

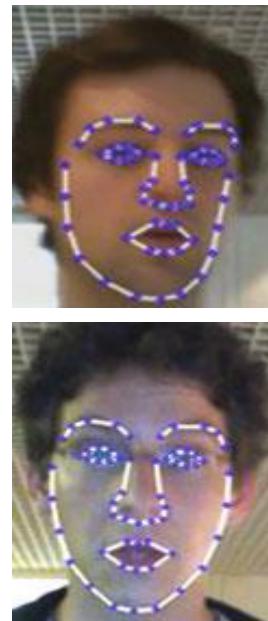
Bildanalyse

- Information automatisch aus Bildern generieren

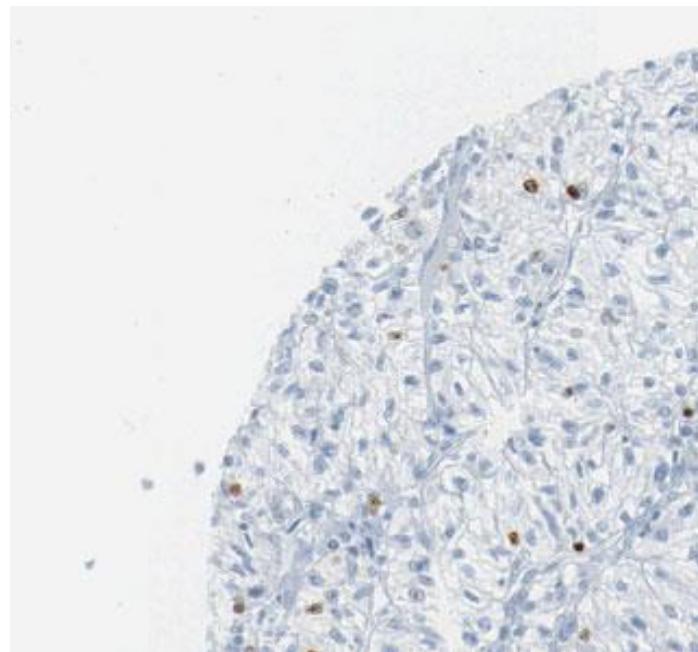
Echtzeit 3d Erkennung



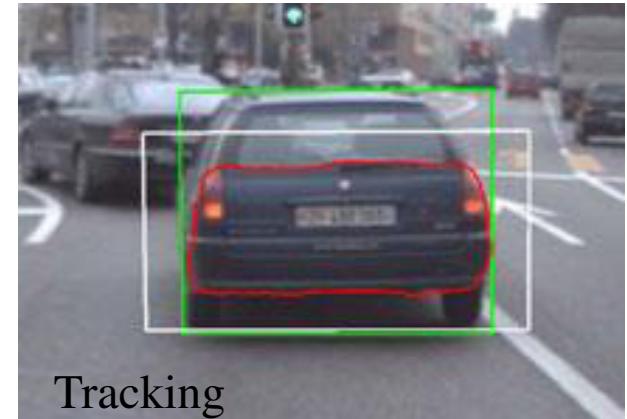
Gesichtserkennung



Erkennung von Tumorzellen



Tracking



Bewegungserkennung



Segmentierung aus
Bewegung



Wieso ist das wichtig?

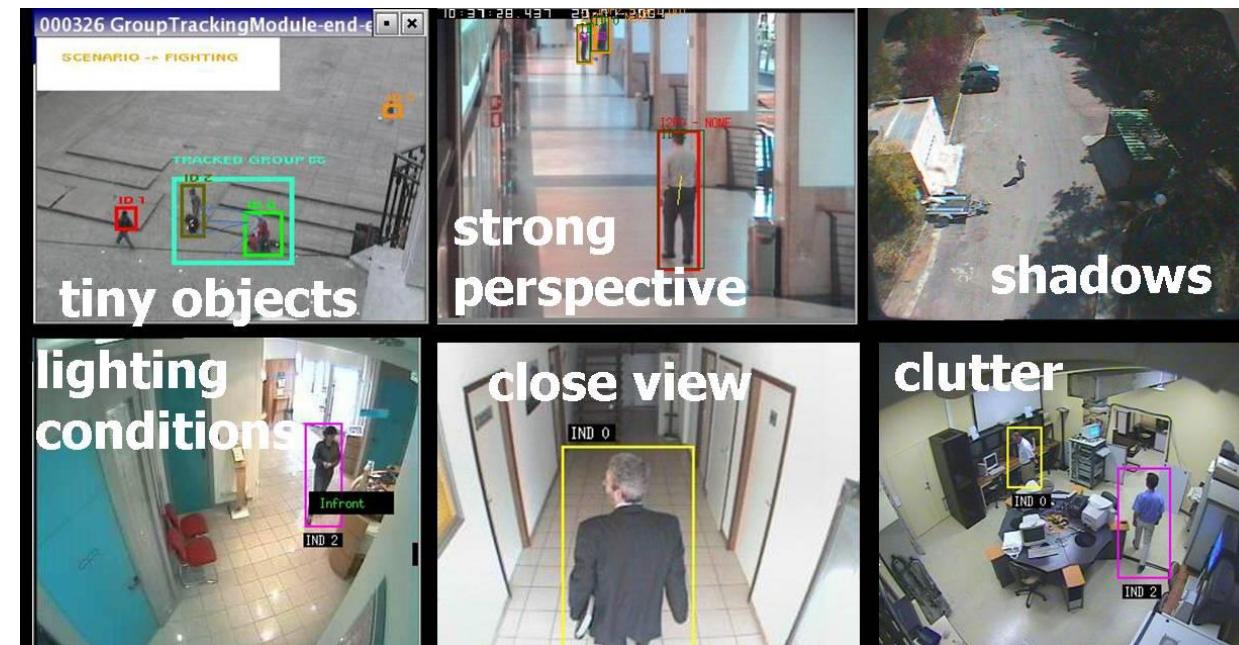
- Menschen nehmen einen sehr großen Teil ihrer Umwelt über Bilder wahr
- alltägliche Entscheidungen auf der Basis von Bildern sind z.B.
 - wie sich im Straßenverkehr verhalten
 - welches Produkt zu kaufen
 - sich nach einer Gebrauchsanleitung verhalten
- andere, wichtige Entscheidungen sind z.B.
 - Diagnose von Krankheiten
 - Qualität von Produkten oder Bauteilen

Wieso Bildanalyse?

- Viele Entscheidungen sind schneller, präziser und/oder zuverlässiger zu treffen, wenn sie automatisiert werden, z.B.
 - Fahrerassistenzsysteme im Automobilbau
 - Qualitätssicherung in der Fertigung
 - Überwachung in sicherheitskritischen Orten (Flughäfen, Fussballstadien,...)
 - Diagnose und Therapieunterstützung in der Medizin
 - Exploration in der Geologie
- Problem: was ist die Information, die in den Bildern steckt, und wie kann sie gefunden werden?

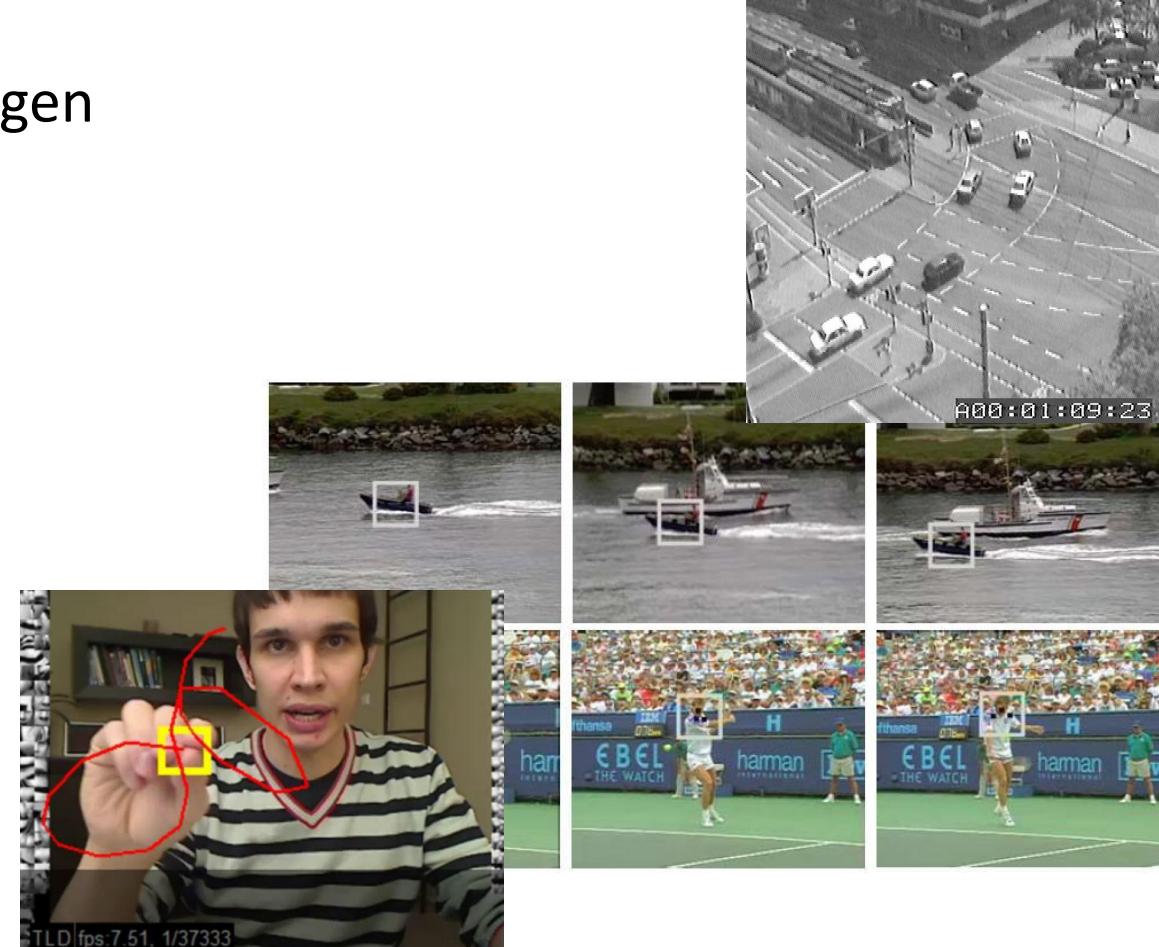
Beispiele: Objektdetektion

- Ziel: automatische Kontrolle kritischer Situationen
 - Menschen
 - Fahrzeuge
- Beispiele
 - Kollisions-verhinderung
 - Personen-kontrolle



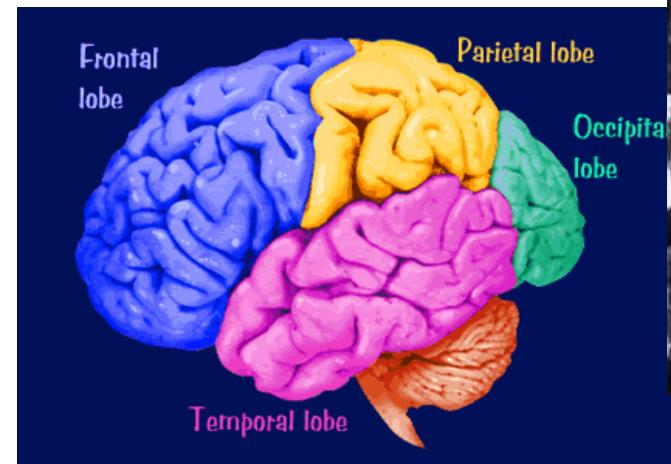
Tracking

- Ziel: Situationsentwicklung verfolgen
 - Menschen, Fahrzeuge, Gesten
- Beispiele
 - Gestenerkennung
 - Erkennung von Stimmungen
 - Personen verfolgen
 - Fahrzeuge verfolgen
 - Content Tracking



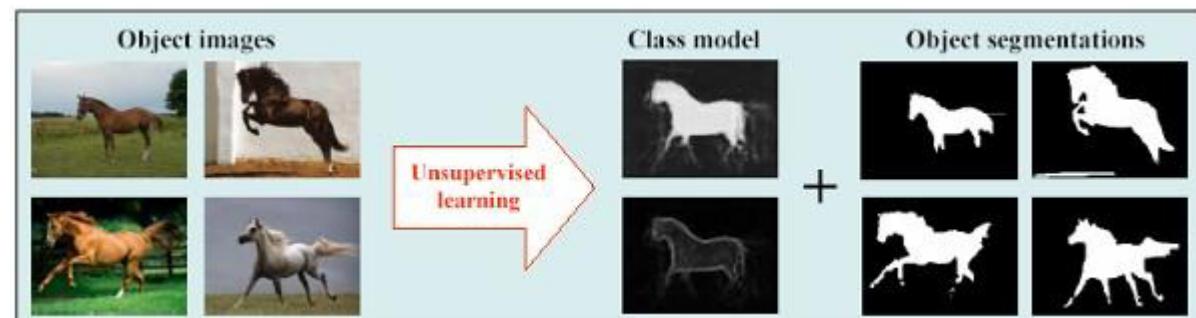
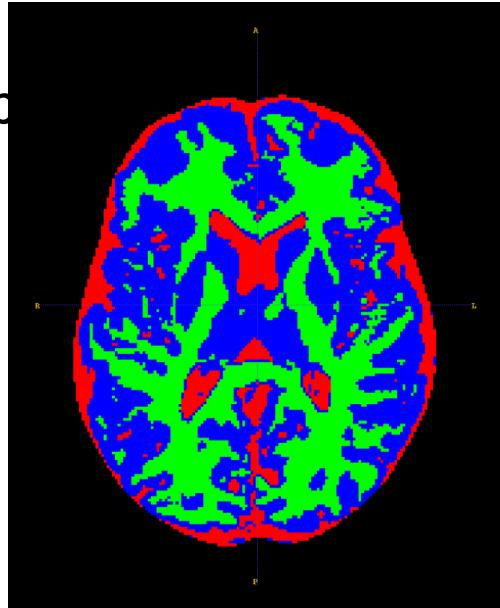
Objekt Identifikation

- Ziel: Individuen identifizieren
 - Personen, Werkstücke, ...
- Beispiele
 - Personenidentifikation
 - Astronomie
 - Medizin



Objektextraktion

- Ziel: interessante Objekte finden, Pixel von Objekten gruppieren
- Beispiele
 - Merkmale für Bildersuche generieren
 - Vorbereitung von Klassifikation in Bildern
 - Medizin



Bildanalyse ≠ Bildbearbeitung!

- Bildbearbeitung (z.B. mit Adobe Photoshop)
 - hat die Verbesserung eines Bilds zum Ziel
 - was Verbesserung ist, wird durch *Bearbeiter/in* beurteilt
 - Entscheidung über den Einsatz der Methoden durch Bearbeiter/in
- Bildverarbeitung und Bildanalyse
 - *Algorithmus* beschreibt, was Bildinformation ist
 - erfordert exakte Definition, was diese Information ist
 - erfordert validierbaren Algorithmus zu Extraktion dieser Information

Grundlagen der Bildverarbeitung

- Vorstufe für Analyseaufgaben
 - Störungen aus Bildern entfernen
 - Wahrnehmbarkeit von Inhalten verbessern
 - Bild in Komponenten zerlegen
- Was werden Sie lernen?
 - Beschreibung und erste Methoden zu Information in Bildern
 - Benötigte algorithmische und mathematische Methoden dazu

Information in Bildern

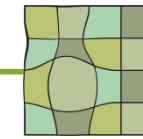
- Ein Bild kann viele Bedeutungen haben:
 - Sind Menschen abgebildet?
 - Wo sind Fahrzeuge zu sehen?
 - Ist Mr. X auf dem Bild?
 - Wie spät ist es?
 - Wo ist die Aufnahme gemacht worden?
- Erste Schritt einer Bildanalyse
 - Genaue Spezifikation der gesuchten Bedeutung



Informationen in Bildern

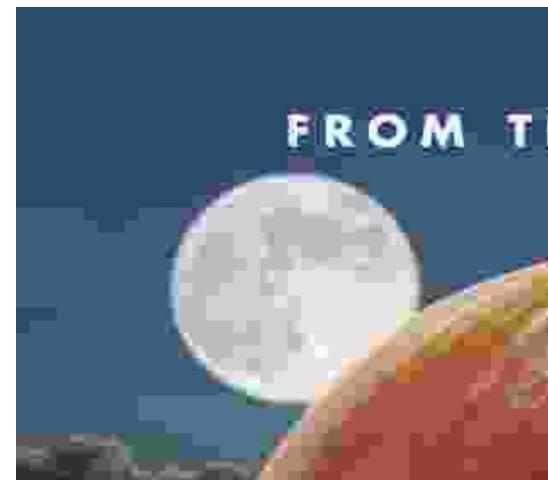
- Information in Bildern ist nicht leicht zu finden
- Beispiel Videokompression
 - Datenmenge DVD Video pro Bild = 720x576 RGB Pixel = ca. 1,2 MB pro Bild
 - 24 Bilder/sec, DVD Speicherkapazität ca. 8.5 GB
 - unkomprimiert können ca. 7000 Bilder, d.h. 5 Minuten Film gespeichert werden
 - um einen Film auf DVD zu speichern, muss ca. 97% der Daten weggeworfen werden (nur wesentliche Information bleibt)





Aufgaben

- Definition von „wesentlich“
- Methode finden, die die Information aus dem Bild „heraustrennt“
- Effiziente Repräsentation für die Speicherung der übriggebliebenen Information
- ...und das muss alles immer funktionieren (d.h., ohne dass man das Resultat jedes Mal überprüfen kann)



Wie macht man das?

- Kreativität und Disziplin
 - Ideen entwickeln, welche allgemeinen Merkmale für bestimmte Bedeutungen in Bildern zutreffen
 - mittels eines geeigneten Modells in einen Algorithmus umsetzen
- Werkzeuge
 - Algorithmen und Programmiersprachen
 - Mathematik



Tracking people in broadcast sports

Formale Beschreibung

- Fragestellung: Welche Bedeutung ist einem gegebenen Bild zuzuordnen
- Formale Problembeschreibung:
 - Bedeutung m produziert ein Bild b : $b=h(m)$
 - Suche inverse Funktion h^{-1} , so dass m aus b gewonnen werden kann
- Aufgaben
 - Bild störungsfrei machen: „ideales“ h finden
 - Eindeutige Umkehrung von h definieren

Die semantische Lücke

Bild

- Helligkeit
- Farbe
- Musterung



Interpretation

- Name (Klasse)
- Existenz/Nichtexistenz von Objekten
- Eigenschaften von Objekten



„Ente (2CV)“

gleiche Bedeutung

Der Sprung über die Lücke

- Die gesuchte Funktion h muss die Pixel auf die Bedeutung abbilden
 - ähnliche Bedeutungen im Bild müssen ähnliche Resultate von h produzieren
- Wenn das (für manche Fälle) gelingt,
 - dann ist das Bildverarbeitungsverfahren generell für diese Fälle anwendbar (also nützlich)
 - dann weiß man für diese Fälle, was den Informationsgehalt ausmacht
 - dann ist das Verfahren vorhersagbar
- Welche Information wird benötigt?

Bildinformation auf Pixelebene

- Charakteristische Farbe und Helligkeit
 - aber: Variation durch Beleuchtung
- Charakteristische Musterung (Textur)
 - aber: Variation durch Skalierung und Projektionswinkel



Bildinformation: Merkmale

- Merkmale, die den abgebildeten Objekten angehören
 - Ecken und Kanten
 - Regionen
- gut geeignet für
 - Tracking-Aufgaben
 - Kombination zu Merkmalen mit mehr Semantik



Bildinformation: Kontext

- Bilder lassen sich nur im Kontext interpretieren (A-Priori-Wissen)
 - Bedeutungskontext (Bildklasse, erwartete Objektformen)
 - Anfragekontext
 - Räumlicher Kontext (Relation zu andern Objekten)
 - Zeitlicher Kontext (Bildfolge)
- Bild- und Kontextinformation müssen redundant sein, um auch bei Störungen oder fehlenden Objektdetails interpretierbar zu sein.



Informationsträger

- Es sind Pixelgruppen, die die Information tragen
 - wie findet man die Gruppen
 - Segmentierung des Bildes
 - Unterschied zwischen Segmenten und Objekten
 - nicht alles in einer Pixelgruppe ist Information
 - Störungen (Rauschen, Artefakte)
 - für die Fragestellung irrelevante Information
 - dieselbe Information kann unterschiedlich abgebildet sein
 - Suche nach Invarianten (z.B. was charakterisiert ein Auto)

Bildverarbeitung

Bildverarbeitung ≠ Informatik

- Wissen über die Bedeutung in Bildern stammt aus
 - Signalverarbeitung (Elektrotechnik)
 - Kognitionspsychologie
 - Neurobiologie
- Wissen über die kontrollierte Verarbeitung stammt aus
 - Signalverarbeitung (Elektrotechnik)
 - Algorithmik (Informatik)
 - Algebra und Numerik (Mathematik)

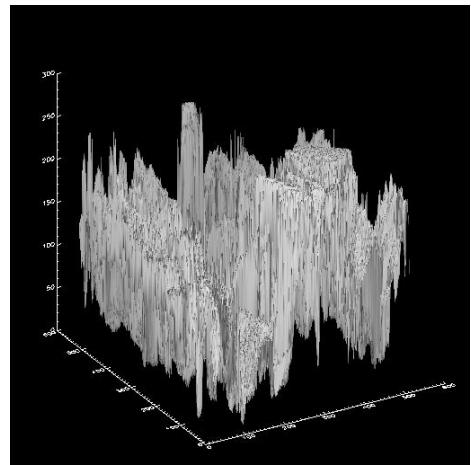
Bsp Mathematik: Lineare Operatoren

- Bildkompression (DVD-Beispiel): Bild wird als Vektor \mathbf{b} von Pixeln betrachtet
- Komprimierbares Bild ist ein neuer Vektor \mathbf{k} , bei dem an den ersten Stellen die relevante Information steht
- \mathbf{k} kann durch Matrixmultiplikation von \mathbf{b} mit einer invertierbaren Matrix \mathbf{H}_{komp} erzeugt werden: $\mathbf{k} = \mathbf{H}_{komp} \times \mathbf{b}$
- Kompression: \mathbf{k} berechnen und unwichtigen Vektorelemente wegwerfen.
- Dekompression: Auffüllen der fehlenden Stellen von \mathbf{k} mit Nullen und Multiplikation mit der Inversen von \mathbf{H}_{komp}



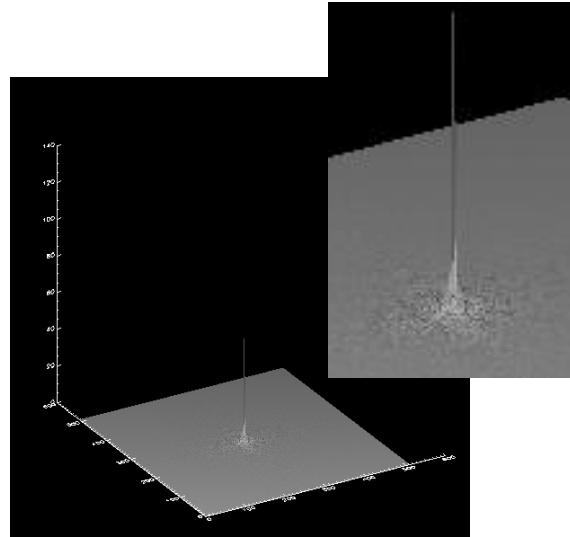
Originalbi
ld

Geht das?



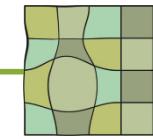
als Funktion
 $grauwert(x,y)$

alle Werte < 0.01 gelöscht
(80% der Werte!) und multipliziert
mit \mathbf{H}_{FT}^{-1}

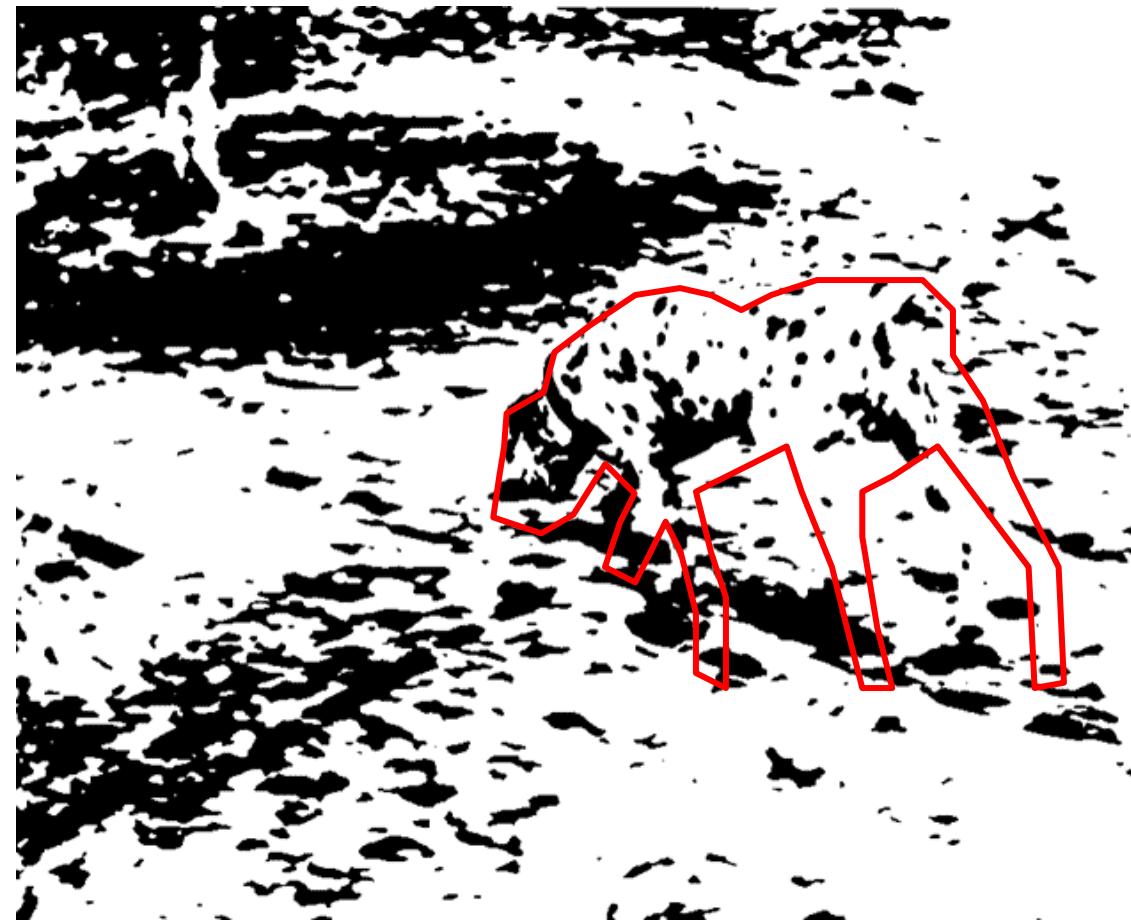


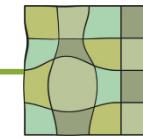
multipliziert mit einer
invertierbaren Matrix



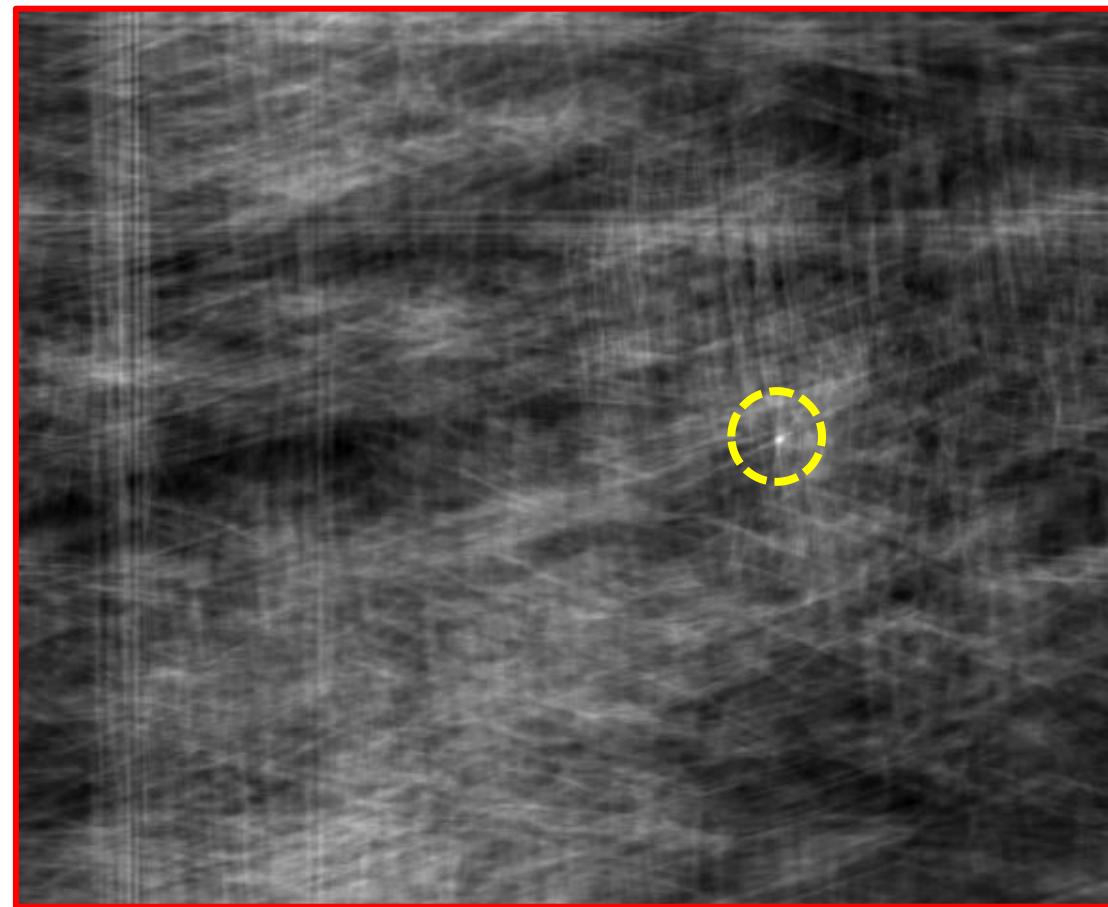
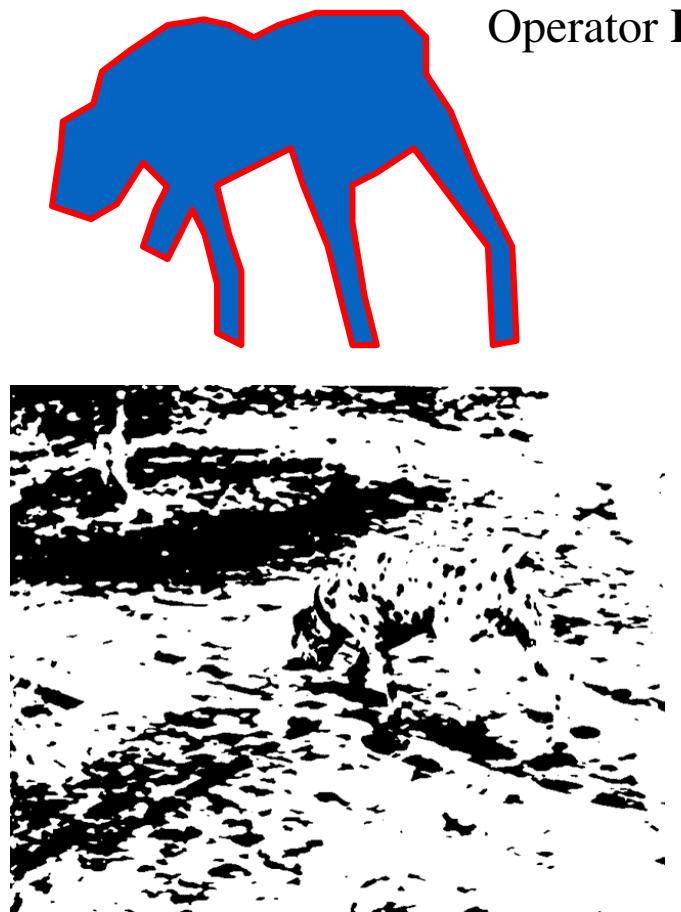


Lineare Operatoren II: Das Einbringen von Kontextwissen





Modellgestützte Interpretation



Der Unterschied zwischen den beiden Operatoren

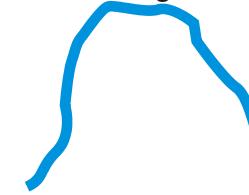
- „Kompressionsoperator“
 - der lineare Operator ist für alle Bilder gleich
 - Bedeutung ergibt sich „aus dem Bild heraus“
 - Komplexere Bedeutungen werden erst nachher eingebracht
- Hundesuchoperator
 - ein Modell des gesuchten Hundes wird gebraucht
 - Bedeutung ergibt sich aus dem Vergleich der Erwartung mit dem Bildinhalt

Bottom-Up vs. Top-Down-Strategie

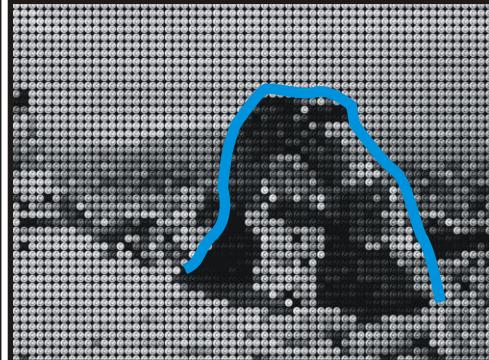


von den Daten zum Modell

Modell des Bogens



vom Modell zu den Daten



Modelle

Modellierung beschreibt Umwandlungen, Störungen und Verluste bei der Abbildung eines Objekts

Modellierung

- ...des Bildentstehungsprozesses: Bildrestauration
- ...der Wahrnehmbarkeit: Bildverbesserung
- ...der Informationsträger: Segmentierung
- ...der Objektzuordnung: Klassifikation

Bildverarbeitung und Bildanalyse

- Bildverarbeitung:
 - Trennung von Information und Artefakten (Störungen).
 - Speicherung, Kompression und Transfer von Bildern
 - Verbesserung der Wahrnehmbarkeit von Bildern.
- Bildanalyse/Bildverstehen
 - Identifikation von Bedeutung tragenden Merkmalen.
 - Berechnung von Merkmalswerten.
 - Zuordnung von Bedeutung.

