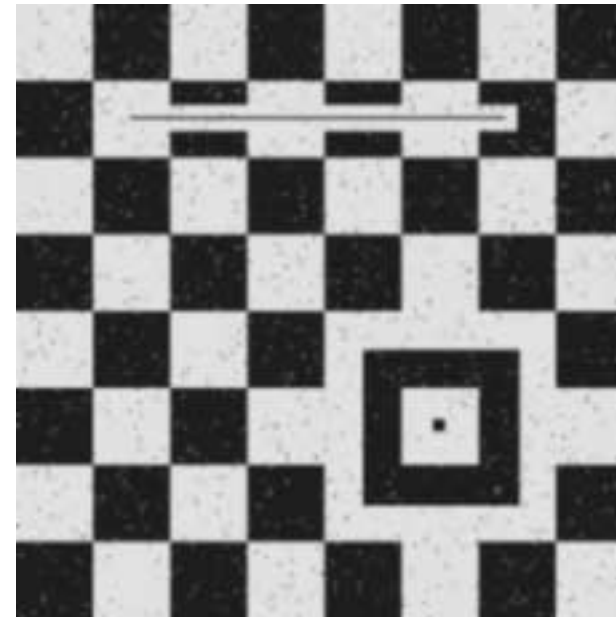
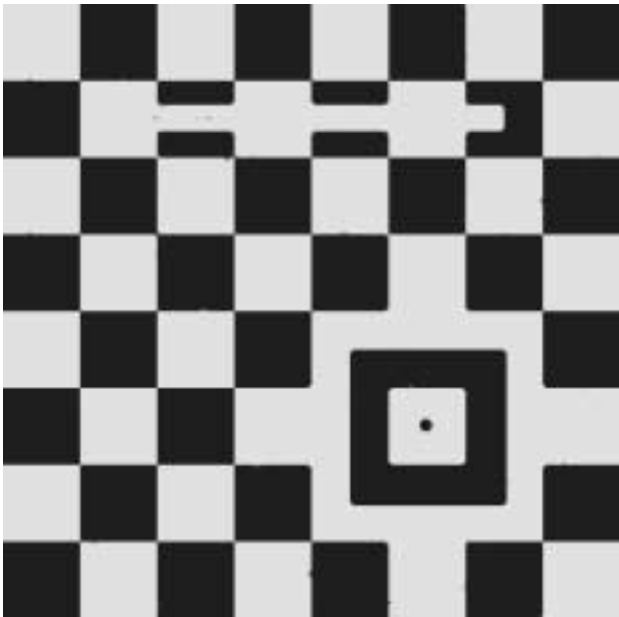
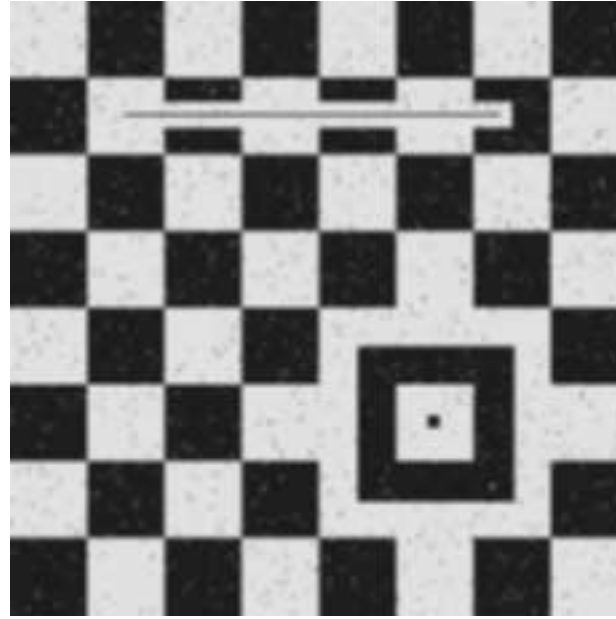
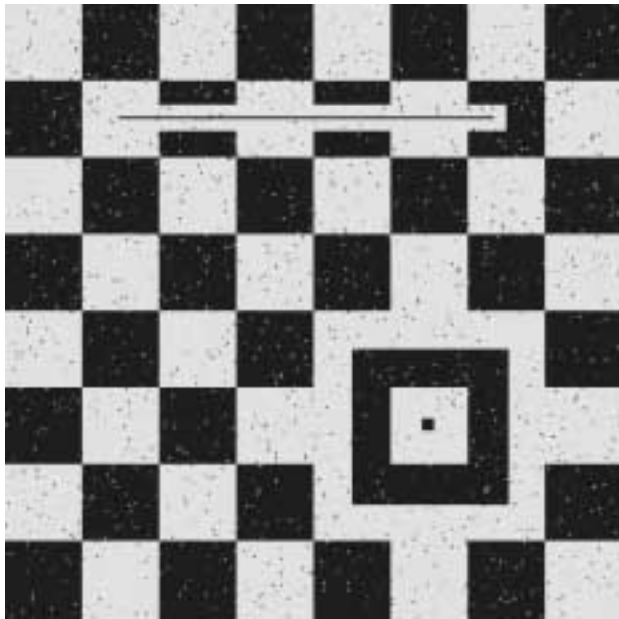
## Beispiele von einfachen Algorithmen (Rauschen mit zufälliger Verteilung)

Von oben links: Originalbild, 3 x 3 lokaler Mittelwert, 3 x 3 Median, 3 x 3 Gaussisches Filter.

Merke: je stärker die Glättung des Rauschens, desto stärker die Glättung der Kanten und der Kontrastverlust von schmalen Objekten (s. horizontale Linie).

Beim Median werden die Ecken abgerundet, und Punkten und Linien mit Dicke 1 Pixel verschwinden.

Generell hat das gaussische Filter die beste, Median die schlechteste Leistung.



## Beispiele von einfachen Algorithmen (Salz und Pfeffer Rauschen)

Von oben links: Originalbild, 3 x 3 lokaler Mittelwert, 3 x 3 Median, 3 x 3 Gaussisches Filter.

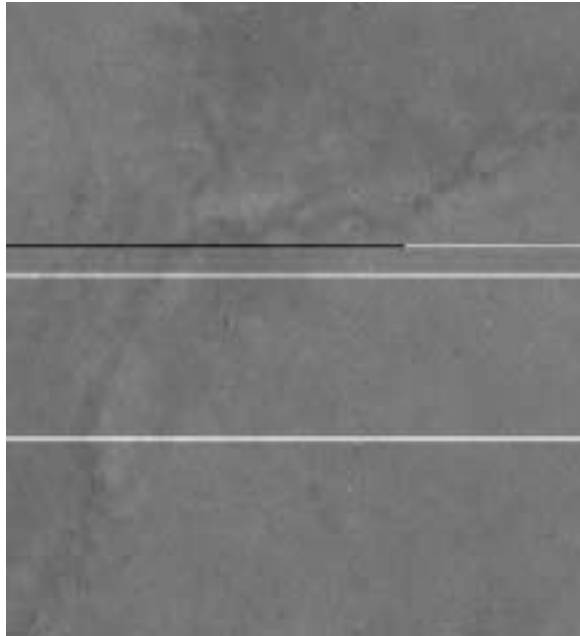
Merke: mit der Glättung wird das Salz und Pfeffer Rauschen breiter, es verliert an Kontrast aber verschwindet nicht.

Median ist ideal für solches Rauschen, hat aber die in der vorherigen Figur erwähnten Nachteile.

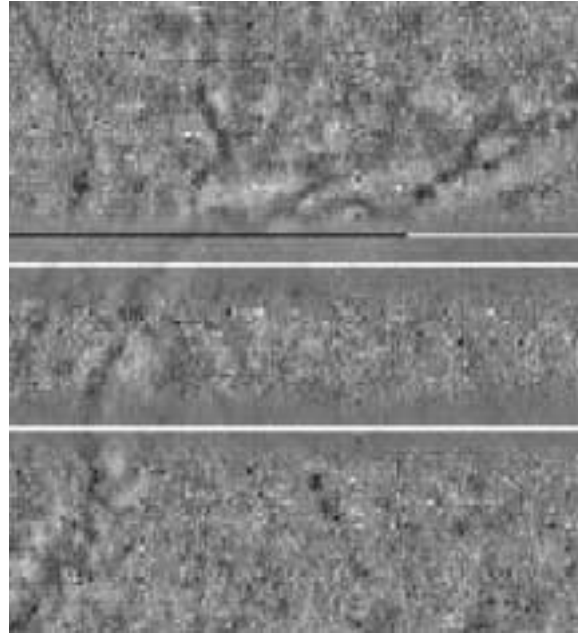


## BEISPIEL: spezifische Bildrestauration (MOMS-02/D2 Satellitendaten)

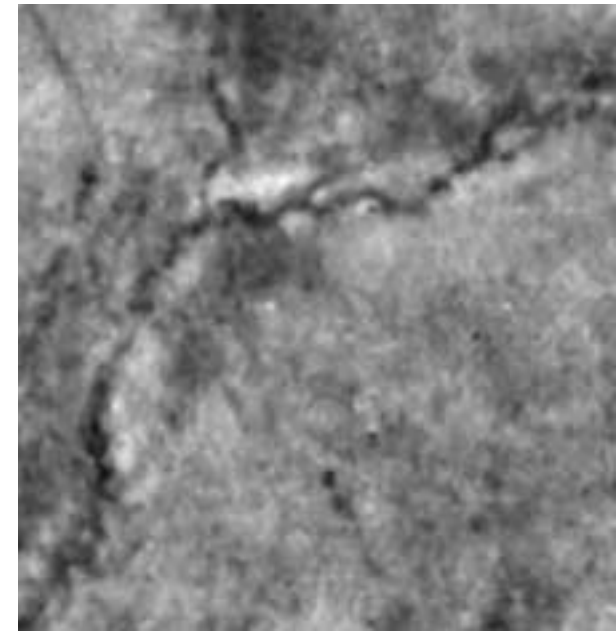
- Sensor mit 3 Kanälen: Vor- und Rückblick (13.5m Auflösung), Nadir (4.5m Auflösung)
- Grauwertbereich: nur 50 Grauwerte
- Positives und negatives “Spike”-Rauschen (Salz und Pfeffer), “Pattern”-Rauschen
- Defekte Linien im Nadir-Kanal
- Unterschiedliche Helligkeit in der linken und rechten Hälfte des Nadirkanals



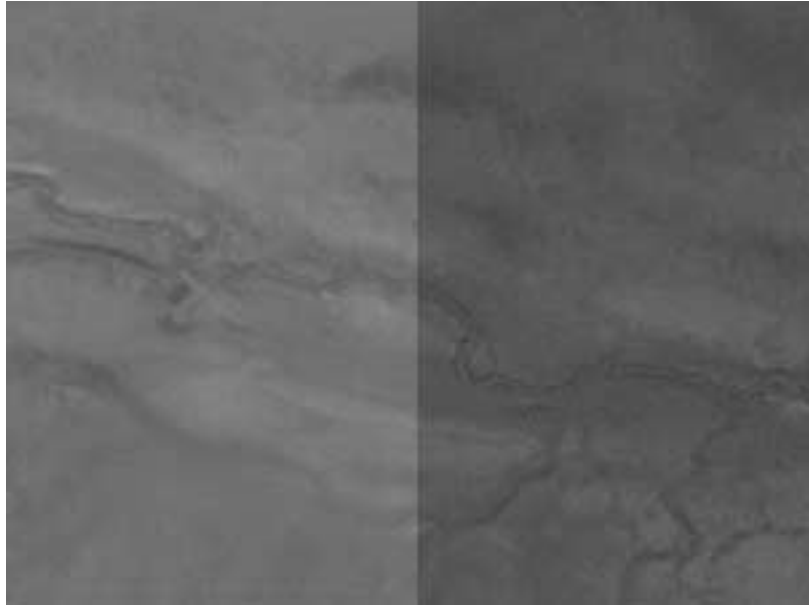
Original (Nadir)



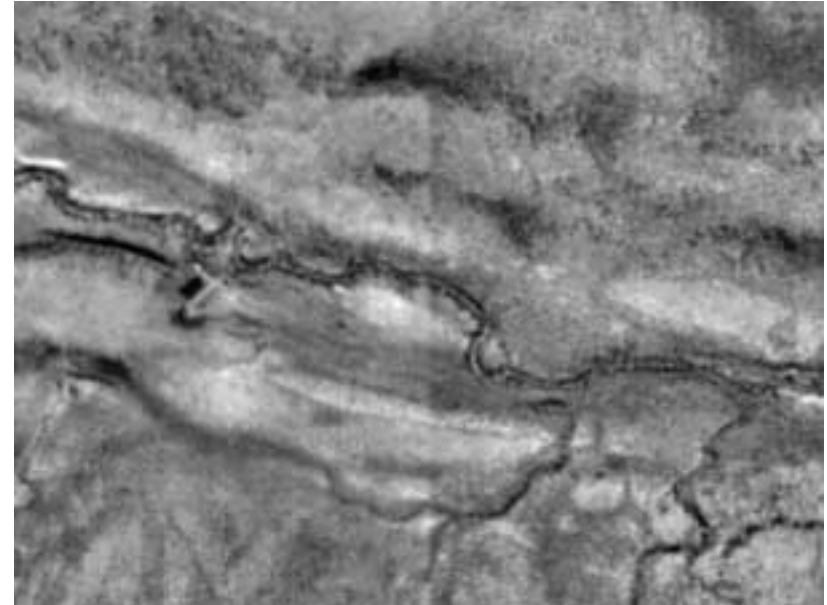
Nach Kontrastverstärkung (um Rauschen sichtbar zu machen)



Nach Verarbeitung



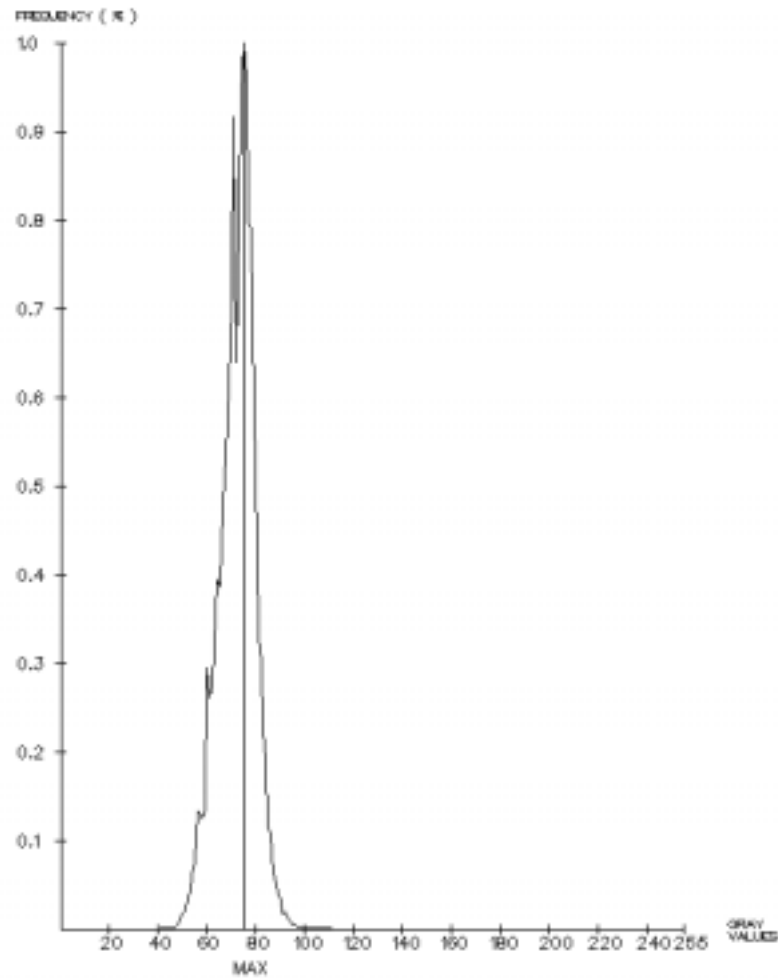
**Original (Nadir)**



**Nach Verarbeitung**

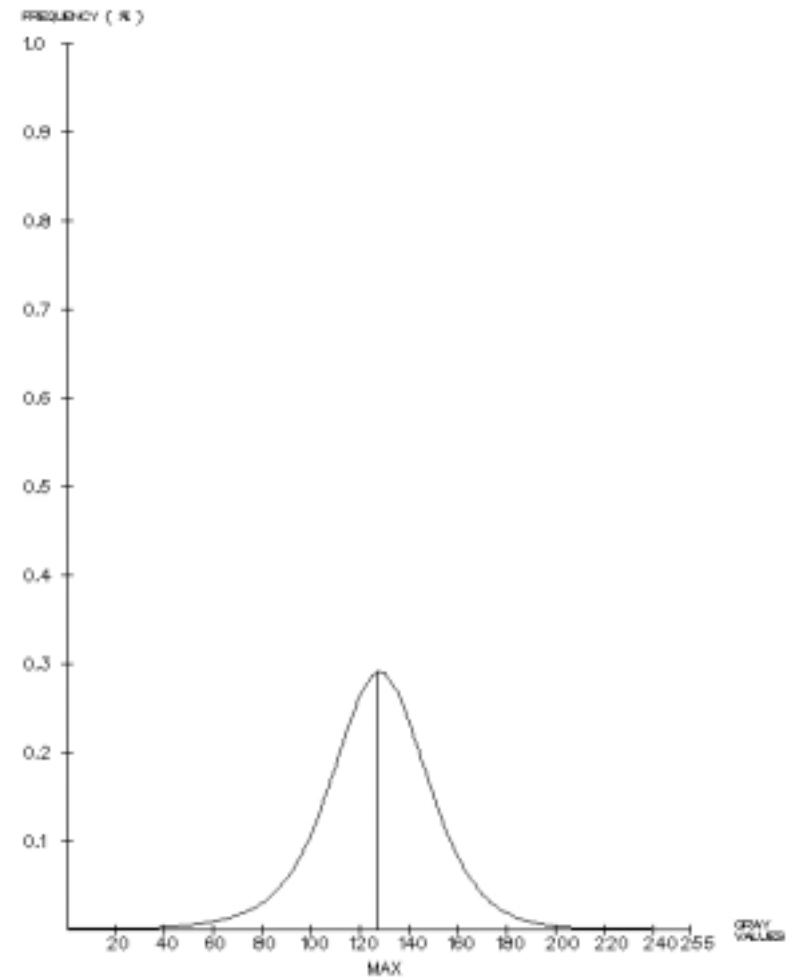
## Histogramme des Vorblick - Kanals

CHANNEL\_6(FORE)  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =24168096  
SCALE FACTOR =6.017



Originalbild

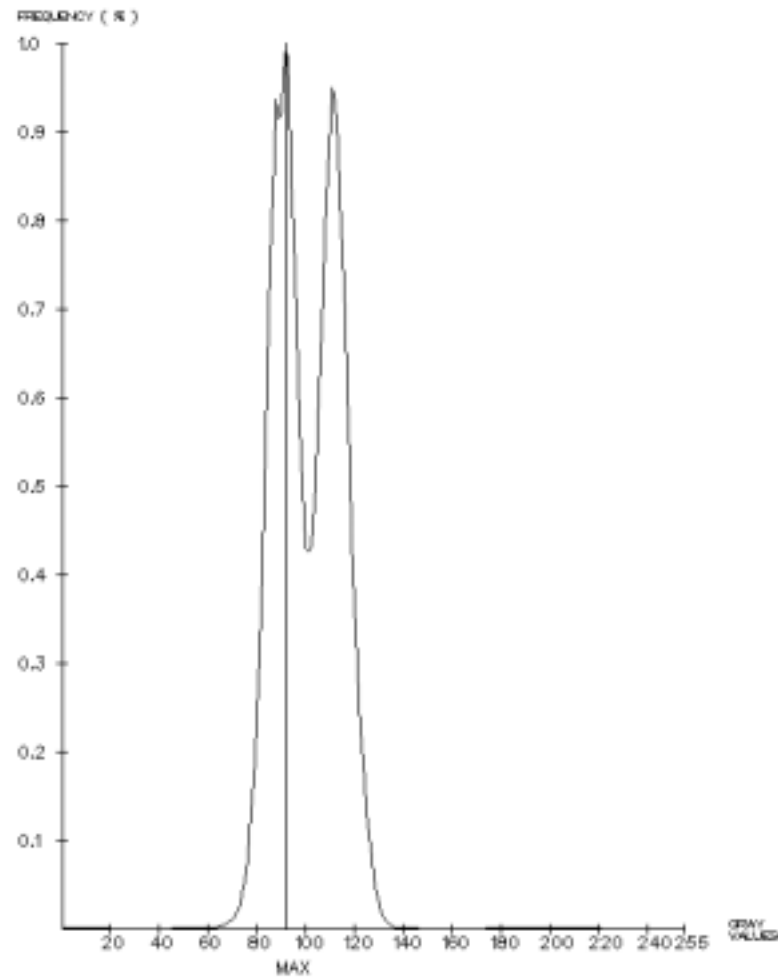
CHANNEL\_6\_PREPROCESSED  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =24168096  
SCALE FACTOR =5.017