Was sollten Sie mitbringen?

- Neugier
 - wie Bilder „funktionieren“.
 - wie man das algorithmisch beschreiben kann.
- Kreativität und einen disziplinierbaren Basteltrieb.
- Keine (allzu große) Angst vor der Mathematik.
- Programmierkenntnisse.

Was sollten Sie heute gelernt haben?

- Digitale Bilder = Information aus Pixeln
- Was ist Bildanalyse und Bildverarbeitung
- Modelle, Top-Down- und Bottom-Up-Ansatz
- Prozesse der Bildverarbeitung
 - Bildrestaurierung, -verbesserung, Segmentierung, Klassifikation

Literatur

Deutschsprachig:

- K.D. Toennies, *Grundlagen der Bildverarbeitung*, Pearson, 2005.
- B. Jähne, *Digitale Bildverarbeitung*, 5. Auflage, Springer, 2002.
- Vorlesungsfolien: <http://isgwww.cs.uni-magdeburg.de/bv/gbv/bv.html>

Englischsprachig:

- G. Baxes, *Digital Image Processing: Principles and Applications*, J. Wiley & Sons, 1994.
- K. Castleman, *Digital Image Processing*, Prentice-Hall, 1996
- R. Gonzales, R. Woods, *Digital Image Processing*, Addison-Wesley, 1992
- M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis, and Machine Vision*, Addison-Wesley, 1998.

Scheine, Prüfungen etc.

- Prüfung
 - Prüfungsvoraussetzungen laut Website
 - Klausur 2h
- Unbenotete Leistungen
 - Prüfungsvoraussetzungen in diesem Semester erfüllt
 - Klausur in diesem Semester oder im Folgesemester bestanden
 - Praktische Leistungen aus den Übungen gehen in die Klausurbewertung mit ein

Übungen

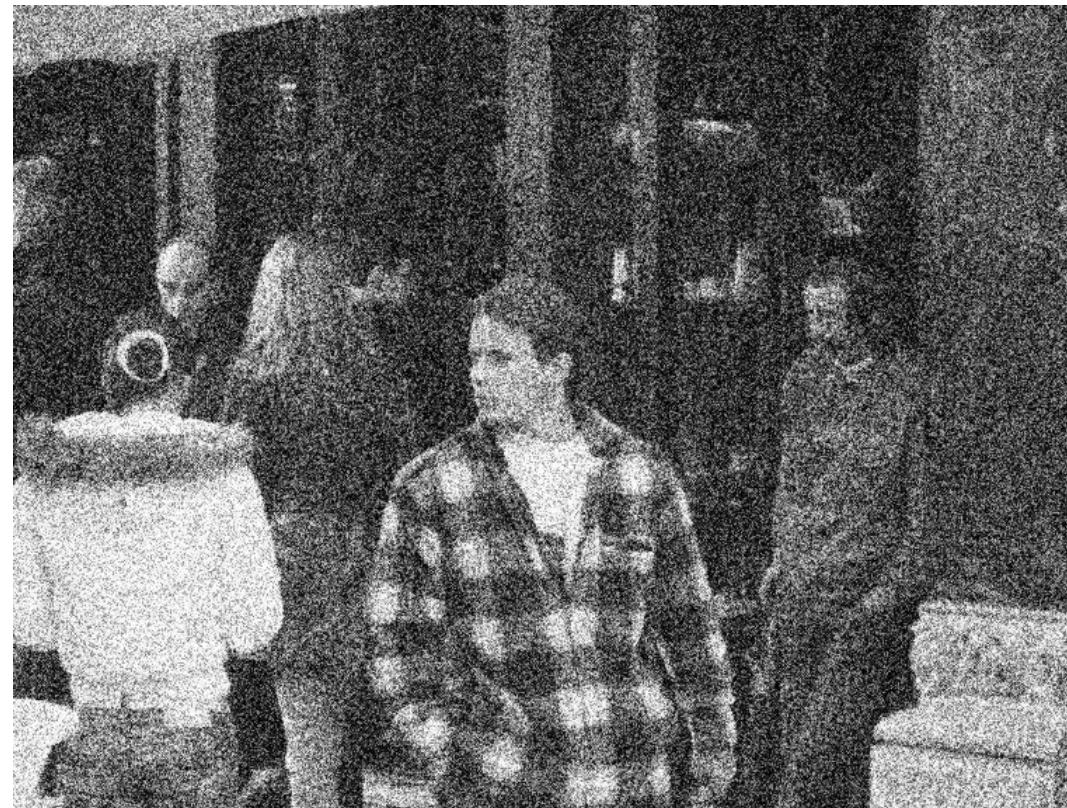
- Details auf wwwisg.cs.uni-magdeburg.de/bv (dort unter „Lehre“, „Grundlagen der Bildverarbeitung“)
- Theoretische und praktische Übungen
 - Details dazu in den Übungsgruppen
- Programmieren in Matlab
 - Matlab-Tutorial auf der LV-Seite (pdf-Datei)
 - schon mal durchlesen für die erste Übung

Übungszeiten und Eintragung

- Übungszeiten (ab 20.10.2015, Raum G29-335/K059):
 - Di, 11-13, G29-K059, und 13-15, G29-335
 - Mi, 13-15, G29-335
 - Do, 09-11, G29-335
- Informationen zu Vorlesung und Übung, sowie Eintragung in die Übungsgruppen (ab 12.10.2015 , 19 Uhr, also gleich) unter
 - wwwisg.cs.uni-magdeburg.de/bv
 - dort unter „Lehre“, „Grundlagen der Bildverarbeitung“

Famous Last Question

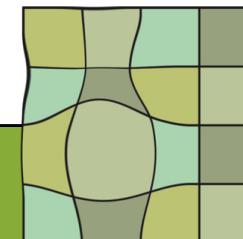
*Was könnte/sollte getan werden, um die Anzahl
der Personen
zu zählen?*



Grundlagen der Bildverarbeitung

Digitale Bilder

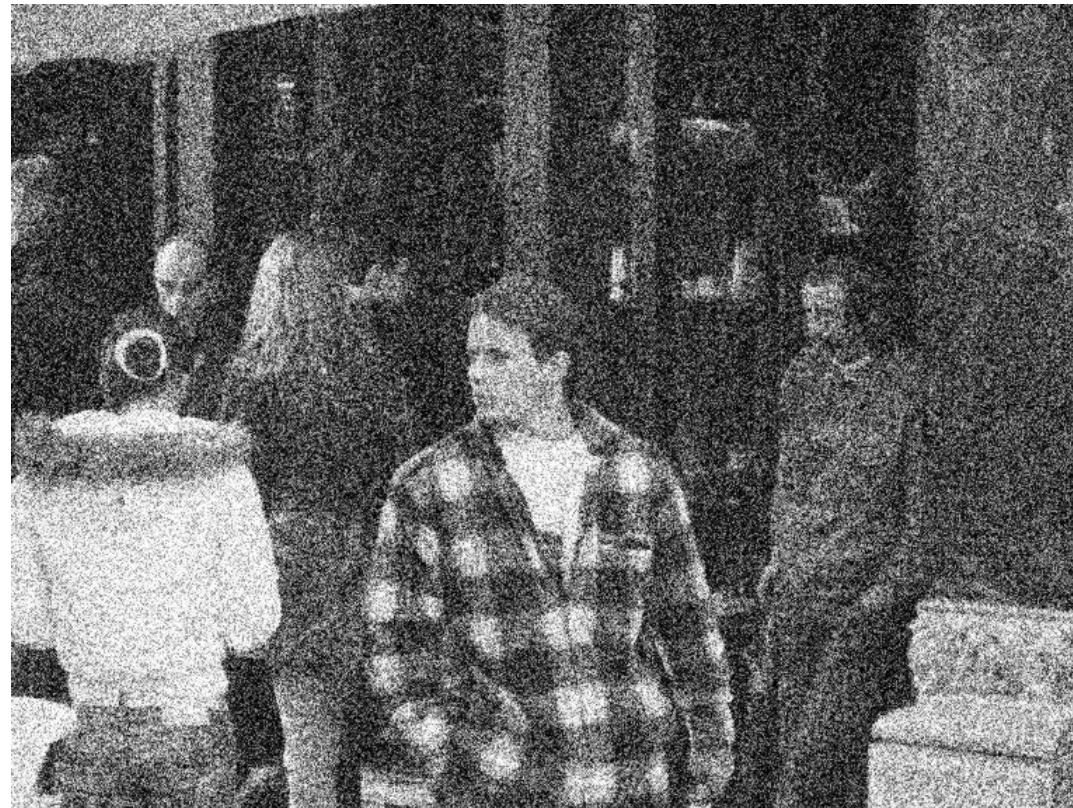
Prof. Dr. Klaus Tönnies



Bildverarbeitung
&
Bildverstehen

Famous Last Question

***Was könnte/sollte getan werden, um
die Anzahl
der Personen
zu zählen?***



Digitale Bilder

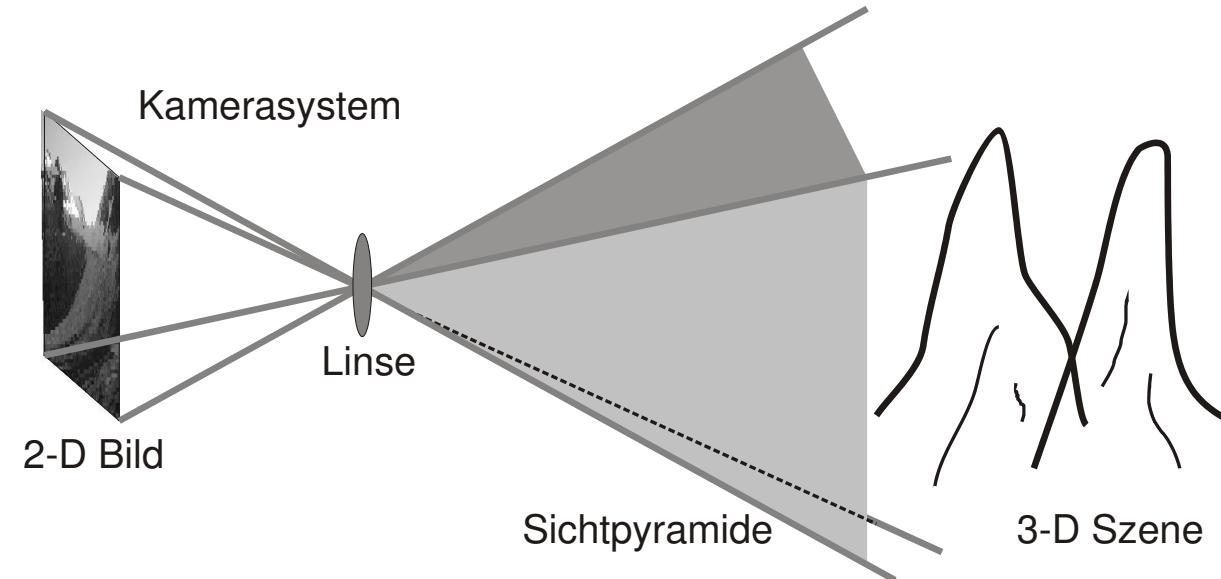
- Was ist ein digitales Bild und welche Information wird gespeichert
 - Digitales Foto
 - Grundlegende Eigenschaften
 - Bildformate

Digitales Foto

- Das Bild besteht aus einer Menge von Bildelementen (**Pixel** von „picture element“).
- Definitionsbereich: Ausdehnung in x- und y-Richtung, sowie Wellenlänge:
 $x_{\min} \leq x < x_{\max}$, $y_{\min} \leq y < y_{\max}$, $\lambda_{\min} \leq \lambda < \lambda_{\max}$.
- Wertebereich: Intensität
 $R_{\min} \leq R(x,y,\lambda) < R_{\max}$
- Definitions- und Wertebereich sind beschränkt.

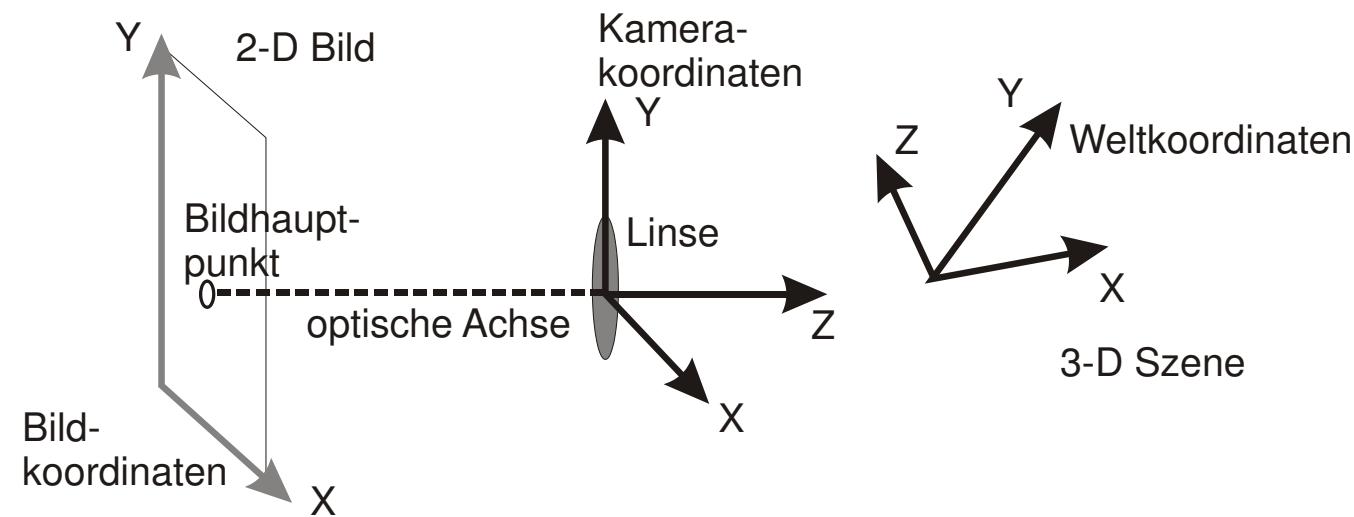


Projektion bei der Aufnahme

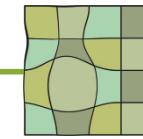


- Alle Punkte außerhalb der Sichtpyramide werden nicht abgebildet.
- Alle verdeckten Punkte in der Sichtpyramide werden nicht abgebildet.
- Von den abgebildeten Punkten geht die Tiefeninformation verloren.

Koordinatensysteme



- *Weltkoordinatensystem:* 3-D, reell, relativ zu einem Punkt in der Welt.
- *Kamerakoordinatensystem:* 3-D, reell, relativ zur Kamera.
- *Bildkoordinatensystem:* 2-D, reell oder ganzzahlig, begrenzt, relativ zum CCD-Chip.



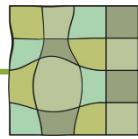
Transformation und Projektion

- Transformation: Rotation und Translation des Kamerakoordinatensystems relativ zum Weltkoordinatensystem:

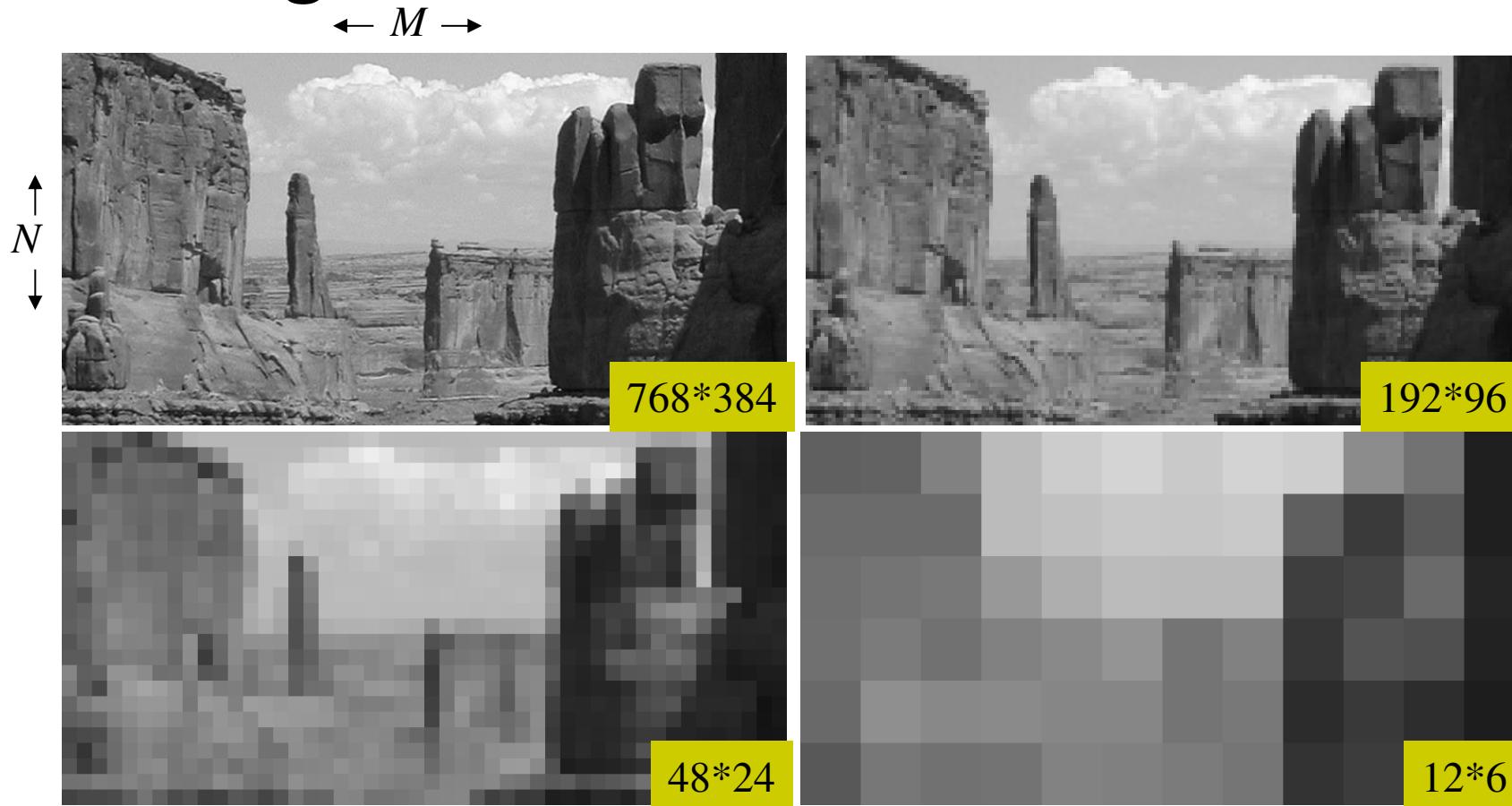
$$(x_w \quad y_w \quad z_w \quad 1) = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}.$$

- Projektion der Kamerakoordinaten in die Bildkoordinaten:

$$(x_i \quad y_i \quad -f \quad 1) := \begin{pmatrix} -f \frac{x_c + d_x}{z_c} & -f \frac{y_c + d_y}{z_c} & -f & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 0 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}.$$



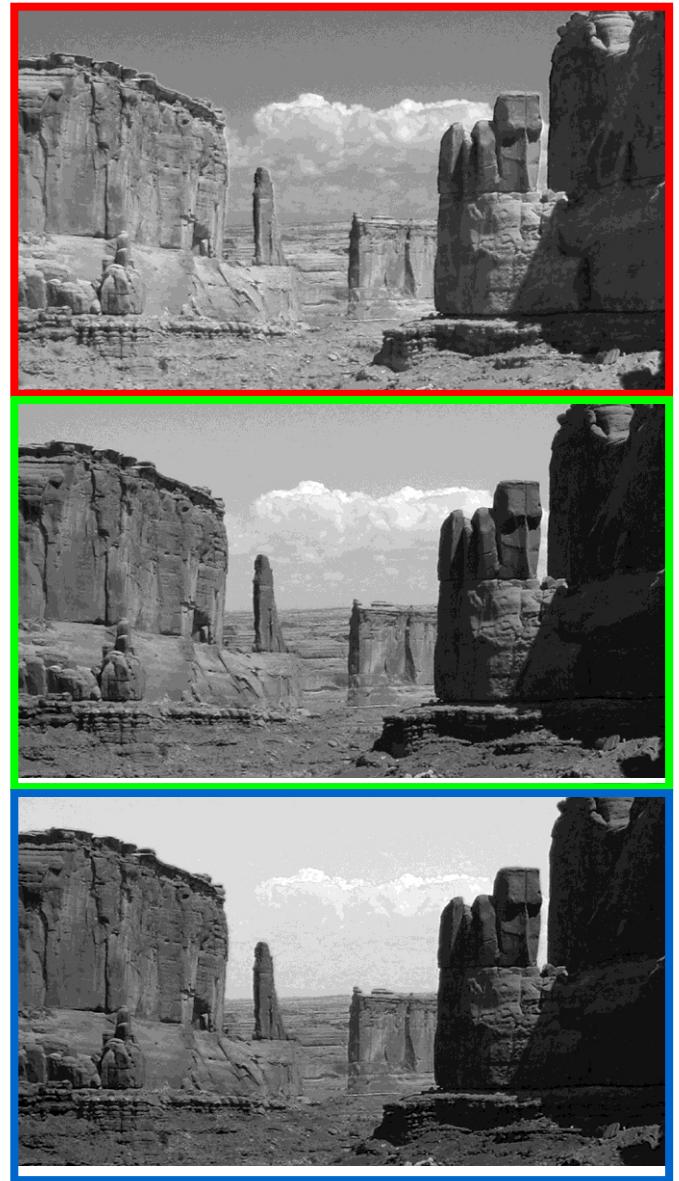
Ortsauflösung



$$m = \left\lfloor M \cdot \frac{x - x_{\min}}{x_{\max} - x_{\min}} \right\rfloor \text{ und } n = \left\lfloor N \cdot \frac{y - y_{\min}}{y_{\max} - y_{\min}} \right\rfloor \text{ für } x_{\min} \leq x < x_{\max}, y_{\min} \leq y < y_{\max}$$

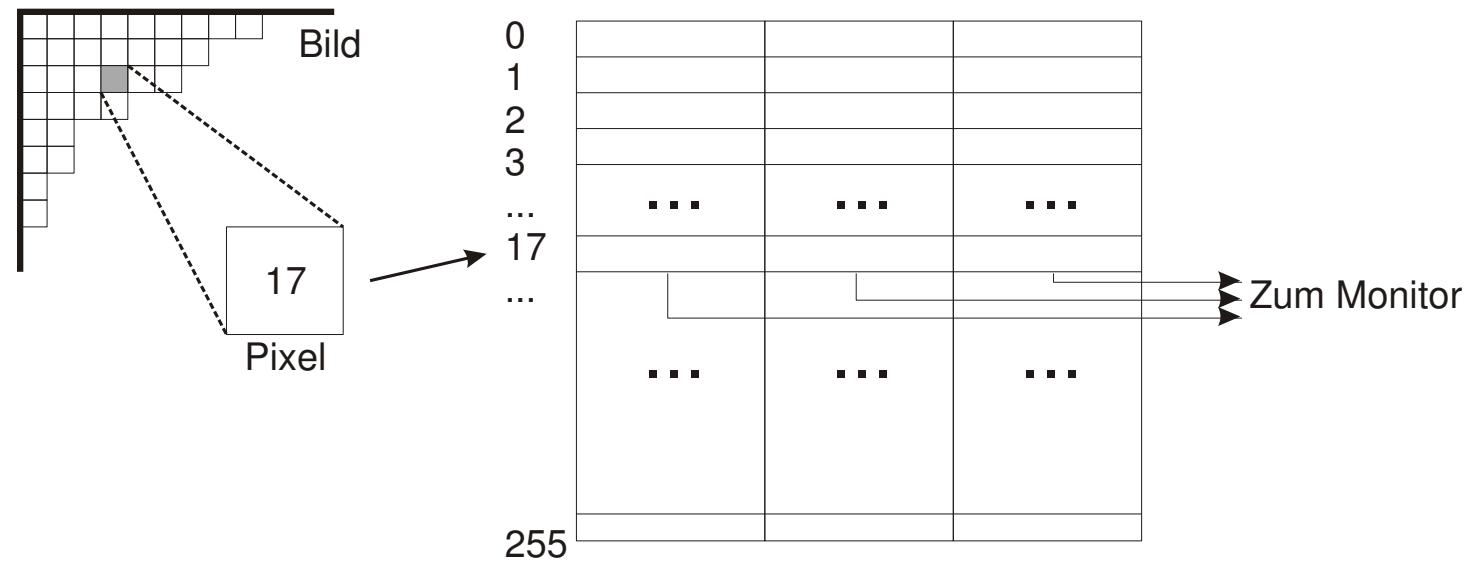
RGB Farbmodell

- Jede Farbe wird durch ein Tripel (rot, grün, blau) repräsentiert.



Repräsentation eines Bildes

- Es werden (meist) 256 Intensitätsstufen unterschieden (= 1 Byte)
- Farbe wird
 - als Tripel (r,g,b) repräsentiert.
 - oder über einen Index in eine Farbtabelle repräsentiert.
- Video-Lookup-Table (VLT)



Kontrastauflösung



256



64



16

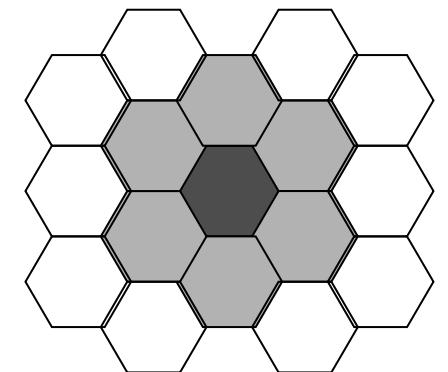
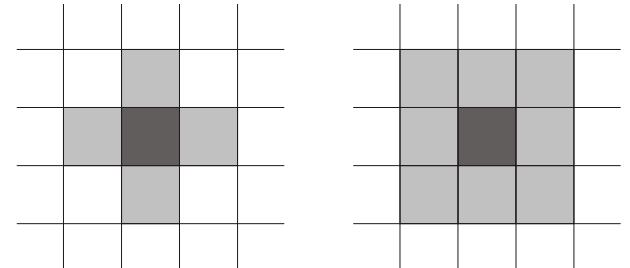


4

Reduzierung der Kontrastauflösung mindert die Erkennbarkeit weniger als die der Ortsauflösung

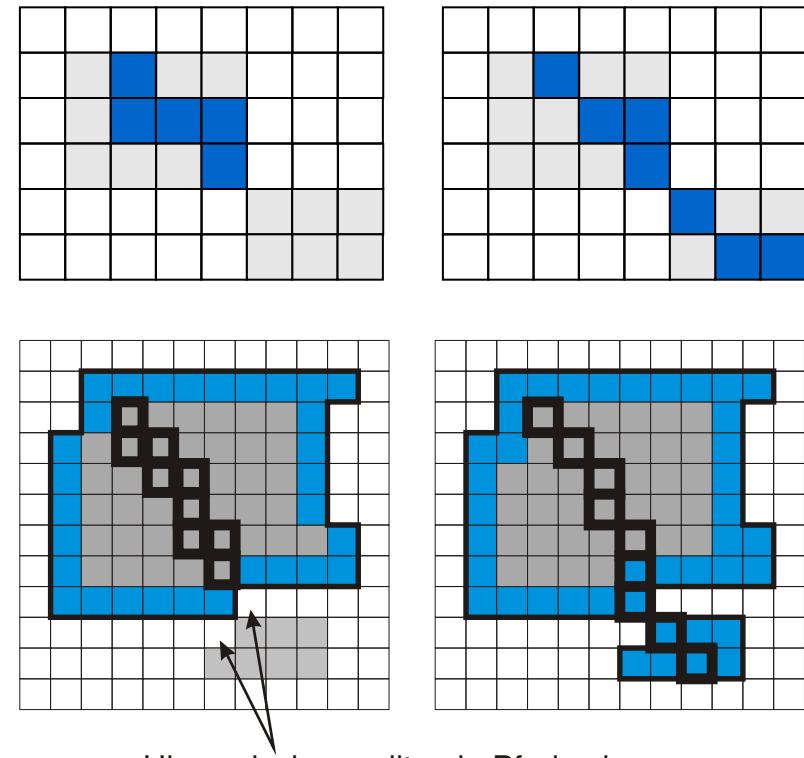
Nachbarschaft

- Quadratische (rechteckige) Gitter:
 - *4-Nachbarschaft*: Pixel haben eine gemeinsame Seite.
 - *8-Nachbarschaft*: Pixel haben eine gemeinsame Seite oder einen gemeinsamen Eckpunkt.
- Hexagonale Gitter:
 - *6-Nachbarschaft*: Pixel haben eine gemeinsame Seite und zwei gemeinsame Eckpunkte.

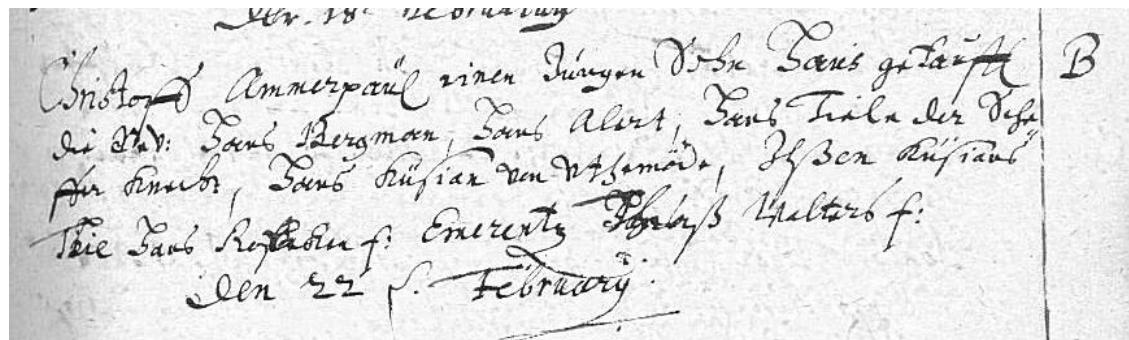


Pfad und zusammenhängendes Gebiet

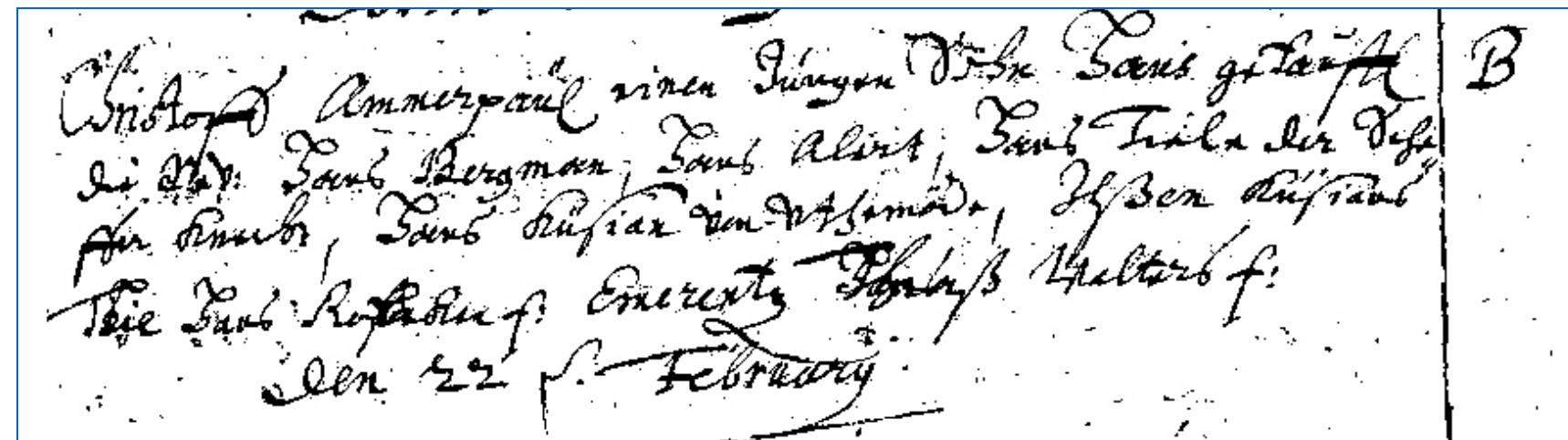
- *Pfad*: Folge von benachbarten Pixeln, die eine Homogenitätsbedingung erfüllen.
 - 4-benachbarte Pfade unterscheiden sich von 8-benachbarten Pfaden.
- *Zusammenhängendes Gebiet*: Menge aller Pixel zwischen denen Pfade existieren.
- Die Nachbarschaftsdefinitionen in Vorder- und Hintergrund müssen unterschiedlich sein.