Nach Verarbeitung

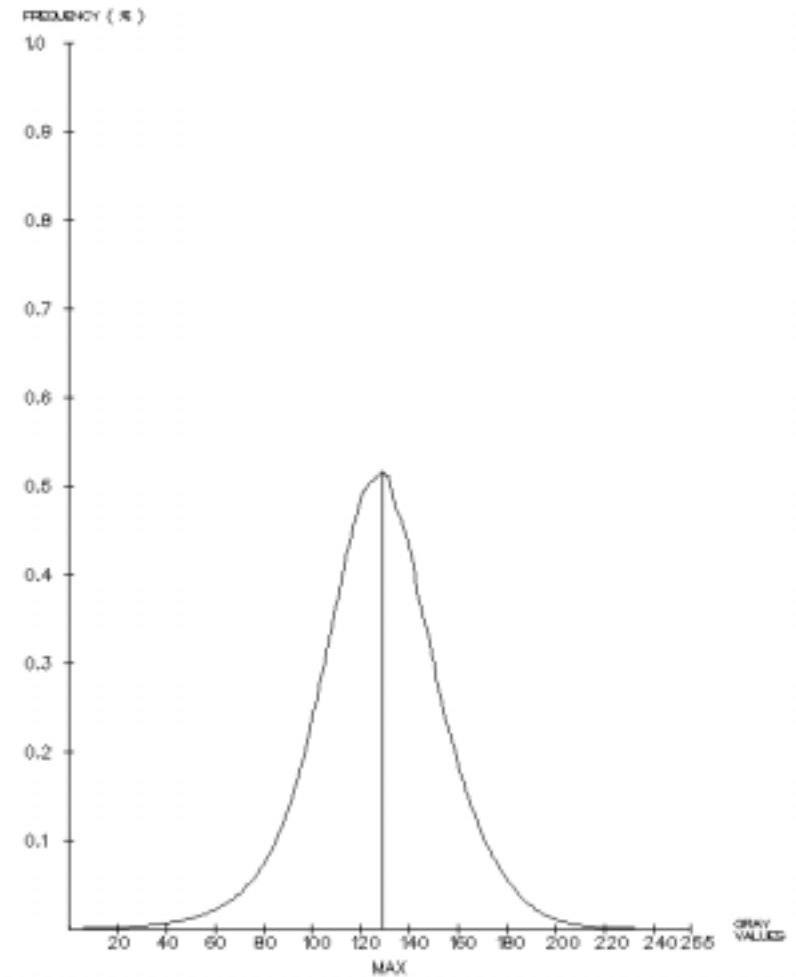
## Nadir - Kanal

MOMS\_NADIR\_5A\_ORIGINAL  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =67436784  
SCALE FACTOR =29.941



Originalbild

MOMS\_NADIR\_5A\_PREPROCESSED  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =67436784  
SCALE FACTOR =29.941



Nach Verarbeitung

# SIGMA-FILTER

- **Bildmodel:** Bild besteht aus homogenen Regionen
- **Adaptives Filter.** Ersetz das zentrale Pixel mit Mittelwert der Nachbarschaft **ABER** zur Berechnung des Mittelwerts werden nur Pixel mit kleiner Differenz zum zentralen Pixel benutzt, d.h. Pixel, die zu derselben Grauwertregion gehören
- **Prozedur:**
  - Berechne in  $N \times N$  Nachbarschaft die Grauwertstandardabweichung  $\sigma$
  - Berechne die absolute Differenz  $D_i$  jedes Pixels vom zentralen
  - Finde  $m = \text{Anzahl } D_i < 2 \sigma$
  - Wenn  $m < \text{Schwellenwert } T \Rightarrow \text{zentrales Pixel} = \text{Mittelwert aller Nachbar}$ , sonst  
zentrales Pixel = Mittelwert der  $(m + \text{zentral})$  Pixel
  - $0 \leq T \leq N^2 - 1$ . Je grösser  $T$ , desto grösser die Fläche des Rauschens, das eliminiert wird, z.B.  $T = 1, 2$  entfernt 1 und 2 Pixel grosses “Salz und Pfeffer”- Rauschen.  
**Vorschlag:**  $T \approx N$

# GLÄTTUNG DURCH GEWICHTUNG MIT DEN INVERSEN GRADIENTEN

- **Bildmodel:** Bild besteht aus homogenen Regionen
- **Adaptive Berechnung des lokalen Mittelwerts.** Benutzte zur Berechnung alle Pixel, aber gib kleines Gewicht zu den Pixeln, die nicht zur gleichen Region wie das zentrale Pixel gehören, d.h. gewichtete Mittelwertbildung
- **Prozedur:**
  - Berechne die absolute Differenz (Gradient)  $D_i$  jedes Pixels vom zentralen
  - Jedes Pixel hat Gewicht  $w_i = 1 / (D_i + e)$ ,  $e$  ...kleine positive Zahl
  - Normalisiere die Gewichte, so dass ihre Summe 0.5 ist. Das Gewicht des zentralen Pixels ist auch 0.5
  - Anwendung des Filters iterativ (normalerweise 2 - 5 Iterationen)



(A)



(B)



(C)



(D)

**Beispiele von Restauration: (A) Originalbild, (B) anisotropische Diffusion (lokales, adaptives, iteratives Verfahren), (C) 5 x 5 lokaler Mittelwert, (D) Gaussfilter ( $\sigma = 1.5$ )**

# **DIGITALE BILDHERVORHEBUNG**

## **Rauschenreduktion**

- **Lokaler Mittelwert** (ein schneller Algorithmus ist in McDonnell, 1981 gegeben)
- **Median** (für schnelle Algorithmen s. Chaudhuri, 1983, 1990, Huang, 1979)
- **Gaussisches Filter**
- **Andere Filter** wurden im Kapitel über Bildpyramiden und Restauration gegeben

## **Histogrammequalisierung**

- **Ziel ist die Kontrastverstärkung.** Versucht die Grauwerte so zu verteilen, so dass alle Grauwerte dieselbe Häufigkeit haben (flaches Histogramm). Mit diskreten Bildern dies kann nur approximiert werden. Die Transformation der Grauwerte ist von folgender Formel gegeben:



$$g_t = \max \left( 0, \text{rint} \left[ \frac{2^q \times CH(g)}{n^2} - 1 \right] \right)$$

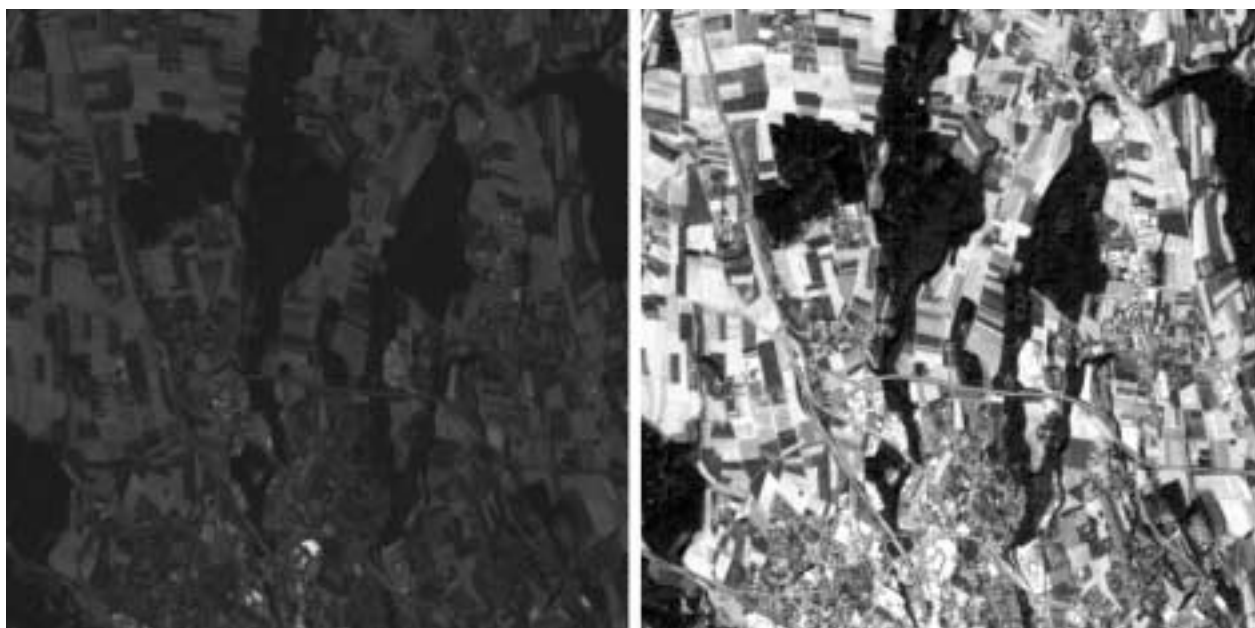
**$g, g_t$       ...alter, neuer Grauwert**

**$CH(g)$       ...Wert des kumulativen Histogramms am Grauwert  $g$**

**$q$               ...Anzahl der Quantisierungsbits**

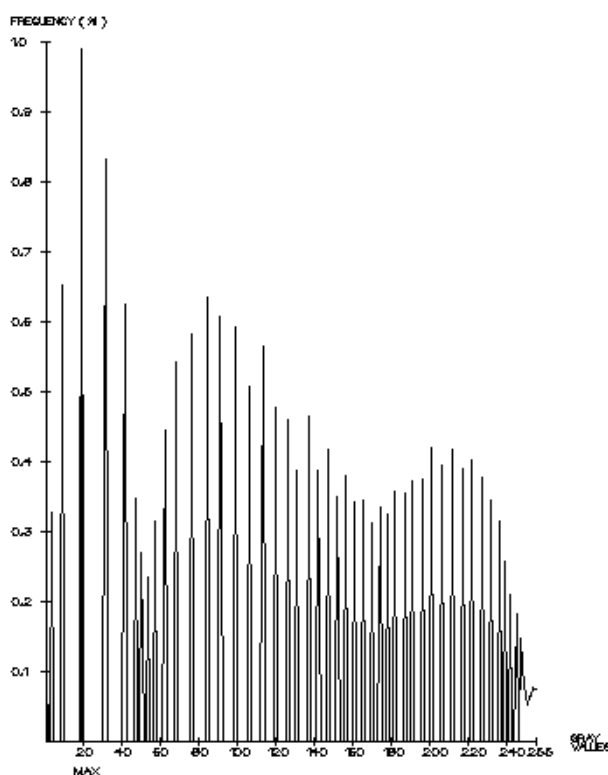
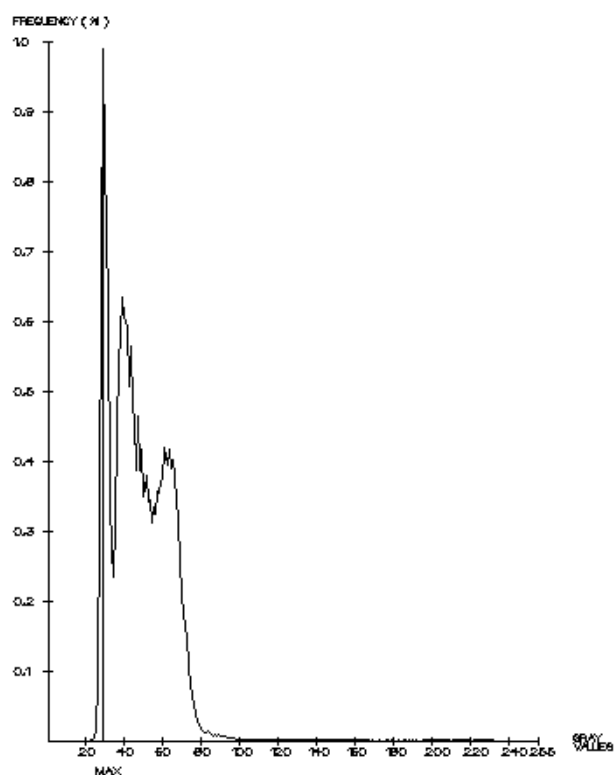
**$n^2$             ...Anzahl Pixel im Bild**

- **Das Verfahren kann gut den Kontrast verstärken, ist aber ein bisschen brutal und führt oft zu Sättigung**
- **Das Eingabe-Histogramm kann auch zu einem beliebig definierten Histogramm equalisiert werden (s. Gonzalez und Wintz, 1977).**



SPOT  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =262144  
SCALE FACTOR =20.000

SPOT\_HISTOGRAM\_EQUALISATION  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =262144  
SCALE FACTOR =20.000



**Oben: links ein teil eines SPOT-Bildes, rechts nach Histogrammequalisierung.**

**Unten: die entsprechenden Histogramme.**

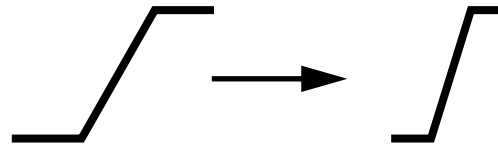
# KANTENVERSTÄRKUNG

Sie kann erreicht werden durch

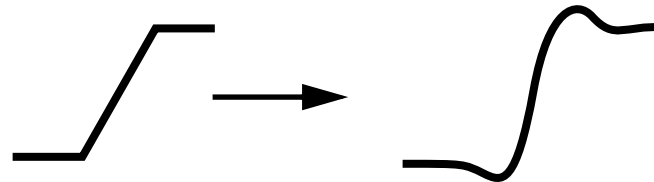
1. Kontrastverstärkung, 2. gezielte Verstärkung der Kanten nur

Erste Methode kann durch Histogrammequalisierung, Wallis Filterung etc. erreicht werden. Die zweite Methode hat zwei Alternativen

a) Mach die Kanten steiler (Breite verkleinern, ohne Kontrast zu erhöhen),  
z.B. mit kantenerhaltender Glättung (EPS)



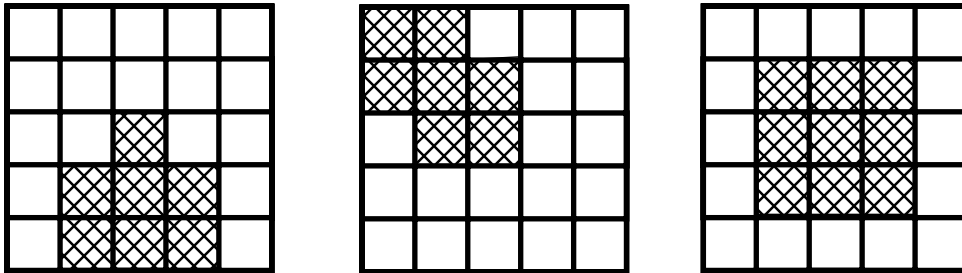
b) Verstärke den Kantenkontrast nur lokal, z.B. mit Wallis oder High-Boost-Filter



Lokale Kontrastverstärkung verursacht ein “Overshoot”, das oft visuell störend wirkt

## KANTENERHALTENDE GLÄTTUNG (Edge Preserving Smoothing)

- In Wirklichkeit Kanten werden stark verstärkt, aber Glättung (Rauschenreduktion) ist sehr klein
- Benutzt 9 Masken innerhalb einer 5 x 5 Nachbarschaft: 4 pentagonale (in den Hauptrichtungen), 4 hexagonale (in den diagonalen Richtungen), und eine zentrale 3 x 3 Maske



- In jede Maske wird Mittelwert und Standardabweichung der Grauwerte berechnet
- Zentrales Pixel bekommt als Grauwert den Mittelwert der Maske mit der kleinsten Standardabweichung

## **Nachteile:**

- **Glättung ist schwach**
- **Homogene Grauwertregionen werden kreiert auch in Regionen von kleiner Grauwertvariation**
- **Alle Details kleiner als die Maskgrösse werden eliminiert**

## **Vorteile:**

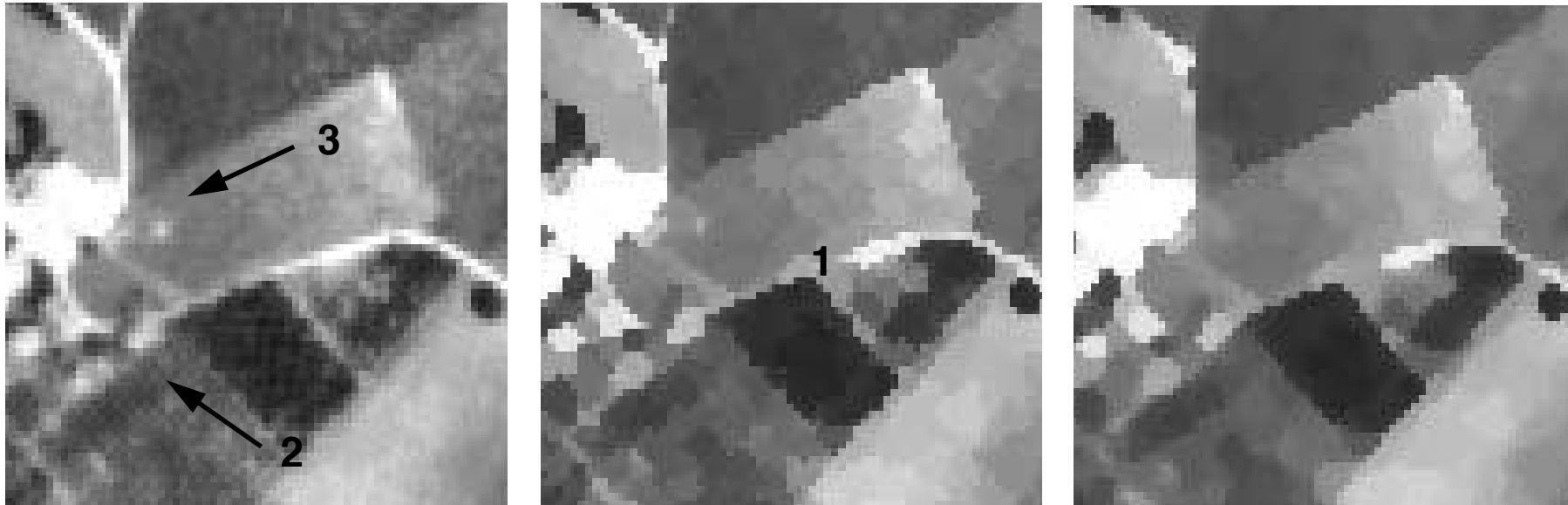
- **Sehr gute Kantenverstärkung ohne ihre Position zu ändern => kreiert fast “step-edges”**
- **EPS kann verbessert werden, z.B.**

**if Differenz der max und min Standardabweichung der 9 Masken klein,**

**if min Standardabweichung > Schwellenwert, then keine Grauwertänderung**

**else benutze den Mittelwert der 3 x 3 oder 5 x 5 zentralen Maske**

**else benutze den Algorithmus wie er ist**



**Links: Originalbild. Mitte: nach EPS mit einer Iteration. Rechts: wie Mitte nach einer 3 x 3 Median um die Kanten zu homogenisieren (Verlauf der Kanten glätten). 1 ...Kanten werden verstärkt. 2 ...kleine Objekte verschwinden, 3 ...homogene Regionen werden gebildet auch dort wo die Grauwertvariation klein ist.**

## HIGH-BOOST FILTER

- $I_{HB} = a I - I_L = (a - 1) I + I_H$

Mit  $I_{HB}$ ,  $I_L$ ,  $I_H$  ...Bild nach High-Boost-, Tiefpass- und Hochpass-Filterung,

$a = \text{Real} \geq 1$

- Für  $a = 1$ , Ergebnis ähnlich wie bei Laplace
- Je grösser  $a$ , desto ähnlicher ist das Ergebnis am Originalbild, höher der Kontrast, kleiner die Kantenverstärkung



**Links: Originalbild. Mitte: High-Boost Filter mit  $\alpha = 1.5$ . Rechts: High-Boost Filter mit  $\alpha = 2.5$ .**



## WALLIS FILTER

Ziel ist den Mittelwert (Helligkeit) und besonders die Standardabweichung (Kontrast) eines Bildes an vorgegebenen Sollwerten anzupassen (Wallis, 1976). Die Operation des Wallis-Filters ist gegeben durch

$$f(x, y) = \frac{[g(x, y) - m_g] c \cdot s_f}{[c \cdot s_g + (1 - c) \cdot s_f]} + b \cdot m_f + (1 - b) \cdot m_g$$

oder

$$f(x, y) = g(x, y) \cdot r_1 + r_0 \quad r_1 = \frac{c \cdot s_f}{[c \cdot s_g + (1 - c) \cdot s_f]} \quad r_0 = b \cdot m_f + (1 - b - r_1) \cdot m_g$$

### Diskussion des Wallis Filters

Die Soll- sowie die Ist-Werte des Mittelwerts und der Standardabweichung werden innerhalb Teilbilder (lokale Nachbarschaften) berechnet. Der Filter ist adaptiv (benutzt die lokale Ist-Werte), lokal und nichtlinear.

**$g(x,y)$ ,  $f(x,y)$  ...Original, gefiltertes Bild**

**$m_g$  ...Ist lokaler Mittelwert (in einer  $i * j$  Nachbarschaft)**

**$s_g$  ...Ist lokale Standardabweichung**

**$m_f$  ...Soll Mittelwert (gültig für alle Teilbilder), Vorschlag ca. 127 wählen (Mitte des Grauwertbereiches)**

**$s_f$  ...Soll Standardabweichung (gültig für alle Teilbilder), Vorschlag  $40 \leq s_f \leq 70$   
je kleiner die Teilbilder, desto kleiner  $s_f$  sein sollte, um das Rauschen nicht zu verstärken**

**$c$  ...Kontrastverstärkungskonstante,  $c \in [0, 1]$ , Vorschlag  $0.7 \leq c \leq 1$**

**$c = 1 \Rightarrow$  neue Standardabweichung =  $s_f$**

**$c = 0 \Rightarrow r_1 = 0$  (gefiltertes Bild hat konstante Grauwerte)**

**Um Rauschenverstärkung zu vermeiden,  $c$  sollte klein ist. Hat viel stärkeren Einfluss auf Rauschen als  $s_f$**

**b ...Helligkeitskonstante,  $b \in [0, 1]$ , Vorschlag  $0.5 \leq c \leq 1$**

**$b = 1 \Rightarrow$  neuer Mittelwert =  $m_f$**

**$b = 0 \Rightarrow$  neuer Mittelwert bleibt  $m_g$**

**Die Dimensionen  $i, j$  der Teilbilder ist proportional zur Grösse der Objekte, die hervorgehoben werden sollten. Das klassische Wallis Filter hat  $c = 1, b = 1$ .**

## **IMPLEMENTATIONSASPEKTE**

**1. Das Bild wird in Teilbilder mit Dimensionen  $(i, j)$  geteilt**

**2. Definiere die Distanzen zwischen den Teilbildzentren  $dx, dy$**

**$dx \leq j ; dy \leq i$  ( $dx, dy$  oft so gewählt dass Teilbilder 50% Überlappung haben)**

**3. Falls notwendig, modifiziere 1) und 2) für den rechten und unteren Rand**

**4. Berechne in jedem Teilbild die Parameter  $r_1, r_0$**

**5. Transformiere jedes Pixel mit Parametern  $r_i$ , die von den  $r_1, r_0$  Werten der 4 benachbarten Teilbilder bilinear interpoliert sind**

**Ziel von Punkt 5. und Teilbildüberlappung => Unterschiede an Rändern der Teilbilder vermeiden**

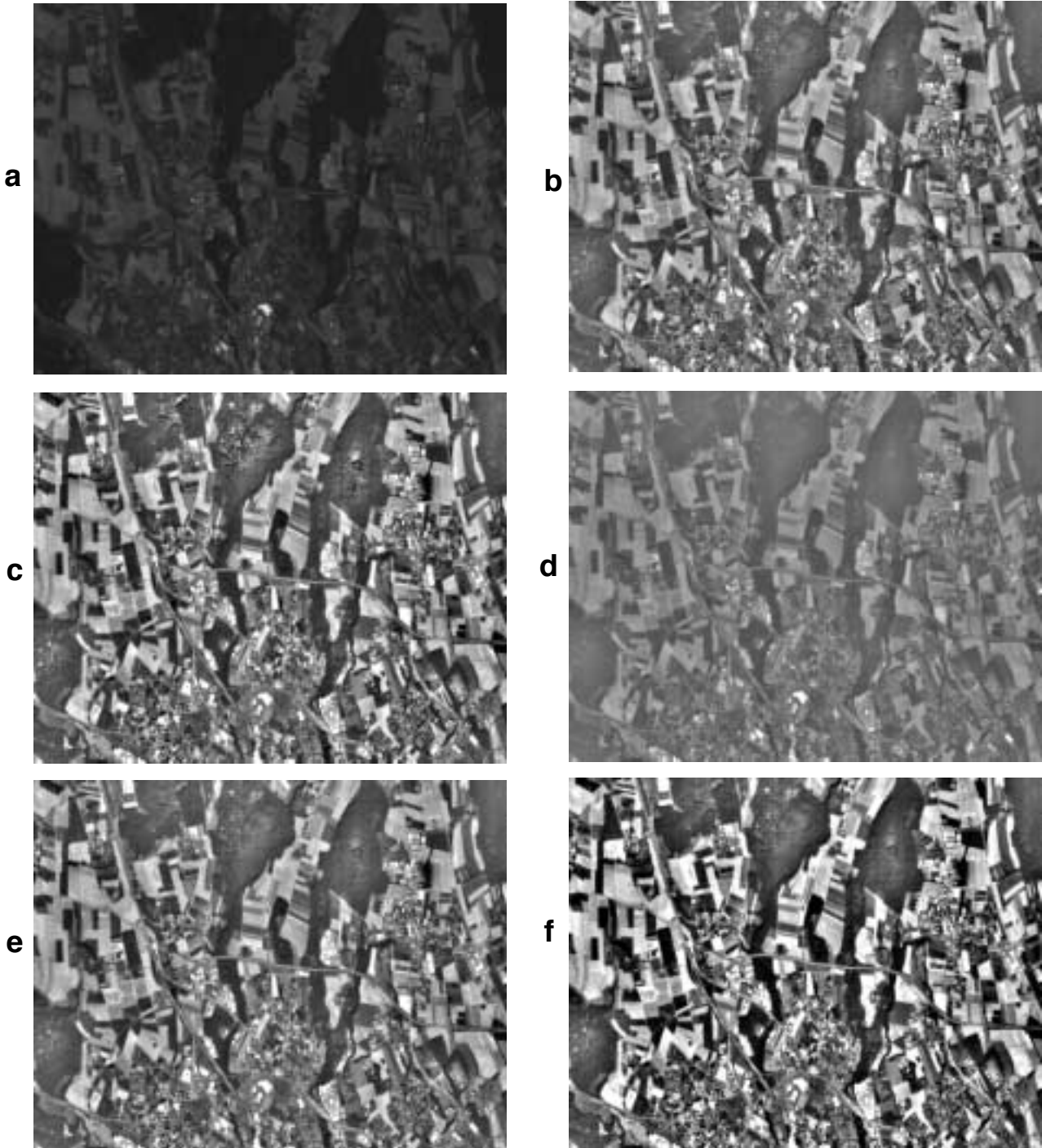
## **Anwendungen von Wallis Filter**

- **Sehr vielfältig wegen seiner Flexibilität**
- **Tiefpass-, Hochpass-, Bandpass-Filterung**
- **Kontrast- und Kanten-Verstärkung (wichtig für Merkmalsextraktion und automatische DTM Generierung)**
- **Radiometrische Equalisierung von Bildern (falls alle Bildern mit denselben Wallis Parametern gefiltert wurden ; Vorschlag für Parameter: Teilbildgrösse klein,  $m_f = 127$ ,  $b = 1$ ,  $c = \text{ca. } 1$ ,  $s_f = \text{nicht sehr hoch}$ ). Nützlich in:**

**Vergleich von multitemporalen Bildern, Mosaikierung von Bildern, Korrektur von Streifen**

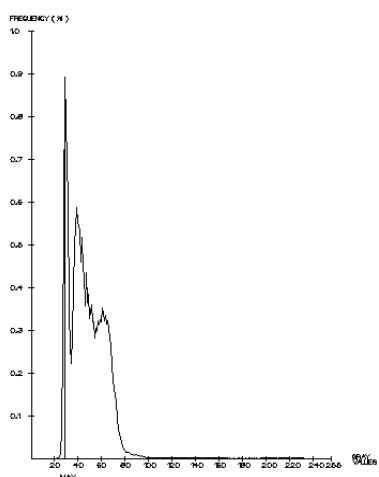
## Beispiele von Wallis Filter

- Effekt von unterschiedlichen  $c$  und  $s_f$  Werten

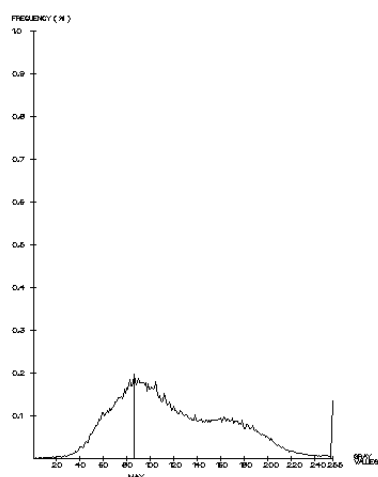


**SPOT Satelliten-Bilder: a) Original, b) Wallis mit  $i = j = 50$ ,  $m_f = 120$ ,  $b=1$ ,  $s_f = 65$ ,  $c = 0.96$ , c) wie b) mit  $c = 11.0$ , d) wie b) mit  $c = 0.8$ , e) wie b) mit  $s_f = 75$ , f) wie b) mit  $s_f = 127$ .**

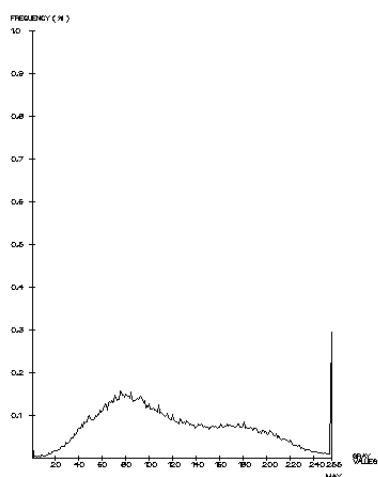
SPOTROI  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000



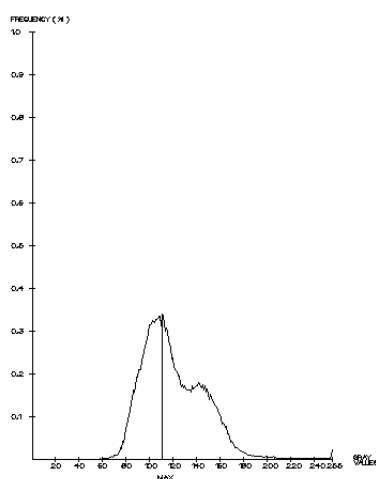
SPOTROIENH1  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000



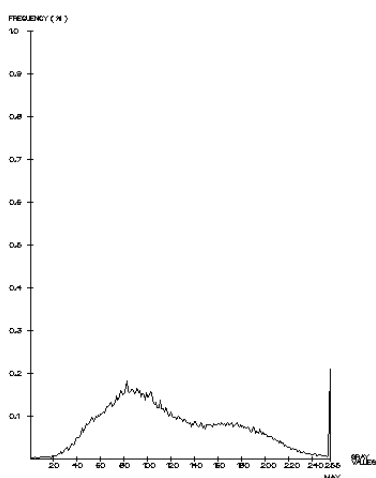
SPOTROIENH2  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000



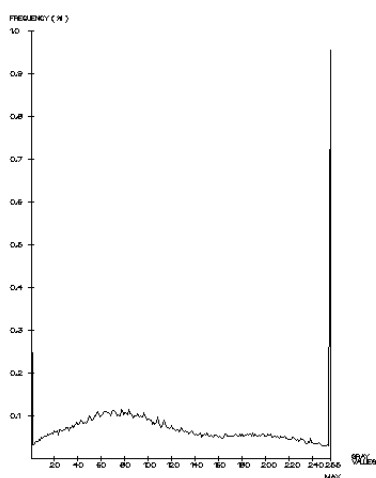
SPOTROIENH3  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000



SPOTROIENH4  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000

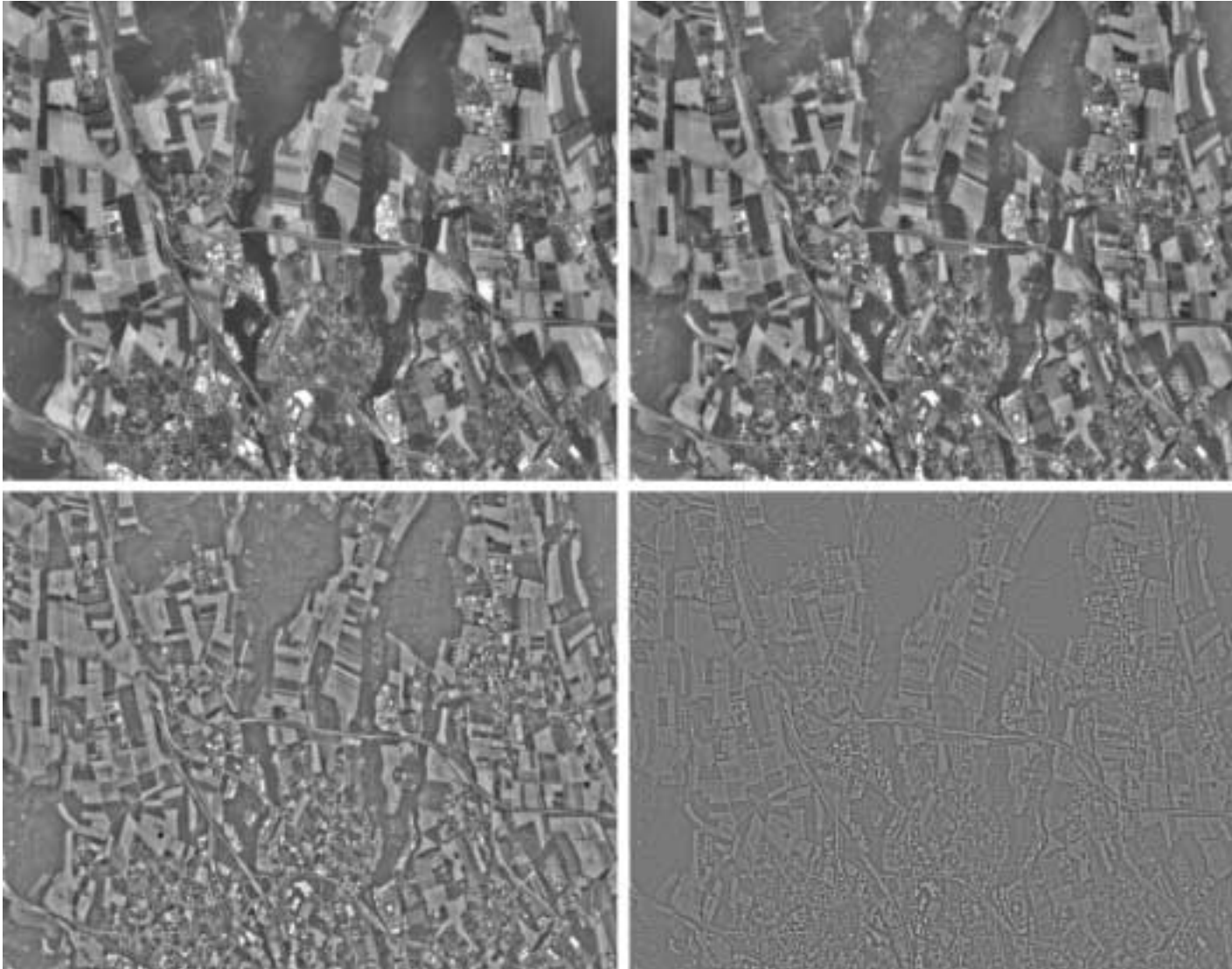


SPOTROIENH5  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =205824  
SCALE FACTOR =8.000



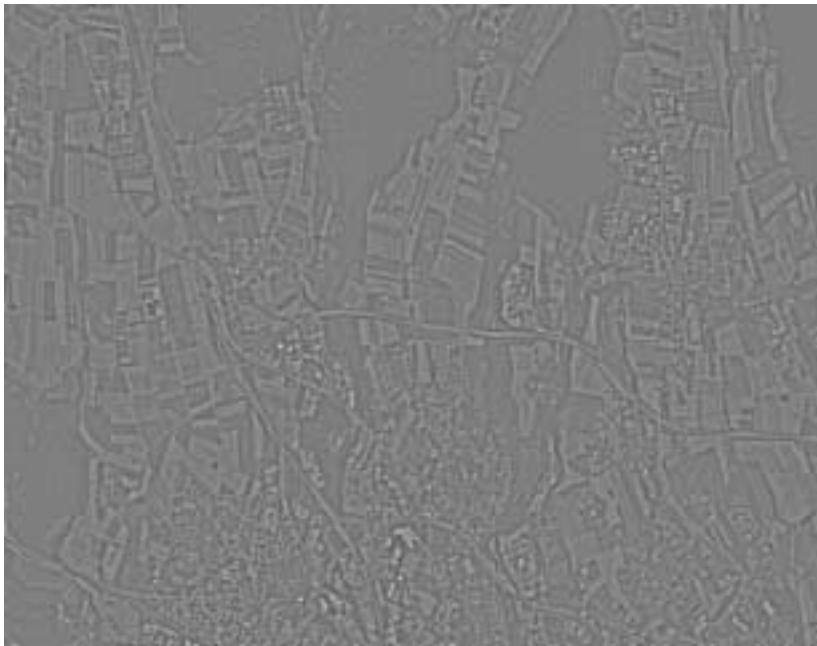
**Die entsprechende Histogramme der Bilder der vorherigen Figur.**

- Effekt von unterschiedlichen Teilbildgrößen

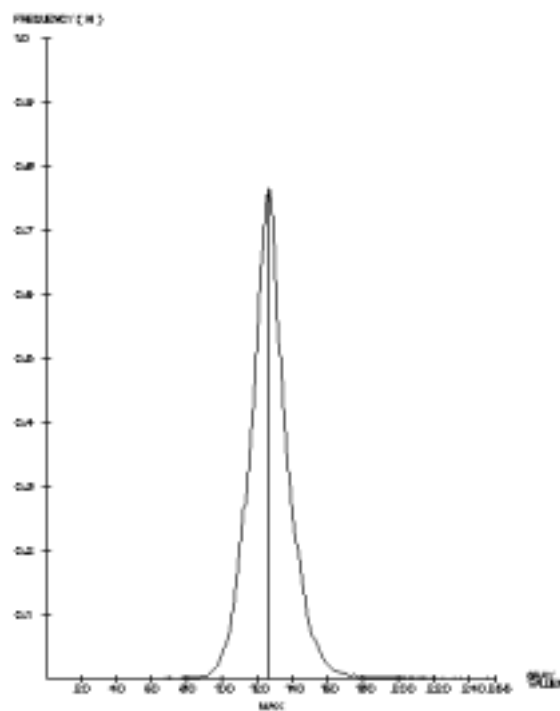


Zeilenweise von oben links: Teilbildgrösse ( $i = j = 75, 30, 14, 4$ ). In allen Bildern  $m_f = 121$ ,  $s_f = 65$ ,  $b = 1$ ,  $c = 0.9$

- Wallis als Vorstufe für Kantendetektion



SPOTOLENHT2  
H-INTERVAL =1 GRAY VALUES  
NUMBER OF PIXELS =306824  
SCALE FACTOR =16.000



Oben: Ergebnis mit Parameter  $i = j = 5$ ,  $m_f = 127$ ,  $s_f = 60$ ,  $b = 1$ ,  $c = 0.82$ . Unten: das entsprechende Histogramm.



## 2.9. MORPHOLOGISCHE OPERATIONEN

### MORPHOLOGISCHE OPERATIONEN MIT BINÄREN BILDERN

- **Strukturelles Element**
  - Entspricht der Maske in der Filterung ; Form kann beliebig sein
  - Hat einen Ursprung (Bezugspunkt) ; Position des Ursprungs beliebig
  - Normalerweise ist Form symmetrisch und Ursprung am Maske-Zentrum

- **Annahme**

**Bild = Vordergrund (schwarz) + Hintergrund (weiss) ; Objekt = Vordergrund**

**Strukturelles Element (SE) = schwarz**

- **Nur 2 Operatoren (SE) notwendig: schwarze M.P. und weisse M.P.**



**Schwarze Mrs. Packmann**



**Weisse Mrs. Packmann**

## **REGEL DER MATHEMATISCHEN MORPHOLOGIE**

- 1. Mrs. Packmann sucht überall im Bild für Essen, d.h. ihrer Mund (= Ursprung des SE) sucht in alle Pixel des Bildes.**
- 2. Mrs. Packmann frisst Objekte derselben Farbe wie sie. Sie beginnt aber zu fressen, nur wenn sie durch Treffen (Überlappung) mit Objekten anderer Farbe gereizt wird.**
- 3. Wenn sich die weisse/schwarze M.P. und Vordergrund (schwarz)/Hintergrund (weiss) überlappen, dann werden weisse/schwarze Pixel unter dem Mund gefressen und werden respektiv zu schwarz/weiss.**
- 4. Je dicker Mrs. Packmann, desto mehr Pixel frisst sie (grössere Überlappung)**
- 5. Wenn sie Hunger hat, geniesst sie mehrere Gänge (= Iterationen). Es gibt aber ein Unterschied. Je mehr Gänge, desto grösser die Portionen für die weisse M.P. (mehr Dilatation), und je kleiner für die schwarze M.P. (weniger Erosion)**

# **MORPHOLOGISCHE OPERATIONEN (MO)**

(beziehen sich auf dem Bildvordergrund)

**1.Dilatation = weisse M.P., d.h.**

**Hintergrund (weiss) wird erodiert => Vordergrund (schwarz) wird dilatiert**

**2.Erosion = schwarze M.P., d.h.**

**Vordergrund (schwarz) wird erodiert => Hintergrund (weiss) wird dilatiert**

**3.Öffnen (Erosion + Dilatation): erst schwarze M.P., dann weisse M.P.**

**4.Schliessen (Dilatation + Erosion): erst weisse M.P., dann schwarze M.P.**

## **EFFEKT DER MORPHOLOGISCHEN OPERATIONEN**

- **Dilatation**

**Vergrößerung der Objekte, Löcher können geschlossen werden, benachbarte Objekte können verschmolzen werden.**

- **Erosion**

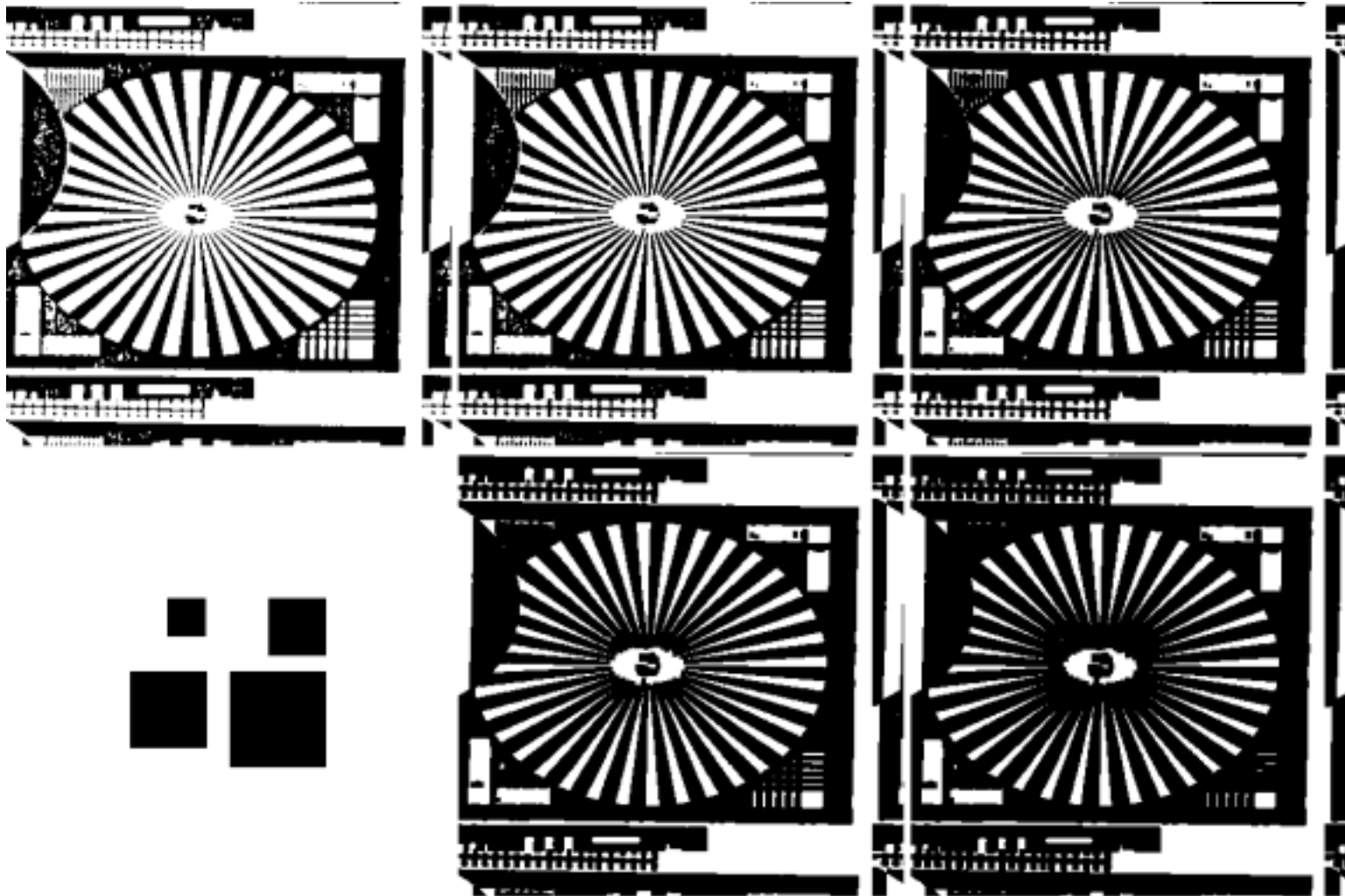
**Verkleinerung der Objekte, Löcher werden grösser, zusammenhängende Objekte können getrennt werden.**