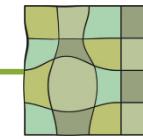


Bildanalyse und Zusammenhängende Gebiete



Geeignete Homogenitätskriterien erzeugen zusammenhängende Gebiete, deren Merkmale mit der gesuchten Semantik korrelieren





Abstandsmaße

$$D_e(x_1, y_1; x_2, y_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

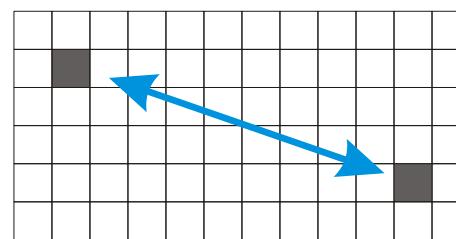
Euklidische Distanz

$$D_4(x_1, y_1; x_2, y_2) = |x_1 - x_2| + |y_1 - y_2|$$

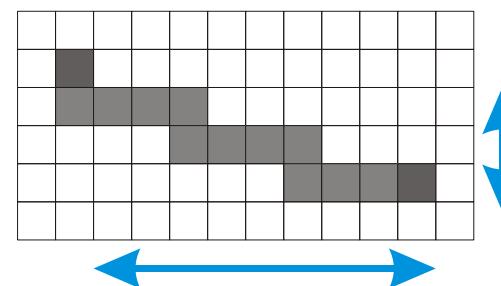
Cityblock-Distanz

$$D_8(x_1, y_1; x_2, y_2) = \max(|x_1 - x_2|, |y_1 - y_2|)$$

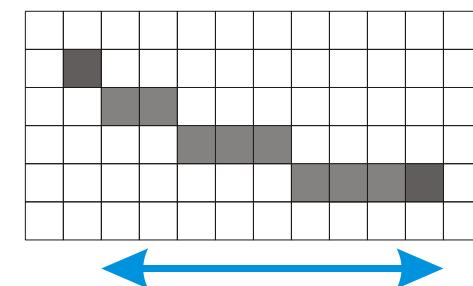
Schachbrett-Distanz



Euklidische Distanz



Cityblock-Distanz



Schachbrettdistanz

D_4 bzw. D_8 ist die kürzeste Distanz zwischen zwei Pixeln unter 4- bzw. 8-Nachbarschaft

Exkurs: Normen

- Die drei Distanzmaße sind Vektornormen.
- Die Norm $\|\mathbf{a}\|$ eines Vektors \mathbf{a} ist eine Zahl.
- Bedingungen für Normen:

$$\|\vec{a}\| \geq 0, \|\vec{a}\| = 0 \Rightarrow \vec{a} = \vec{0}$$

$$\|c \cdot \vec{a}\| = |c| \cdot \|\vec{a}\|$$

$$\|\vec{a} + \vec{b}\| \leq \|\vec{a}\| + \|\vec{b}\|$$

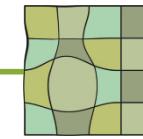
- L_p-Normen:

$$\|\vec{a}\|_p = \left(\sum_{i=1}^N |a_i|^p \right)^{\frac{1}{p}}, \|\vec{a}\|_p \geq \|\vec{a}\|_q, \text{ falls } p < q$$

- L₁-Norm: Cityblock-Distanz, L₂-Norm: euklidische Distanz, L_∞-Norm: Schachbrett-Distanz

Gängige Bildformate

- Bilder können unterschiedlich komfortabel gespeichert werden.
- Die meisten Graphikprogramme und Graphikentwicklungsumgebungen bieten Filter zum Lesen und Schreiben gängiger Formate an.
- Gängige Formate sind zum Beispiel
 - PPM, PGM, PBM: portable pixel/grey/bit map (SUN)
 - BMP: Windows Bitmap
 - PNG: Portable Network Graphics
 - TIFF, TIF: Tagged Image File Format
 - JPG, JPEG: Joint Photographers Expert Group



PPM, PGM (, PBM)

- Reihe von proprietären, sehr einfachen Formaten von SUN
- Keine Datenkompression
- ASCII- oder als Binärdatei
- Struktur:
 - id: P1,P4 (PBM); P2,P5 (PGM); P3,P6 (PPM)
 - Bildbreite, Bildhöhe
 - Maximaler Grauwert
 - Pixelwerte
 - (Kommentarzeilen, beginnend mit #)
- Beispiel (Grauwertdatei)

```
P2
10 5
10
0 0 0 0 0 0 0 0 0 0
1 2 2 2 2 2 2 1 1 1
2 4 4 4 7 7 7 7 7 7
1 2 2 2 2 10 10 10 10 2
0 0 0 0 0 0 0 0 0 0
```

BMP

- Von Microsoft mit Windows eingeführt
 - Unterstützt unterschiedliche Bittiefen von 1,4,8,16,24, oder 32 Bit per Pixel (bpp)
 - Gebräuchlich sind 8 Bit (für Grauwertbilder) und 24 (für 3-Kanal-Farbbilder)
 - Bilddaten werden durch Run-Length-Encoding komprimiert (VL 7 Bildkompression)
 - Es können Lookup-Tabellen (bei Bittiefe 1,4,8) für die Repräsentation von Farbbildern gespeichert werden.
- Es existieren mehrere Versionen des BMP-Formats (Version 3,4,5)
 - Die meisten BMP-Reader unterstützen Version 3 und 8 bzw. 24 bpp-Repräsentationen.
 - Transparenzkanal (Alpha-Kanal) ist in Version 3 nicht vorgesehen (wohl aber in Version 4 und 5), auch wenn einige Programme das letzte Byte einer 32-bpp-Repräsentation so interpretieren.

PNG

- Ersatz für GIF, aber ohne Möglichkeit zur Speicherung von Animationen (eigenes Format MNG; wurde nie standardisiert)
- wird von Web-Browsern unterstützt
- Pixelformate
 - Farbpalettenbilder mit 8-Bit-Pixeln
 - 1-, 4-, 8-, 16-Bit Grauwertbilder
 - 3×8 oder 3×16-Bit-Farbbilder (RGB)
 - Alpha-(Transparenz-)kanal mit 8 bzw. 16-Bit
- Kompression
 - Zweistufig: Vorfilterung durch Differenzbildung, anschließend ZIP-Kompression (Kombination aus Lempel-Ziv und Huffman-Kodierung)
 - Kompressionsrate besser als BMP, auch – durch Vorfilterung – besser als bei GIF

TIFF

- Sehr variables (und komplexes) Format
 - Ein oder mehrere Bilder (z.B. auch Thumbnail-Bild und original)
 - Bildpunkte können Einzelwerte (Grauwerte) oder Gruppen von Einzelwerten (z.B. RGB) sein
 - Bildwerte können Integer oder Floats sein
 - unterstützt verschiedene Farbmodelle
 - unterstützt eine Reihe von (verlustfreien) Kompressionsmethoden
 - Alpha- (Transparenz-) Werte können gespeichert werden
- Realisation
 - Nach Identifikation (u.a. durch Magic Number 42) folgt Offset-Angabe zum ersten Bild
 - Bilder werden durch eine Folge von Tags repräsentiert
 - Tags bestehen aus ID (z.B. Anzahl der Pixel je Zeile) gefolgt von den tag-spezifischen Daten
 - Nach dem Bild folgt Offset zum nächsten Bild
- Verwendung
 - überall da, wo hochwertige Bilder benötigt werden (z.B. Druckindustrie)

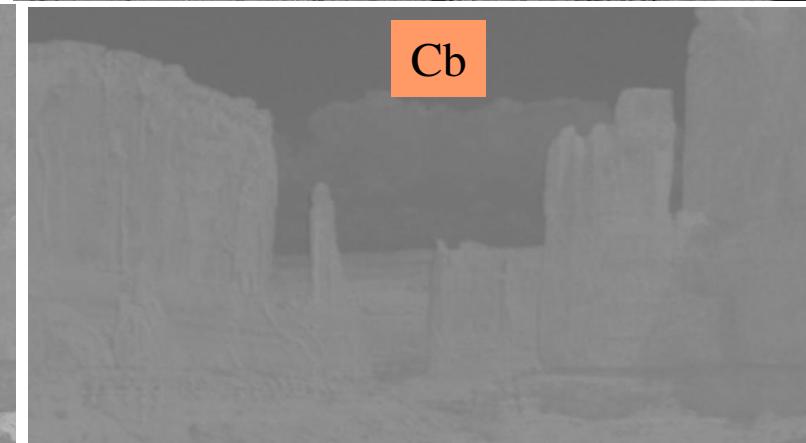
JPEG

- Standardisiertes Format zur effizienten, aber verlustbehafteten Speicherung von Bildern
- Speicherung von Grauwert- und Farbbildern
- Farbbildrepräsentation als YCbCr-Repräsentation (PAL)
 - Y: gewichtete Kombination von R,G,B (entspricht etwa dem Grauwertbild)
 - Cb, Cr: um Y-Wert korrigierter Blau- bzw. Rot-Anteil im Bild (Farbabweichungen)
- Algorithmus zur Erzeugung der JPEG-Repräsentation (Details in VL 7: Kompression)
 - Umrechnung in YCbCr-Raum
 - Verlustbehaftete Filterung der Cb-, Cr- Anteile
 - Einteilung des Y-Bilds in 8×8 -Blöcke und verlustbehaftete Kompression dieser Blöcke
 - Verlustfreie Kompression des Restsignals

Exkurs YCrCb-Farbraum

- Abbildungsvorschrift

$$\begin{pmatrix} Y \\ Cr \\ Cb \end{pmatrix} = \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$



JPEG Blockkompression

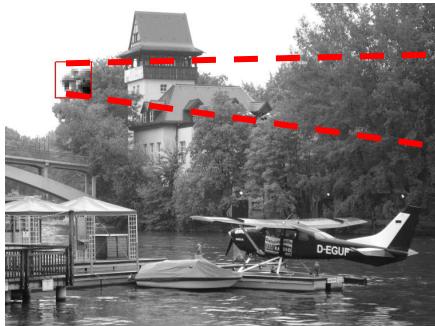
- Bild wird Block für Block komprimiert gespeichert
- Blockgröße ist 8×8 Pixel
- Kompression des Blocks durch Transformation in effizientere Repräsentation

Blockzerlegung



Transformation des Blocks

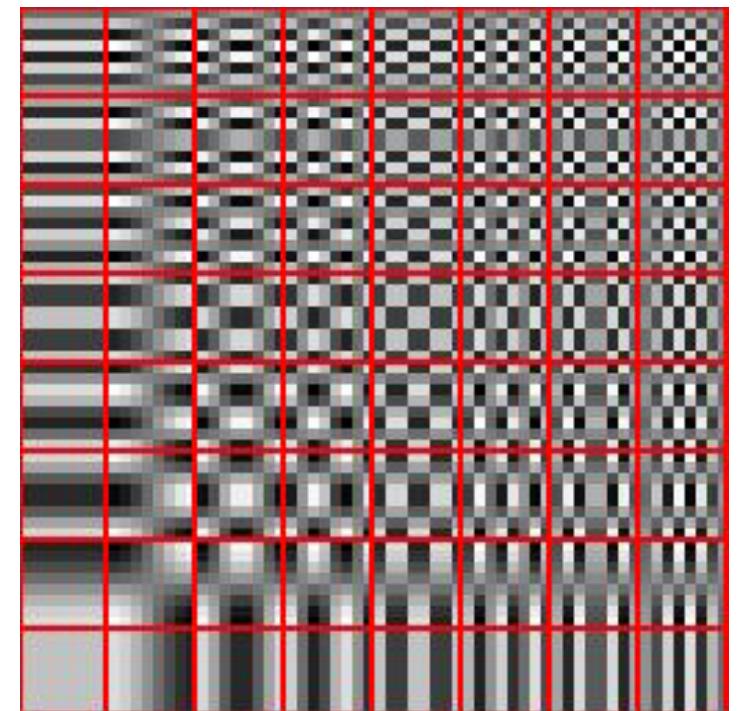
- Bildung gewichteter Summen mit sogenannten Basisfunktionen
- Es gibt soviel Basisfunktionen, wie es Pixel in einem Block gibt
- Jede Gewichtung mit einer Basisfunktion ergibt genau einen Wert



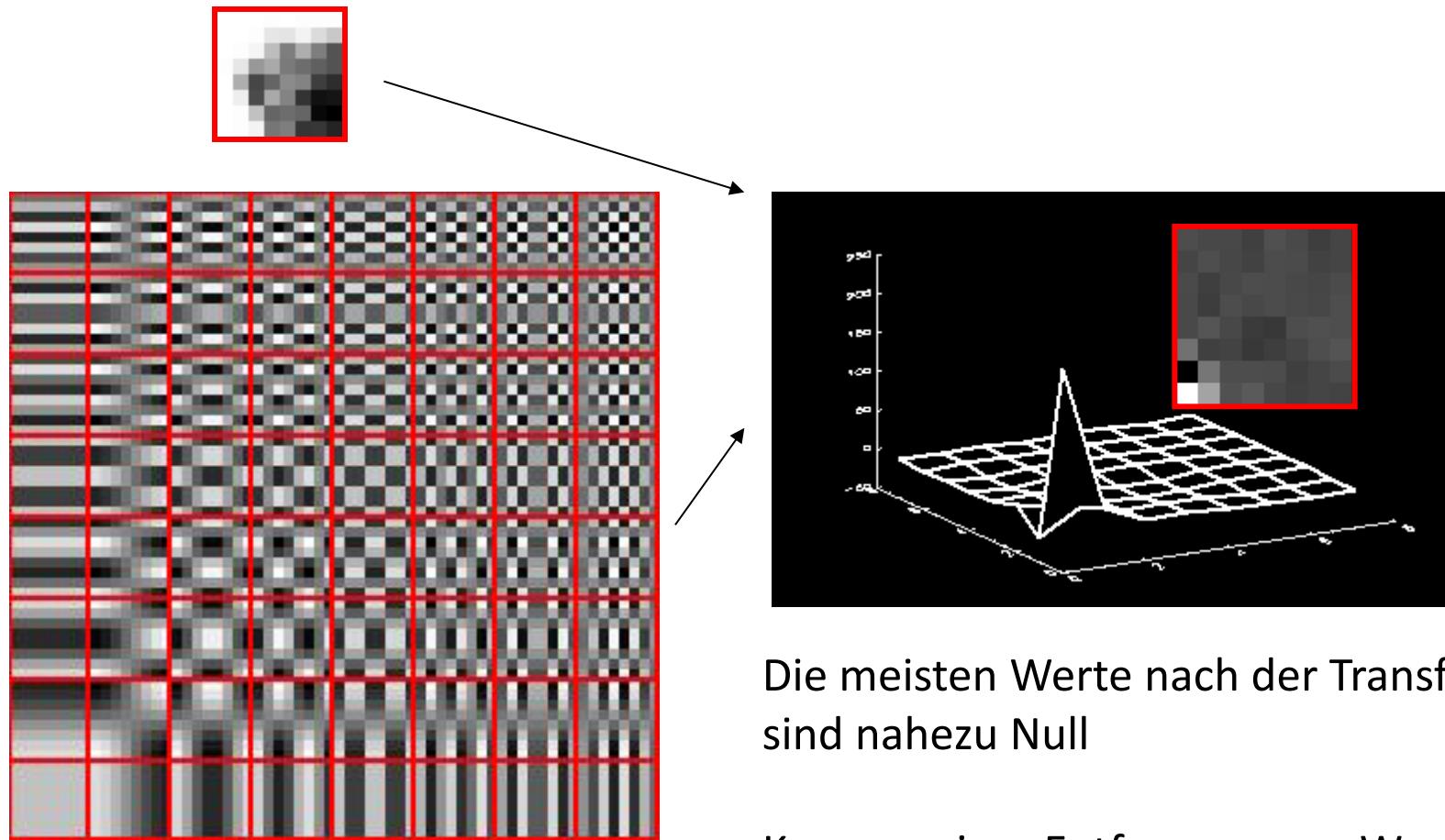
*



Die 64 Basisfunktionen (je 8×8 Pixel groß)
für die JPEG-Kompression



Resultat

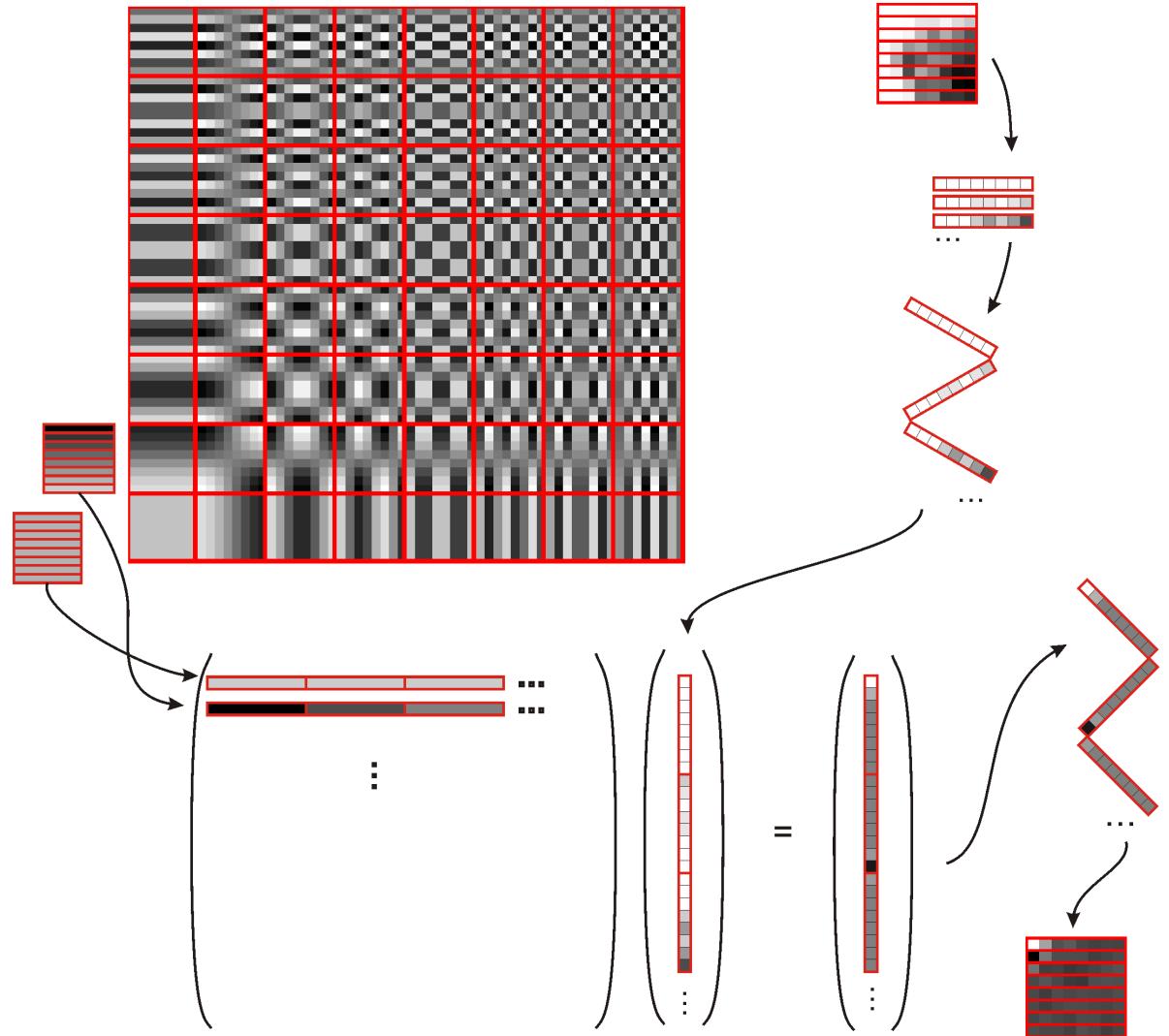


Die meisten Werte nach der Transformation
sind nahezu Null

Kompression: Entfernung von Werten <

Berechnung

- Bild als Vektor
- Basisfunktionen als Sequenz von Vektoren in Matrix
- Operation als Matrixmultiplikation $\mathbf{Ax}=\mathbf{b}$



Fragen

- Wie verändert sich das Bild, wenn Werte nahe Null **gelöscht** werden?
- Wie kann man aus dem transformierten Bild das Original zurückgewinnen (**Invertierung** von A)
- Trennt die Basis A **zuverlässig** wichtige und unwichtige Elemente (Operator für relevante Bildanteile)?
- Ist die Transformation für **beliebige Bilder** (und auch für beliebige Größen der Basismatrizen) ausführbar?

Was Sie heute gelernt haben sollten

- Was sind digitale Bilder, wie entstehen digitale Bilder
- Grundlegende Eigenschaften von digitalen Bildern
- Gängige Bildformate
- Einstieg in lineare Bildtransformationen

Famous Last Question

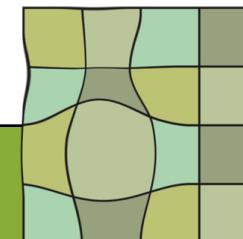
Wie könnte diese Information “unsichtbar” in das RGB-Bild eingebettet werden?



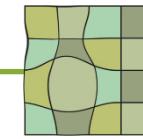
Grundlagen der Bildverarbeitung

Fouriertransformation

Prof. Dr. Klaus Tönnies

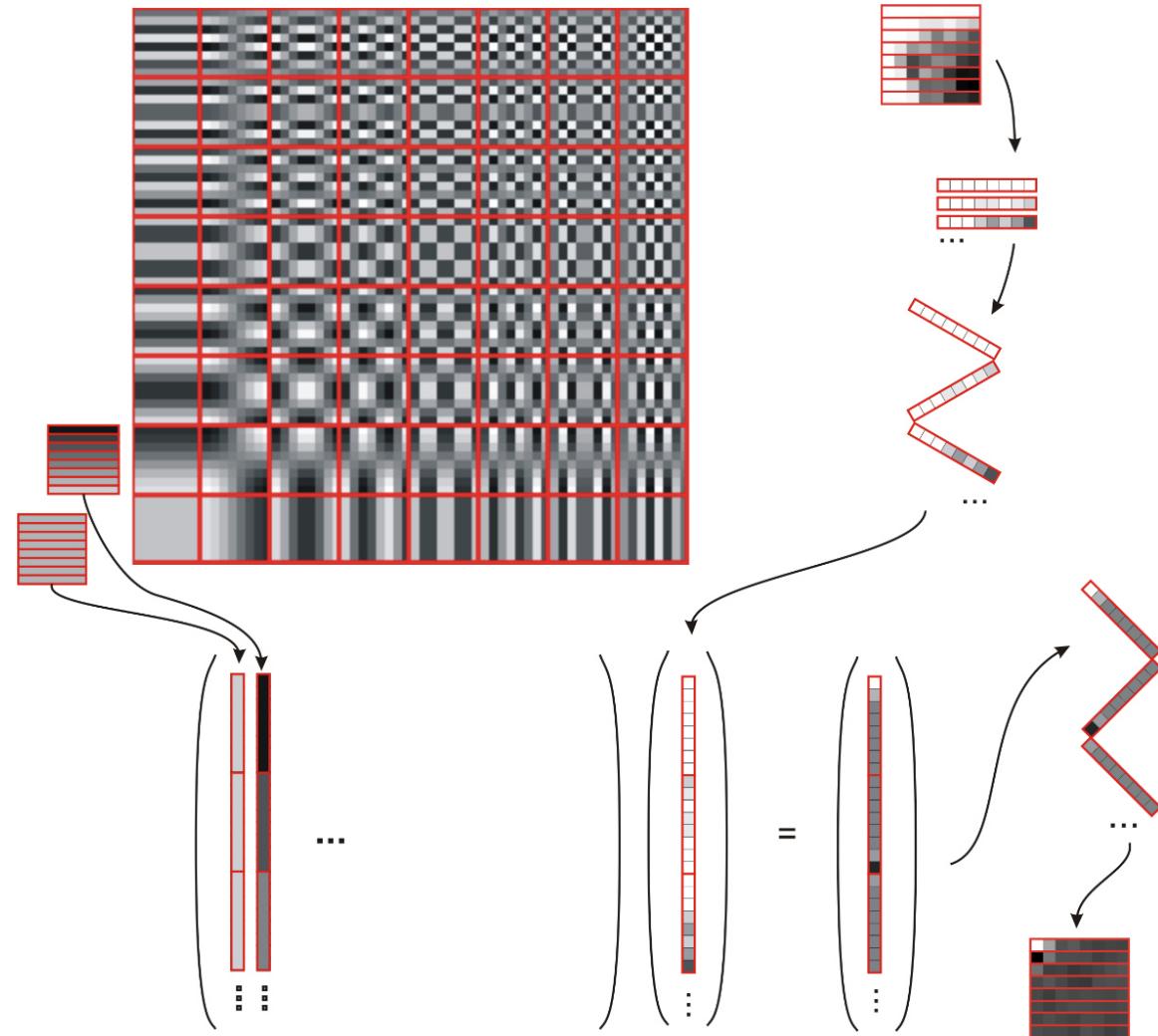


Bildverarbeitung
&
Bildverstehen



Erinnerung JPEG-Kompression

- Wie lassen sich die Basen entwickeln?
 - Sinnvolle Zerlegung
 - Invertierbarkeit
- Geht das für beliebige Blockgrößen?
- Kann man noch mehr damit machen?

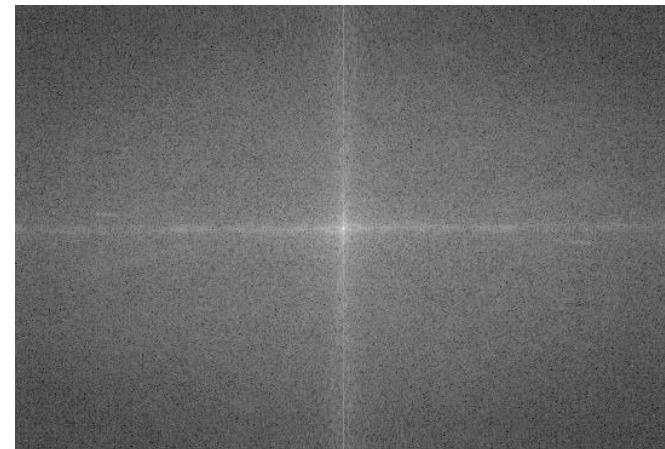


Fouriertransformation

- Vektor- und Funktionsbasen
- Transformation zwischen Basen
- Basisfunktionen der Fouriertransformation
- 1-D und 2-D Fouriertransformation und grundlegende Eigenschaften
- Darstellung der Komponenten der Fouriertransformierten

Fouriertransformation

- Beschreibung einer beliebige Funktion als Summe von gewichteten periodischen Funktionen (*Basisfunktionen*) mit unterschiedlicher Frequenz (*Frequenzraumrepräsentation*).
- Anwendungen
 - Beschreibung des Informationsverlusts bei Digitalisierung
 - Restauration von linearen Störungen
 - Rekonstruktion von Bildern aus Projektionen
 - Schnelle Filterung

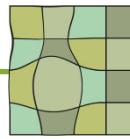


Basisfunktionen

- Bilder können als zweidimensionale Funktion $f(m,n)$ aufgefasst werden.
- Jede Funktion kann als Summe von gewichteten Basisfunktionen $b_{u,v}$ aufgefasst werden:

$$f(m,n) = \sum_{v=0}^{M-1} \sum_{u=0}^{N-1} w_{u,v} \cdot b_{u,v}(m,n)$$

- Die Wichtungen $w_{u,v}$ bilden eine neue Funktion $w(u,v)$, die $f(m,n)$ zusammen mit den Basisfunktionen genau beschreibt.
- Die Auswirkung von Veränderungen auf die Wichtungen (z.B. Filterung) hängen von den Basisfunktionen ab.
- Die Fouriertransformation ist die Transformation von einer Ortsbasis in eine Frequenzbasis.



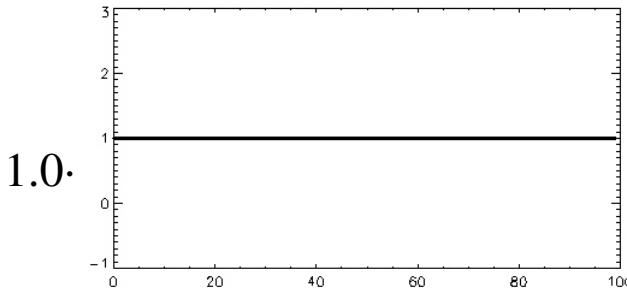
Ortsbasis

- Jede Basisfunktion ist ein Bild, das in einem Pixel den Wert 1 und in allen anderen Pixeln den Wert 0 hat.
- Zwei Basisfunktionen unterscheiden sich dadurch, dass sie den 1-Pixel an unterschiedlichen Stellen haben.
- Vorgeschlagene Basis ist vollständig und eindeutig.

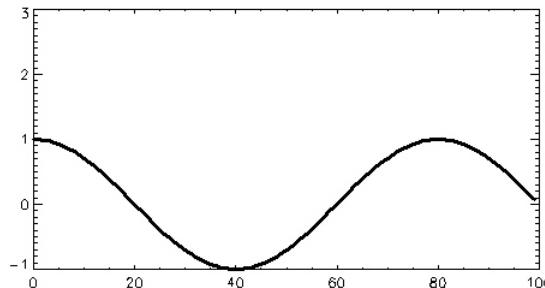
$$f(m, n) = \sum_{v=0}^{M-1} \sum_{u=0}^{N-1} w_{u,v} \cdot b_{u,v}(m, n)$$

Basisfunktionen	Wichtung
	\times
$f(m, n)$	$=$
	\times
	\dots
	\times
$b_{u,v}(m, n)$	$w_{u,v}$

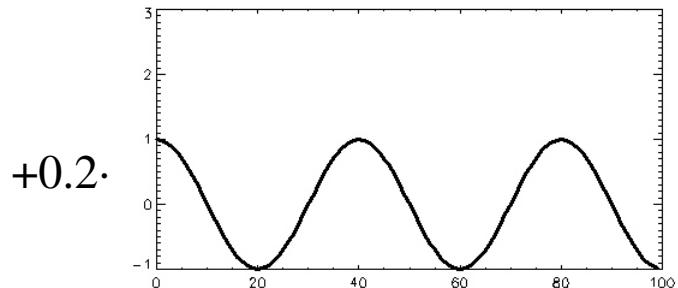
Frequenzbasis (1-D)



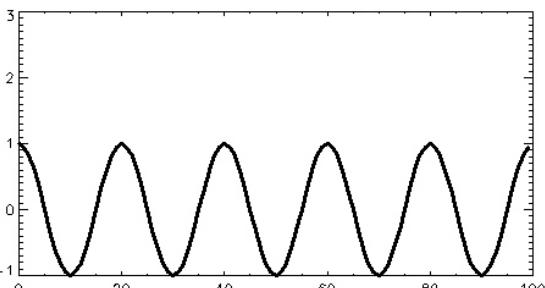
1.0.



+1.0.

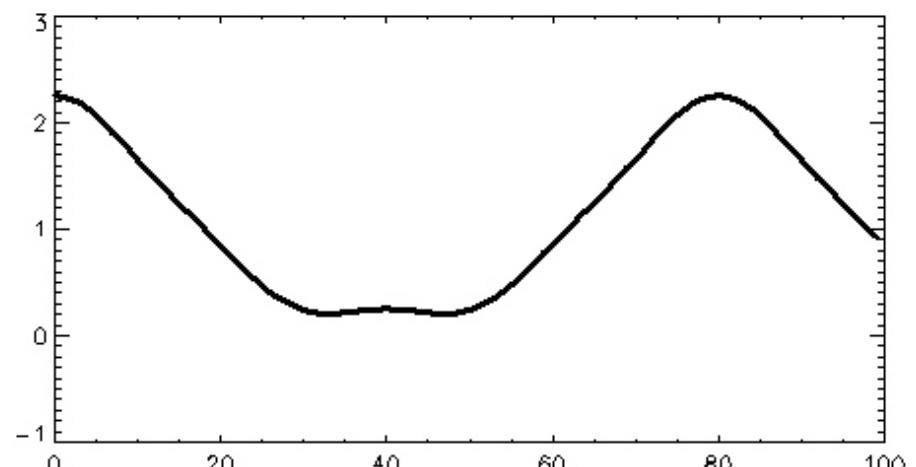


+0.2.