- **Öffnen**

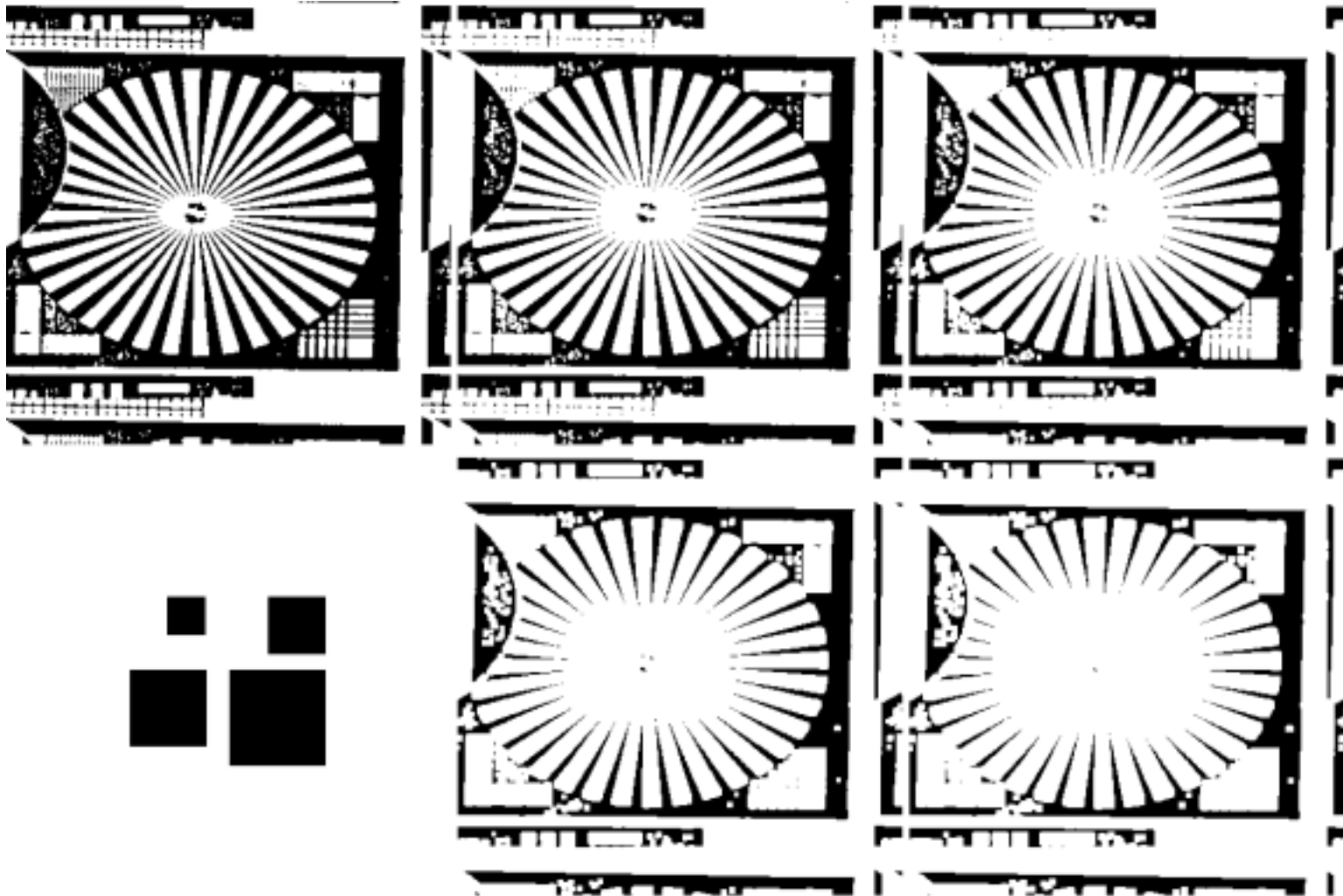
**Grössere Strukturen bleiben erhalten, Linien, Punkte und feine Strukturen werden gelöscht.**

- **Schliessen**

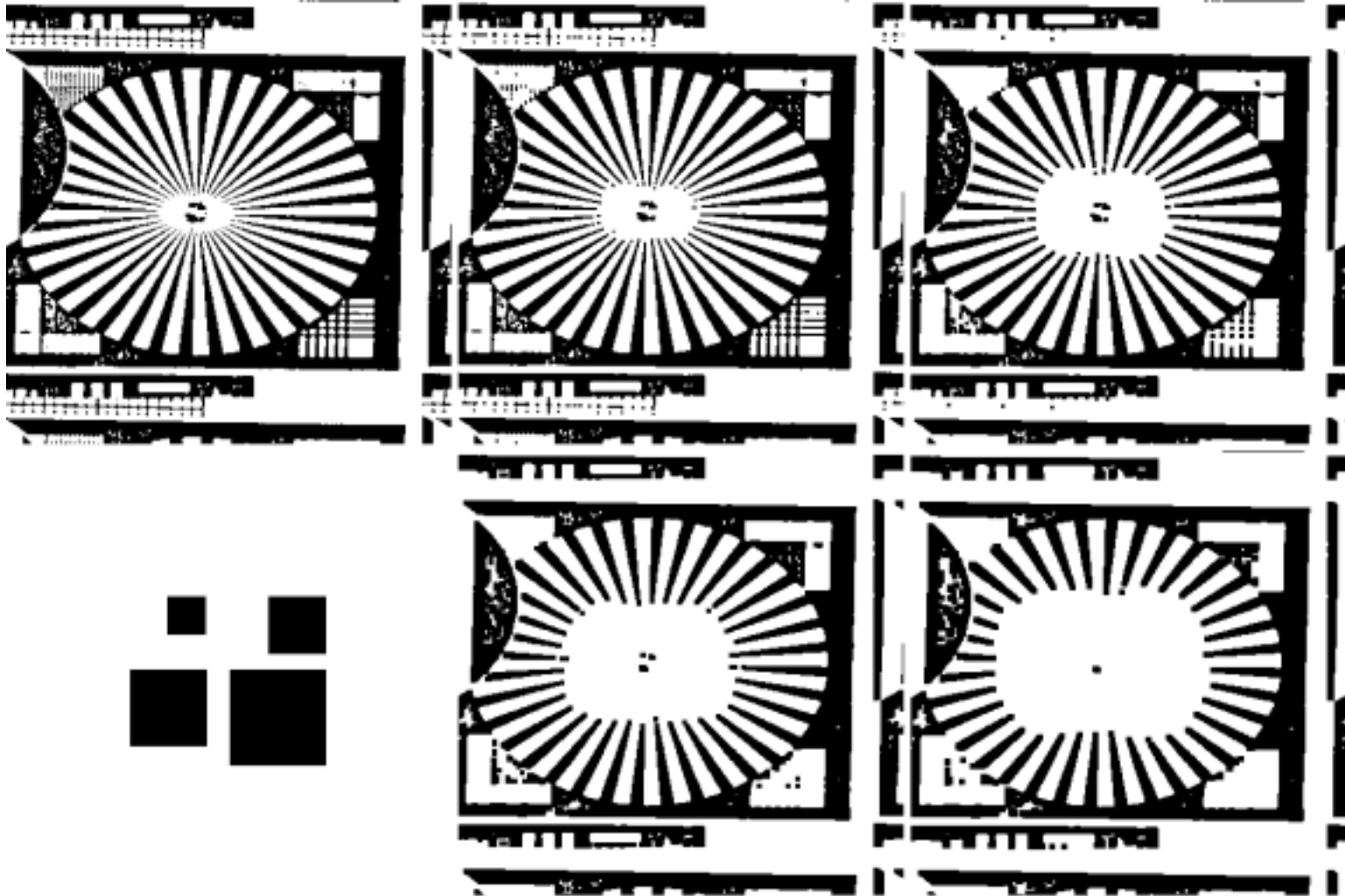
**Kleine Unterbrechungen, Lücken und Risse werden ausgefüllt.**



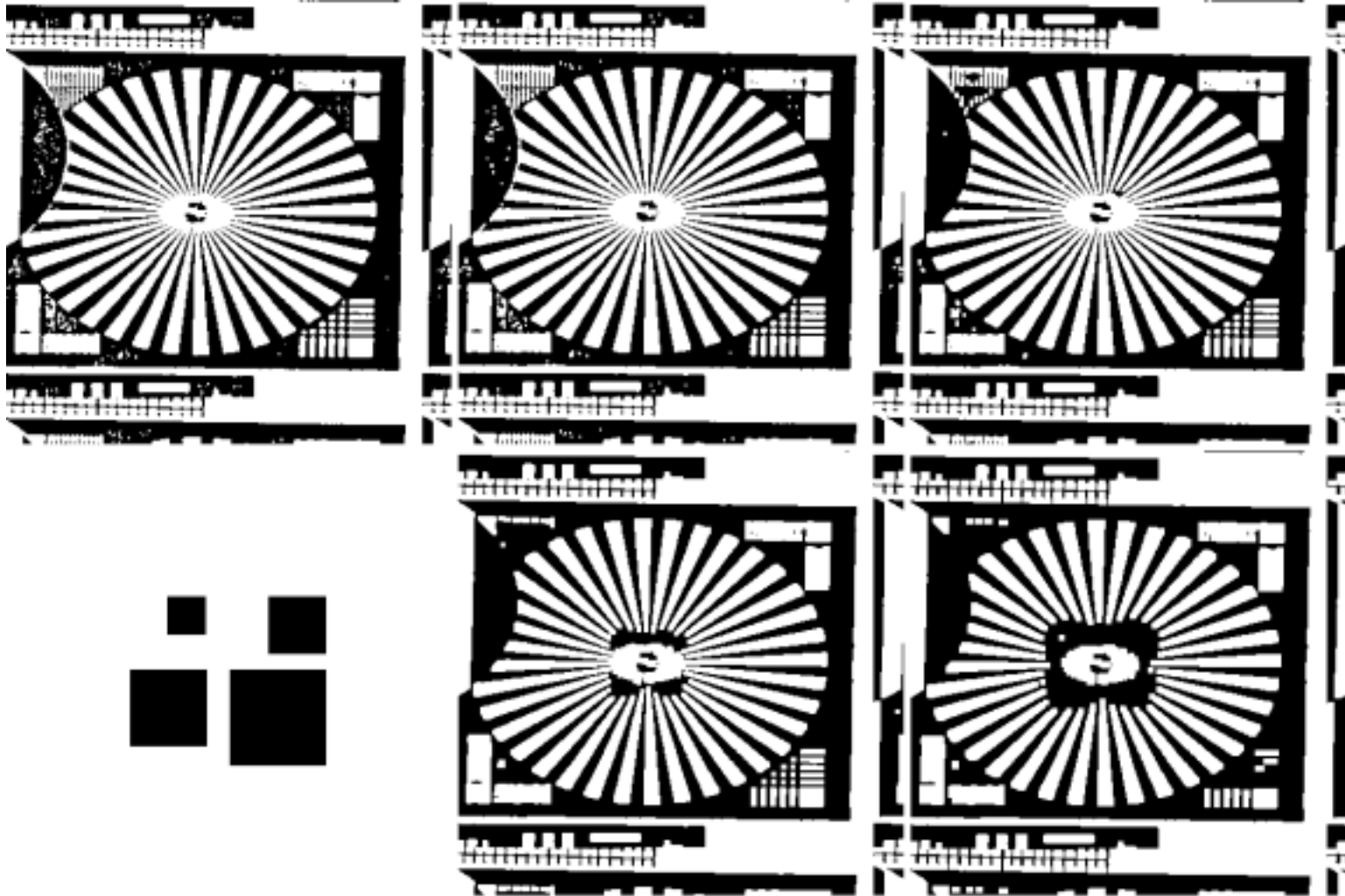
**Dilatation.** Oben links: Original Bild. Übrige Bilder ...Ergebnis der Dilatation mit den strukturellen Elementen (dargestellt als Bilder, Grösse 2 x 2 - 5 x 5 Pixel) unten links.



**Erosion. Oben links: Original Bild. Übrige Bilder ...Ergebnis der Dilatation mit den strukturellen Elementen (dargestellt als Bilder) unten links.**

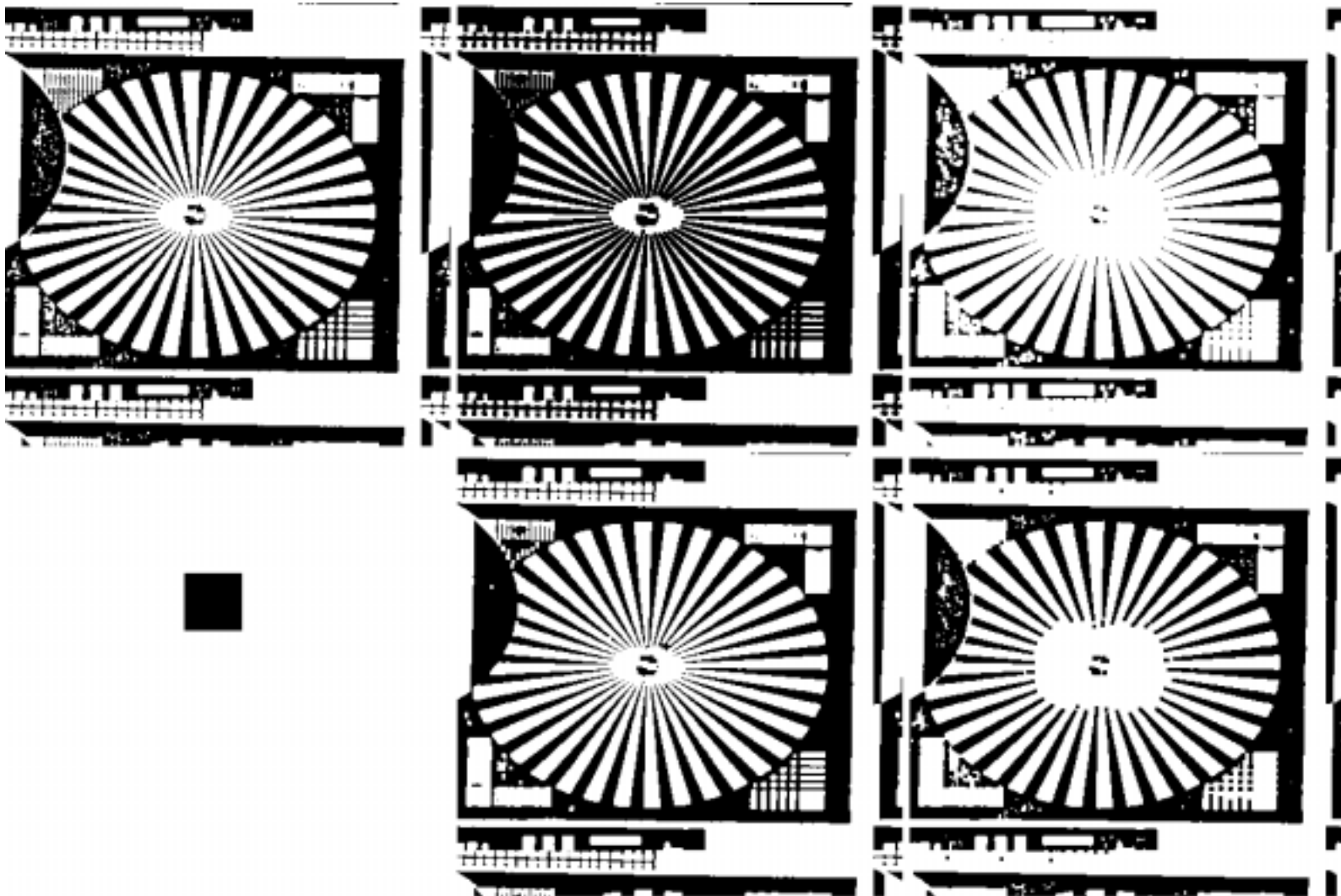


Öffnen. Oben links: Original Bild. Übrige Bilder ...Ergebnis der Dilatation mit den strukturellen Elementen (dargestellt als Bilder) unten links.



Schliessen. Oben links: Original Bild. Übrige Bilder ...Ergebnis der Dilatation mit den strukturellen Elementen (dargestellt als Bilder) unten links.

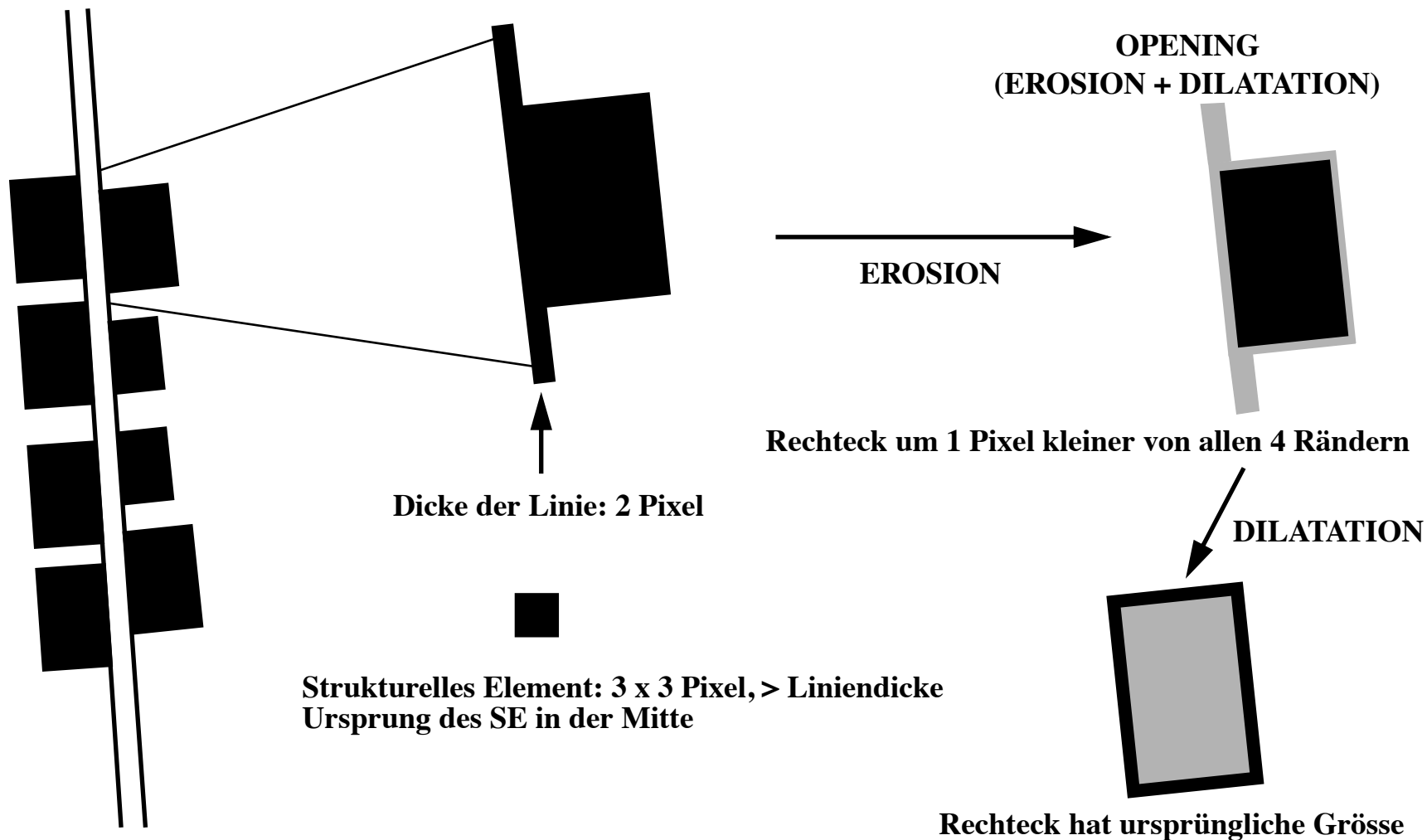




Vergleich von morphologischen Operationen mit binären Bildern und einem 3 x 3 strukturellen Element (unten links). Von oben links: Originalbild, Dilatation, Erosion, Schliessen, Öffnen.