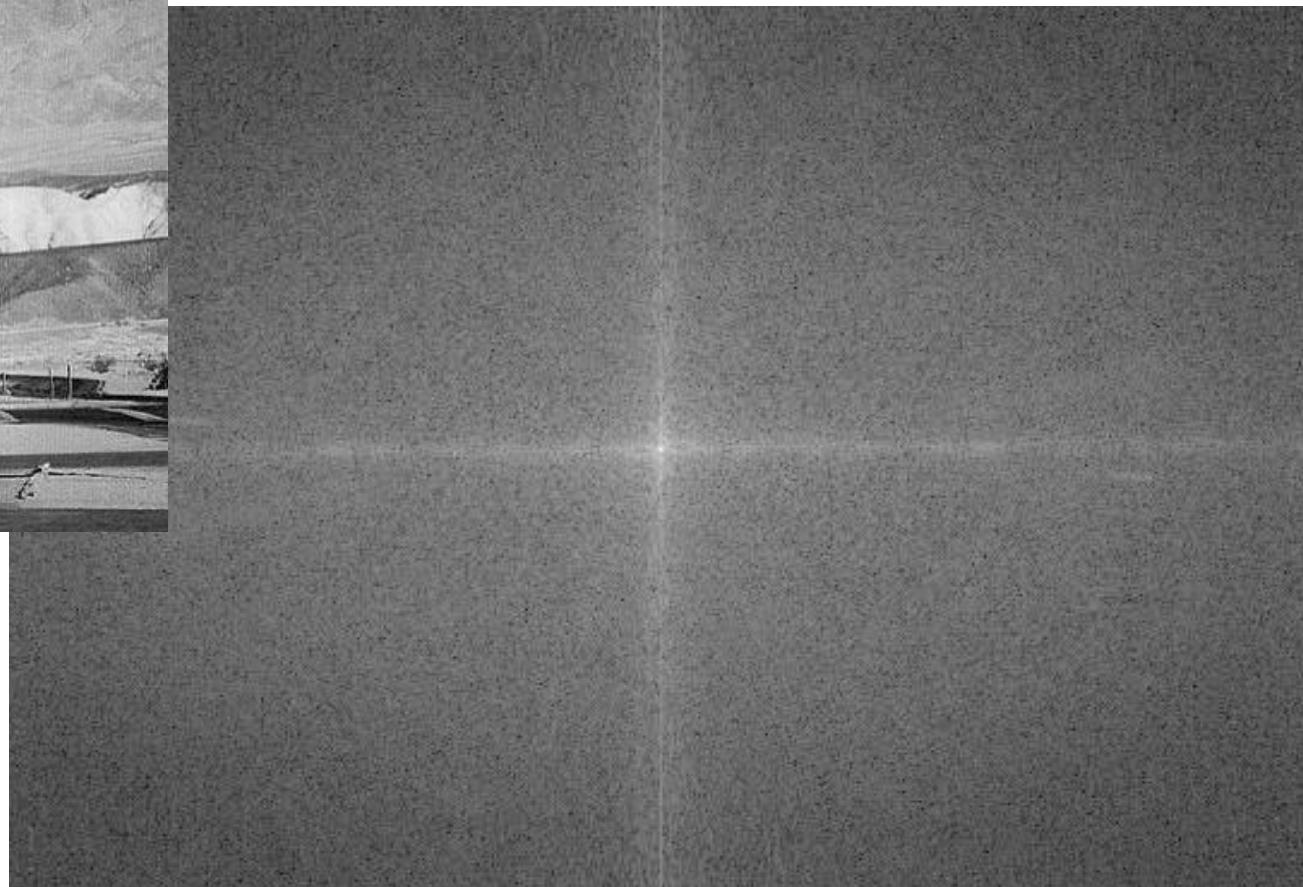
+0.05.

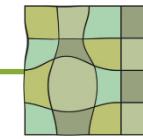
=



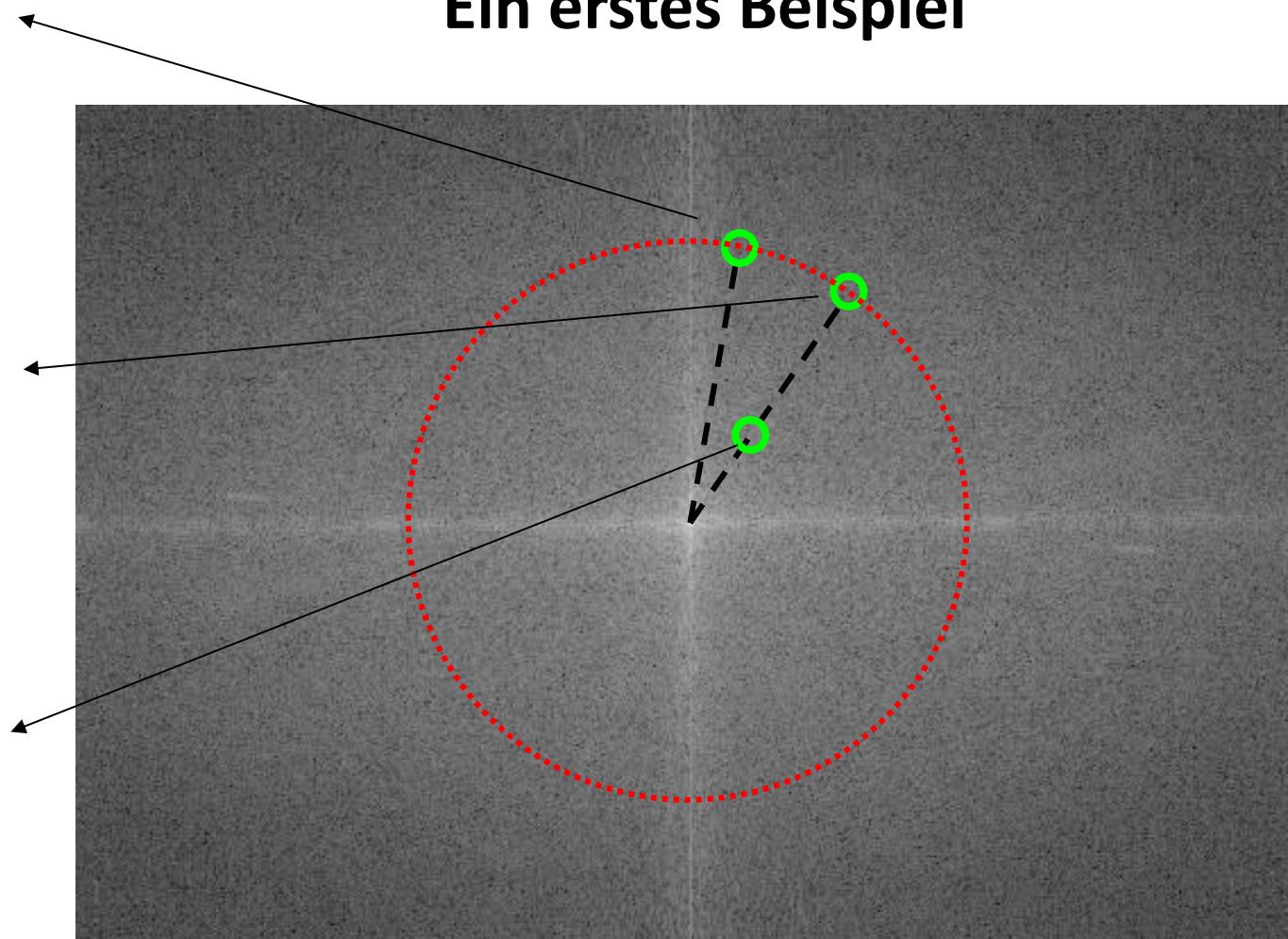
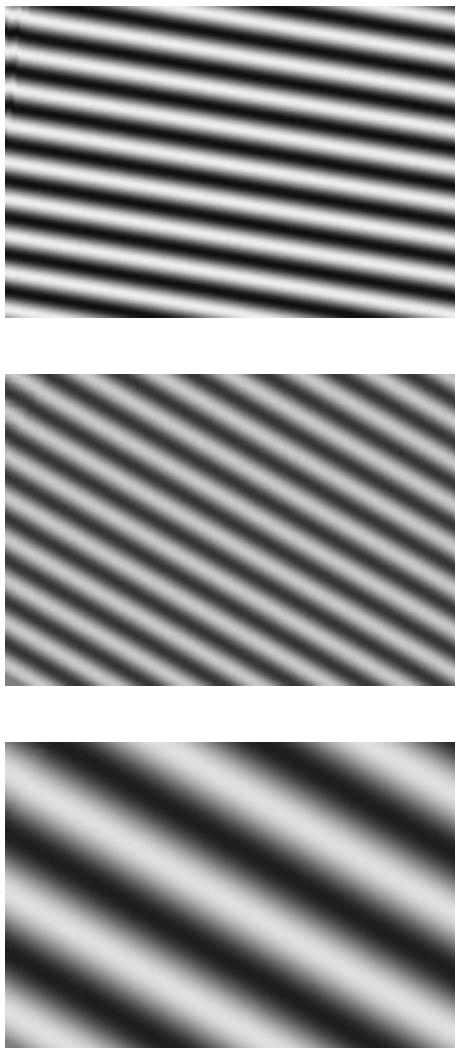
Vollständig? Eindeutig?

Ein erstes Beispiel

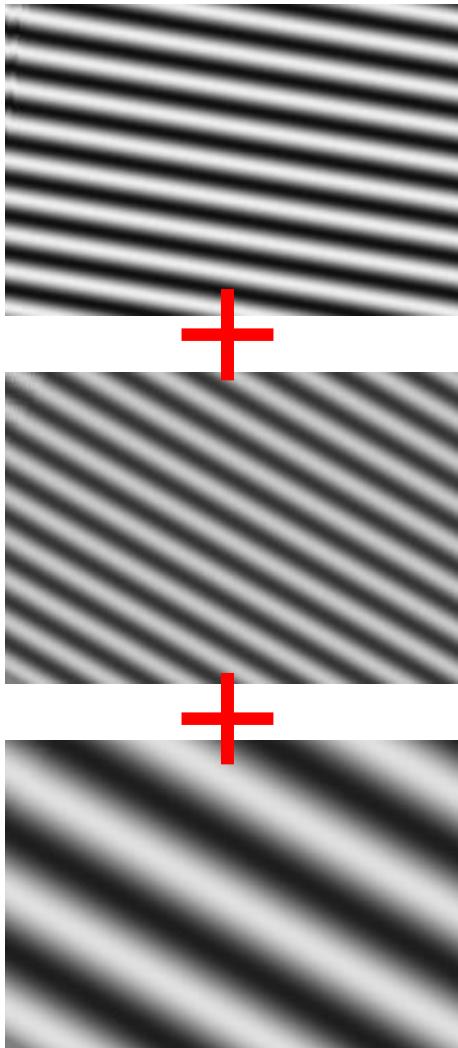




Ein erstes Beispiel

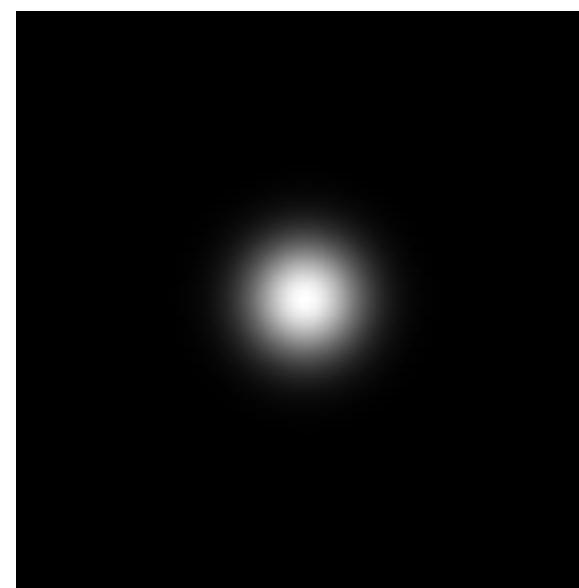
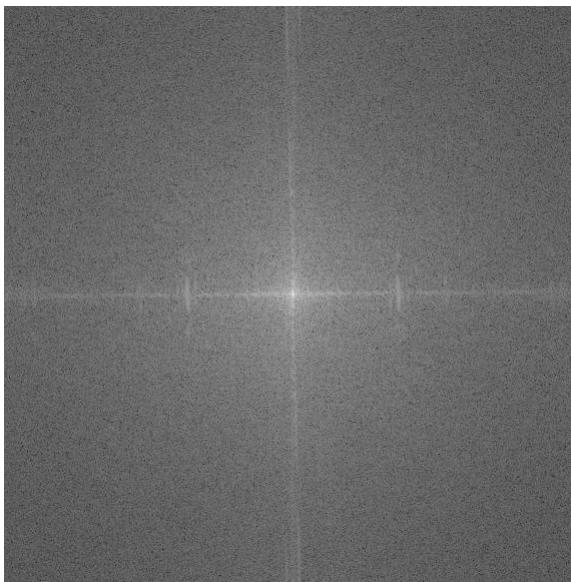


Ein erstes Beispiel

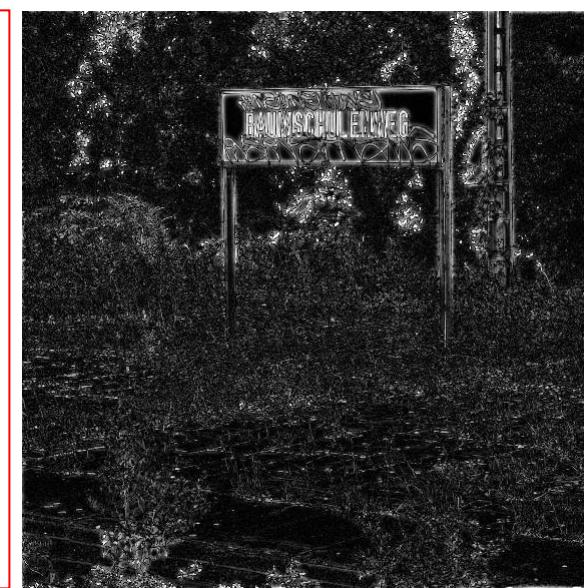
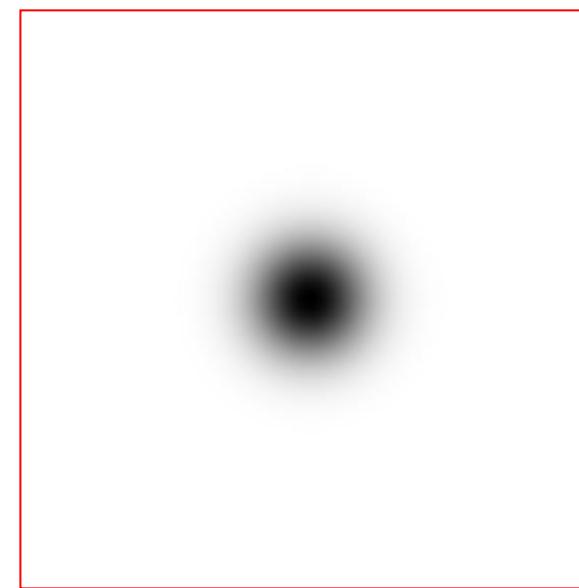
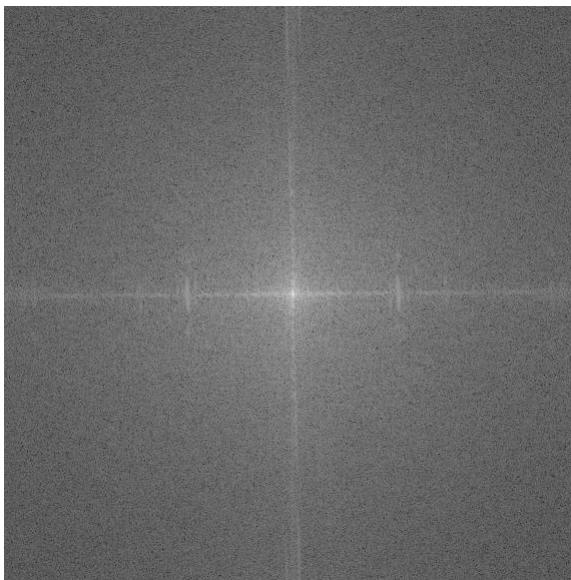
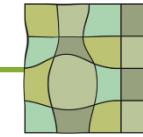


+ ... =





Erste Experimente



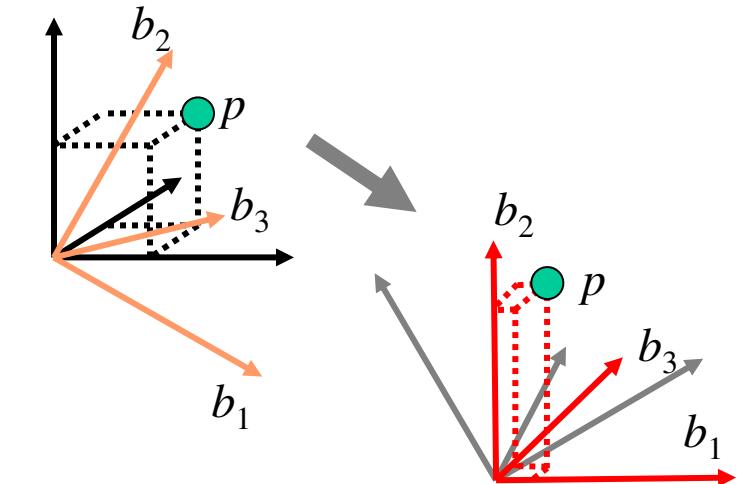
Erste Experimente

Invertierbarkeit von $Ax=b$

- Die Transformation ist einfach invertierbar, wenn die Basisfunktionen in A eine *orthogonale Basis* bilden.
 - Was ist Orthogonalität für Funktionen?
 - Wann bilden Basisfunktionen eine orthogonale Basis?
- Transformation: „Projektion“ der Funktion auf die neue Basis.
- Inverse Transformation: Projektion auf die alte Basis.
- Orthogonalität, Projektion, Transformation etc. sind Begriffe aus der *Vektoralgebra* und können auch für Funktionen genutzt werden.

Orthogonale Vektorbasis

- Zwei Vektoren sind orthogonal, wenn ihr Skalarprodukt Null ist.
- N Vektoren bilden eine Basis für einen N -dimensionalen Raum, wenn sie alle orthogonal zueinander sind.
- Projektion eines Vektors p auf einen Basisvektor b ist durch das normierte Skalarprodukt zwischen ihnen gegeben.
- Beispiel:
 - Koordinatenvektoren eines dreidimensionalen Raums.



$$\vec{b}_i \cdot \vec{b}_j = 0, \quad \text{für } i, j = 1, 2, 3 \wedge i \neq j$$

$$\vec{p}^B = \begin{pmatrix} \vec{p} \cdot \vec{b}_1 & \vec{p} \cdot \vec{b}_2 & \vec{p} \cdot \vec{b}_3 \\ \|\vec{b}_1\| & \|\vec{b}_2\| & \|\vec{b}_3\| \end{pmatrix}$$

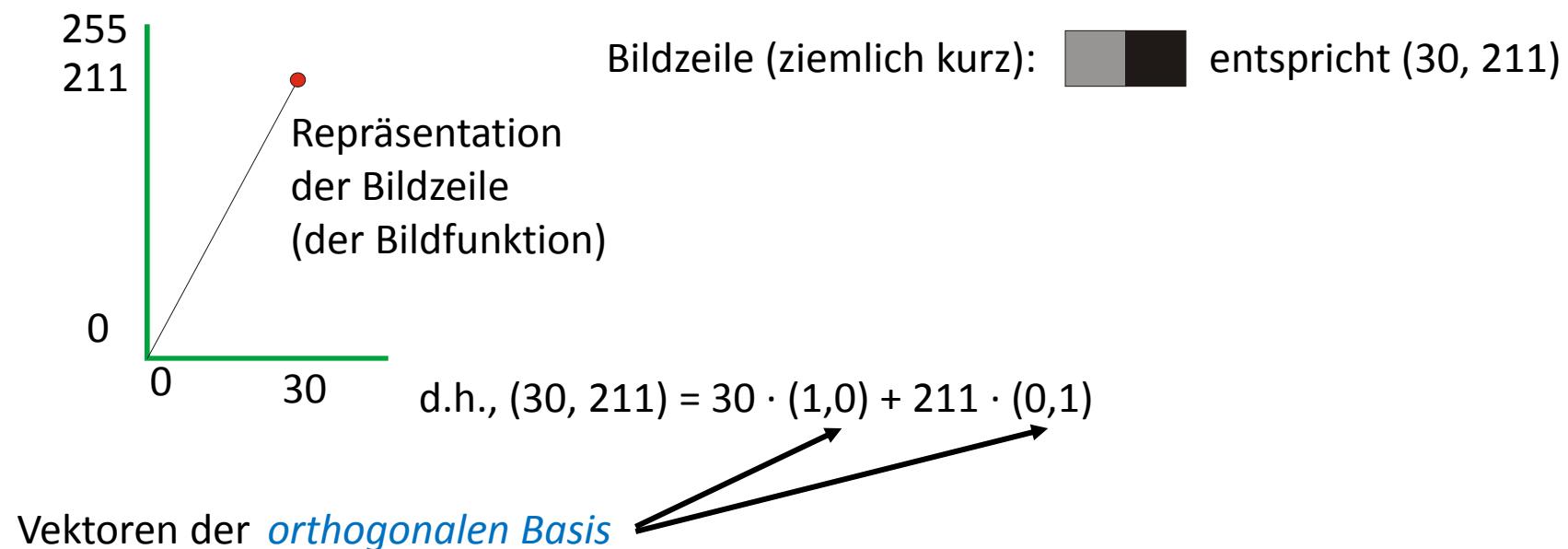
Funktionen statt Vektoren

- Eine Funktion $f(x)$ ist auch durch vollständige Aufzählung aller Werte eindeutig definiert: $f(x) = \{f(x_1), f(x_2), \dots\}$
- Skalarprodukt zwischen Funktionen f_1 und f_2 :
 - Reeller, unbeschränkter Definitionsbereich $\int_{-\infty}^{\infty} f_1(x)f_2(x)dx$
 - Reeller, beschränkter Definitionsbereich $\int_{x_{\min}}^{x_{\max}} f_1(x)f_2(x)dx$
 - Ganzzahliger, unbeschränkter Definitionsbereich $\sum_{n=-\infty}^{\infty} f_1(n)f_2(n)$
 - Ganzzahliger, beschränkter Definitionsbereich $\sum_{n=0}^{N-1} f_1(n)f_2(n)$
- Orthogonale Basen können wie für Vektoren definiert und genutzt werden.

Funktion als Vektor (Beispiel)

Ein Bild kann als Funktion mit endlichem Definitionsbereich aufgefasst werden.

Anzahl der Bildelemente = Anzahl der Funktionswerte = Vektordimension.



Transformation in eine andere Basis

- Die Transformation in eine andere Basis ist invertierbar,
 - wenn sie *orthogonal* ist
 - wenn die *Anzahl der Basisvektoren gleich* bleibt.
- Orthogonalität der zwei Funktionen: $[f_1 \bullet f_2](n) = \sum_{n=0}^{N-1} f_1(n) \cdot f_2(n) = 0$
- Jede orthogonale Basis kann durch Rotation (und Translation) aus jeder anderen Basis für die gleiche Funktion erzeugt werden.
- Transformation entspricht einer Rotation (und ggf. Translation):

$$\begin{aligned} T_b[f](u) &= \sum_{n=0}^1 f(n) \cdot b_u(n) \\ b_1(n) &= (\cos \alpha \quad \sin \alpha) \\ b_2(n) &= (-\sin \alpha \quad \cos \alpha) \end{aligned}$$

Transformation in Vektor-Matrix-Schreibweise

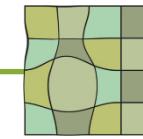
- Funktion kann als Vektor \vec{f} ihrer Funktionswerte repräsentiert werden:

$$\vec{f} = (f(0) \ f(1) \ \dots \ f(N-1))$$

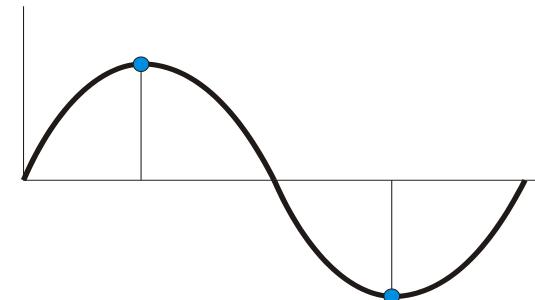
- Basisfunktionen b_u können in einer quadratischen Matrix \mathbf{B} zusammengefasst werden:

$$\mathbf{B} = \begin{pmatrix} b_0(0) & b_1(0) & \dots & b_{N-1}(0) \\ b_0(1) & b_1(1) & & \dots \\ \dots & & & \\ b_0(N-1) & \dots & & b_{N-1}(N-1) \end{pmatrix}$$

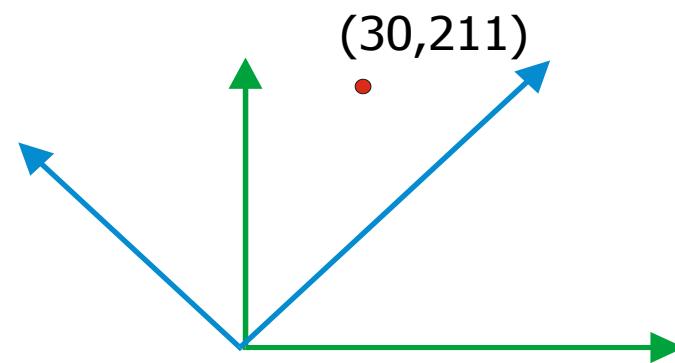
- Transformation auf die neue Basis \vec{F} ist die Multiplikation des Vektors mit der Matrix \mathbf{B} : $\vec{F} = \vec{f} \times \mathbf{B}$



Orthogonale periodische Funktionen



Basis:
 $(1, 1)$
 $(1, -1)$



Transformation:

$$(30, 211) * \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = (241, -181)$$

Rücktransformation:

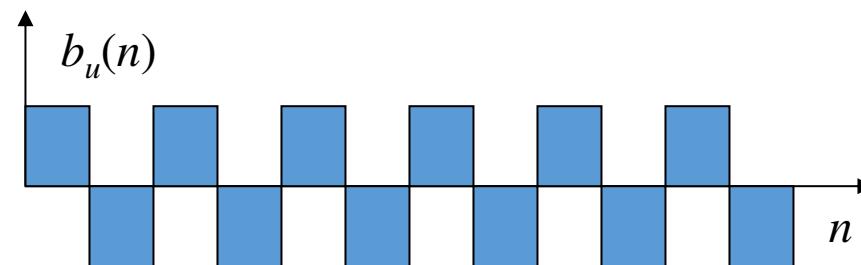
$$(241, -181) * \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^T = (60, 422)$$

Anmerkung:

Das Resultat der Rücktransformation muss skaliert werden, weil die Basis nicht normiert ist.

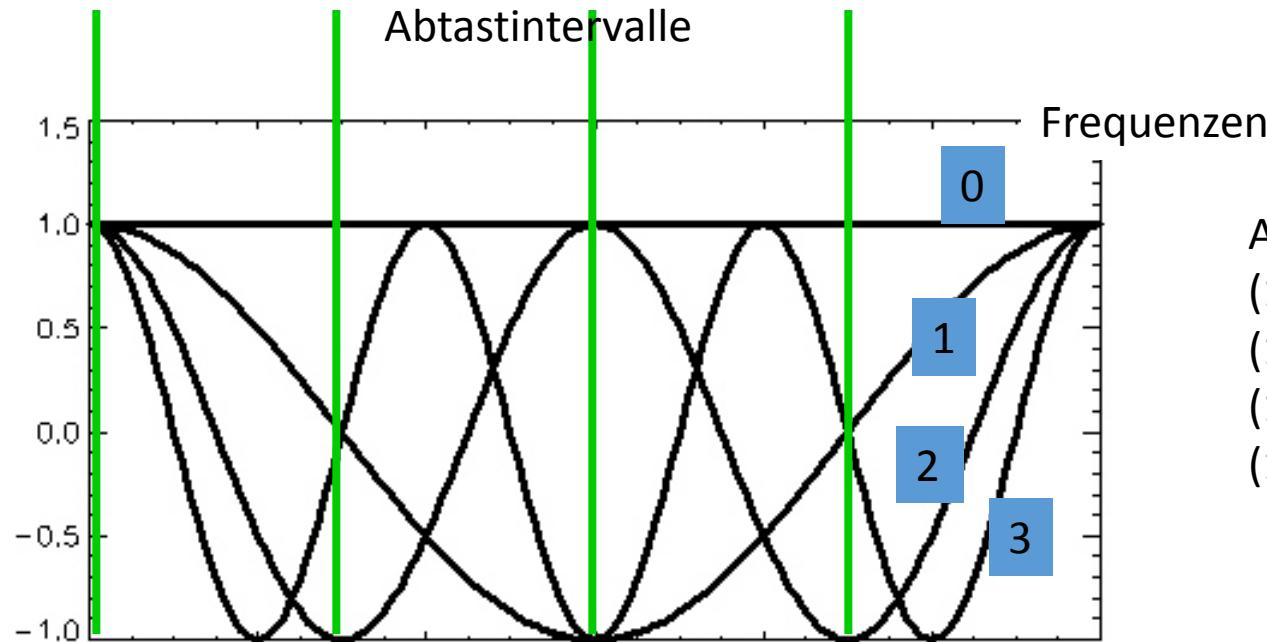
Vollständige Basis

- Basisfunktionen seien Kosinuskurven mit Frequenzen $0,1,2,\dots$: $b_u(n) = \cos(nu \cdot 2\pi / N)$, $u=0,1,2,\dots$
- Problem:
Die maximal repräsentierbare Schwingungsanzahl für eine Funktion mit N Werten ist $N/2$
► Anzahl der Basisfunktionen ist $N/2$.



- Zusätzlich benötigte Basisfunktionen sollten die gleiche Semantik haben.

Beispiel: 4 Basisfunktionen

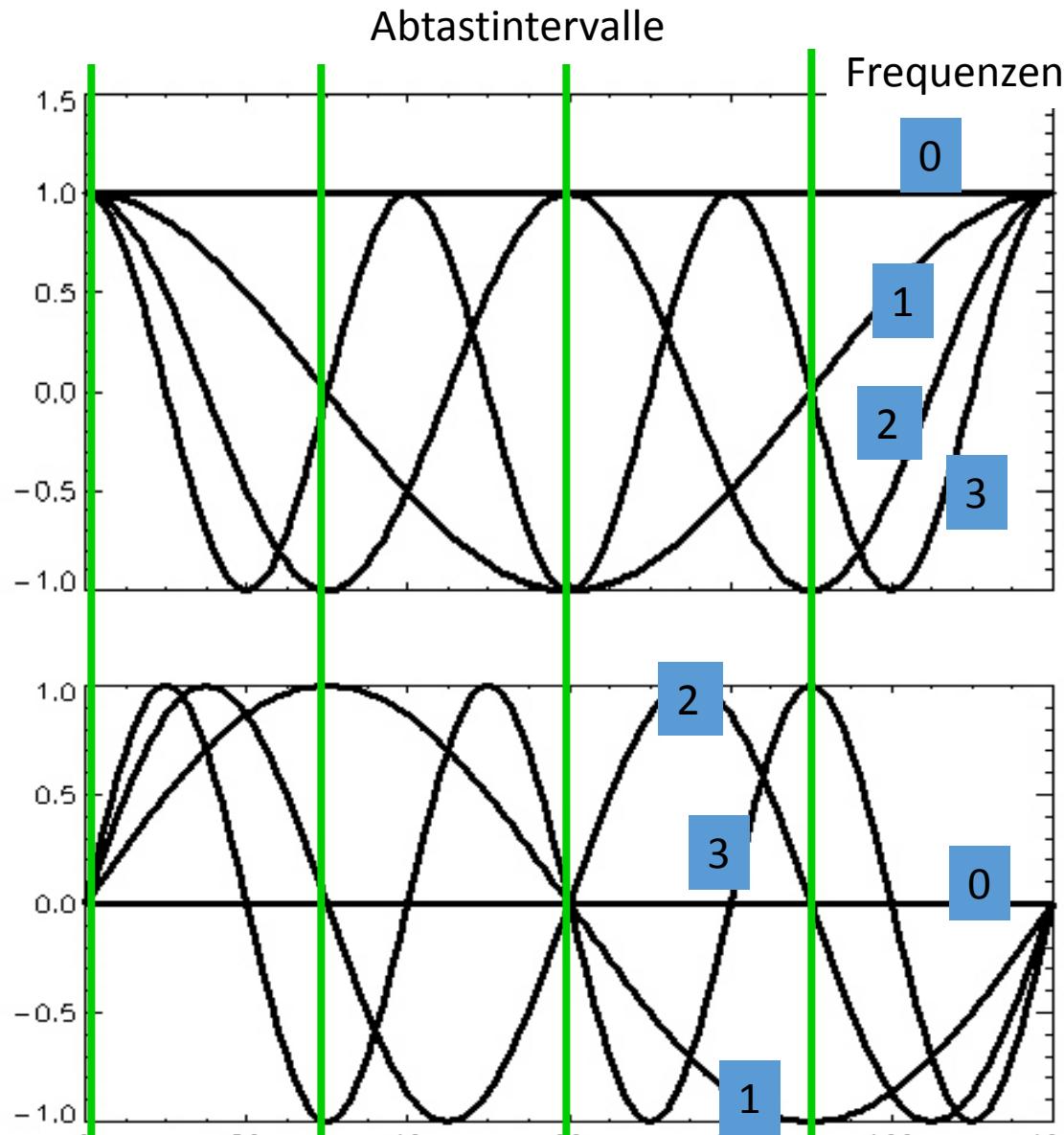


Abgetastete Kosinuswellen:

$$\begin{aligned} & (1 \ 1 \ 1 \ 1) \\ & (1 \ 0 \ -1 \ 0) \\ & (1 \ -1 \ 1 \ -1) \\ & (1 \ 0 \ -1 \ 0) \end{aligned}$$

Lösungen:

- frequenzverschobene Perioden (DCT).
- komplexe periodische Funktionen (FT).



Basisfunktionspaare

Abgetastete Kosinuswellen:

$$\begin{aligned} &(1 \ 1 \ 1 \ 1) \\ &(1 \ 0 -1 \ 0) \\ &(1 -1 \ 1 -1) \\ &(1 \ 0 -1 \ 0) \end{aligned}$$

Abgetastete Sinuswellen:

$$\begin{aligned} &(0 \ 0 \ 0 \ 0) \\ &(0 \ 1 \ 0 -1) \\ &(0 \ 0 \ 0 \ 0) \\ &(0 -1 \ 0 \ 1) \end{aligned}$$

Komplexe periodische Funktionen

1. Projektion auf Vektor von Basisfunktionen:

- $\vec{b}_u(n) = [b_{u,\cos} \quad b_{u,\sin}] = [\cos(nu \cdot 2\pi / N) \quad \sin(nu \cdot 2\pi / N)]$
- $\vec{F}(u) = [\vec{f} \times \mathbf{B}_{\cos} \quad \vec{f} \times \mathbf{B}_{\sin}]$

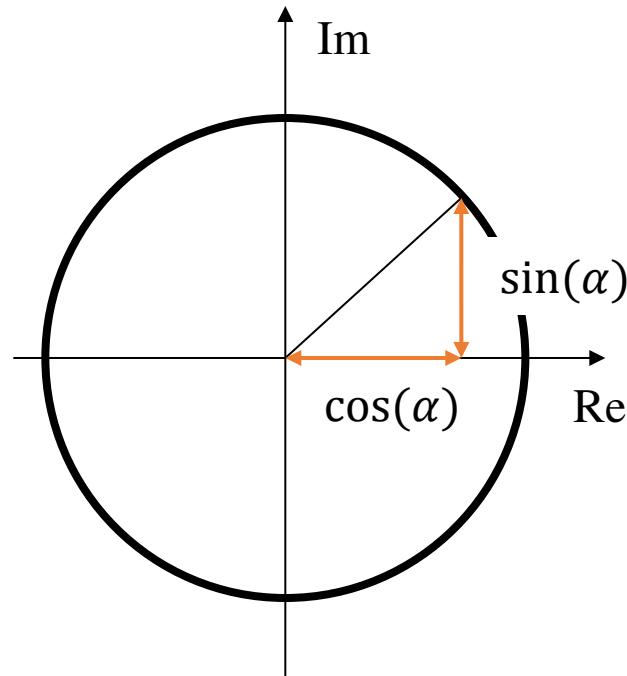
2. Definition des Skalarprodukts:

- Betrachtung der beiden Komponenten des Vektors als Real- und Imaginärteil

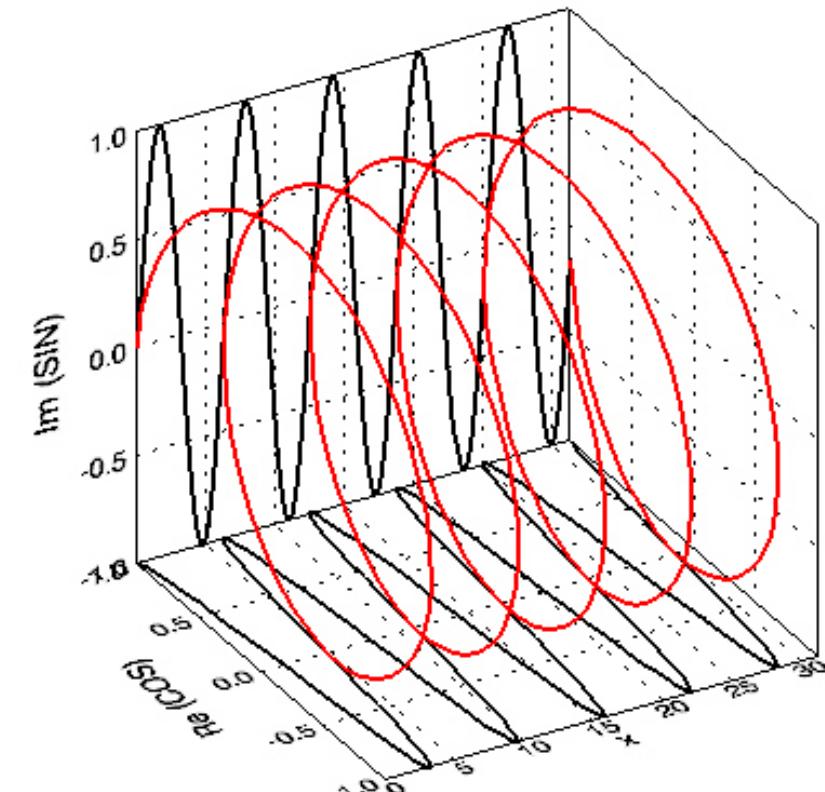
$$\vec{b}_u(n) = \cos(nu \cdot 2\pi / N) + i \cdot \sin(nu \cdot 2\pi / N)$$

- Komplexes Skalarprodukt

$$\cos(\alpha) + i \sin(\alpha)$$



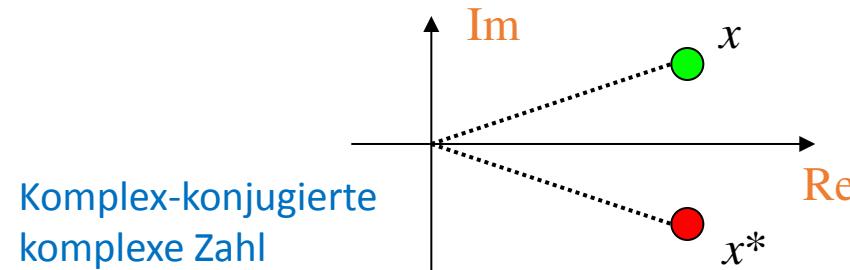
Alle Werte für komplexe Zahlen der Form $\cos(\alpha) + i \cdot \sin(\alpha)$ liegen auf einem Kreis mit Abstand 1 in der komplexen Ebene.



Komplexes Skalarprodukt

Skalarprodukt zwischen zwei Vektoren mit komplexen Elementen:

- Summe der Produkte der Komponenten des ersten Vektors mit der komplex-konjugierten Komponenten des zweiten Vektors.
- Die komplex-konjugierte zu $x=a+i\cdot b$ ist $x^*=a-i\cdot b$.



Skalarprodukt:

$$\vec{x} \bullet \vec{y} = \sum_{i=0}^{N-1} x_i \cdot y_i^* = \sum_{i=0}^{N-1} (\operatorname{Re}(x_i) + i \operatorname{Im}(x_i))(\operatorname{Re}(y_i) - i \operatorname{Im}(y_i))$$

Repräsentation als Exponentialfunktion

- Taylorreihenentwicklung für Kosinus und Sinus:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots \quad \sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

- Taylorreihenentwicklung für e^{ix} :

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots \Rightarrow e^{ix} = 1 + \frac{ix}{1!} + \frac{(ix)^2}{2!} + \frac{(ix)^3}{3!} + \dots$$

- Es gilt daher wegen $i^2 = -1$: $\cos(x) + i \cdot \sin(x) = e^{ix}$
- Phasenverschiebung α kann in komplexen Funktionen als Multiplikation ausgedrückt werden:

$$\cos(x + \alpha) + i \sin(x + \alpha) = \exp(i(x + \alpha)) = \exp(i\alpha) \exp(ix)$$

1-D Basisfunktionen

Bildfunktion: $f(n)$, $n=0, N-1$,

also: N Basisfunktionen

$$b_u(n) = \exp(i \cdot 2\pi/N \cdot n \cdot u), \text{ mit Frequenzen } u=0, N-1$$

$$\text{z.B. } b_0(n) = [(1,0), (1,0), \dots, (1,0)]$$

Transformation **FT** : $\mathbf{FT}(\mathbf{f}) = \mathbf{F} = \mathbf{f} \times \mathbf{B}$ (Vektor-Matrix-Schreibweise)

$$F(u) = \sum_n f(n) \exp\left(-i \cdot 2\pi/N \cdot nu\right), \text{ für alle } u=0, N-1$$

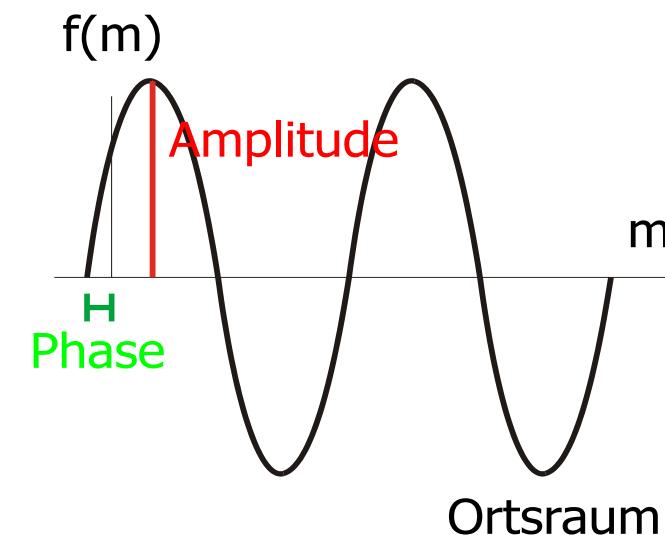
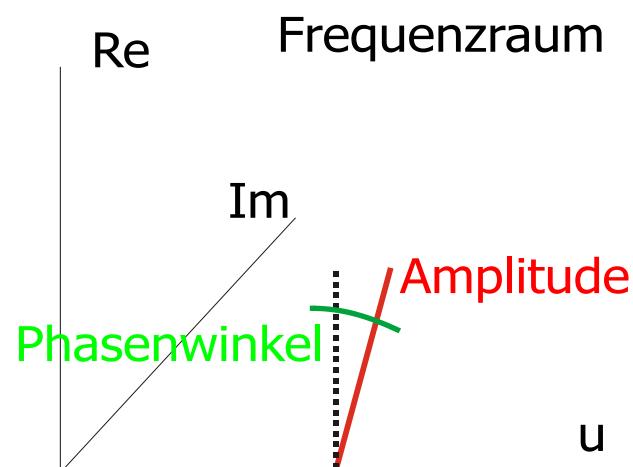
Rücktransformation **FT⁻¹** : $\mathbf{FT}^{-1}(\mathbf{F}) = \mathbf{f} \times \mathbf{B}^T$ (Vektor-Matrix-Schreibweise)

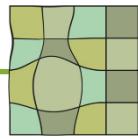
$$f(n) = \frac{1}{N} \sum_u F(u) \exp\left(i \cdot 2\pi/N \cdot nu\right), \text{ für alle } n=0, N-1$$

Skalierungsfaktor, weil die Basisfunktionen nicht normiert sind.

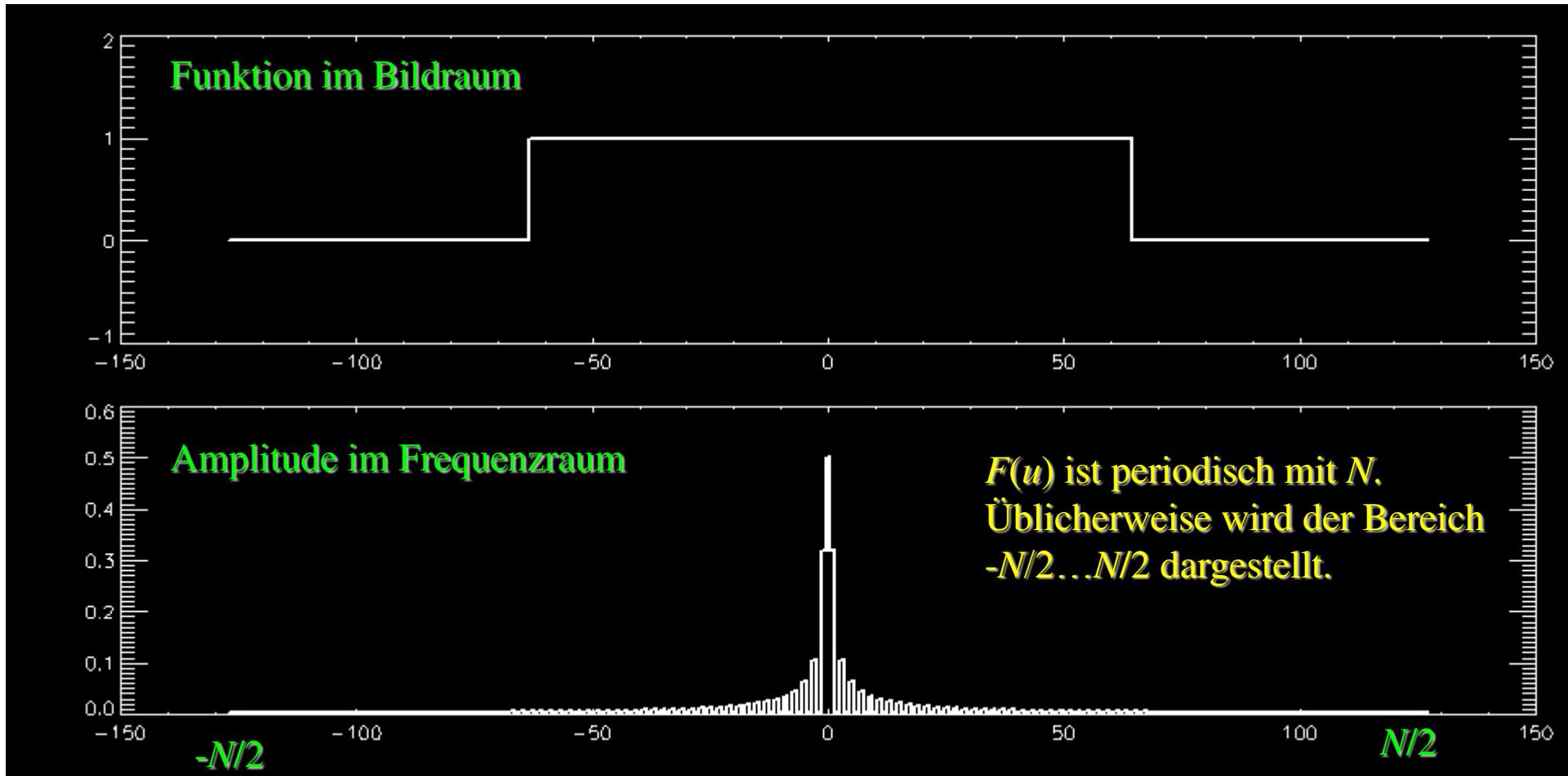
Phase und Amplitude

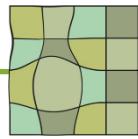
- Das Resultat der Fouriertransformation ist eine komplexe Funktion $F(u)$.
- Der **Betrag** eines Funktionswerts ist die **Amplitude** und der **Winkel** zur reellen Achse ist die **Phase** zur Gewichtung der betreffenden Basisfunktion



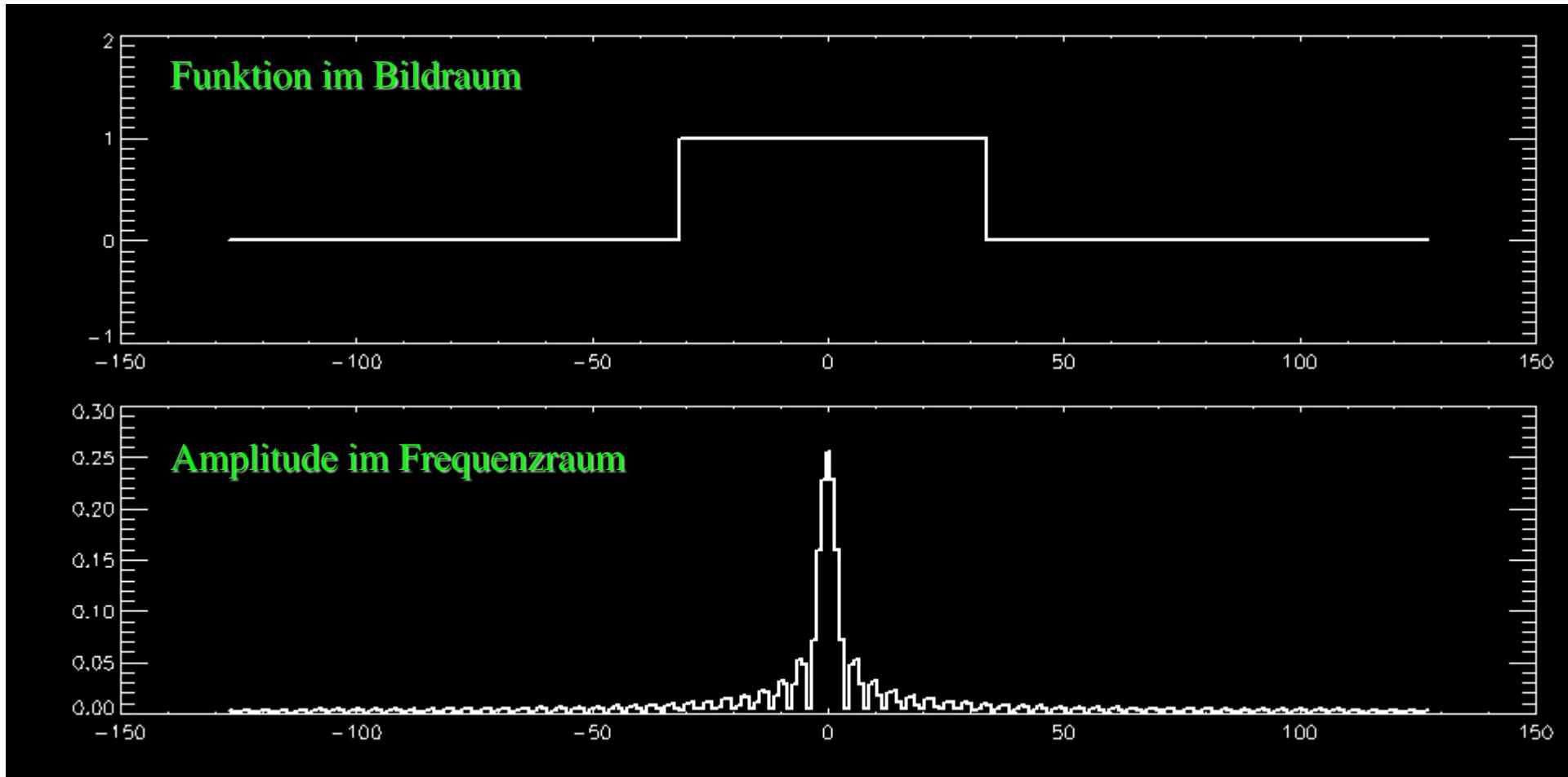


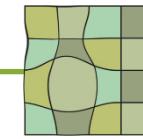
Beispiel 1-D Fouriertransformation



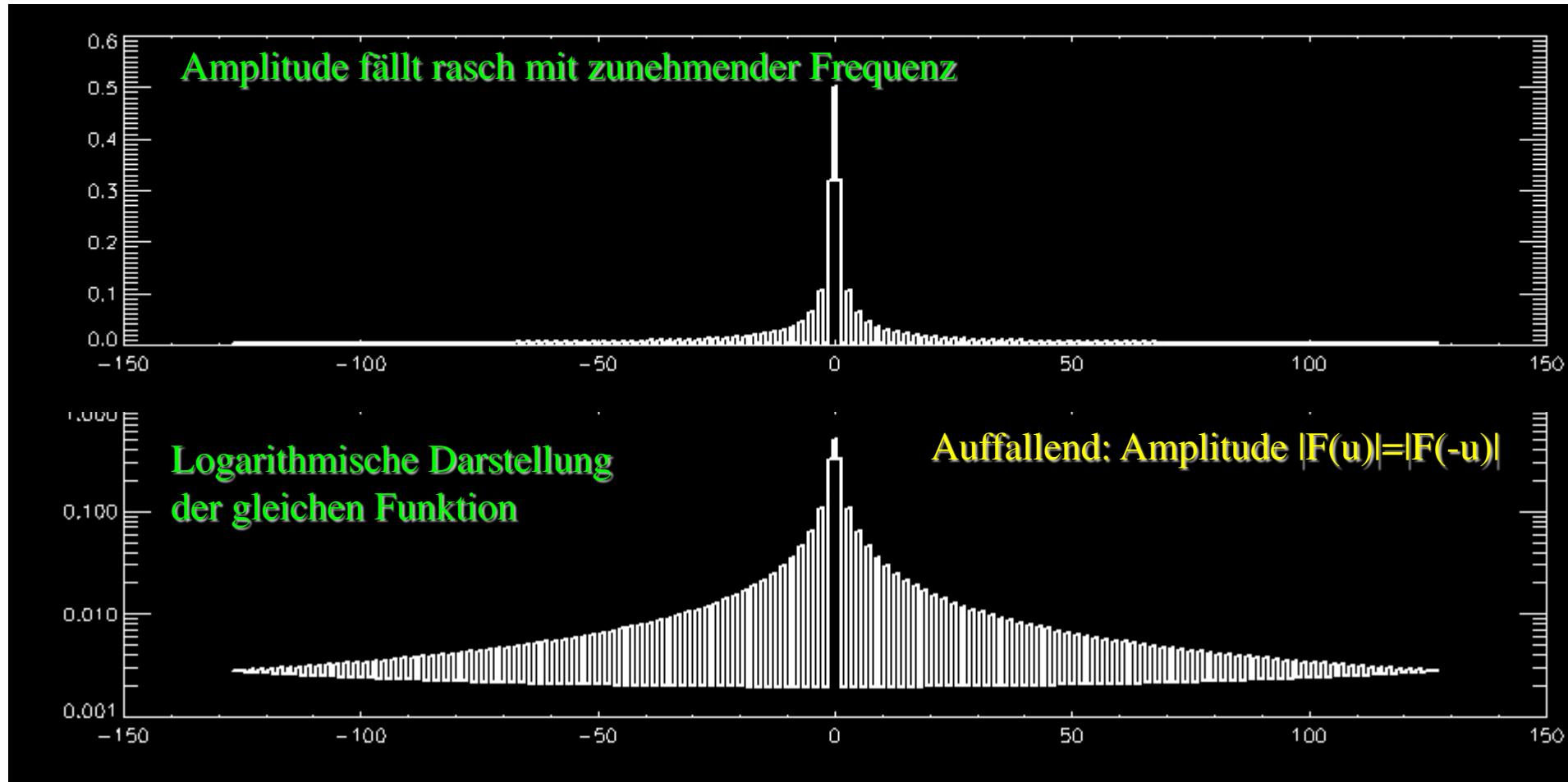


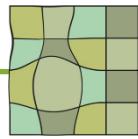
Beispiel 1-D Fouriertransformation



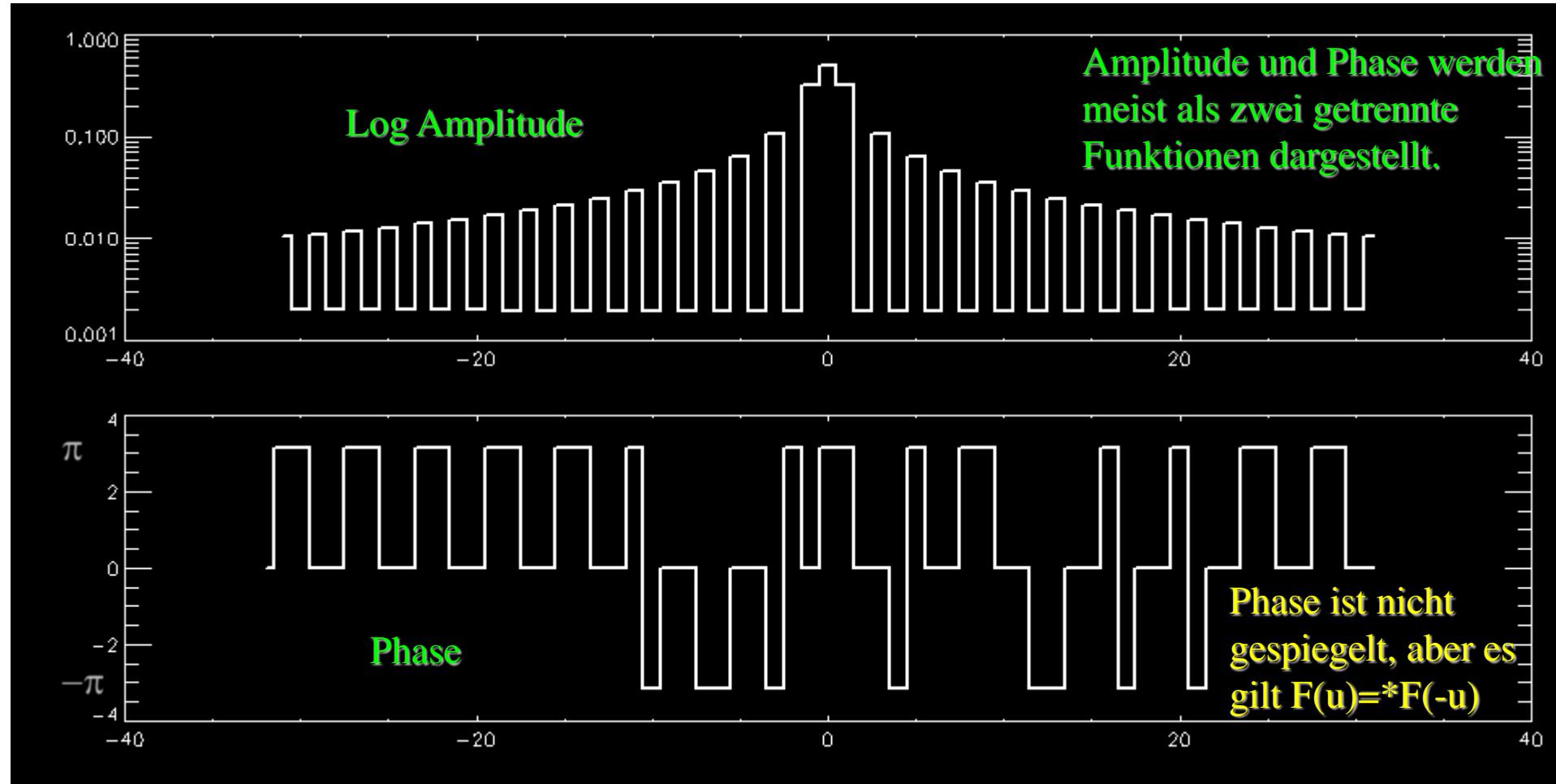


Logarithmische Darstellung der Amplitude

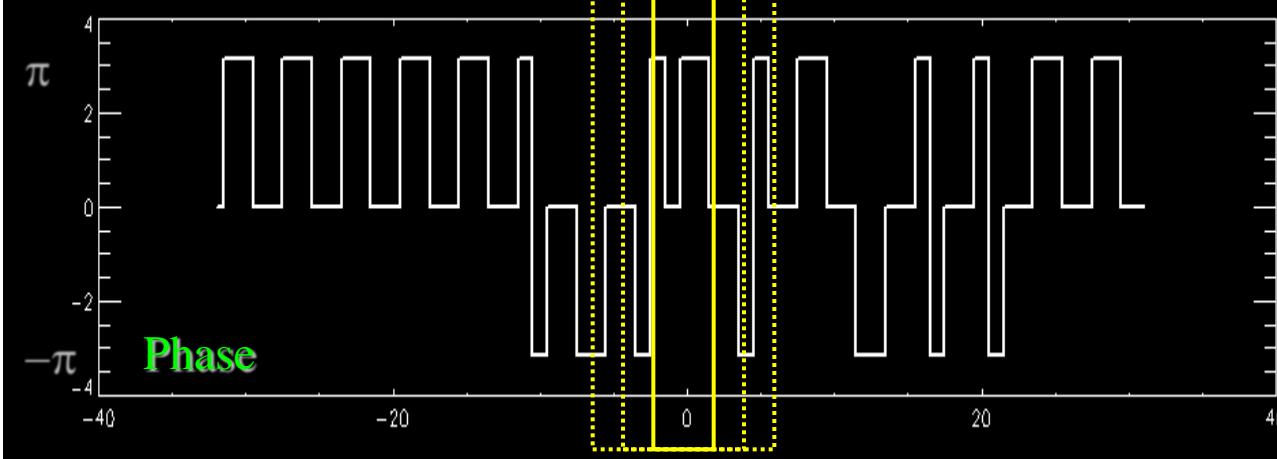
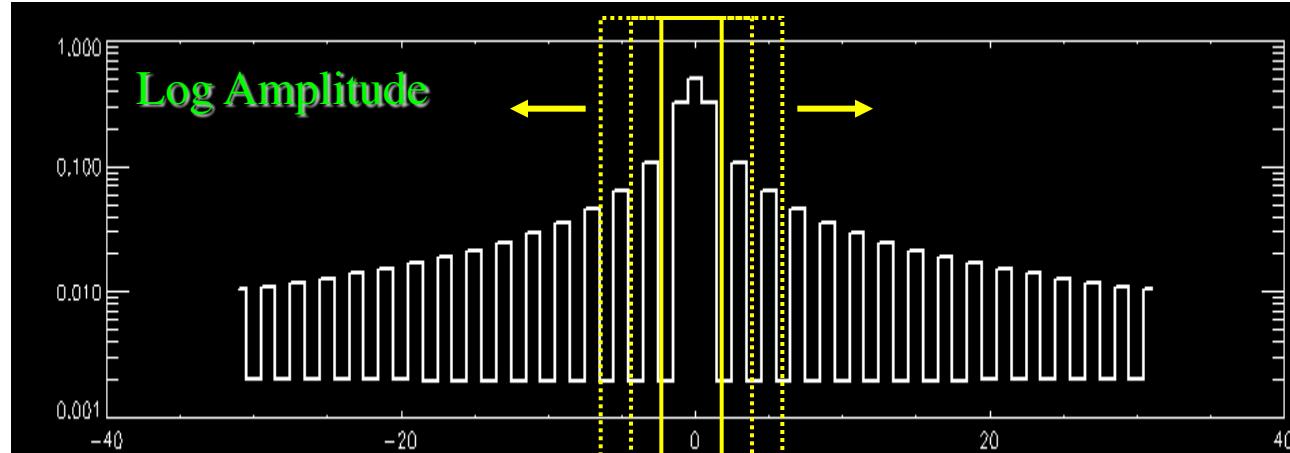




Amplitude und Phase

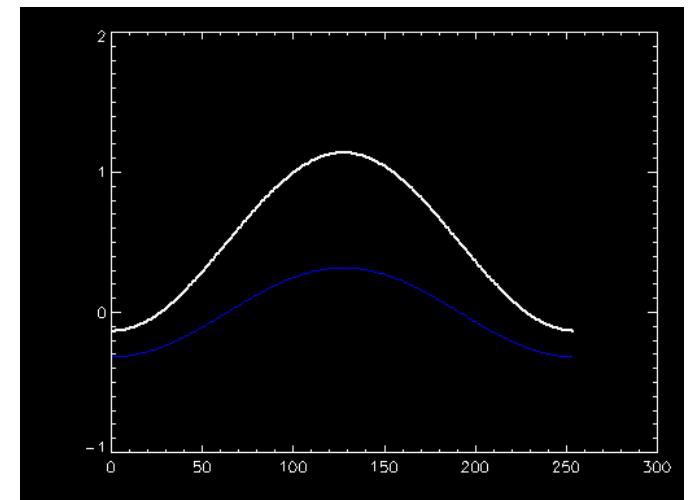


Rücktransformation

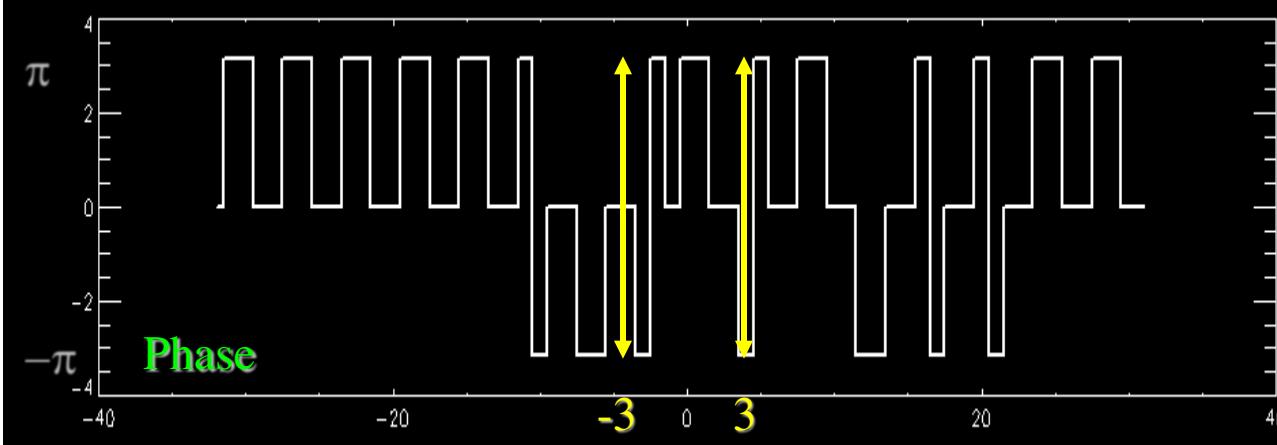
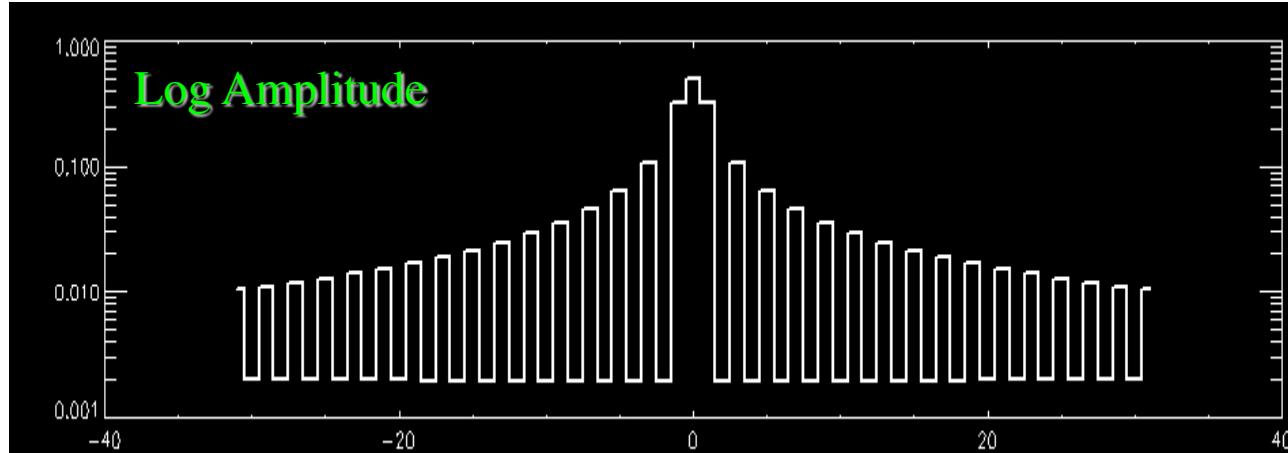


Summation der mit Amplitude und Phase modifizierten Sinus/Kosinuskurven.