## Beispiel morphologischer Operationen

Anwendung: Trennung von Gebäuden und Strassen in digitalisierte Ebene Situation der Landeskarte 1:25,000



## MO ZWISCHEN GRAUWERTBILDERN UND BINÄREN SE

- **Dilatation (max Filter)**

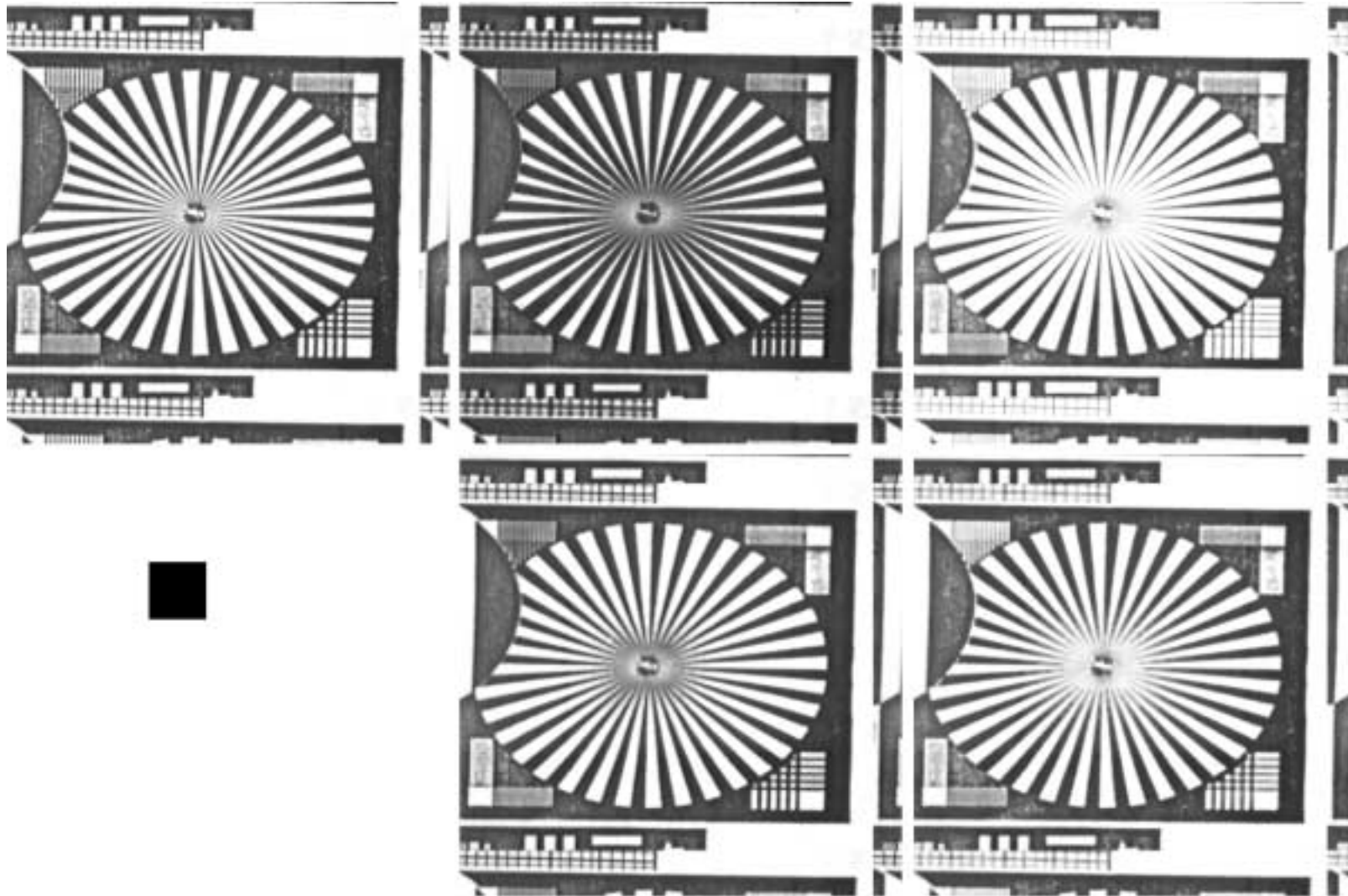
**Pixel unter dem SE-Ursprung = max. Wert der Bildpixel, die sich mit dem SE überlappen**

- **Erosion (min Filter)**

**Bildpixel unter dem SE-Ursprung = min. Wert der Bildpixel, die sich mit dem SE überlappen**

- **Öffnen (min/max Filter)**

- **Schliessen (max/min Filter)**



**Vergleich von morphologischen Operationen mit Grauwertbildern und einem 3 x 3 strukturellen Element (unten links). Von oben links: Originalbild, Dilatation, Erosion, Schliessen, Öffnen.**

## **Anwendungen von morphologischen Operationen**

**Nichtlineare Filterung ; Kantendetektion ; Rauschenreduktion ; Bildhervorhebung ; Repräsentation, Glättung und Erkennung von Formen ; Texturanalyse ; Skelettierung ; Kodierung ; Ausfüllen von geschlossenen Polygonen etc.**

## **Vorteile von morphologischen Operationen (MO)**

- **Breite Anwendung in sehr unterschiedlichen Aufgaben ;**

**Viele komplexe Aufgaben gelöst basierend auf einige einfache Operationen => niedrige Kosten**

- **Kompakte und systematische Repräsentation von Algorithmen mit Hilfe von MO**
- **Einfache Parallelisierung der Operationen**
- **Einfache Implementation, schnelle Operationen**

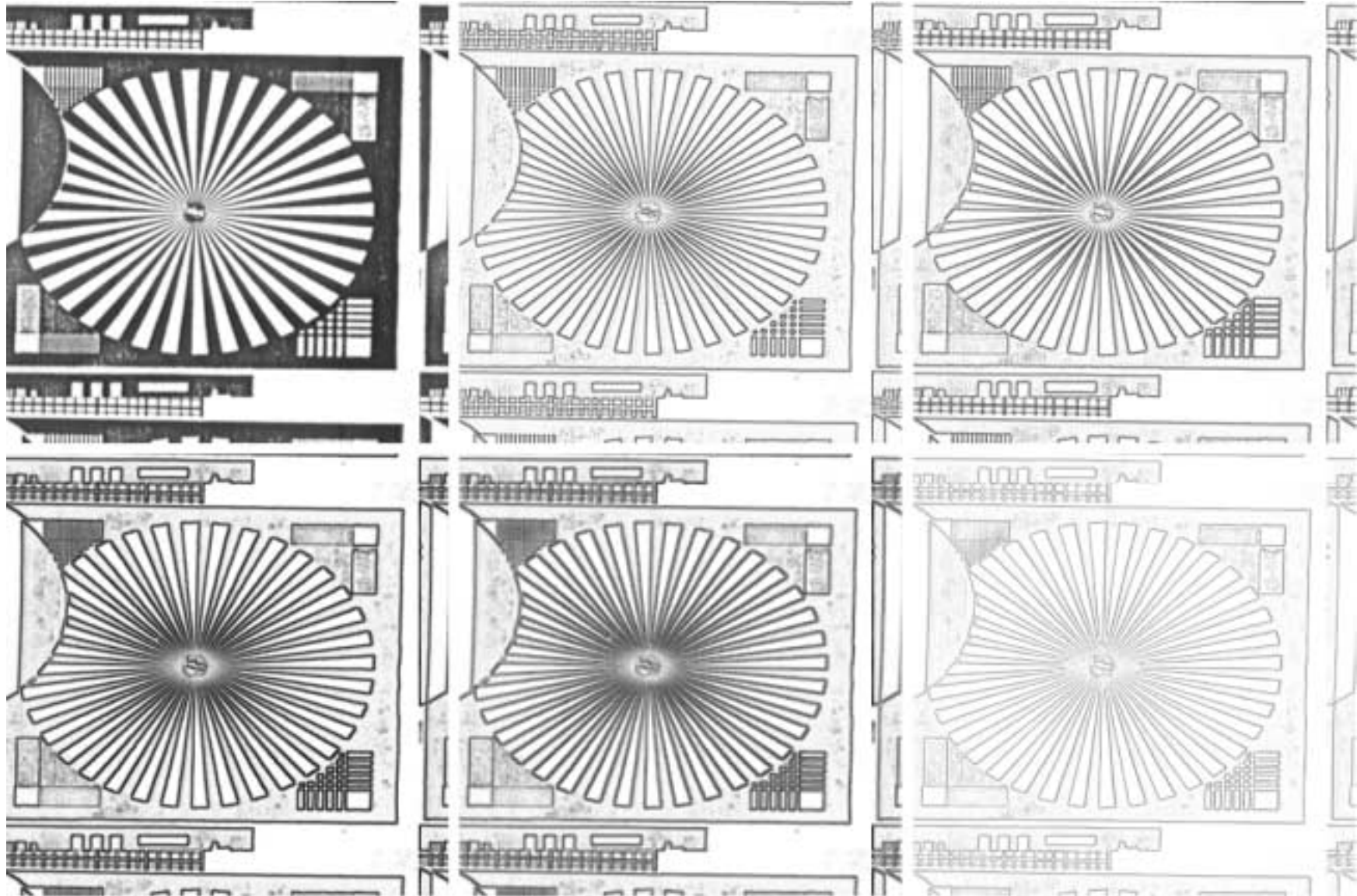
**Wesentliche Operationen: Addition, Subtraktion, Vergleiche, logisches AND/OR**

# BEISPIELE VON ANWENDUNGEN

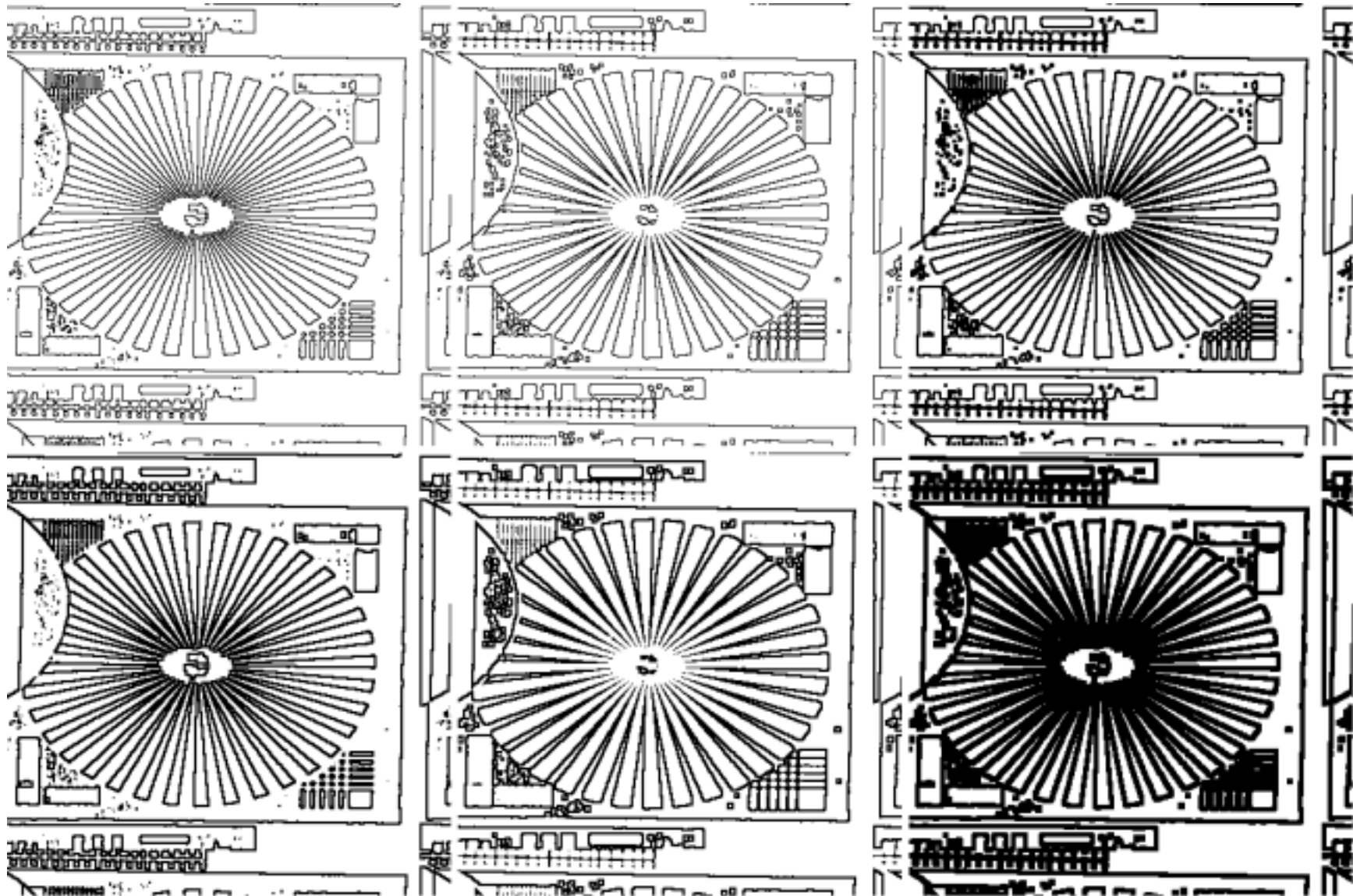
## Kantendetektion

- Anwendung auf binären oder Grauwertbilder. SE normalerweise symmetrisch und klein (3 x 3 - 5 x 5 Pixel)
- Beispiele von Kanten - Operatoren ( $G$  ...Eingabebild):
  - $O1 = \text{Dilatation}(G) - G$
  - $O2 = G - \text{Erosion}(G)$
  - $O3 = \text{Dilatation}(G) - \text{Erosion}(G)$
  - $O4 = \max(O1, O2)$
  - $O5 = \min(O1, O2)$

Operatoren 3-5 geben einen symmetrischen Response in Bezug auf die Kante



Von oben links: Original Grauwertbild, die 5 Kantendetektoren mit einem 3 x 3 SE



Das Originalbild der vorherigen Figur wurde binarisiert und dann die drei ersten Kanten-Operatoren angewendet. Oben: Ergebnis mit 3 x 3 SE. Unten: Ergebnis mit 5 x 5 SE



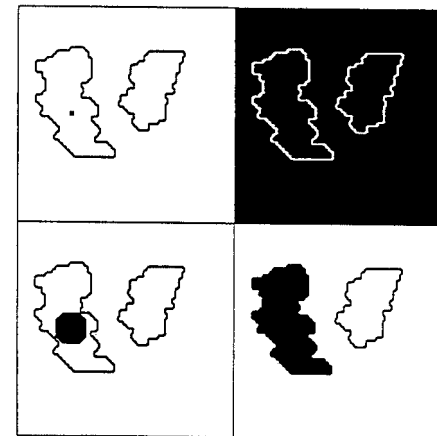
## “Region Filling”

- Gegeben: eine binäre Kontur (von Breite  $w$ ) einer geschlossenen Region, und ein Startpunkt  $(x,y)$  in der Region  
Aufgabe: Finde und markiere alle Pixel in der Region
- Prozedur:

Wahl eines symmetrischen, binären SE mit Grösse  $n$  ( $n \leq (2w+1)$ ,  $n \geq 3$ ). Der Ursprung des SE, sowie dessen 4- oder 8-Nachbar (je nachdem was beim binären Kontur verwendet ist), sollen dem SE gehören.

Mit  $X$  ...binäres Eingabebild,  $F$  ...ausgefüllte Fläche,  $X_c$  ...komplementär (inverses) von  $X$ ,  $\emptyset$  ...leere Menge,  $\cap$  ...Schnittoperation, der Algorithmus lautet:

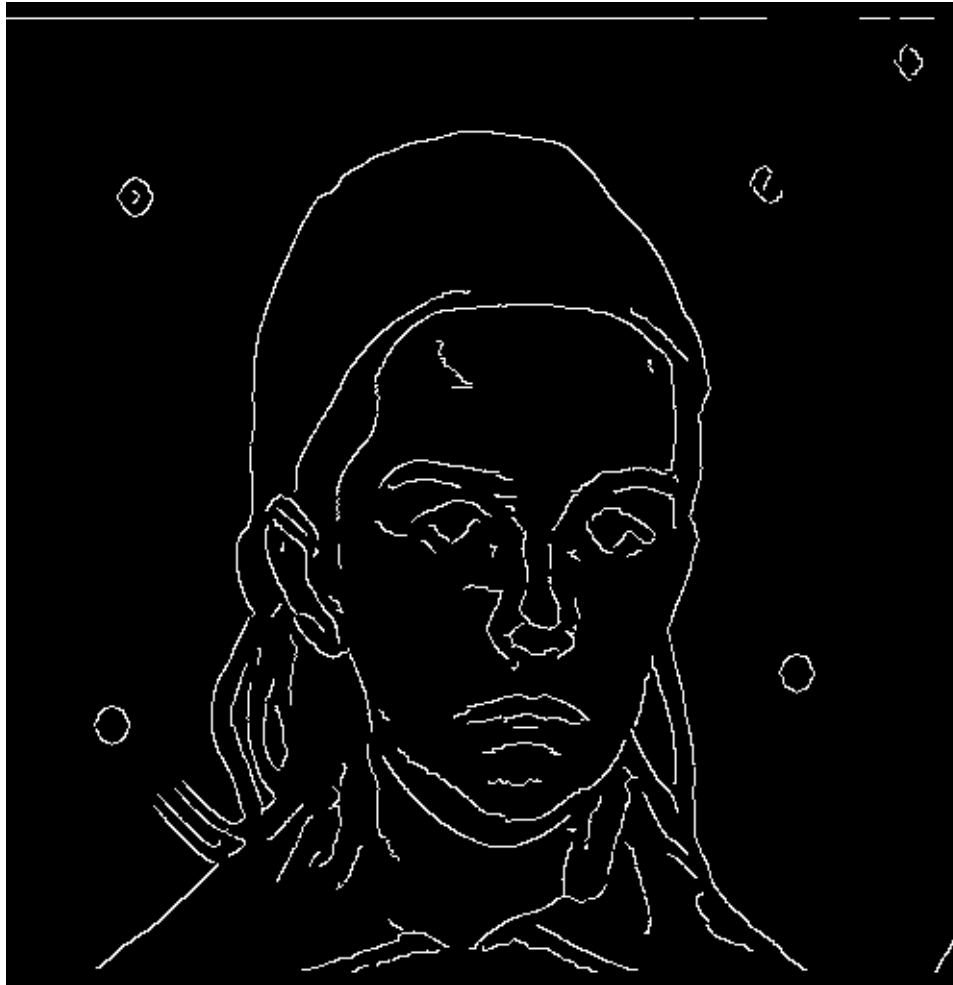
```
i = 0
Fi = { (x,y) }
10 continue
Fi+1 = Dilatation(Fi)  $\cap$  Xc
if (Fi+1 - Fi) =  $\emptyset$  then
    i = i + 1
    goto 10
```



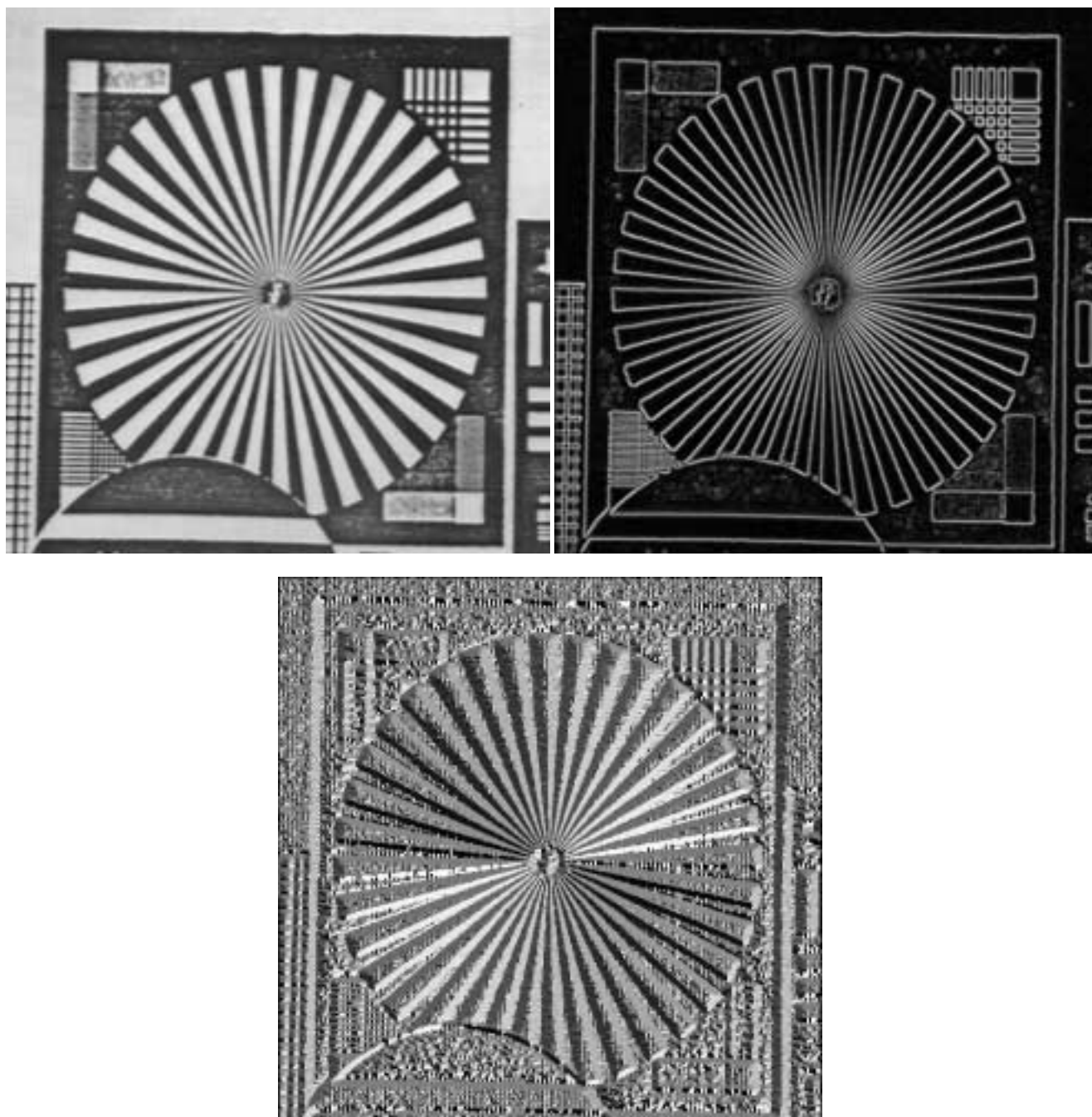
Oben: links  $X$ , rechts  $X_c$ . Unten: links Ergebnis nach einigen Iterationen, rechts Schlussergebnis (nach Maragos, 1987).