Realteil der summierten Wellen

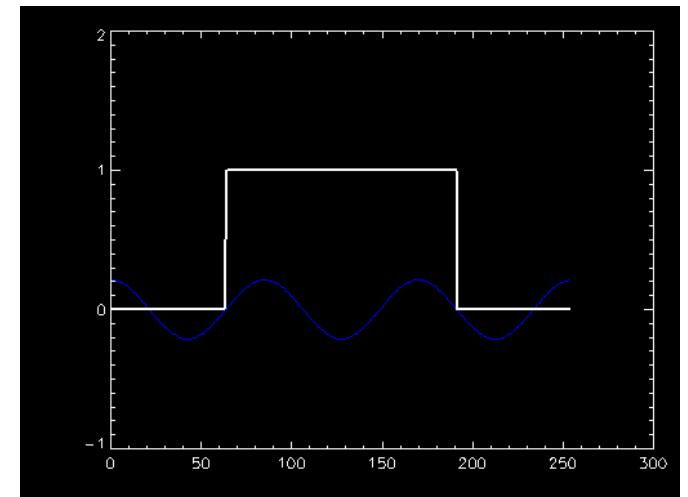


Phasenverschiebung



Summation der mit Amplitude und Phase modifizierten Sinus/Kosinuskurven.

Realteil der summierten Wellen



Was sollten Sie heute gelernt haben?

- Orthogonale Funktionsbasis
- Komplexe periodische Funktionen
- 1-d Fouriertransformation
- Darstellung von Amplitude und Phase

Famous Last Question

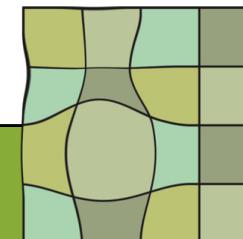
Warum enthält die Fouriertransformierte Funktion $F(u)$ redundante Information?

$F(u)=*F(-u)$, d.h. F ist exakt bestimmt,
falls man die Hälfte der Werte kennt

Grundlagen der Bildverarbeitung

2D – Fouriertransformation

Prof. Dr. Klaus Tönnies



Bildverarbeitung
&
Bildverstehen

2-D Fouriertransformation

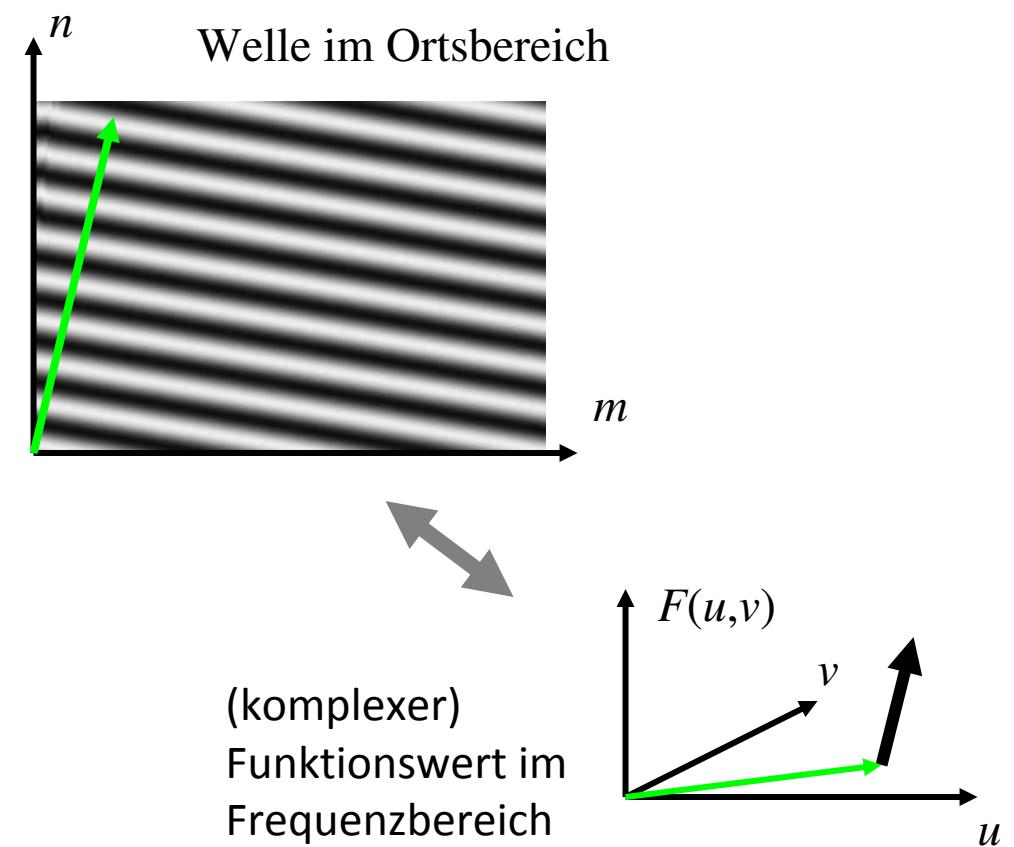
Basisfunktionen sind **Wellen** (Frequenz, Richtung, Amplitude, Phase):

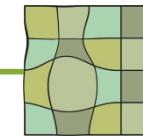
$$\exp\left(i \cdot 2\pi/N \cdot (mu + nv)\right)$$

Richtung ist durch Vektor $(u v)$ gegeben.

Die Basisfunktionen der 2-D Fouriertransformation sind **zerlegbar**:

$$\begin{aligned} \exp\left(i \cdot 2\pi/N \cdot (mu + nv)\right) = \\ \exp\left(i \cdot 2\pi/N \cdot mu\right) \cdot \exp\left(i \cdot 2\pi/N \cdot nv\right) \end{aligned}$$





2-D Fouriertransformationspaar

$$F(u, v) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cdot e^{-i \cdot 2\pi \left(\frac{um}{M} + \frac{vn}{N} \right)}$$
$$f(m, n) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \cdot e^{i \cdot 2\pi \left(\frac{um}{M} + \frac{vn}{N} \right)}$$

Transformationspaar für Bilder
der Größe $M \times N$

Transformationspaar für qua-
dratische Bilder der Größe $N \times N$

$$F(u, v) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} f(m, n) \cdot e^{-i \cdot \frac{2\pi}{N} \cdot (um + vn)}$$
$$f(m, n) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \cdot e^{i \cdot \frac{2\pi}{N} \cdot (um + vn)}$$

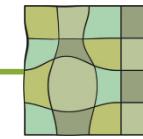
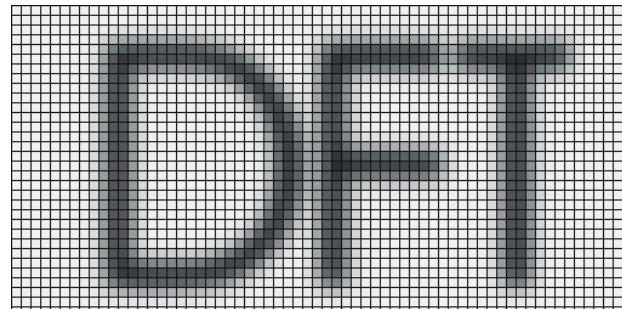


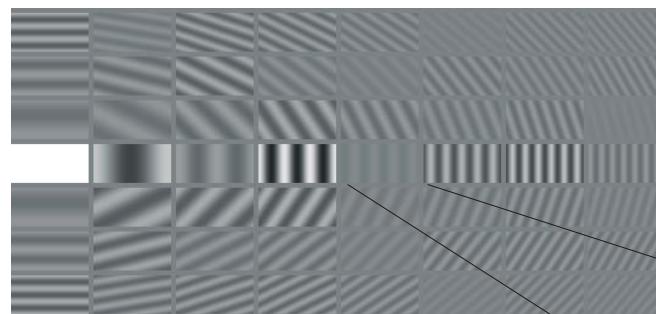
Bild $f(m,n)$



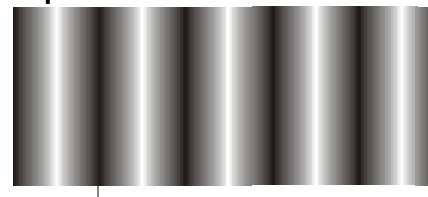
© Jähne, Digitale Bildverarbeitung

Zweidimensionale Fouriertransformation

$$= \frac{1}{MN} \cdot \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \cdot e^{i2\pi \left(\frac{um}{M} + \frac{vn}{N} \right)}$$

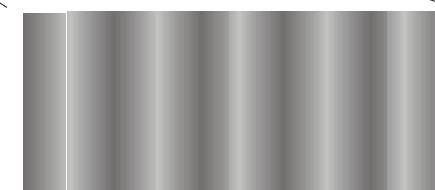


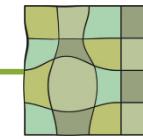
Beispiel einer Basisfunktion:



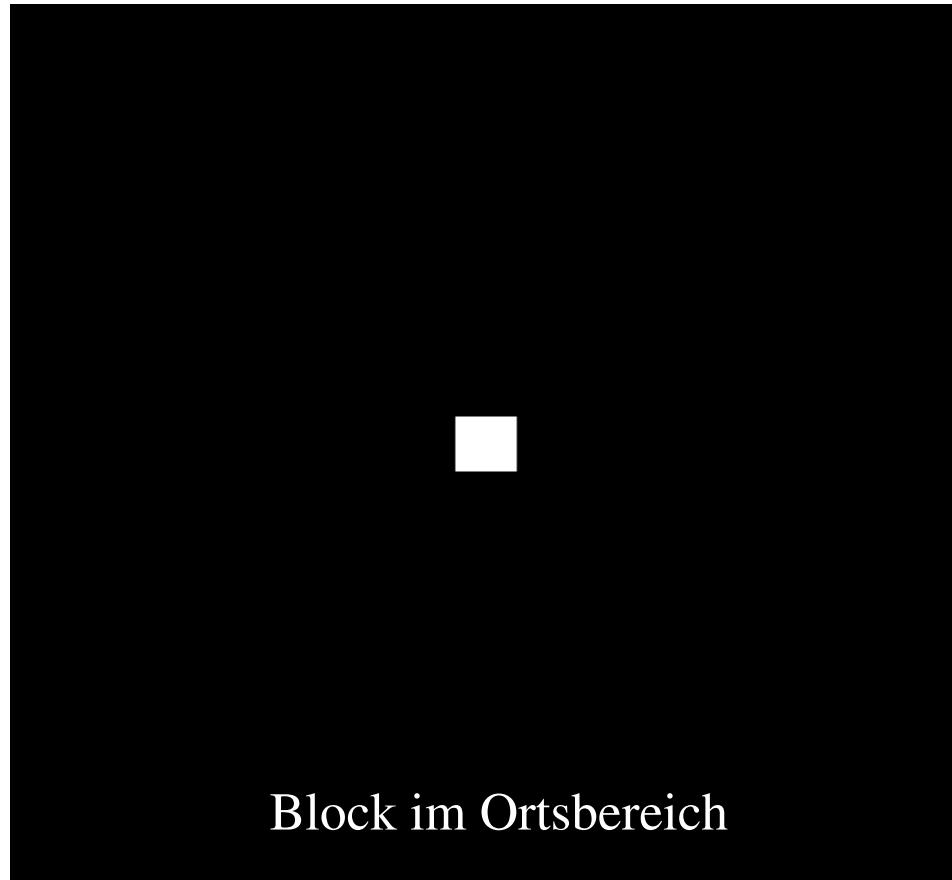
$$\times \quad \boxed{\text{■}} =$$

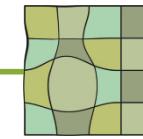
$F(4,0)$



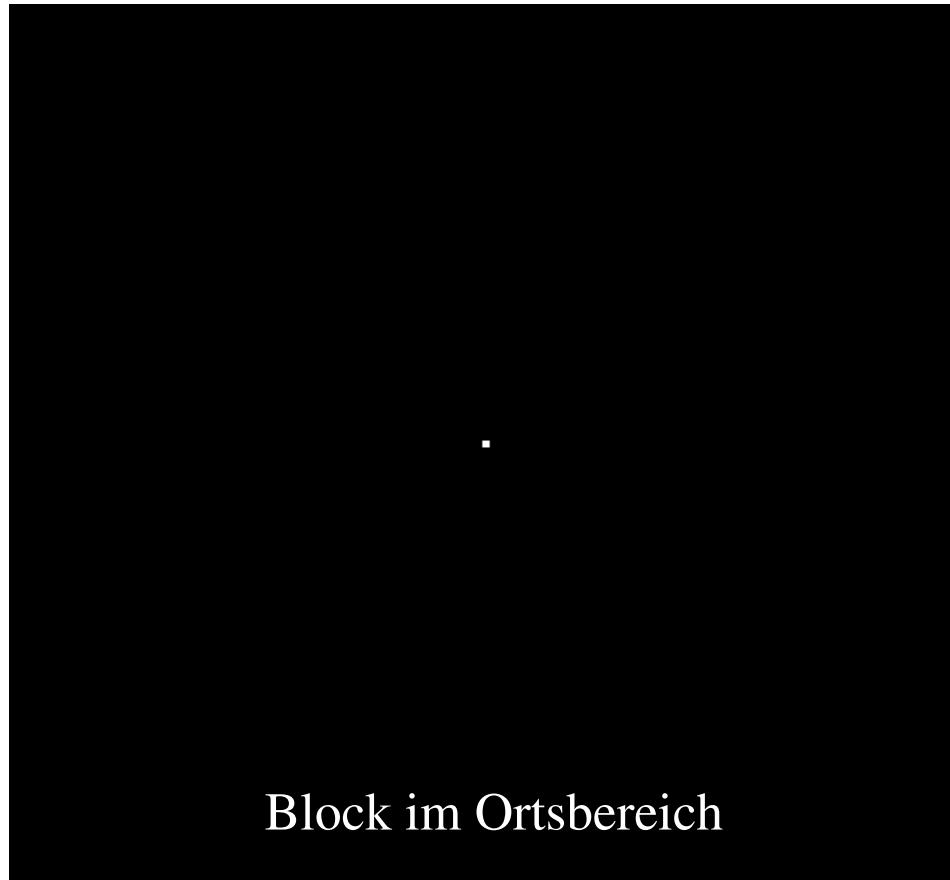


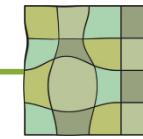
Bilder im Orts- und Frequenzraum





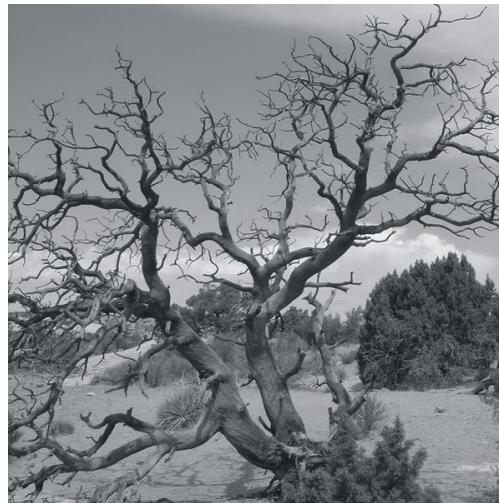
Bilder im Orts- und Frequenzraum



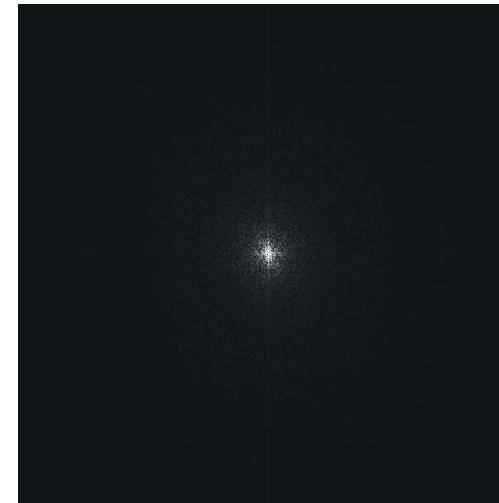


Darstellung der Amplitude

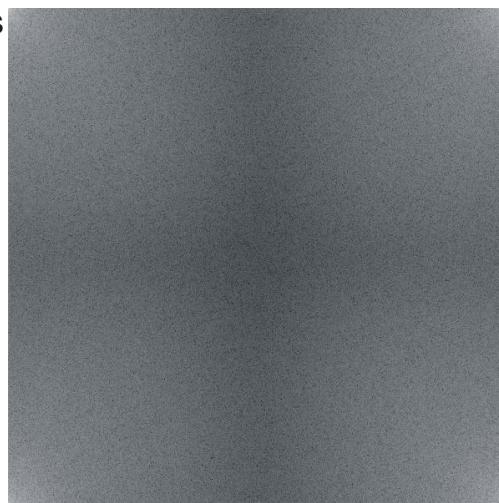
original



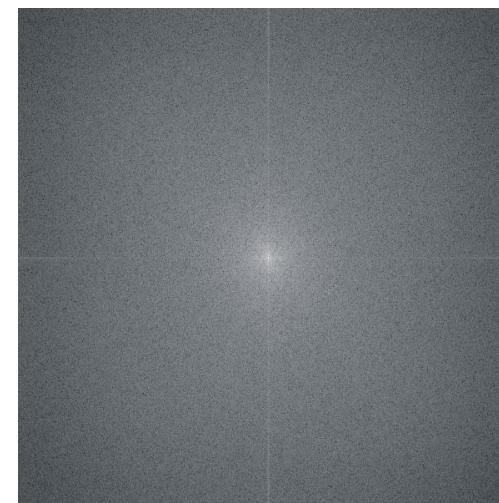
Spektrum
zentriert



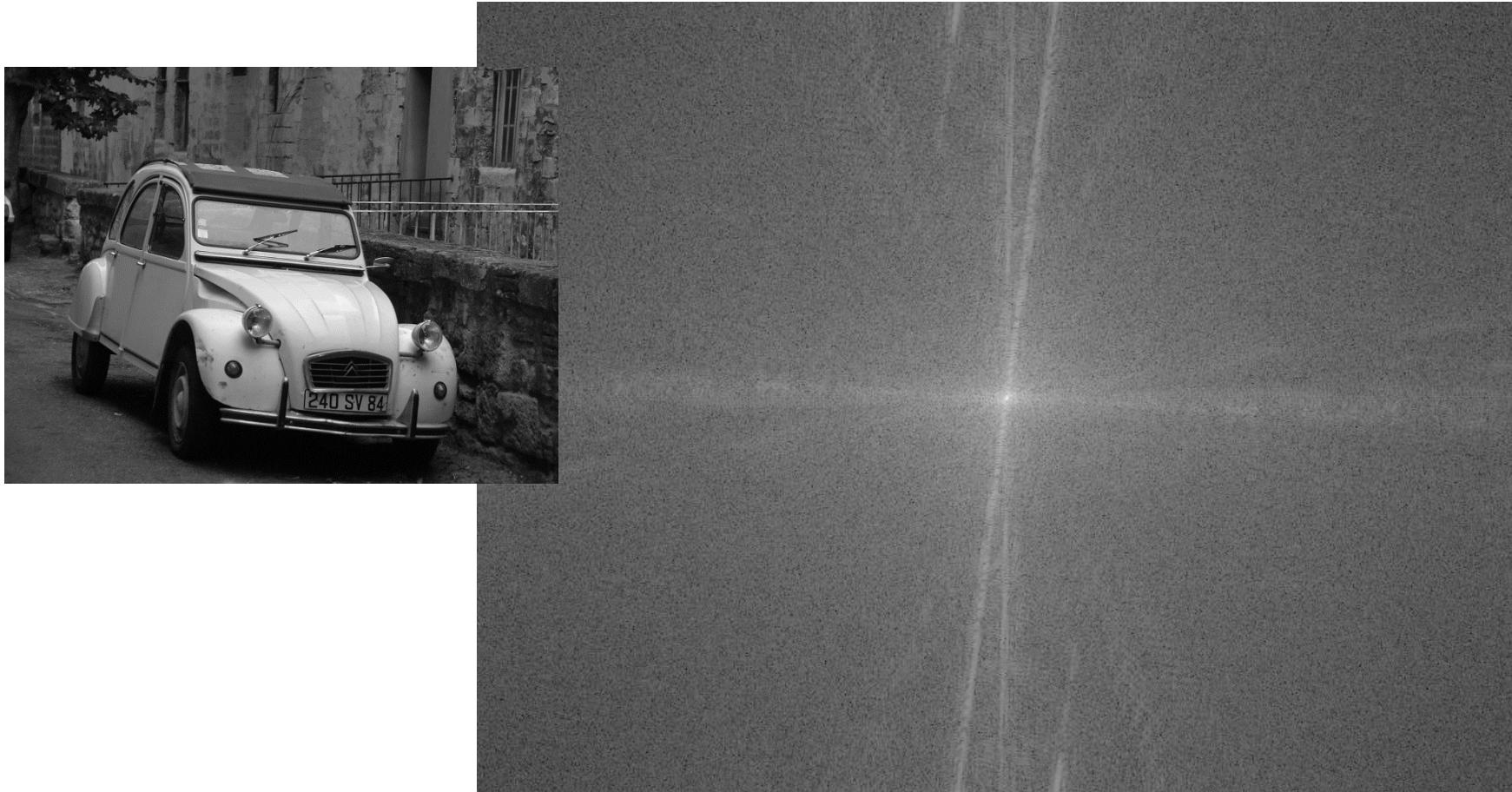
logarithmiertes
Spektrum



logarithmiertes
Spektrum,
zentriert

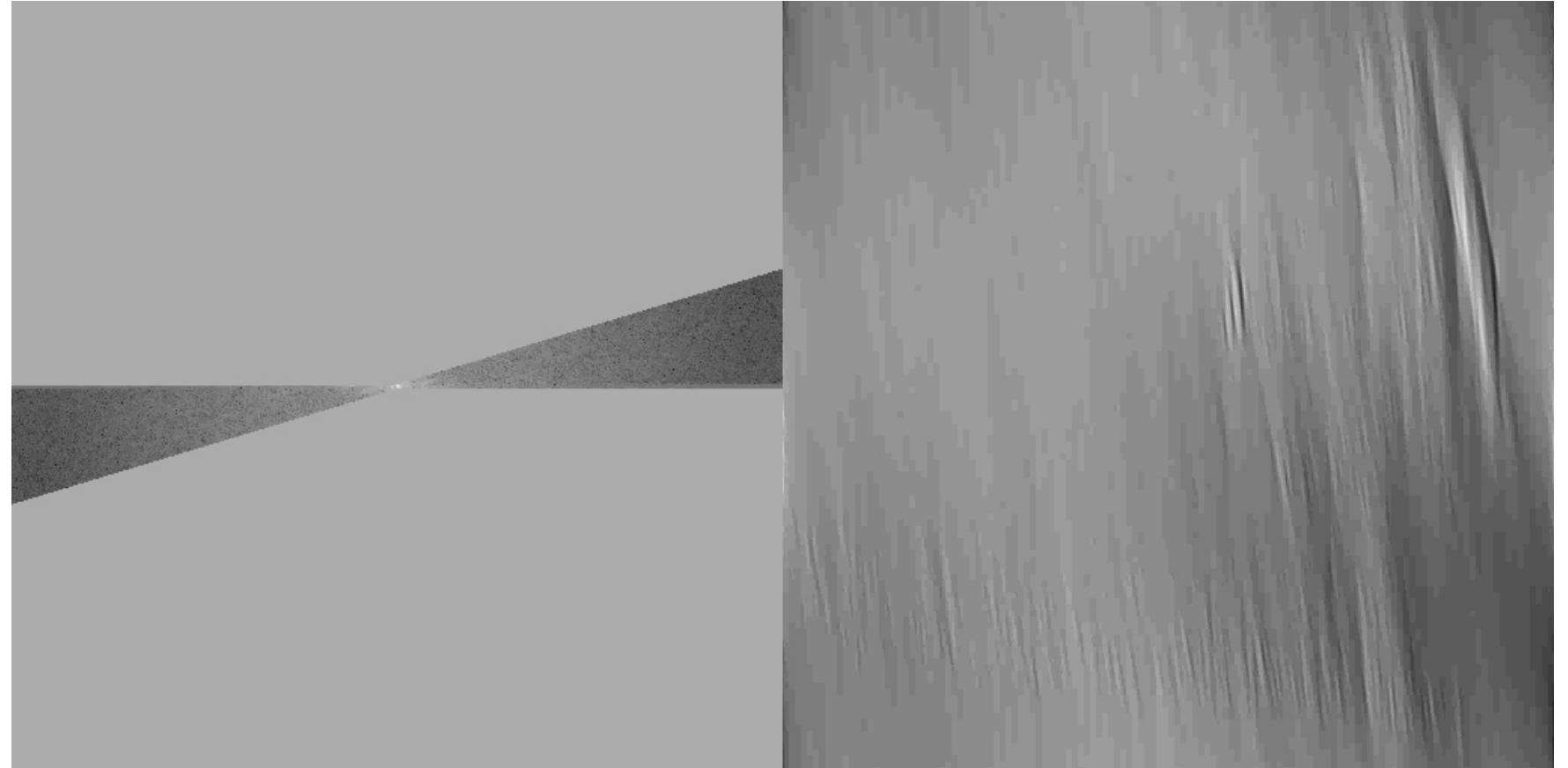


Bilder im Orts- und Frequenzraum



Schrittweise Summation der Komponenten

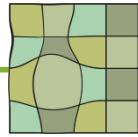
Rekonstruktion



Ausgewählter Bereich im Frequenzraum

Schrittweise Summation der Komponenten





Der Einfluss der Phaseninformation



Amplitude

Rekonstruktion aus
Amplitude allein

Amplitude von
normalverteiltem
Rauschen

Phase



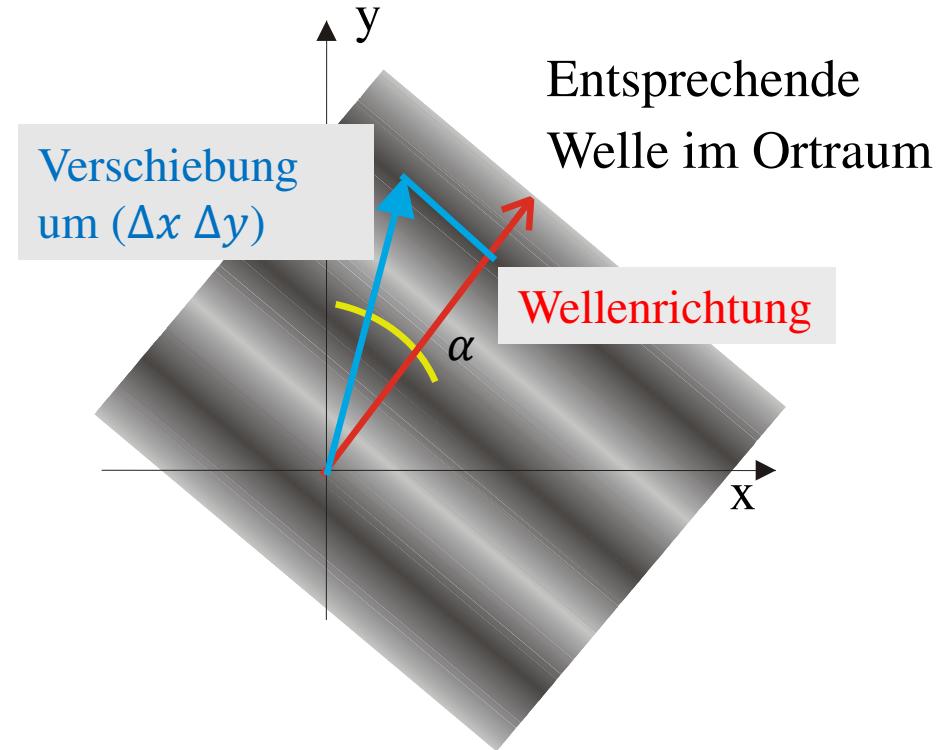
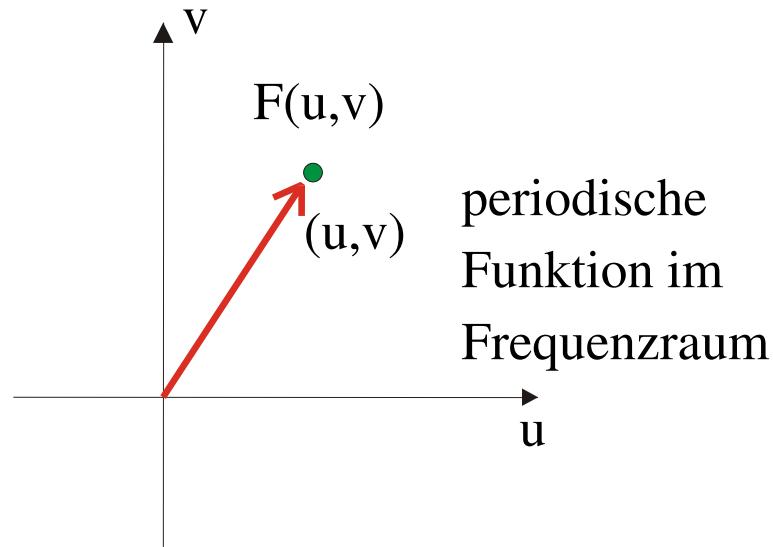
Fouriertransformation für kontinuierliche Funktionen

- Anzahl der Funktionswerte geht gegen unendlich.
- Skalarprodukt zwischen Funktionen f und g ist das Integral der miteinander multiplizierten Funktionen.
- Das Skalarprodukt existiert, wenn die Funktion kontinuierlich und integrierbar ist.
- Fouriertransformation existiert, falls das Skalarprodukt existiert

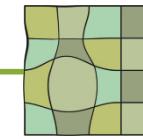
Transformationspaar

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot e^{-i \cdot 2\pi \cdot (ux + vy)} dx dy$$
$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) \cdot e^{i \cdot 2\pi \cdot (ux + vy)} du dv$$

Verhalten: Translation



- Translation im Ortsbereich führt zu einer Translation der zusammensetzenden Wellen.
- Umfang der Translation hängt vom Unterschied zwischen Wellenrichtung ($u v$) und Translationsrichtung ($x y$) ab.
- Im Frequenzbereich bedeutet die Translation eine Phasenverschiebung.