### 3. MERKMALSEXTRAKTION





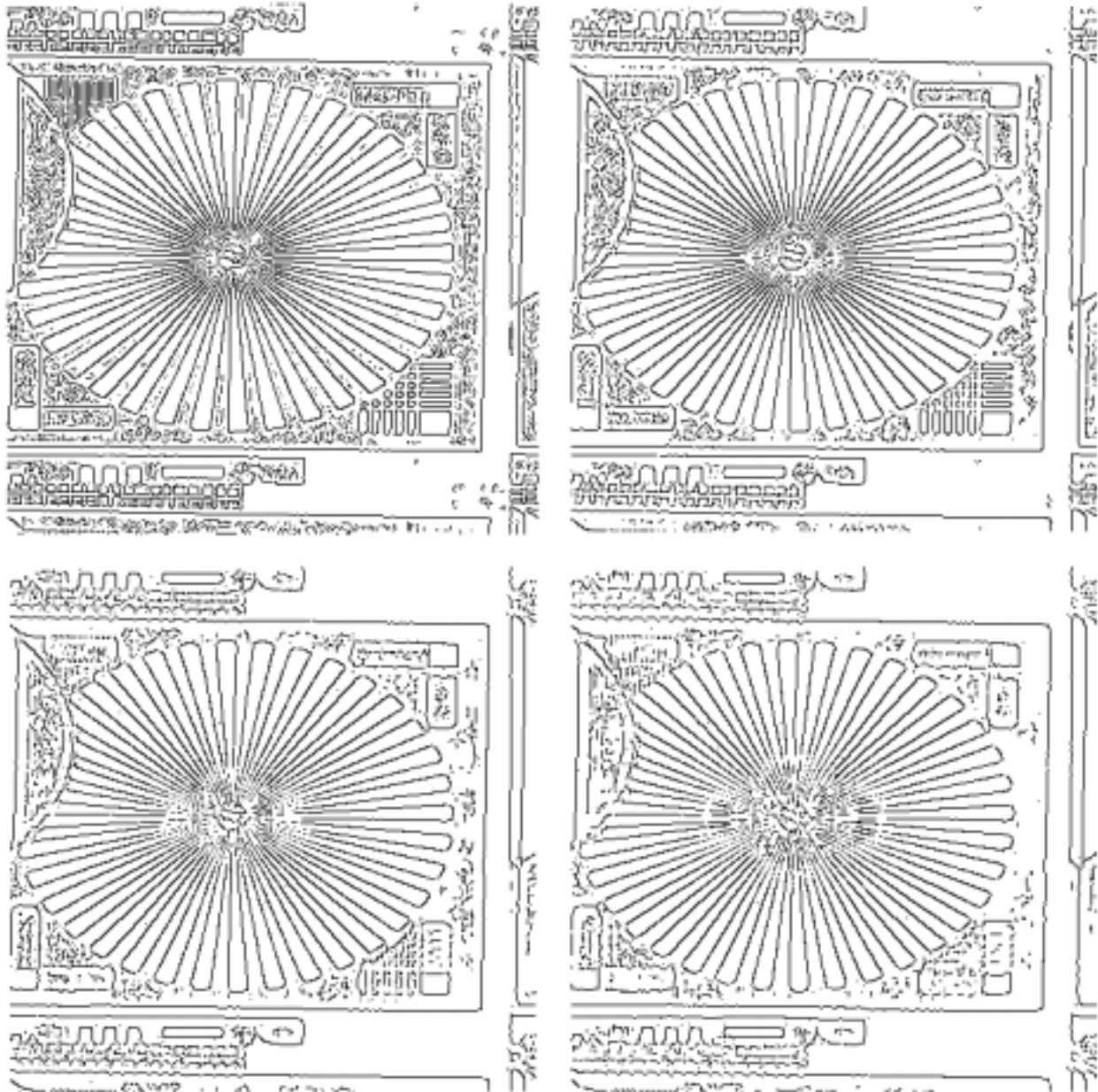
**Verschiedene Typen von Merkmalen. Von oben links (s. vorherige Figur): Originalbild, extrahierte Punkte mit einem Interest Operator, extrahierte Linien mit dem Canny Operator, extrahierte homogene Grauwertflächen mit kantenerhaltender Glättung und Median-Filterung.**



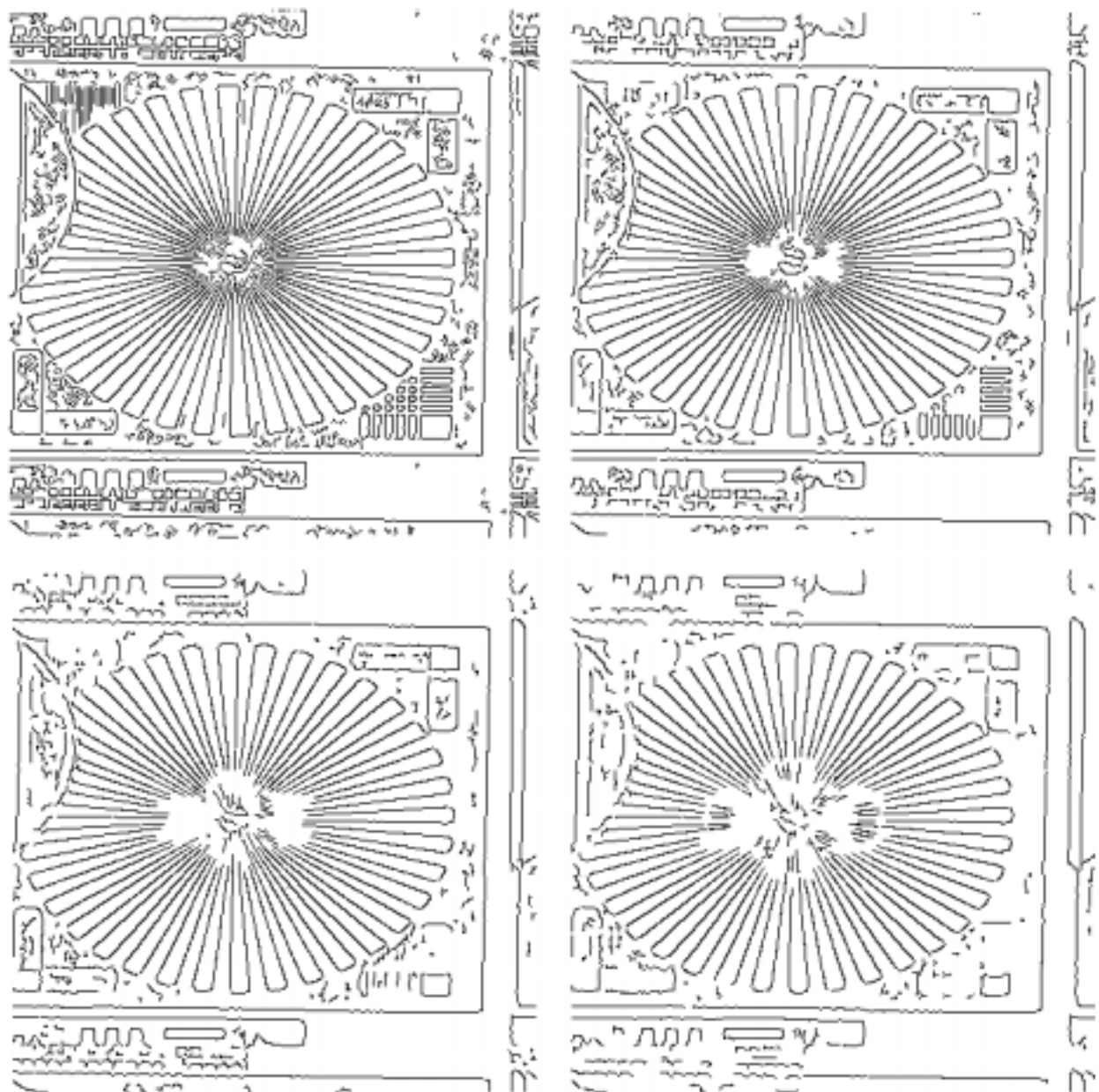
**Kantenextraktion (Beispiel mit dem Sobel Operator). Von oben links: Originalbild, Betrag der Grauwertgradienten, Gradientenrichtung (0 - 180°) kodiert als Grauwert (0-255)**

# ERGEBNISSE VON KANTENDETEKTOREN

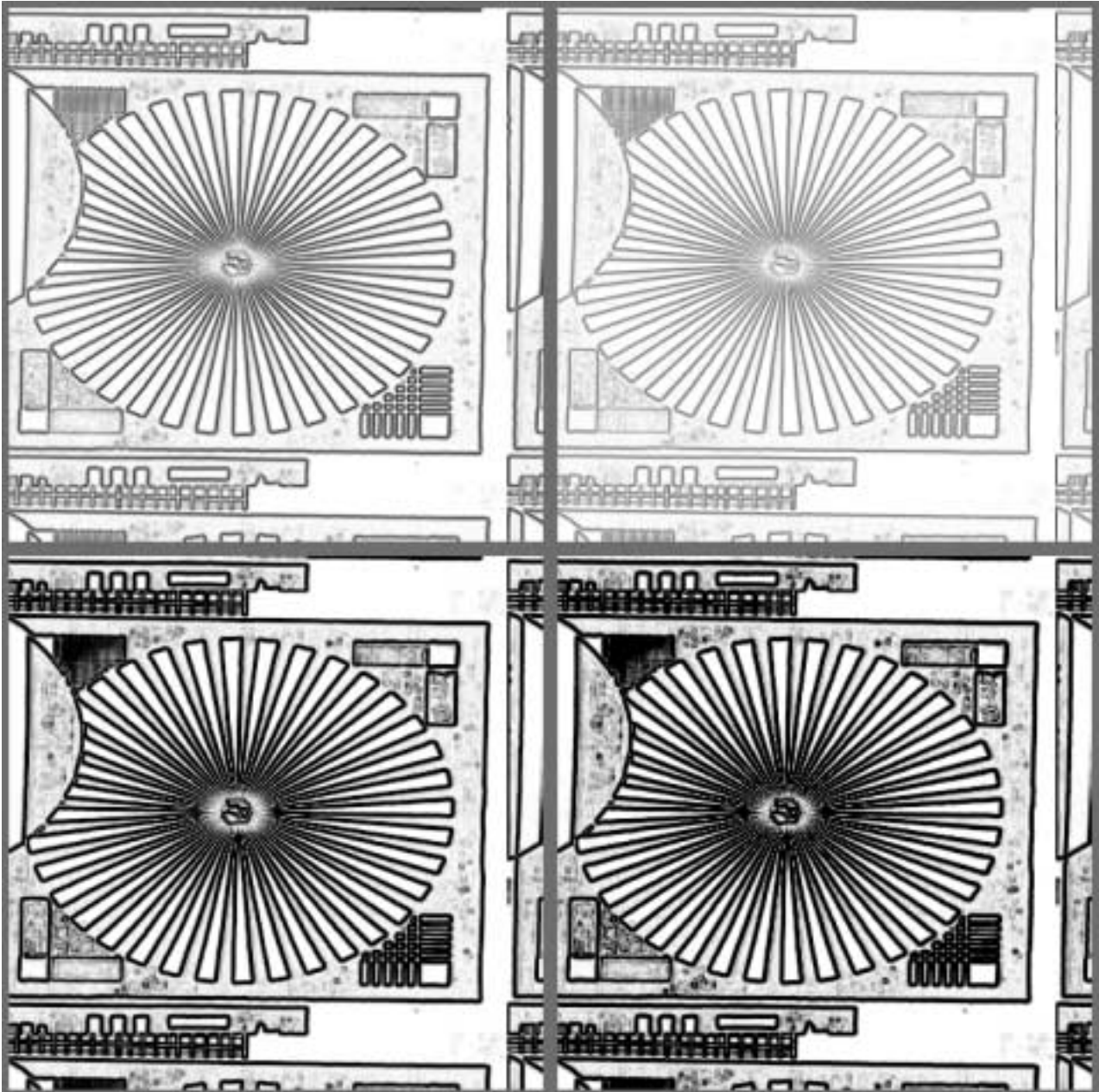
(alle Beispiele beziehen sich auf dem Betrag der Gradienten)



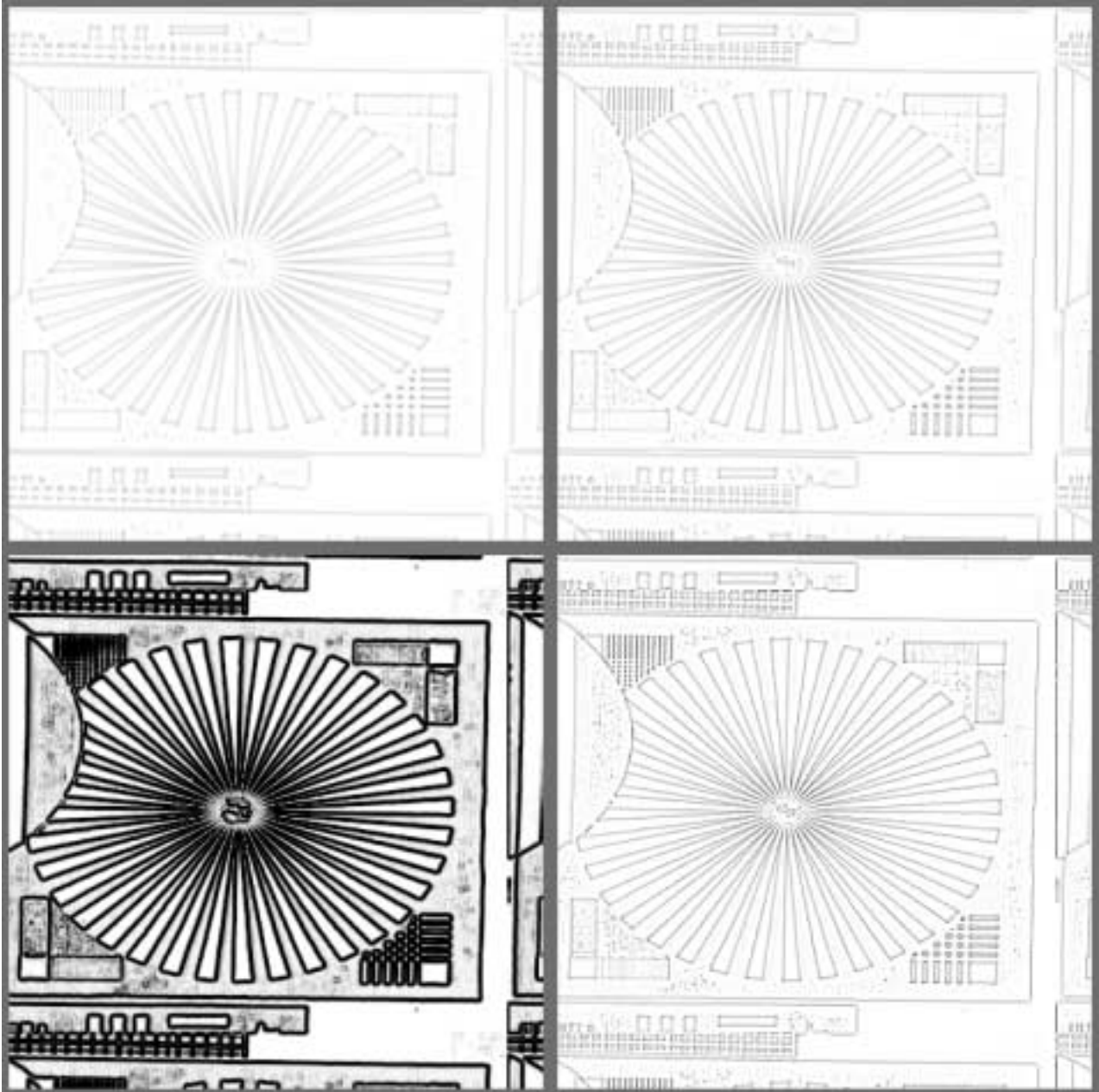
**Canny Operator.** Der Betrag der Gradienten wurde erst binarisiert und dann waren die detektierten Kanten auf 1 Pixel Breite ausgedünnt. Von oben links bis unten rechts wird der Wert von  $\sigma$  graduell grösser, d.h. die Anzahl detektierten Kanten wird kleiner.



**Canny Operator.** Wie vorherige Figur, aber die kurzen Kanten sind hier eliminiert (Annahme: kurze Kanten sind wegen Rauschen).



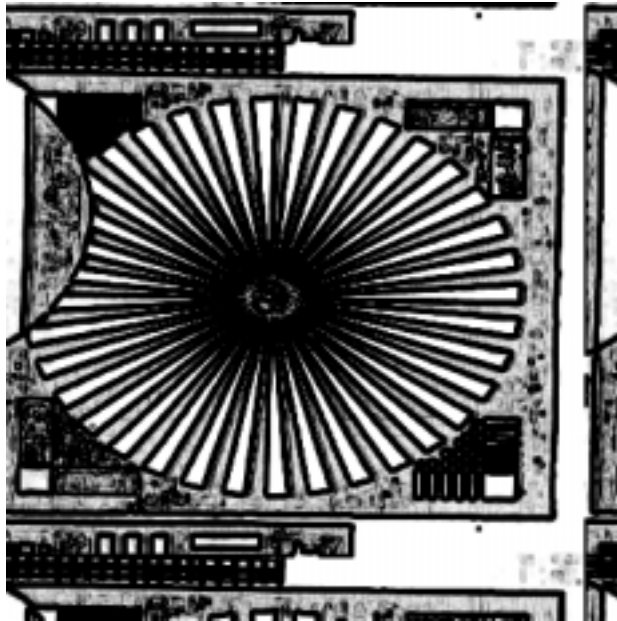
**Betrag der Gradienten. Oben: links diskrete Differenzierung ; rechts Roberts. Unten: links 3 x 3 Prewitt ; rechts 3 x 3 Sobel.**



**Betrag der Gradienten. Oben: links DoG, 7 x 7 Maske,  $\sigma_{\text{klein}} = 1$ ,  $\sigma_{\text{gross}} = 1.6$  ; rechts [Originalbild - (Original nach einer Tiefpassfilterung mit 5 x 5 lokalem Mittelwert)]. Unten: links 3 x 3 “compass” mit 8**

**Masken ; rechts Laplace mit Maske** 
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

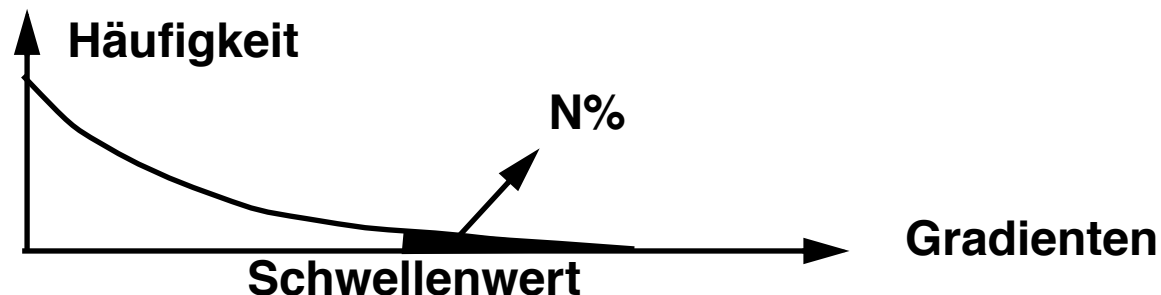




**Betrag der Gradienten. 3 x 3 Kirsch mit 8 Masken.**

## Berechnung des Schwellenwertes zur Binarisierung des Gradientenbildes

- Iterativ, z.B. Messung der Gradienten bei schwachen Grauwertkanten, die extrahiert werden sollen
- Empirisch, z.B. falls Grauwertunterschied zwischen Nachbarn  $> 8$ , dann Kante
- Automatisch aus dem Bild berechnet, z.B. berechne das Histogramm der Gradienten. Schwellenwert gewählt so dass N% (z.B. 10%) der Gradienten grösser als der Schwellenwert ist, d.h. behalte die N% besten Gradienten



- Berechne die Rauschenvarianz in homogenen Regionen und wähle den Schwellenwert so, dass "keine" Kanten wegen Rauschen detektiert werden
- Obige Verfahren können global oder lokal (in Teilbilder) benutzt werden