Berechnung der Phasenverschiebung

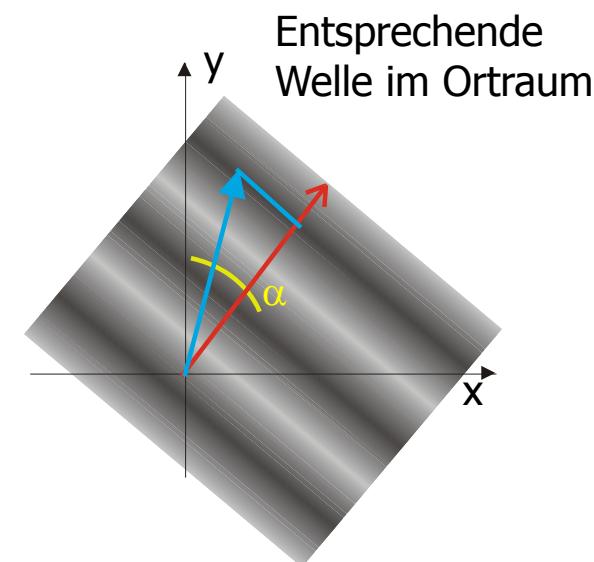
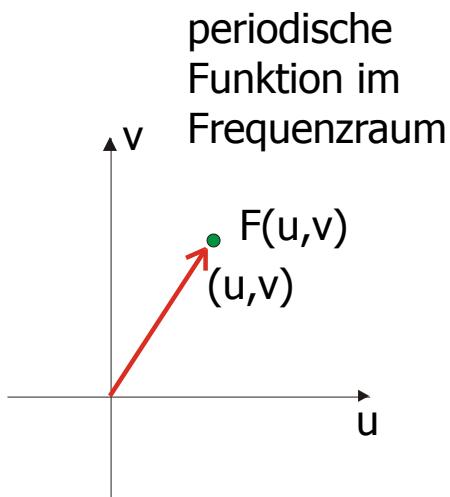
$$p_{u,v} = \frac{|(\Delta x \quad \Delta y)| \cdot \cos(\alpha)}{T_{u,v}}$$

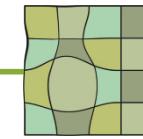
mit $\cos(\alpha)$ - Winkel zwischen Wellenrichtung und Richtung von $(\Delta x \quad \Delta y)$
 $T_{u,v}$ - Wellenlänge = 1/Frequenz

$$\cos(\alpha) = \frac{(\Delta x \quad \Delta y) \bullet (u \quad v)}{|(\Delta x \quad \Delta y)| \cdot |(u \quad v)|}$$

$$T_{u,v} = \frac{N}{2\pi} |(u \quad v)|$$

$$\Rightarrow p_{u,v} = i \cdot \frac{2\pi}{N} \cdot (u\Delta x + v\Delta y)$$

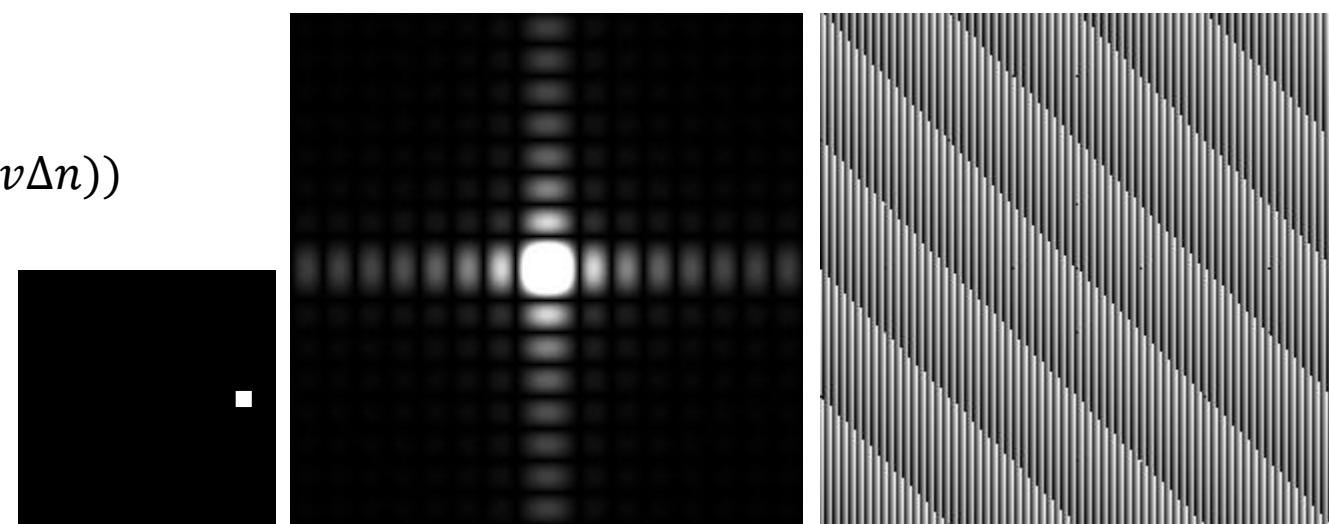
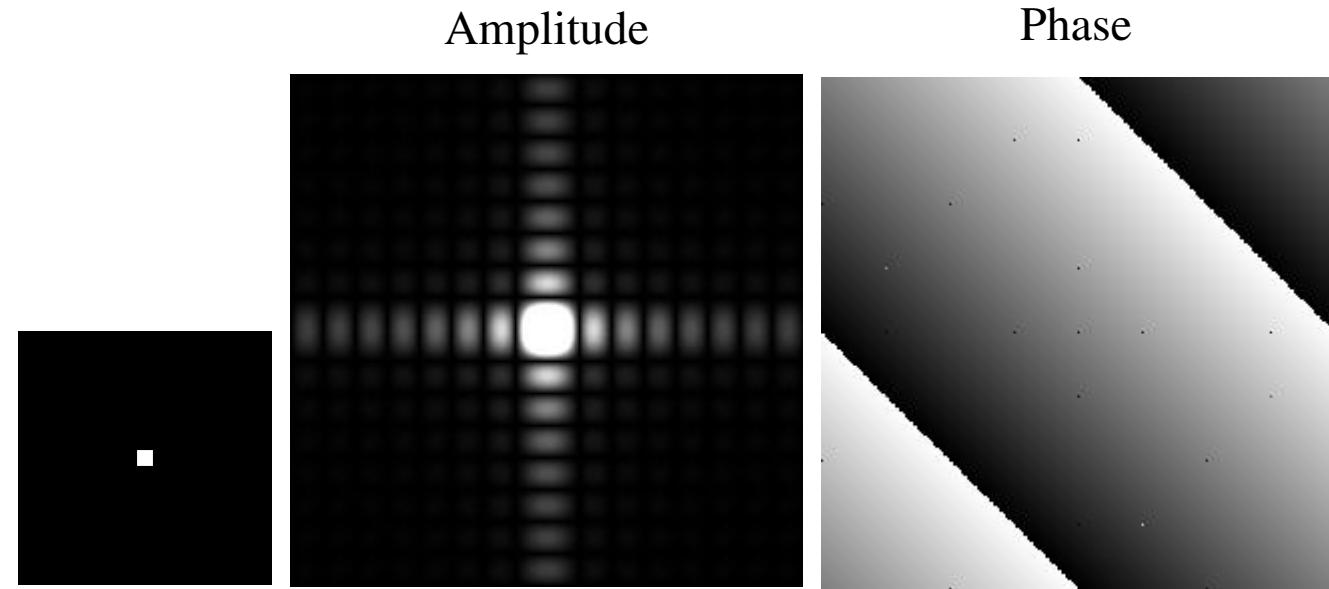




Translation (Beispiel)

Translation um $(\Delta m, \Delta n)$ führt zu einer Phasenverschiebung

$$F'(u, v) = F(u, v) \exp\left(-i \frac{2\pi}{N} (u\Delta m + v\Delta n)\right)$$



Translation um $M/2$

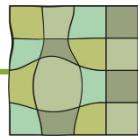
- Für eine diskrete Fouriertransformation auf einer Funktion mit M Werten gilt

$$F\left(u - \frac{N}{2}\right) \Leftrightarrow f(m)\exp\left(i\frac{2\pi}{N}\frac{N}{2}m\right) = f(m)\exp(i\pi m) = f(m)(-1)^m$$

- Für die 2-D Variante gilt entsprechend

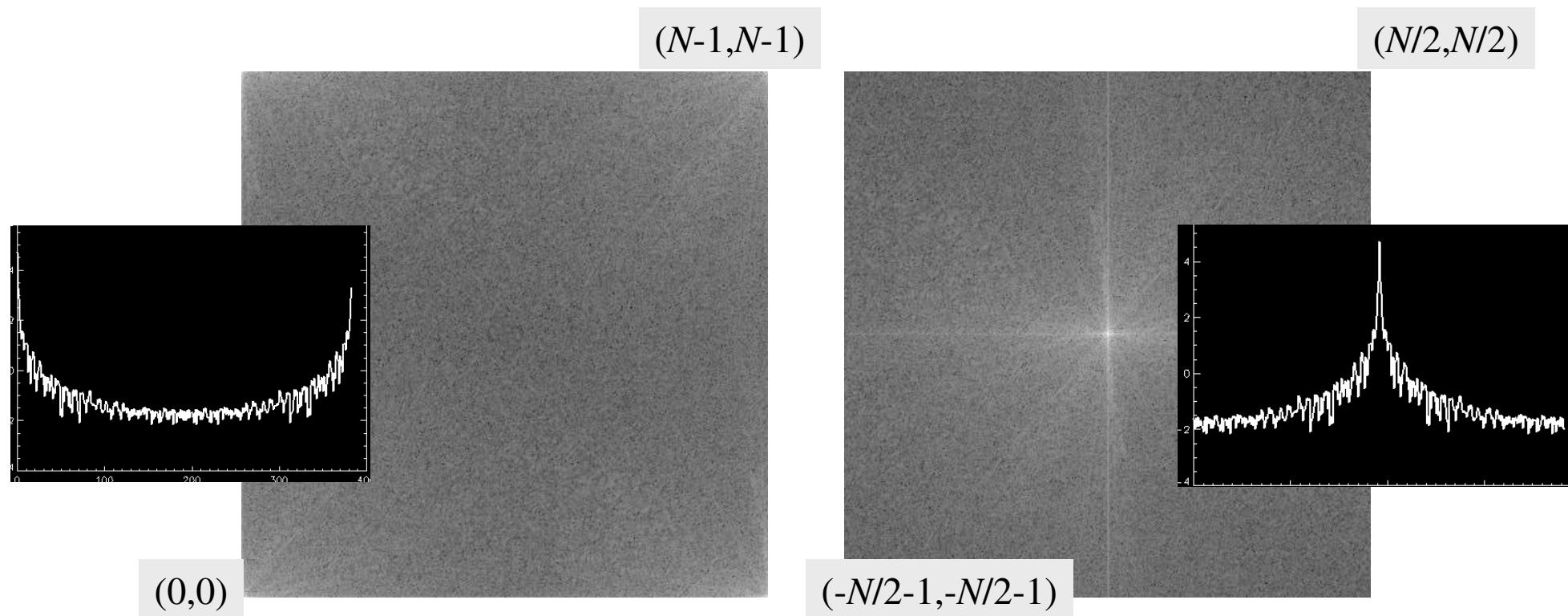
$$\begin{aligned} F\left(u - \frac{N}{2}, v - \frac{N}{2}\right) &\Leftrightarrow f(m, n)\exp\left(i\frac{2\pi}{N}\left(\frac{N}{2}m + \frac{N}{2}n\right)\right) \\ &= f(m, n)\exp(i\pi(m+n)) = f(m, n)(-1)^{m+n} \end{aligned}$$

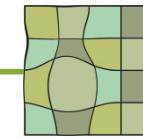
Verschiebung von $(0..N-1, 0..N-1)$ nach $(-N/2..N/2-1, -N/2..N/2-1)$ ändert nur die Phase!



Translation um $M/2$ (Beispiel)

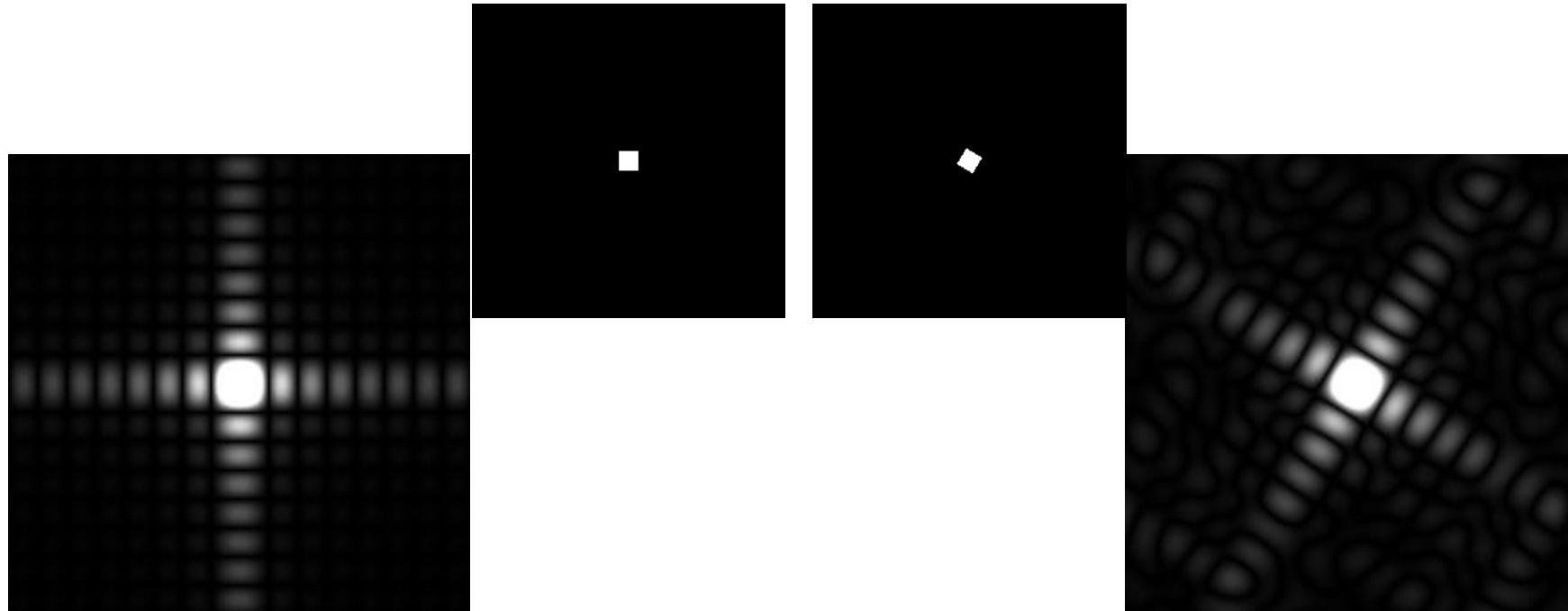
$|F(u,v)|$, $u=-N/2 \dots +N/2-1$, $v=-N/2 \dots +N/2-1$ ist gleichwertig zu $|F(u,v)|$, $u=0, \dots, N-1$, $v=0, \dots, N-1$.

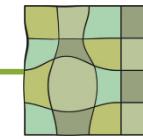




Rotation

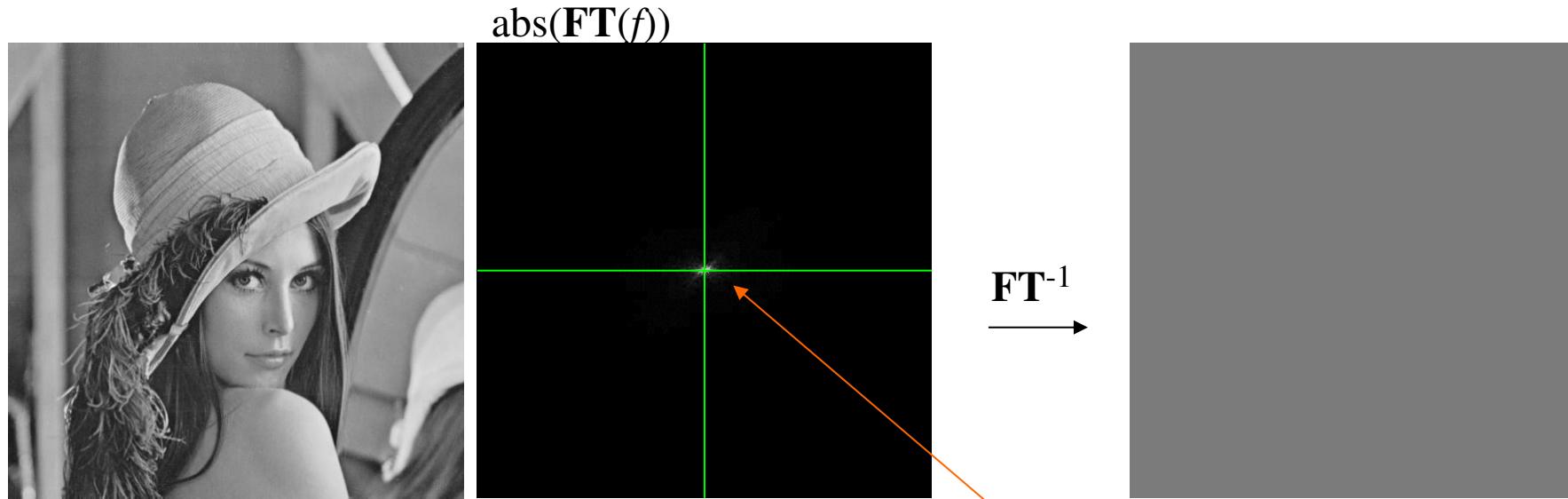
Rotation: $F(u,v)$ wird in gleicher Weise rotiert wie $f(m,n)$.





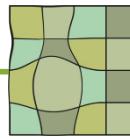
Mittelwert der Funktion

$$\begin{aligned} F(0,0) &= \frac{1}{N^2} \sum_n \sum_m f(m,n) \exp\left(-i \cdot 2\pi \cdot \frac{0n + 0m}{N}\right) \\ &= \frac{1}{N^2} \sum_n \sum_m f(m,n) \exp(0) \\ &= f_{avg} \end{aligned}$$

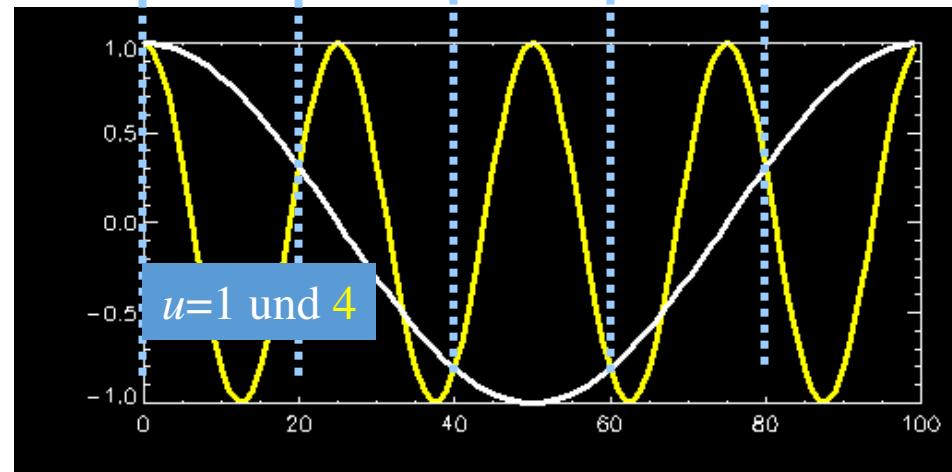
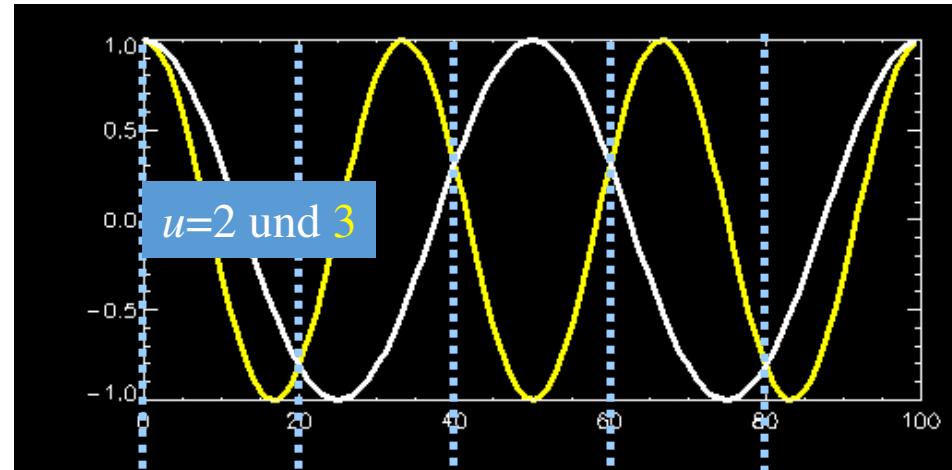


Periodizität und Symmetrie

- Für ein- und zweidimensionale Funktionen mit M bzw. M und N Werten gilt:
 - $F(u) = F(u+M)$, $f(m)=f(m+M)$
 - $F(u,v) = F(u+M,v) = F(u,v+N) = F(u+M,v+N)$
 - $f(m,n) = f(m+M,n) = f(m,n+N) = f(m+M,n+N)$
- Für reellwertige Funktionen f gilt für die Fouriertransformierte:
 - $F(u) = *F(-u)$
 - $F(u,v) = *F(-u,-v)$(reduziert die zu berechnenden Werte auf die Hälfte)
** $x=a-ib$ ist die *komplex-konjugierte* der komplexen Zahl $a=a+ib$.*



Symmetrie der abgetasteten Kosinusfunktion

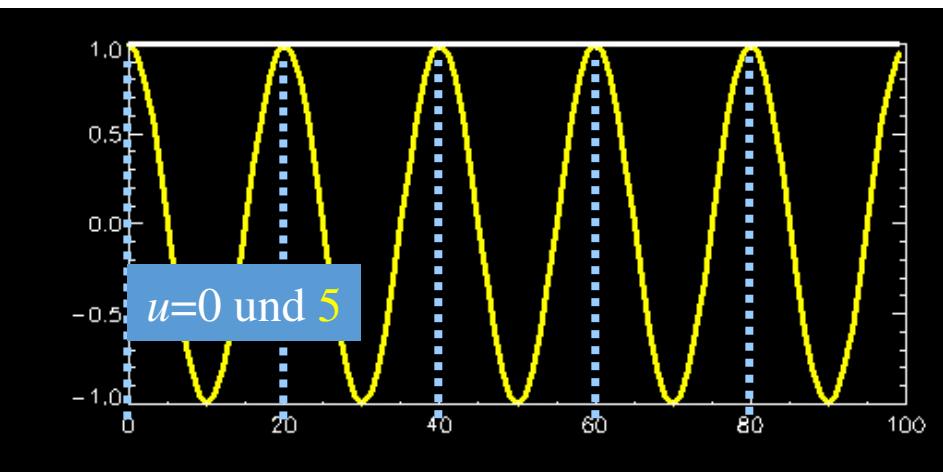


Beispiel für $M=5$:

$$\left| \exp\left(i \frac{2\pi}{5} 2m\right) \right| = \left| \exp\left(i \frac{2\pi}{5} (-2)m\right) \right| = \left| \exp\left(i \frac{2\pi}{5} 3m\right) \right|$$

$$\left| \exp\left(i \frac{2\pi}{5} 1m\right) \right| = \left| \exp\left(i \frac{2\pi}{5} (-1)m\right) \right| = \left| \exp\left(i \frac{2\pi}{5} 4m\right) \right|$$

$$\left| \exp\left(i \frac{2\pi}{5} 0m\right) \right| = \left| \exp\left(i \frac{2\pi}{5} 5m\right) \right|$$

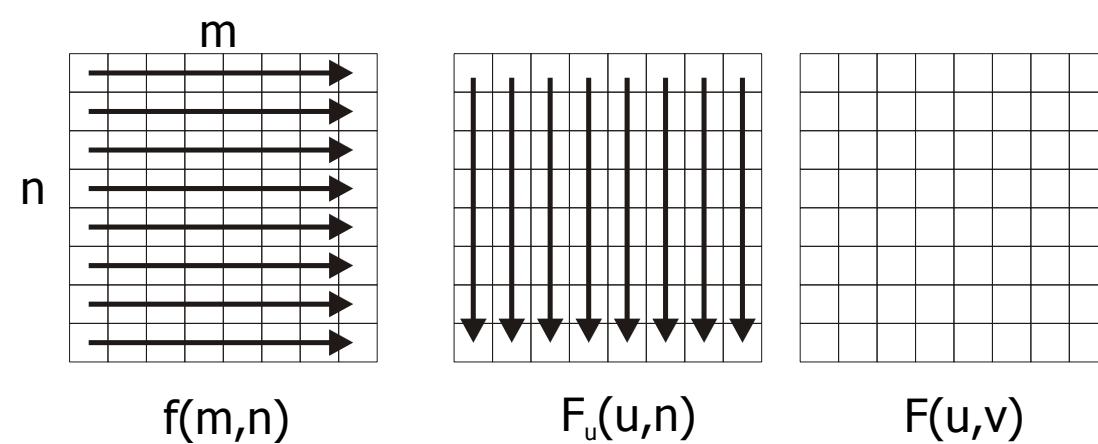


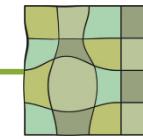
Separabilität

Die Fouriertransformation ist *separabel*, d.h., sie kann zunächst in M-Richtung und anschließend auf diesen Zwischenergebnissen in N-Richtung ausgeführt werden.

$$\begin{aligned}
 F(u, v) &= \frac{1}{N^2} \sum_n \sum_m f(m, n) \exp\left(-i \cdot 2\pi \cdot \frac{um + vn}{N}\right) = \frac{1}{N^2} \sum_n \sum_m f(m, n) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \exp\left(-i \cdot 2\pi \cdot \frac{vn}{N}\right) \\
 &= \frac{1}{N^2} \sum_n \left[\sum_m f(m, n) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right] \exp\left(-i \cdot 2\pi \cdot \frac{vn}{N}\right) \quad \text{kann aus der inneren Summe ausgeklammert werden.} \\
 &= \frac{1}{N^2} \sum_n F_u(m, n) \exp\left(-i \cdot 2\pi \cdot \frac{vn}{N}\right)
 \end{aligned}$$

Reduziert den Berechnungsaufwand von $O(N^4)$ auf $O(N^3)$.



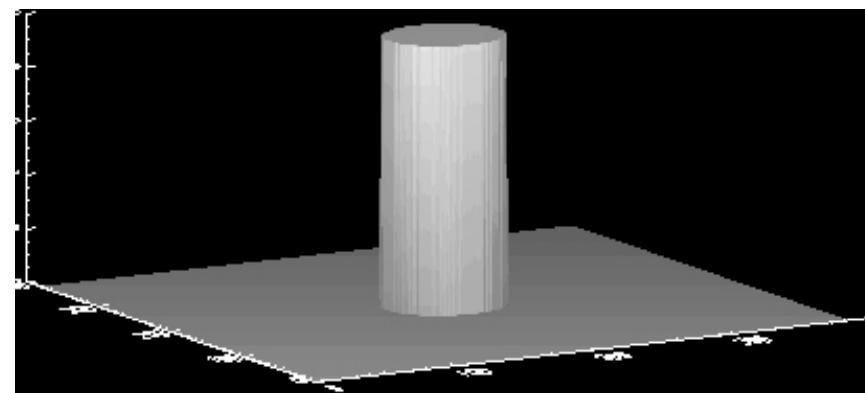


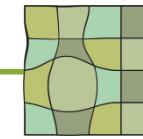
Filterung

- Filterung im Frequenzraum: Veränderung der Funktionswerte vor der Rücktransformation.
- Motivation: Semantische Trennung von Bildern im Frequenzraum, z.B., niedrige Frequenzen zeigen Intensitäten, hohe zeigen Kanten
- Ideales Tiefpassfilter

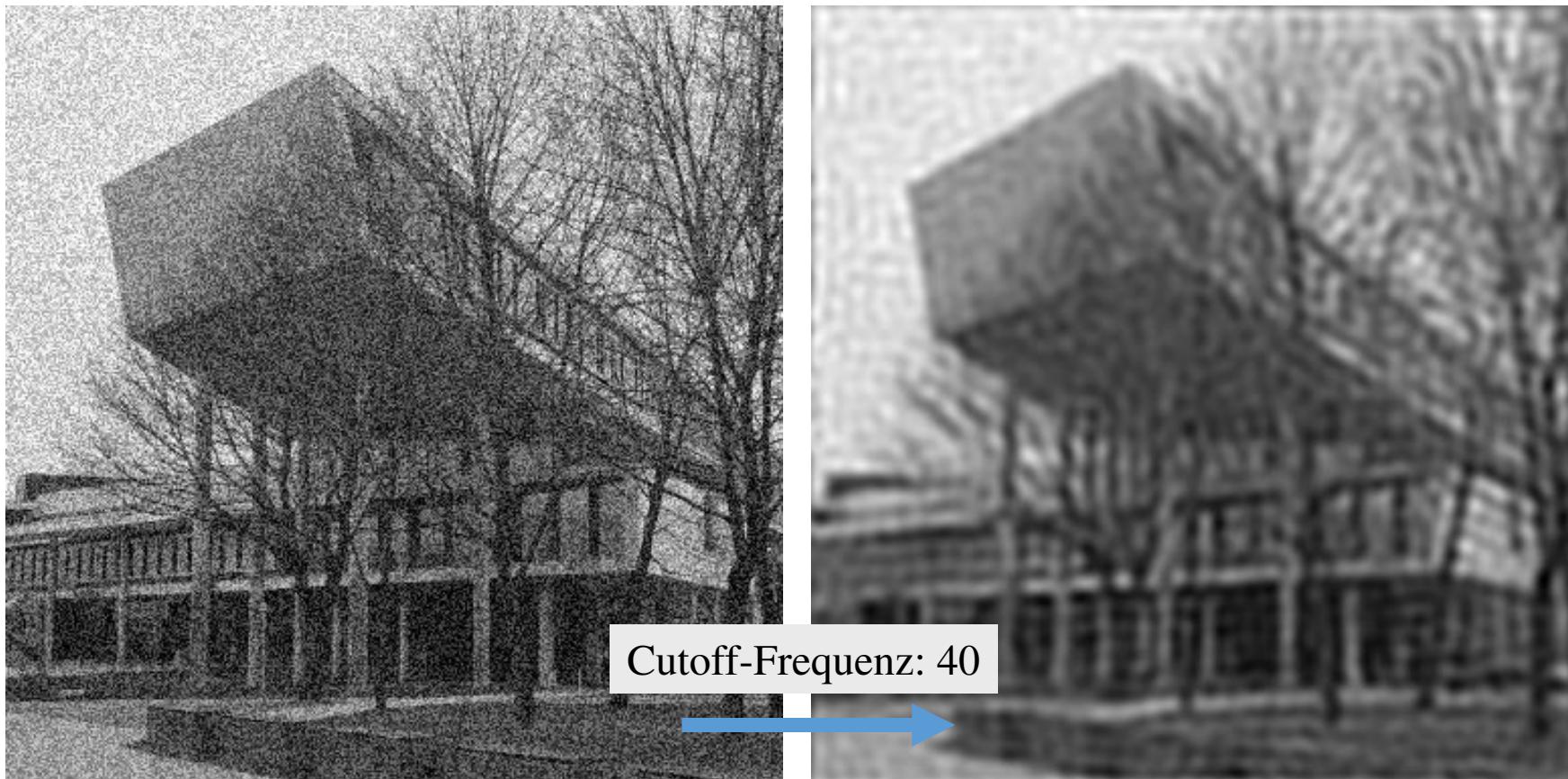
$$H_{F_{\max}}(u, v) = \begin{cases} 1 & , \text{falls } u^2 + v^2 \leq F_{\max}^2 \\ 0 & , \text{sonst.} \end{cases}$$