# KANTENAUSDÜNNUNG

**Zwei Hauptmethoden:**

- **Bestimmung der Mittelachse (Skelett)**

**Nach Binarisierung der Gradienten, finde Mittellinie der ursprünglichen Kantenregion**

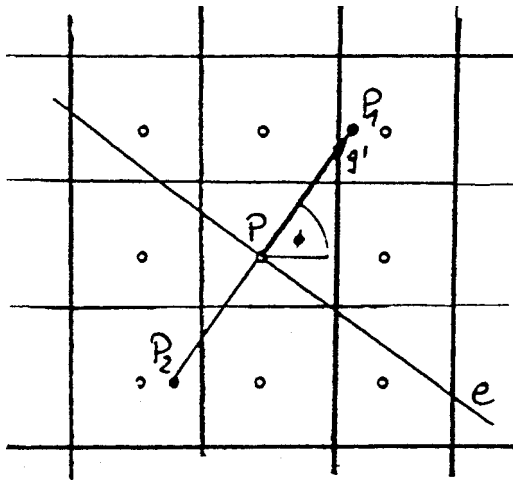
**Ändere Randpixel zu Hintergrund, falls sie keine Endpunkte oder nicht wichtig, um die Kantentopologie beizubehalten**

**Keine gute Kantenlokalisierung, alle Kantenpixel werden gleich behandelt**

- **Non-maxima Suppression**

Wahl der wahrscheinlichsten Kantenpixel vertikal zur Kantenrichtung. Besteht aus zwei Schritten. Für jedes Pixel:

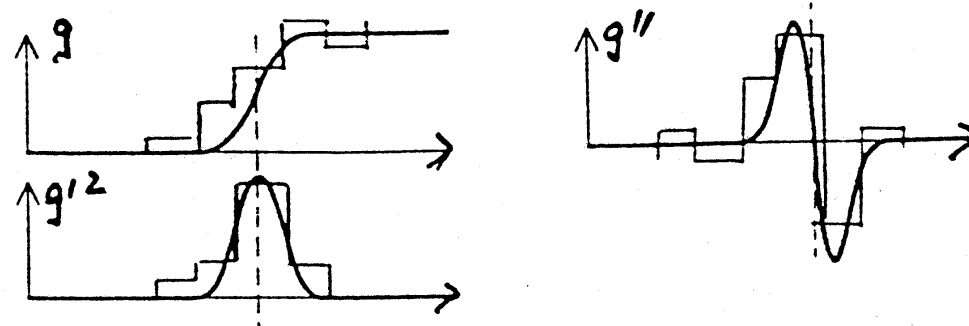
- Wahl der Nachbarpixel in der Gradientenrichtung zum Vergleichen ( $P_1$ ,  $P_2$ )
- Lineare Interpolation der Gradienten an  $P_1$ ,  $P_2$  aus den zwei horizontalen oder vertikalen Nachbar-Kantenpixel und Vergleich der Gradienten an  $P$ ,  $P_1$ ,  $P_2$
- Falls  $P$  kleinere Gradiente als  $P_1$  oder  $P_2$ ,  $P$  wird zu Hintergrund



**Non-maxima Suppression nach Canny, 1983**

**Option - finde Kantenposition mit Subpixel-Genauigkeit => Fit einer Kurve zu den Messungen entlang der Gradientenrichtung, z.B. für verschiedene Eingabedaten**

- **Grauwerte => finde den Krümmungspunkt eines Polynoms 1ten Grades**
- **Gradienten => finde Maximum eines Polynoms 2ten Grades**
- **Zweite Ableitungen der Grauwerte => finde “zero-crossings” eines Polynoms 3ten Grades**



**Subpixel Kantenlokalisierung durch Benutzung der Grauwerte (oben links), Gradienten (unten links), und zweiten Ableitungen der Grauwerte (rechts).**

# VERBINDUNG VON KANTENPIXELN UND SCHLIESSEN VON LÜCKEN

## **Ziel:**

- **Gruppierung aller Pixel, die zur gleichen Kante gehören**
- **Klassifikation der Pixel (End-, Kanten-, Verzweigungs-Pixel)**

## **Prozedur:**

- **Klassifikation basierend auf Anzahl der Nachbar-Kantenpixel**

**1 Nachbar => End-Pixel**

**2 Nachbar => Kanten-Pixel**

**3 oder mehr Nachbar => Verzweigungs-Pixel**

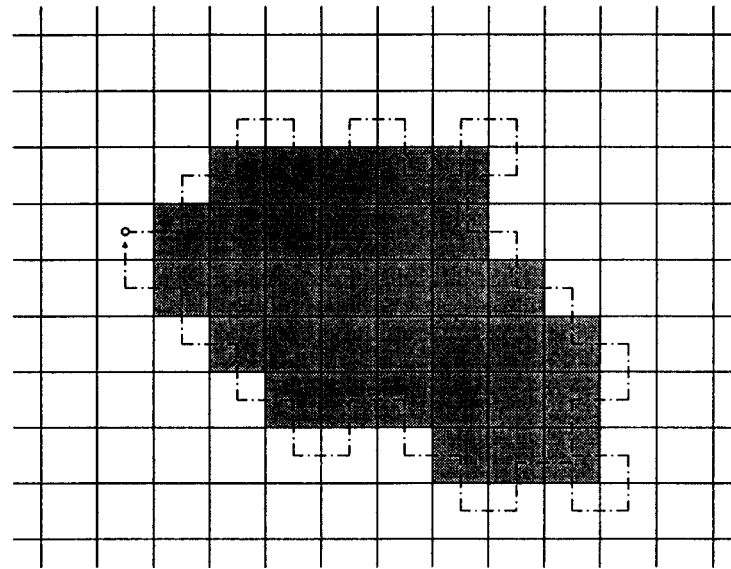
- **Anwendung von Kantenverfolgung in 8er-Umgebung**
- **Für spezielle Fälle wie Geraden**

**Verlange ähnliche Attribute für Nachbar-Kantenpixel, z.B. Gradientenrichtung**

- **Als Option - Überbrückung von kleinen Lücken mit Bildverarbeitungsmethoden,**

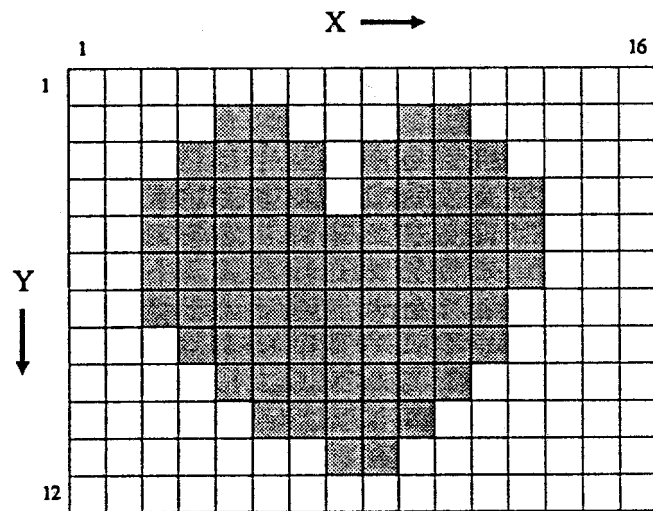
**z.B. mathematische Morphologie, heuristische oder modell-basierte Verfahren**

**Beispiel: Kantenverfolgung (Kombination von Kantenpixeln) in 4er-Nachbarschaft**



**Effiziente Speicherung der Kantenpunkte als Ketten-Code**

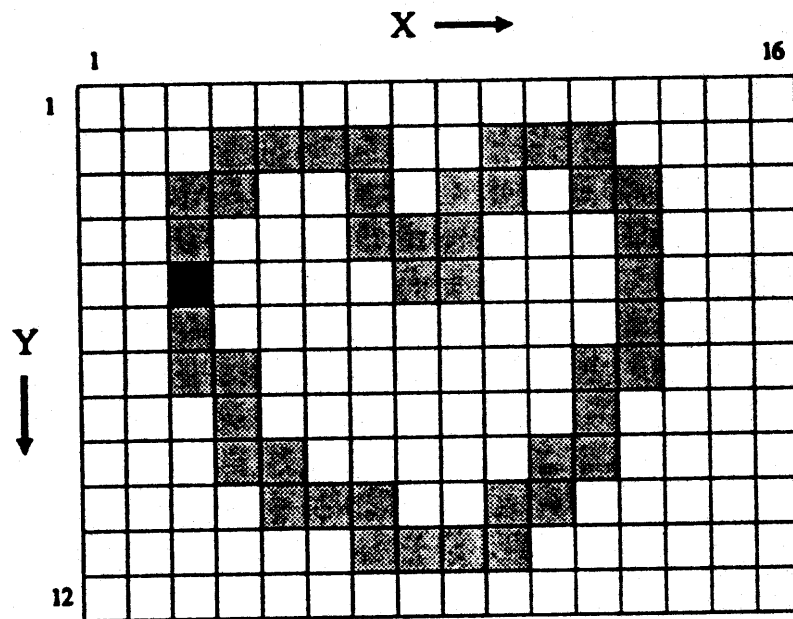
- Kodierung der 8 Nachbar-Richtungen
- Zur Speicherung notwendig (statt x,y Koordinaten aller Kantenpunkte): Anfangspunkt, Anzahl Kantenpunkte, Richtungscode für jeden Kantenpunkt
- Beispiel



Originalbild



# Ketten-Code bei Wahl der 4er- und 8er-Umgebung. Anfangspunkt ist dunkel markiert

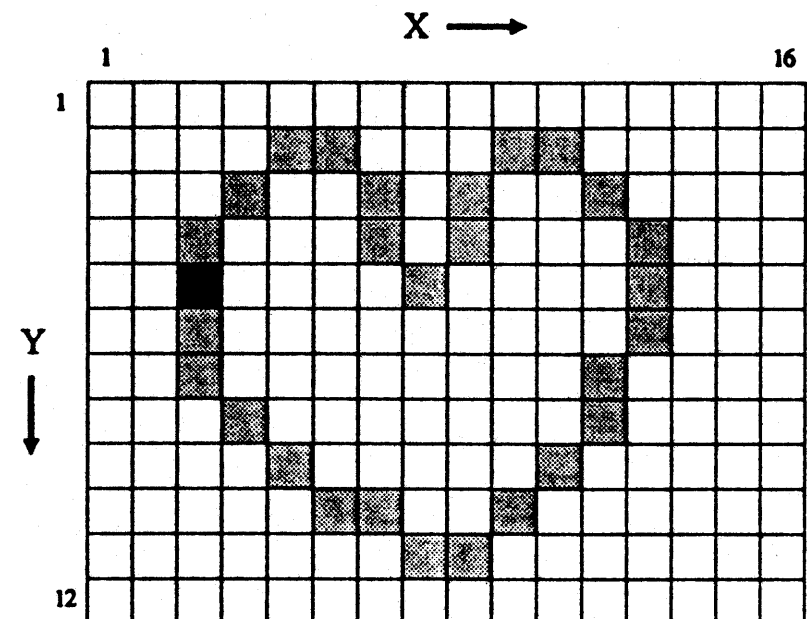


4er-Umgebung. Ketten-Code lautet:

Anfangspunkt:  $x=3, y=5$

Anzahl der Punkte: 44

Richtungen: 2, 2, 0, 2, 0, 0, 0, 6, 6, 0, 6, 0, 2, 2, 0, 2,  
0, 0, 6, 0, 6, 6, 6, 6, 4, 6, 6, 4, 6, 4, 6, 4, 4, 4, 2, 4, 4, 2,  
4, 2, 2, 4, 2, 2



8er-Umgebung. Ketten-Code lautet:

Anfangspunkt:  $x=3, y=5$

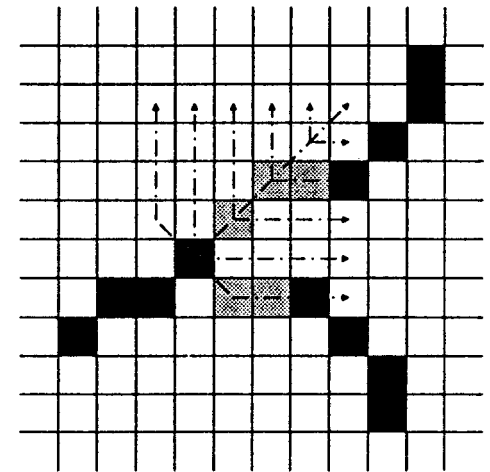
Anzahl der Punkte: 28

Richtungen: 2, 1, 1, 0, 7, 6, 7, 1, 2, 1, 0, 7, 7, 6, 6, 5,  
6, 5, 5, 5, 4, 3, 4, 3, 3, 3, 2, 2

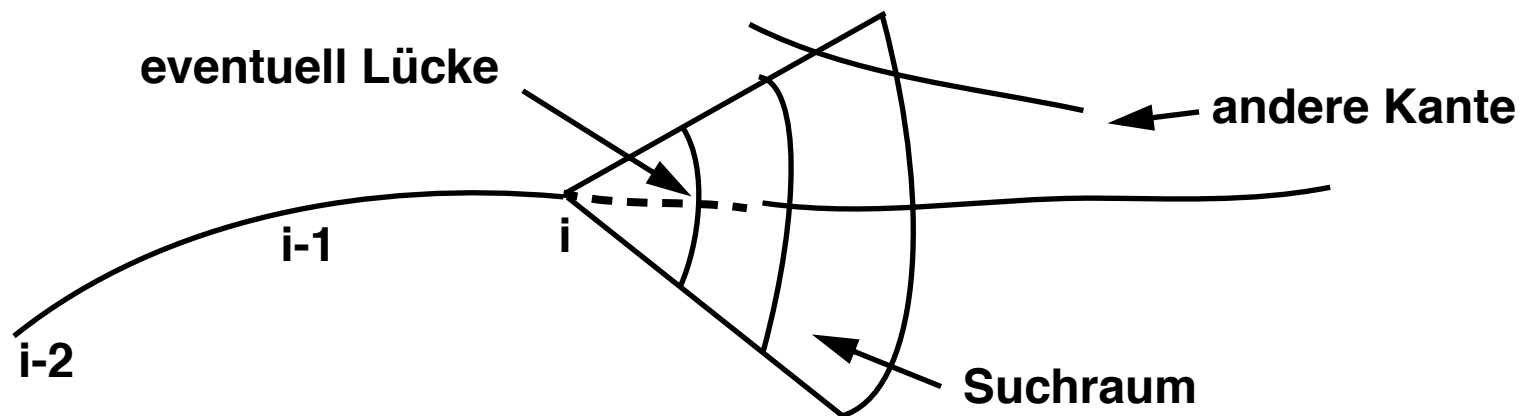
## Schliessen von Lücken

- **Mathematische Morphologie (z.B. Schliessen)**
  - Schliesst Lücken  $< M$  ;  $M$  = Dimension des SE, SE ist symmetrisch
  - Abstand zwischen unterschiedlichen Kanten muss  $> M$  sein, sonst Verschmelzung von Kanten
- **Heuristische Verfahren (auch für Kantenverfolgung), z.B. Lücke soll klein sein, Richtung der detektierten Kante soll nicht viel ändern, d.h. Suchraum = Kreissegment**

Ausgehend von einem Linienendpunkt wurden 11 Suchstrahlen der Länge 4 in einen bezogen auf die Richtung des letzten Linienelements vorwärts orientierten Kegel gerichtet. Es wurden zwei andere Linienendpunkte detektiert und die Verbindungen hergestellt. Die ursprünglich vorhandenen Linienpunkte sind dunkel markiert, die neu hinzugekommenen Linienpunkte etwas heller



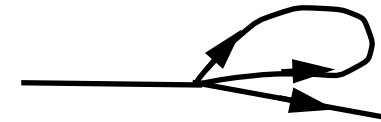
- Ringoperator (für Kantenverfolgung und Schliessen von Lücken)
  - Einen Kreisbogen definieren (oder mehrere in unterschiedlichen Abständen vom letzten Kantenpunkt)
  - Die Grauwerte oder Grauwertgradienten entlang des Bogens untersuchen
  - Mehrere Kreisbögen nützlich, um (a) Lücken zu überbrücken, (b) Verzweigungen erkennen zu können



- Einfacher: Statt Kreisbögen, Geraden vertikal zur prädizierten Kantenrichtung benutzen

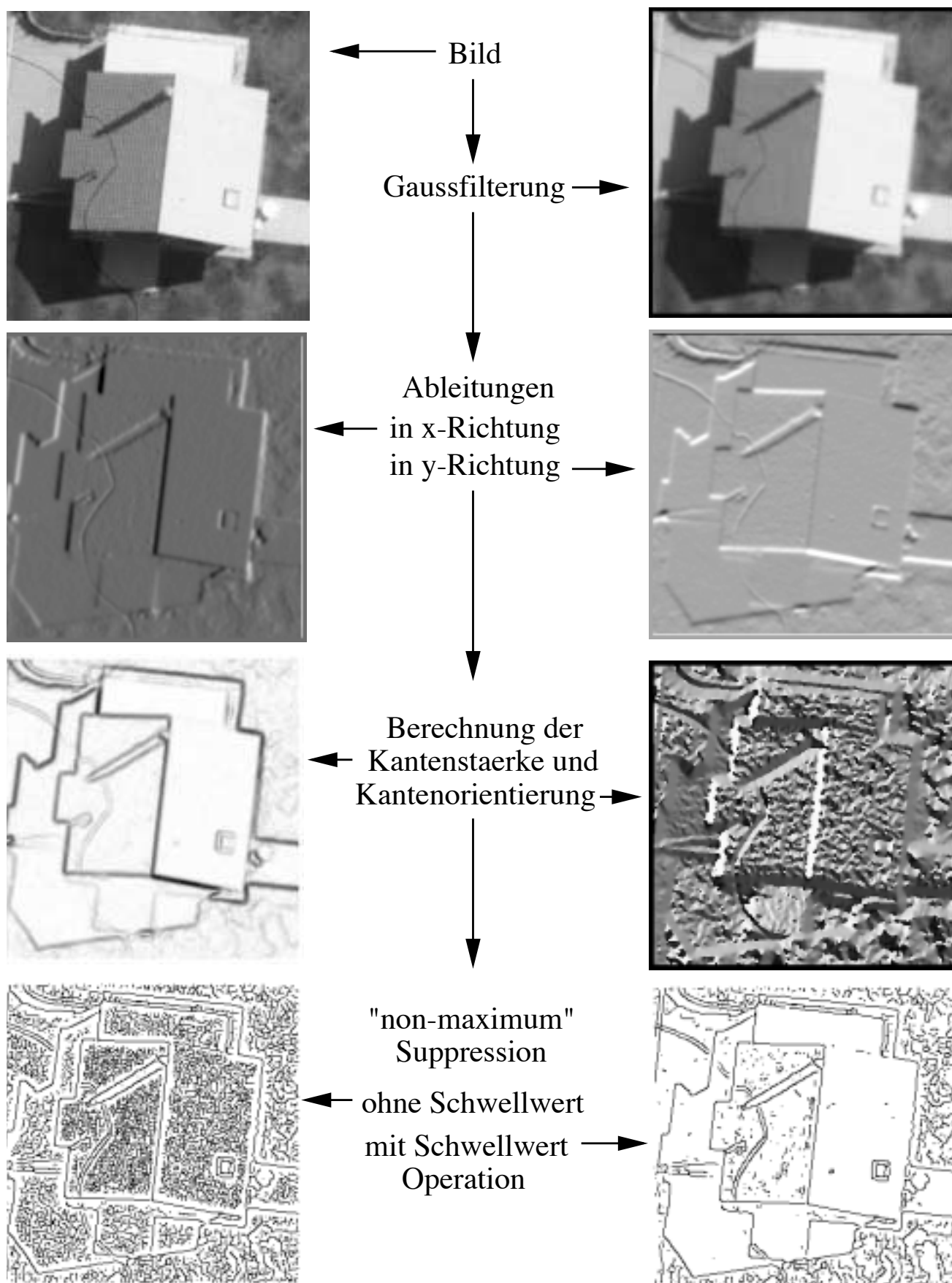
## Probleme der Kantenverfolgung (und allgemein der Kantendetektion)

- **Schwacher Kontrast -> Lücken**
- **Kanten nah aneinander (parallel oder Kanten, die an unterschiedlichen Objekten gehören)**
- **Verzweigungen**
  - **Werden sie erkannt? Wie reagiert der Algorithmus?**
  - **Wird 2maliger Durchlauf von einem Ast erkannt?**
- **Störungen (Schatten, andere Objekte, wie Bäume etc.)**
- **Helligkeits- und Kontraständerungen entlang der Kante**
- **Einflüsse wegen Rauschen, z.B. viele kurze Kanten**

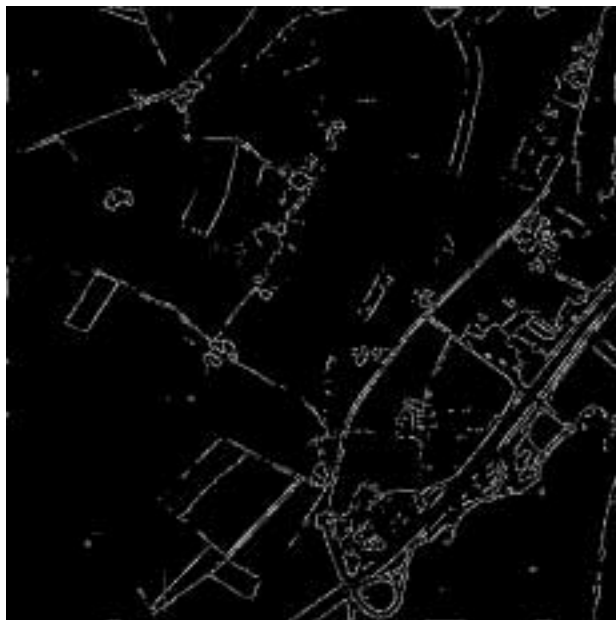
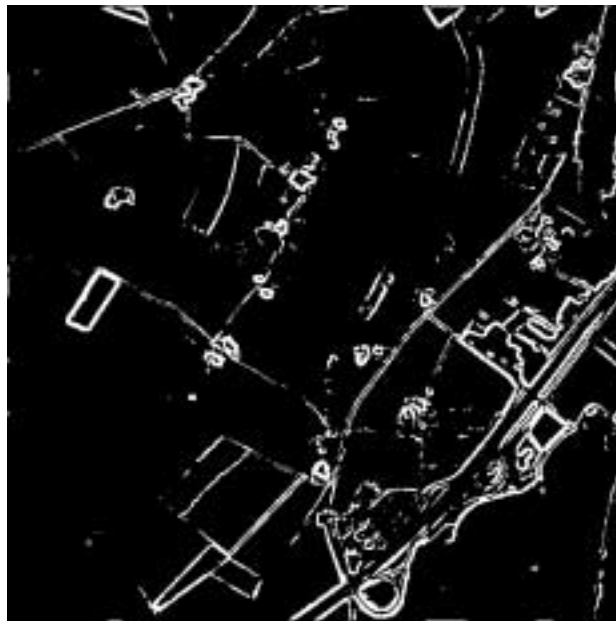
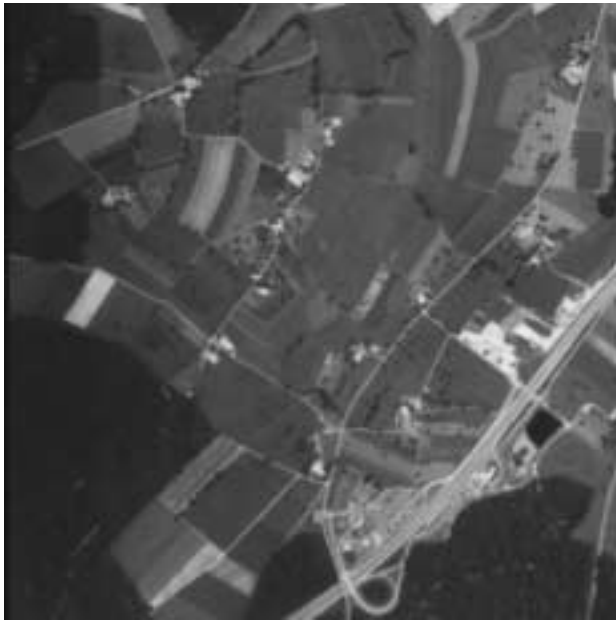


**Als Nachverarbeitung können solche Kanten eliminiert werden**

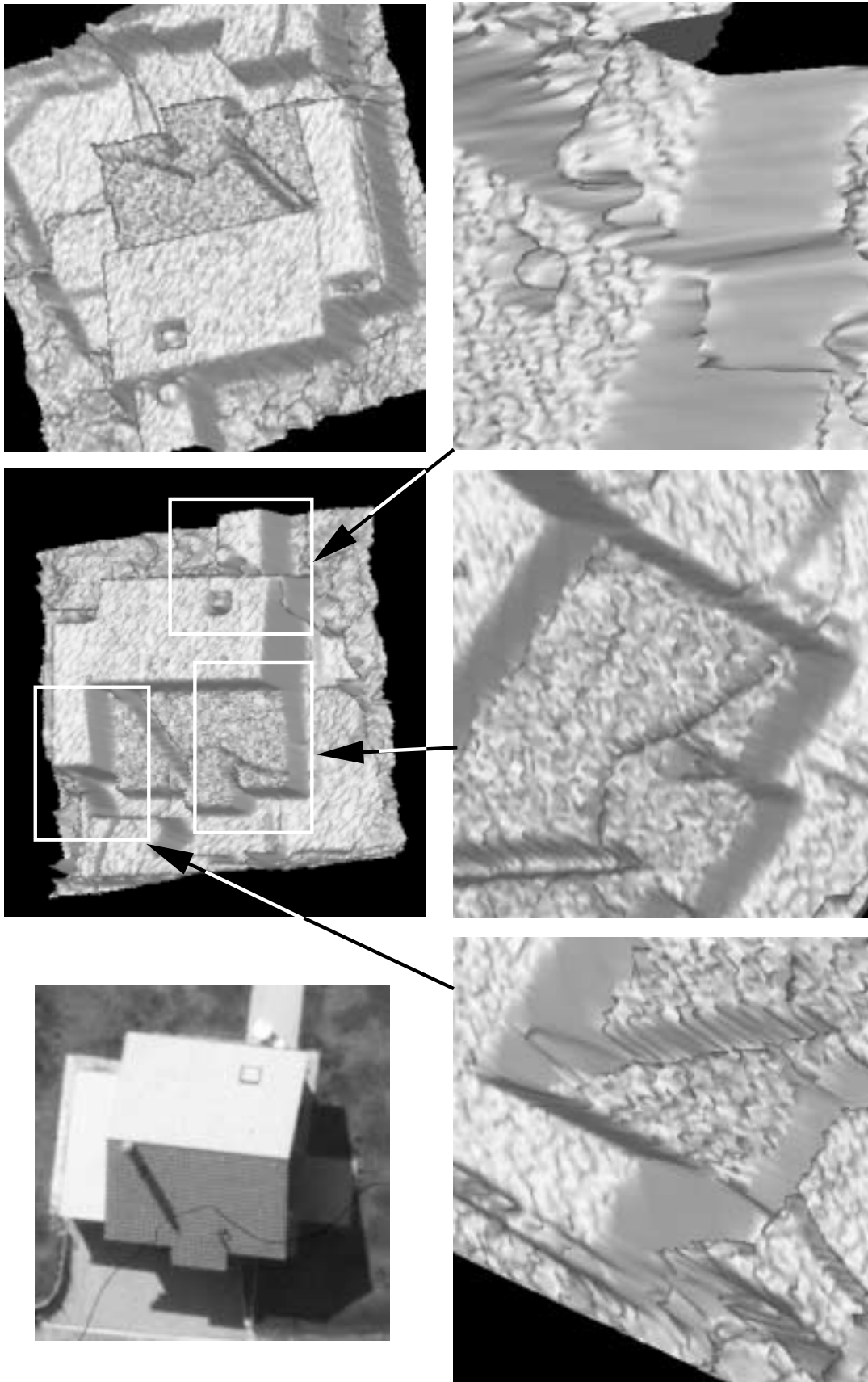
# BEISPIELE UND PROBLEME DER KANTENDETEKTION



Praktische Implementation der Kantendetektion in 2-D



**Zeilenweise von oben links: Originalbild, binarisiertes Bild mit den Sobel-Gradienten, wie früher aber die Kanten sind auf 1 Pixel ausgedünnt, Gruppierung von Kantenpixeln zu Kantensegmenten und Verbindung von unterbrochenen Kantensegmenten (der Algorithmus benutzt die ursprünglichen Gradienten vor der Binarisierung, um die richtigen Verbindungen zu schätzen). Beispiel hergestellt mit dem Public-Domain Paket cantata von Khoros.**



**3-D Darstellung der Grauwerte als Oberfläche. Rechts drei vergrößerte Ausschnitte. Die Probleme mit der Definition der Kanten und dem Rauschen sind sichtbar.**

## **APPROXIMATION / MODELLIERUNG DER KANTEN**

**Nach der Kantenverfolgung, können die Kantenpixel via Kleinste-Quadrate-Funktionen approximiert werden:**

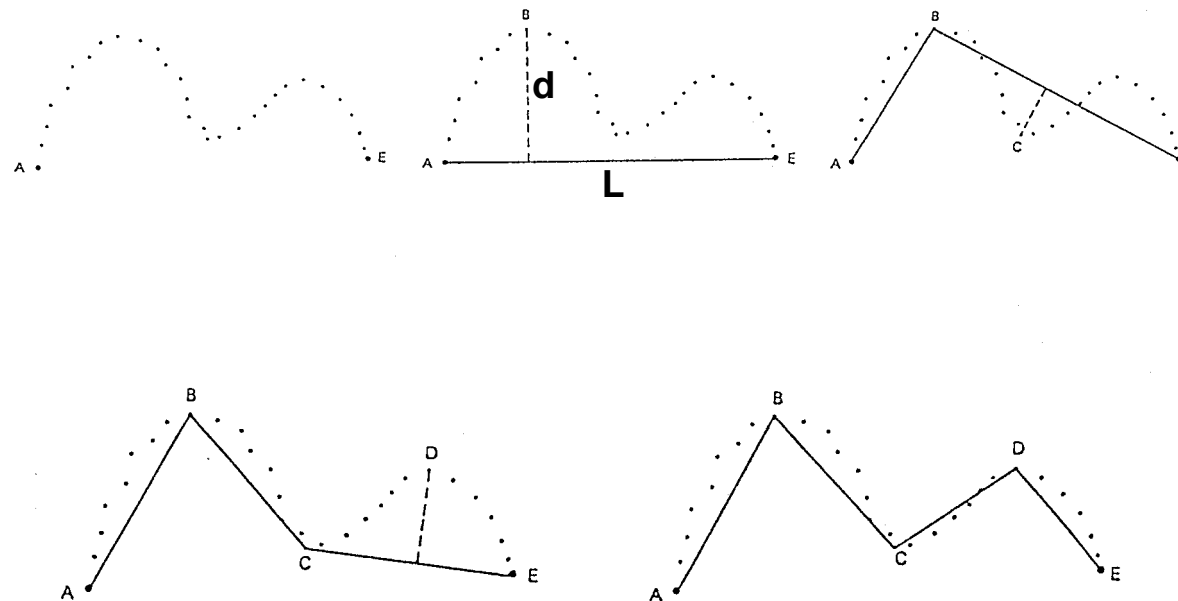
**1. Polynomapproximationen (Geraden, Polynome niedrigen Grades)**

**2. Spline-Approximationen**

**Beispiele zu 1:**

**A. Split Algorithmus für Geraden-Approximation (Pavlidis, 1982, Bässmann und Besslich, 1989, Ramer, 1972)**





- Beginne vom ersten und letzten Kantenpixel (Länge  $AE = L$ )
- Finde den Punkt B mit der maximalen Distanz d von der Linie
- Wenn  $d/L$  (repräsentiert Krümmung) oder  $d > \text{Schwellenwert } T \Rightarrow B$  wird neuer Knoten der Polygonapproximation, sonst stoppe mit diesem Liniensegment
- Verarbeite ähnlicherweise alle Liniensegmente bis keine weitere Teilung in kleineren Liniensegmenten möglich ist

**Für andere Algorithmen über lineare Approximation von 2-D Kurven s. Pavlidis, 1982, Dunhan, 1986, Bässmann und Besslich, 1989**

**B. Fit einer Geraden zu den Kantenpixeln  $x_i$  ,  $y_i$  mit kleinsten Quadraten**

$$y_i + v_{y_i} = a + b \cdot (x_i + v_{x_i}) \Rightarrow \|v_x\|^2 + \|v_y\|^2 = \min$$

### **3.3. EXTRAKTION VON FLÄCHEN, BILDSEGMENTIERUNG**

**1. Allgemein, z.B. teile das Bild in nicht überlappende homogene Regionen**

**2. Spezifisch (thematisch orientiert), z.B. extrahiere alle Wasserflächen**

**Aufgabe der Klassifikation (s. auch multispektrale Klassifikation in Fernerkundung)**

**Methoden zu 1.**

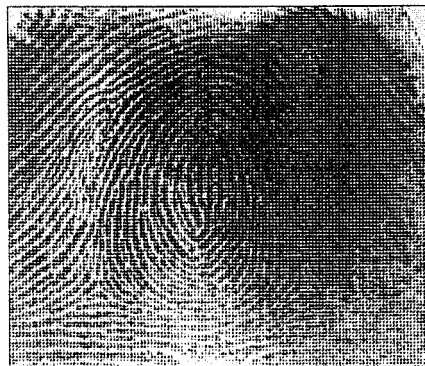
- **Benutzung von extrahierten Kanten (eventuell Schliessung der Kanten -> geschlossene Konturen)**
- **Benutzung von statistischen Grössen der Grauwerte, um homogene Regionen zu finden**
- **Benutzung von anderen Attributen (statt Grauwerten), z.B. Farbinformation, Textur, Kontext, Oberflächenmodelle etc.)**
- **Kombination mehrerer Attribute**

# FLÄCHENEXTRAKTION BASIEREND AUF DIE GRAUWERTE

**“Density-Slicing” - einen Schwellenwert für jede Grauwertregion bestimmen**

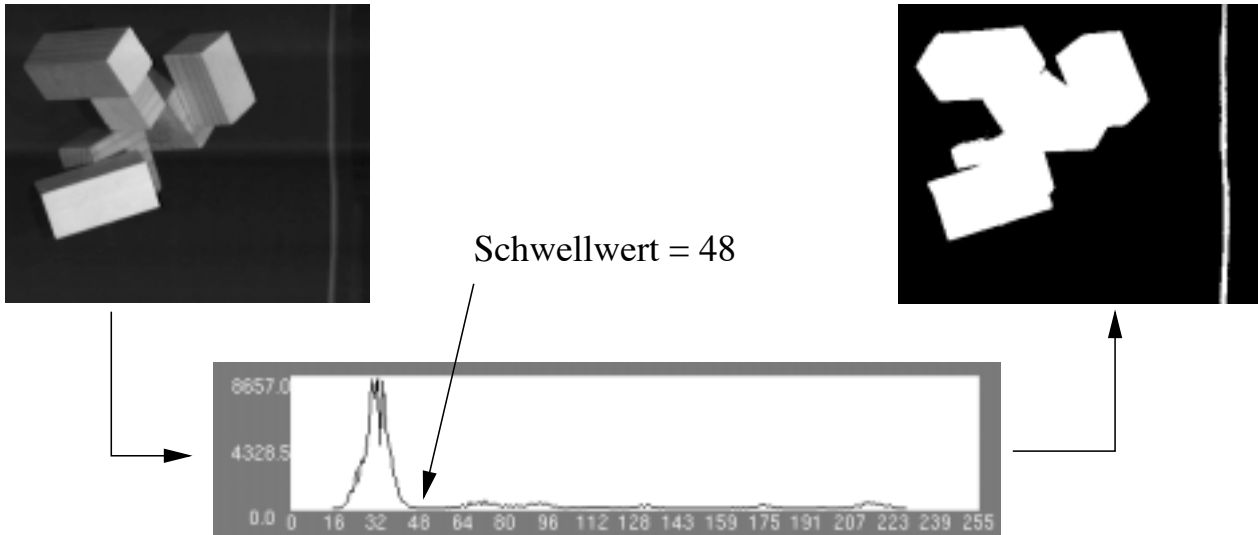
- **Schwellenwerte werden oft mit Hilfe des Histogramms (= Gebirge und Täler) bestimmt -> Schwellenwerte werden bei Tälern gewählt**
- **Oft Annahme -> jede Region im Histogramm hat eine gaussische Form (um überlappende Grauwertgebirge richtig zu trennen)**
- **Der Schwellenwert kann auch adaptiv/lokal berechnet werden (s. Scanner)**
- **Methode anwendar nur für einfache Bilder mit wenigen Grauwerten, die gut von einander getrennt sind**

**Original**

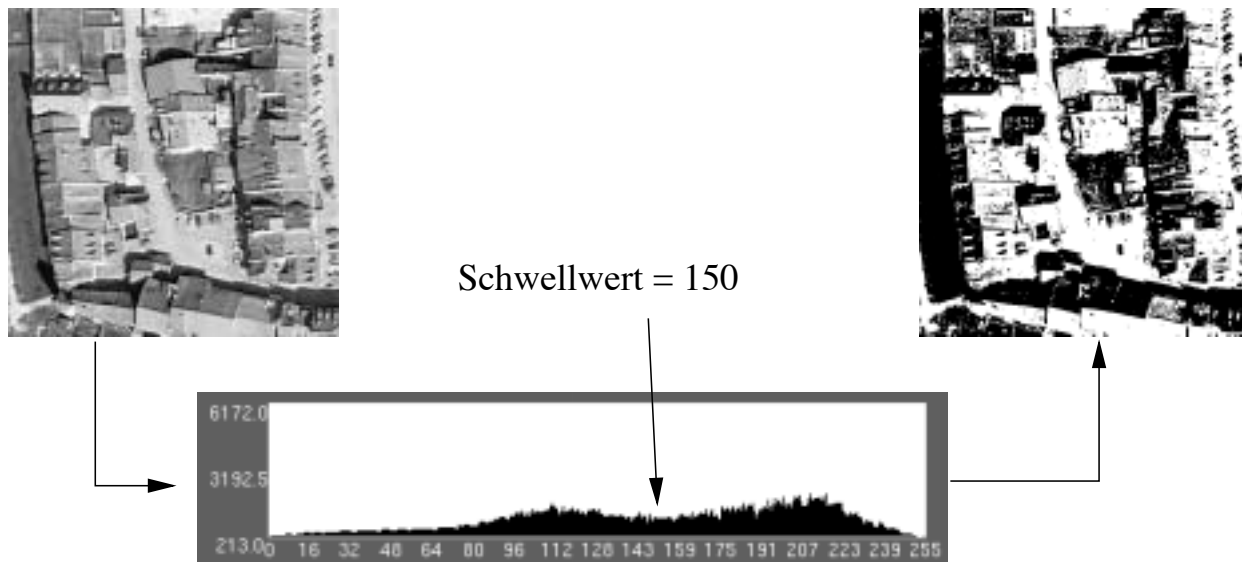


**Binarisiert mit adaptivem Schwellenwert**

(A)



(B)



**Binarisierung durch Schwellenwertoperationen mittels eines Histogramms. (A) Einfache Trennung zwischen Vorder- und Hintergrund. (B) Ein Beispiel, wo es nicht geht.**

## Region Growing

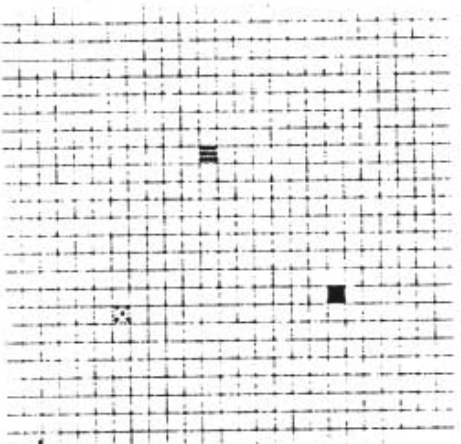
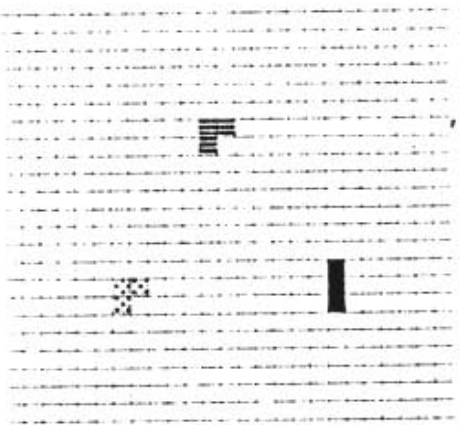
- Wahl von Keimpunkten (Saatzellen), d.h. Region-Initialisierung
- Wahl der Keimpunkte interaktiv oder automatisch. Automatische Bestimmung:
  - viele Punkte, regelmässig verteilt, Überprüfung ob sie auf Kanten liegen -> falls ja, Punkte löschen
  - andere Vorverarbeitungsschritte, z.B. Zentren von Nachbarschaften mit kleiner Grauwertstandardabweichung, oder kleinen Gradienten
- Untersuchung der Nachbarn von jedem Keimpunkt
  - Gehört ein Nachbar  $g$  einer Region? (Einheitlichkeitskriterium)  
z.B. ja, wenn  $\text{abs}(g - \text{Mittelwert der Region}) < \text{Schwellenwert}$  (er kann auch von der Standardabweichung der Region abhängen)
  - Wenn ein Pixel keiner benachbarten Region gehört -> neuer Keimpunkt
- Statistische Grössen der Region neu berechnen

- **Wenn fertig (alle Pixel gehören einer Region) -> eventuell ähnliche benachbarte Regionen zusammenfassen (“Merging”). Notwendig, wenn Anzahl Keimpunkte zu hoch gewählt**
- **Ergebnis abhängig von der Anzahl und Position der Keimpunkte, dem Einheitlichkeitskriterium, aber auch von Startpunkt und Richtung der sequentiellen Verarbeitung -> z.B. andere Ergebnisse wenn das Bild um 90° rotiert wird**
- **Vorverarbeitung mit Bildrestaurationsalgorithmen verbessert das Resultat**
- **Hierarchische Implementation des Verfahrens mit Bildpyramiden vorteilhaft**

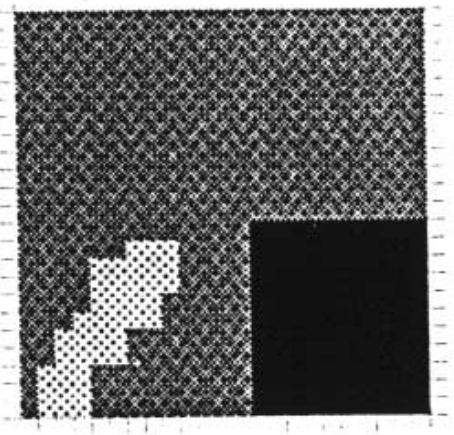
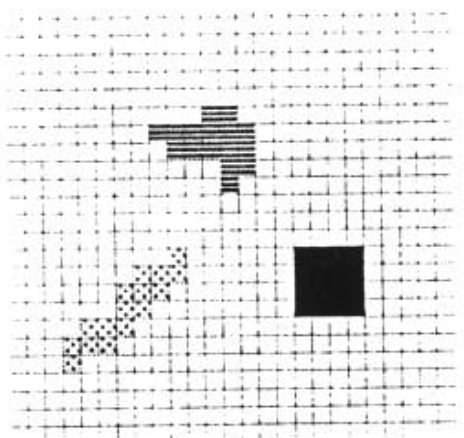
### **Andere Algorithmen**

- **Punkte in homogenen Regionen bestimmen (mit Filtermasken, Förstner Operator, Kantendetektoren etc.)**
- **Benachbarte Punkte in Regionen kombinieren**

Satzzellen



Region Growing



Segmentation

Das Prinzip des Region-Growing