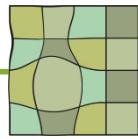
F_{\max} – Cut-Off-Frequenz



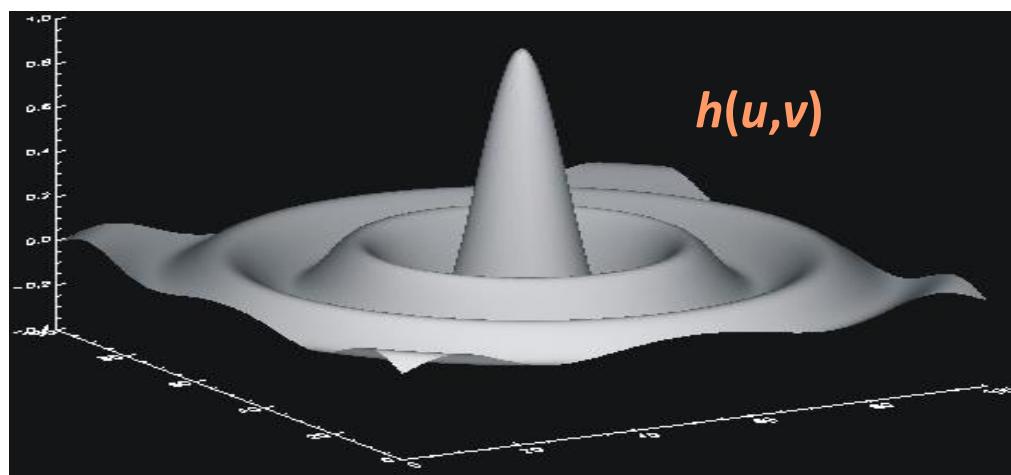
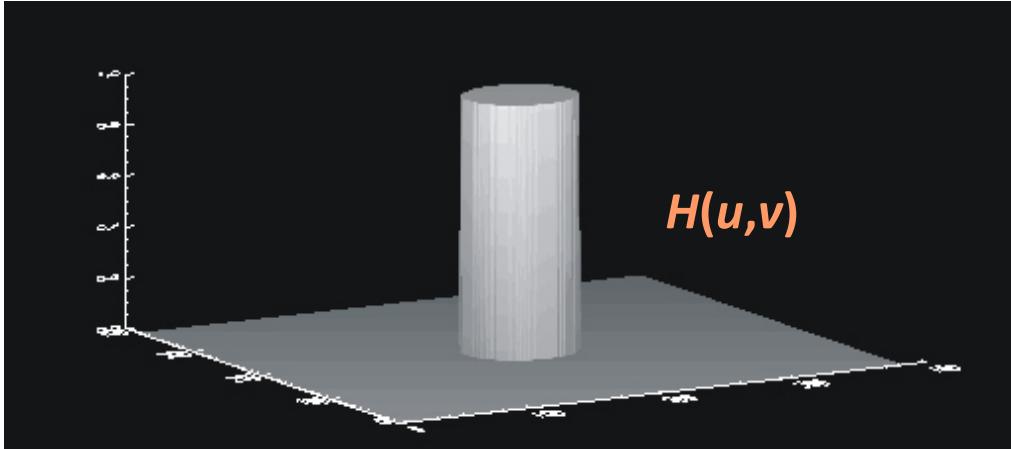
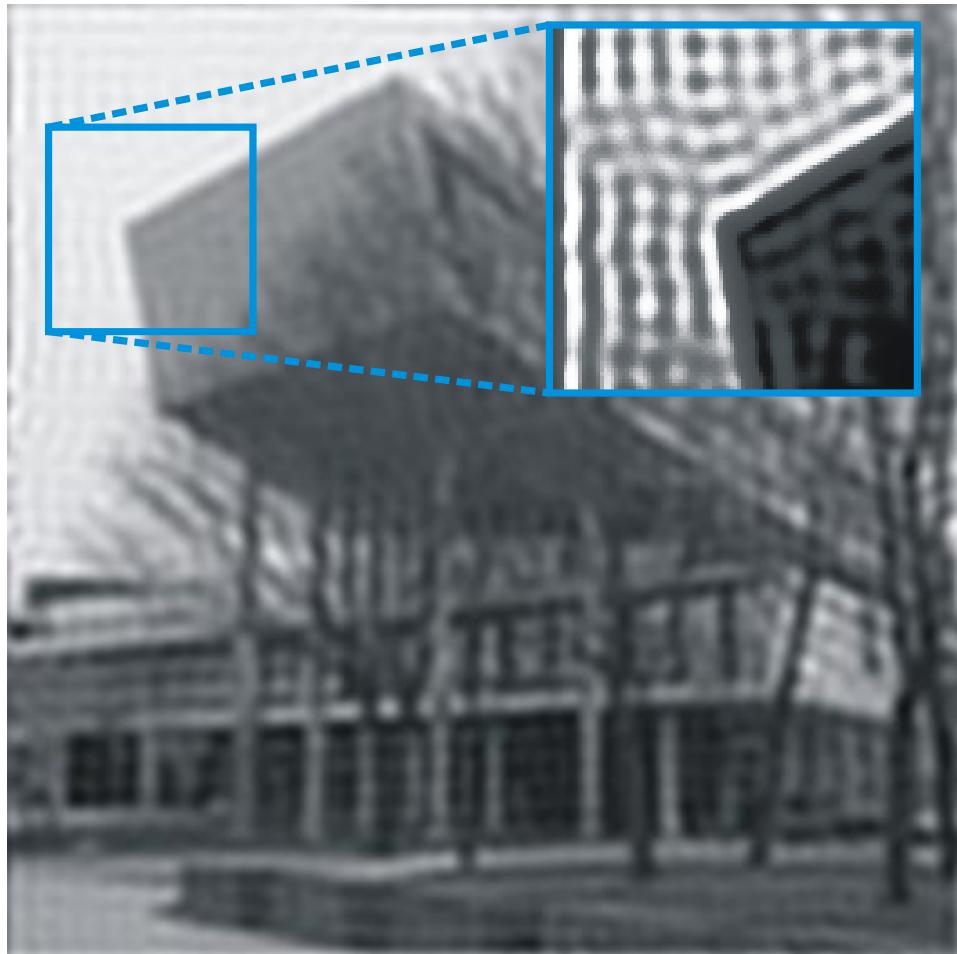


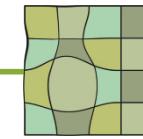
Beispiel Tiefpassfilter



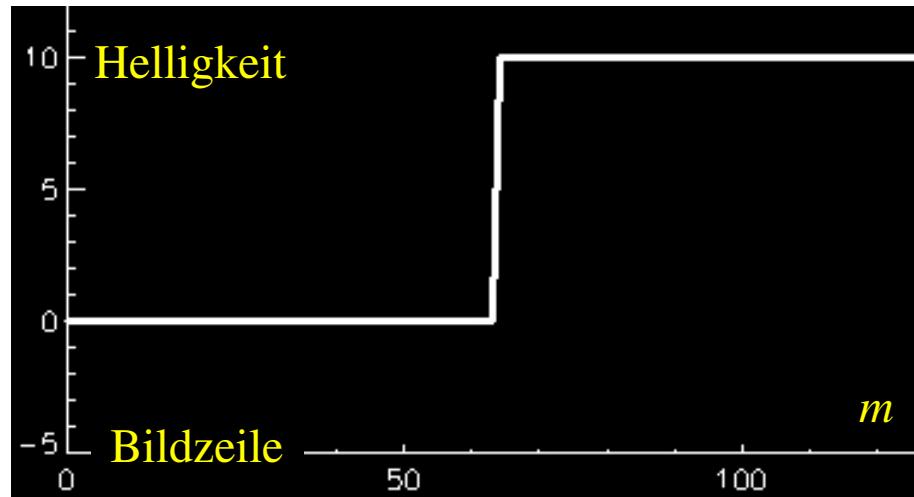


Ringing-Artefakt

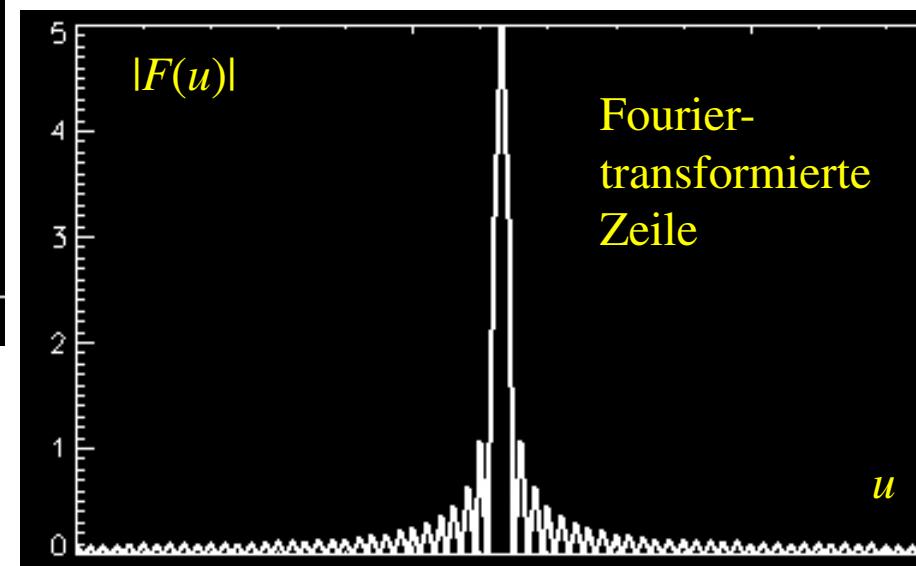




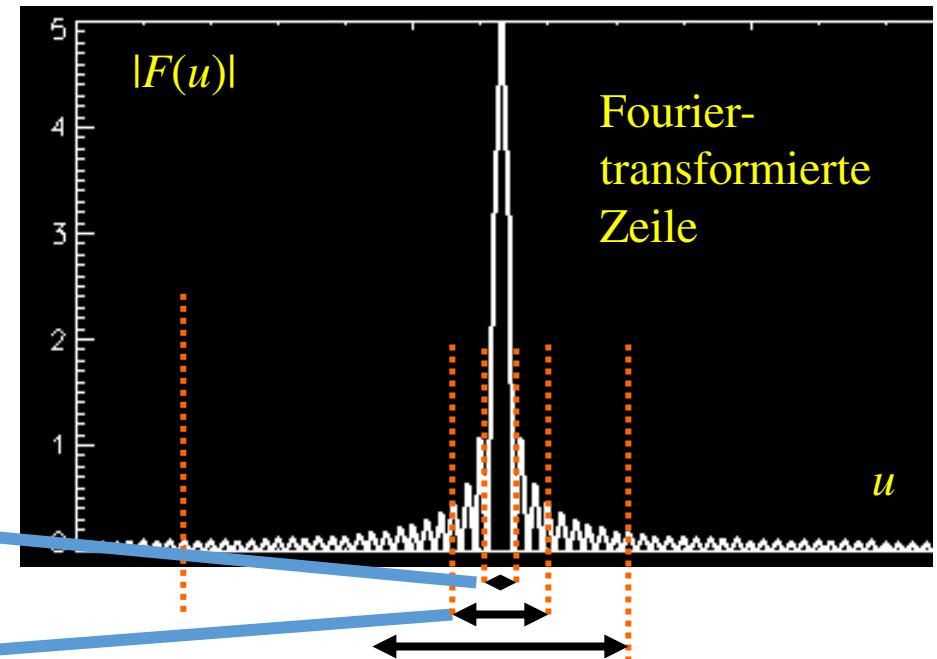
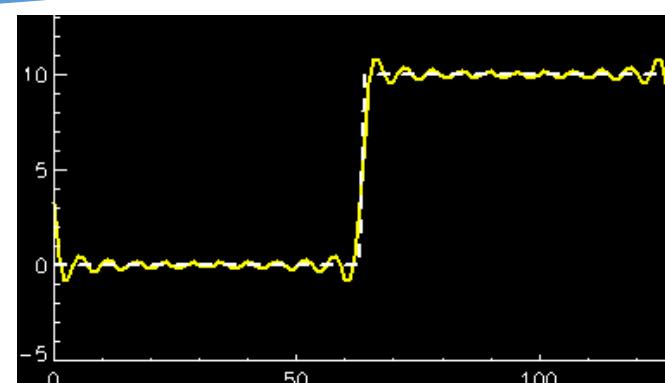
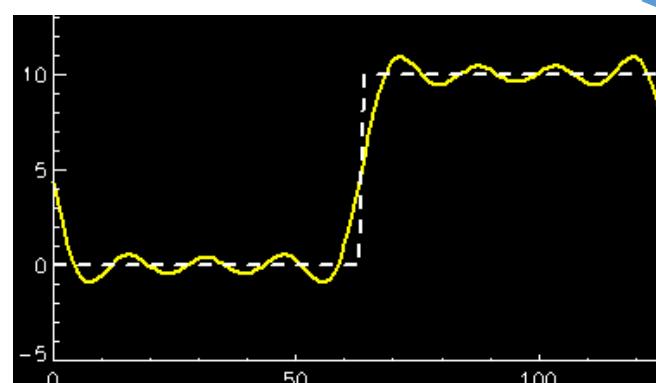
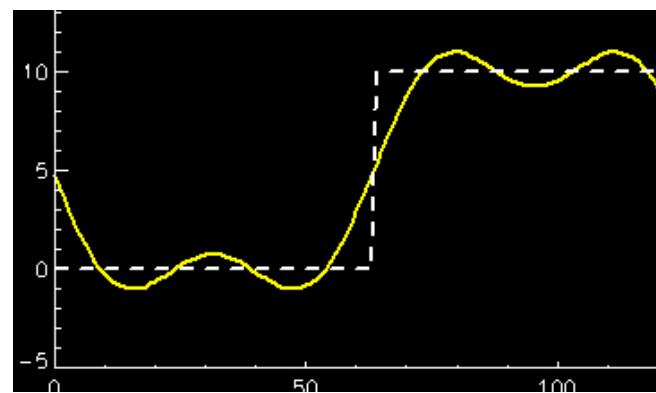
Das Ringing-Artefakt verstehen



Das Ringing-Artefakt entsteht, weil scharfe Kanten durch Wellen **aller** Frequenzen beschrieben werden.

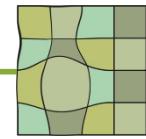


Filterung



Fourier-
transformierte
Zeile

u



Butterworth-Filter

Frequenzen werden nicht gelöscht, sondern nur abgeschwächt.

- Tiefpass-Filter:

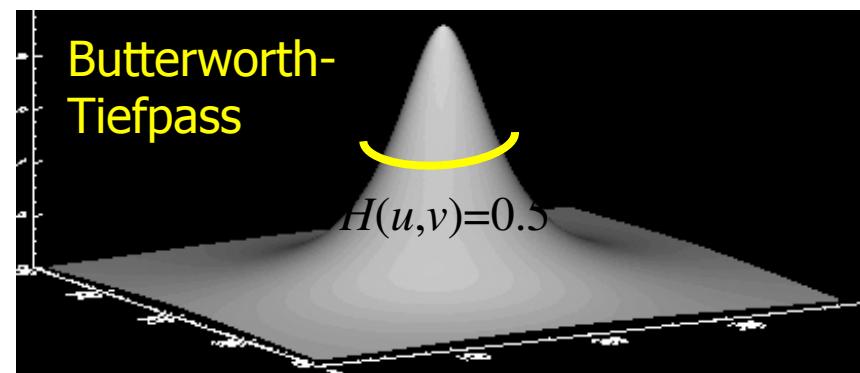
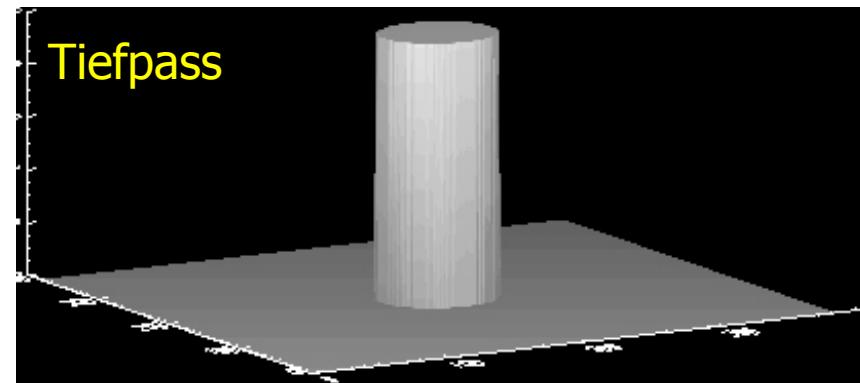
$$H(u, v) = \frac{1}{1 + (D(u, v)/D_0)^{2n}}$$

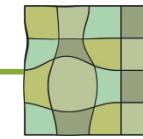
- Hochpass-Filter:

$$H(u, v) = \frac{1}{1 + (D_0 / D(u, v))^{2n}}$$

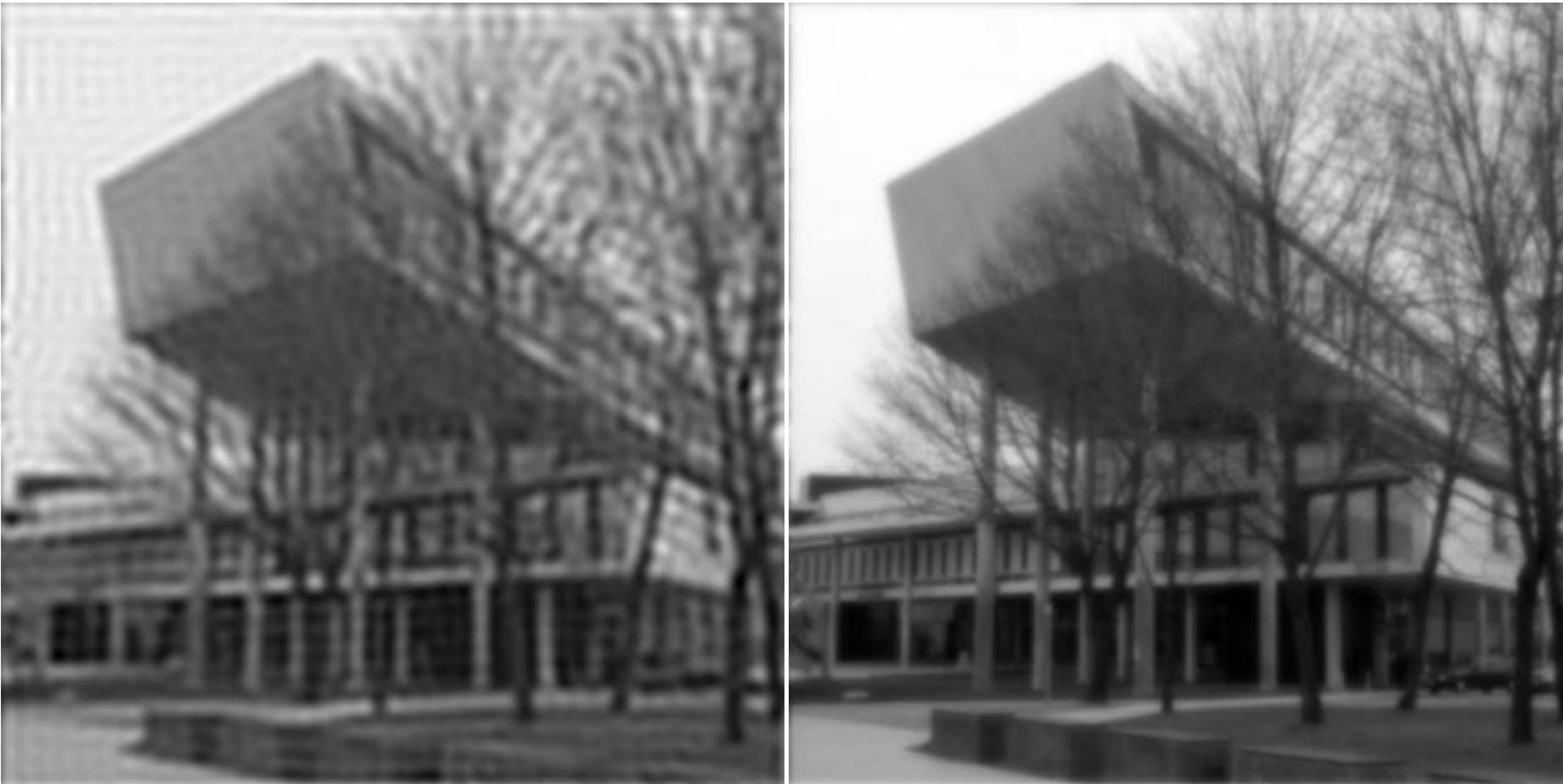
D_0 : Cutoff-Frequenz,

$D(u, v)$: Frequenz, d.h. Abstand
vom Ursprung





Butterworth vs. Einfacher Tiefpass

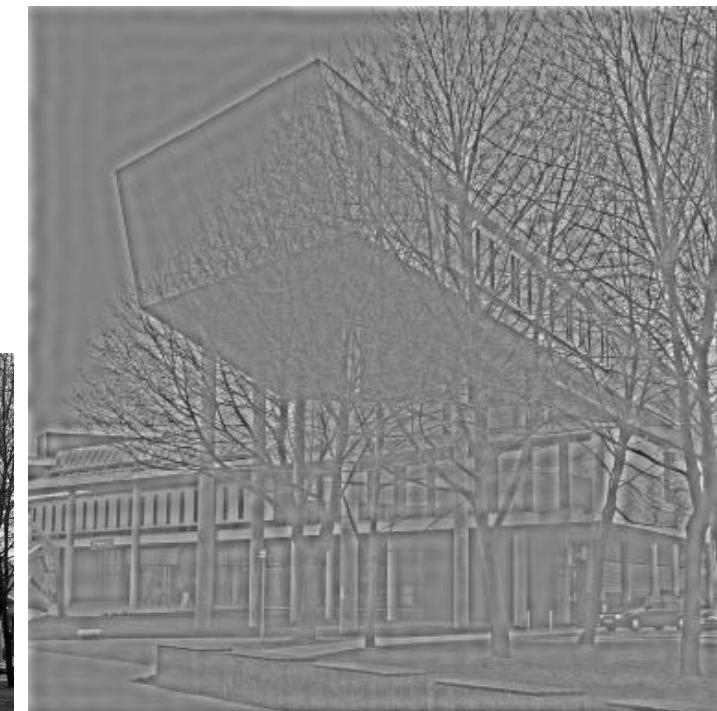


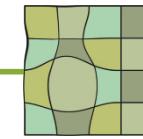
Kantenhervorhebung durch Frequenzraumfilterung

- Kanten weisen mehr hochfrequente Anteile auf wie homogene Gebiete

► Hochpassfilterung

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

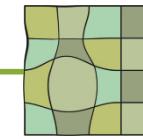




Butterworth-Hochpassfilter



$$H_{F_{\max}}(u, v) = \begin{cases} \frac{1}{1 + \left(\frac{F_{\max}^2}{\sqrt{u^2 + v^2}} \right)^k} & u \neq 0 \vee v \neq 0 \\ 0 & \text{sonst} \end{cases}$$



Filtern im Ortsraum

- Wenn die Fouriertransformation invertierbar ist, dann sollte die Filterung auch im Ortsraum durchführbar sein.

- Wie sieht $\mathbf{FT}^{-1}(F \cdot H)$ aus?

$$\begin{aligned}\mathbf{FT}^{-1}(F(u)H(u)) &= \mathbf{FT}^{-1} \left[\sum_k f(k) \exp\left(-i \cdot 2\pi \cdot \frac{uk}{N}\right) \sum_m h(m) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right] \\ &= \mathbf{FT}^{-1} \left[\sum_k \sum_m f(k) \exp\left(-i \cdot 2\pi \cdot \frac{uk}{N}\right) h(m) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right] \\ &= \mathbf{FT}^{-1} \left[\sum_k \sum_m f(k) h(m) \exp\left(-i \cdot 2\pi \cdot \frac{uk}{N}\right) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right]\end{aligned}$$

(Verschiebeeigenschaft $h(m) \exp(-i \frac{2\pi}{N} uk) = h(m - k)$)



$$\begin{aligned}&= \mathbf{FT}^{-1} \left[\sum_m \sum_k f(k) h(m - k) \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right] \\ &= \mathbf{FT}^{-1} \left[\sum_m \left[\sum_k f(k) h(m - k) \right] \exp\left(-i \cdot 2\pi \cdot \frac{um}{N}\right) \right] \\ &= \mathbf{FT}^{-1} [\mathbf{FT} (\sum_k f(k) h(m - k))] = \sum_k f(k) h(m - k)\end{aligned}$$

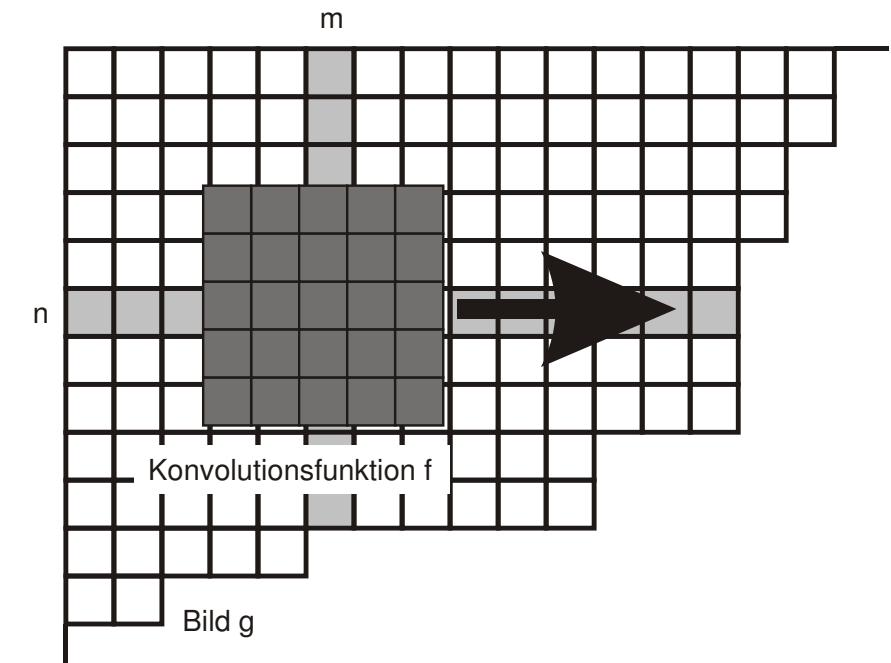
Konvolution (Faltung)

Konvolution (auch: **Faltung**) erzeugt ein neues Bild h durch eine gewichtete Summe von Bildelementen in g :

$$h(m, n) = (g * f)(m, n) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} g(i, j) \cdot f(m-i, n-j)$$

Die Gewichtungsfunktion f heißt **Konvolutionsfunktion** (oder **Faltungsfunktion**)

Die Konvolution entspricht der Filterung durch Multiplikation im Frequenzraum

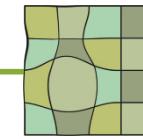


Eigenschaften der Konvolution

- Operatorzeichen: „*“ (bedeutet nicht Multiplikation!)
- Linear
- Verschiebungsinvariant
- Kommutativ: $[g_1 * g_2](m,n) = [g_2 * g_1](m,n)$.
- Assoziativ: $g_1 * ([g_2 * g_3](m,n)) = [g_1 * g_2](m,n) * g_3(m,n)$

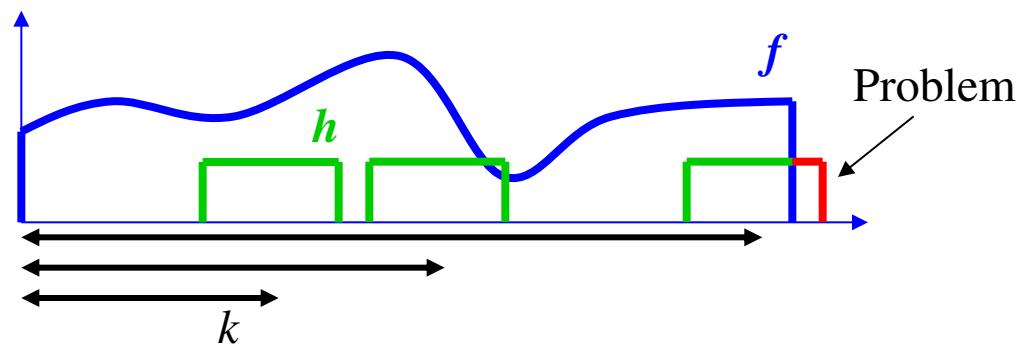
Vorteil der Assoziativität

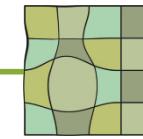
Mehrere Störoperatoren können zu einem gemeinsamen Operator zusammengefasst werden.



Konvolutionskern (Faltungskern)

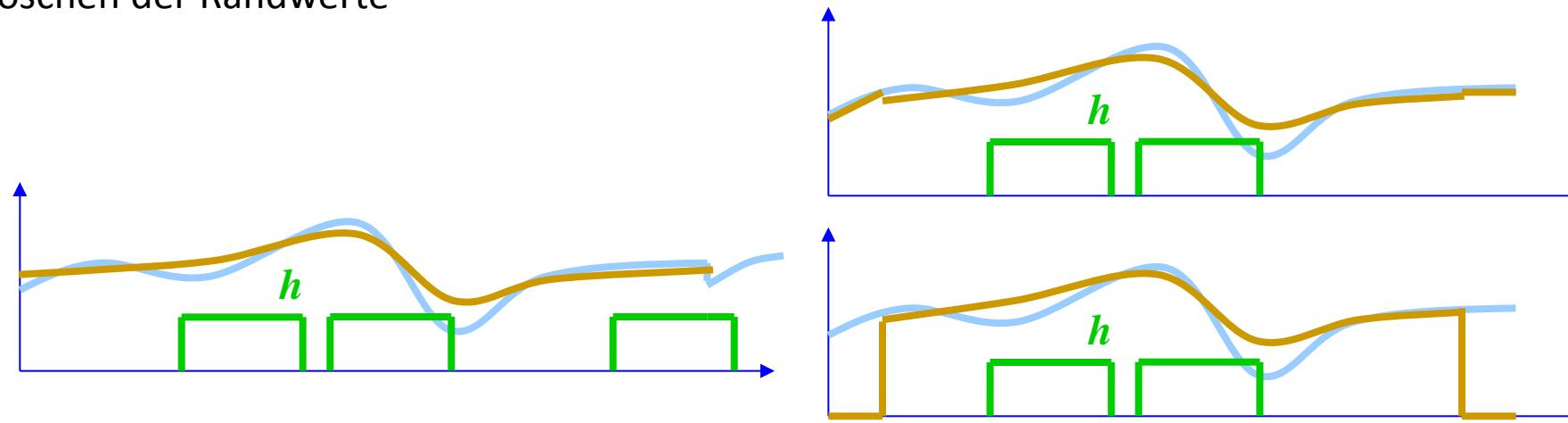
- Konvolution (1D): $[f^*h](m) = \sum_{k=-\infty, \infty} f(k) \cdot h(m-k)$
- Die meisten Konvolutionsfunktionen h sind nur in einem kleinen Intervall von Null verschieden (Konvolutionskern).
- Mit $h(n) \neq 0$ für $-K \leq n \leq K$ ist: $[f^*h](m) = \sum_{k=-K, K} f(k) \cdot h(m-k)$.
- Problem: **Rand** von f .

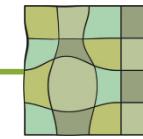




Konvolution am Bildrand

- Lösung 1: periodische Fortsetzung des Bildes
(diese Lösung wird automatisch gewählt, wenn im Frequenzraum multipliziert wird)
- Lösung 2: Rand ist undefiniert
 - Beibehaltung des ursprünglichen Resultats
 - Löschen der Randwerte





$$* \quad \square$$

Konvolution am Bildrand



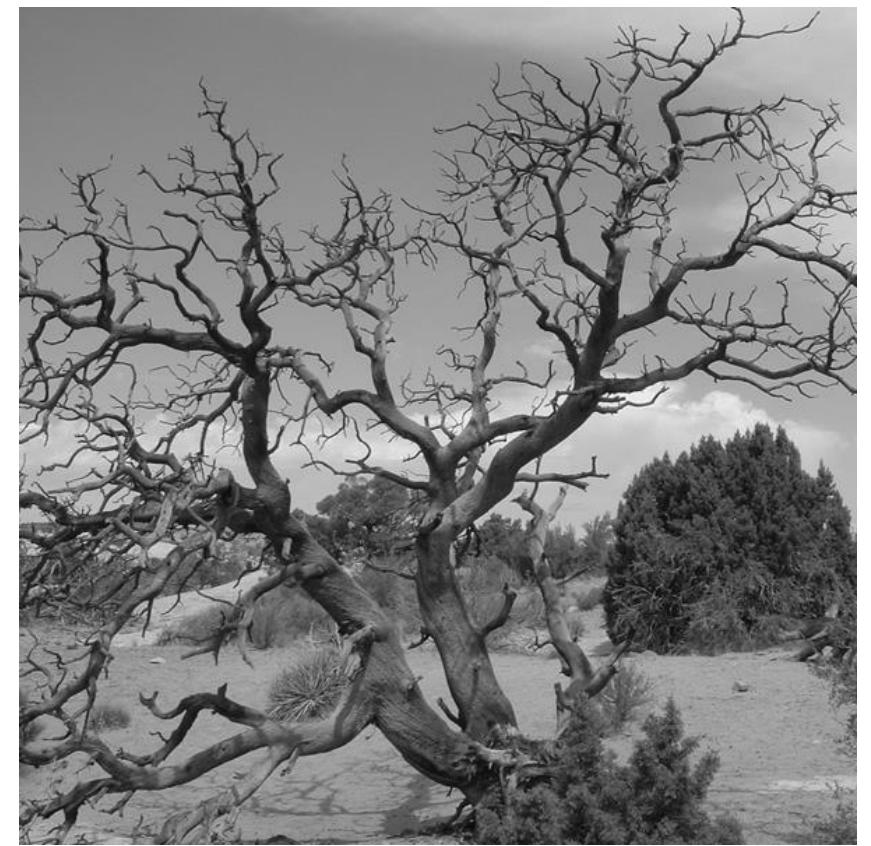
Was sollten Sie heute gelernt haben?

- 2D-Fouriertransformation
- Bedeutung von Frequenz, Amplitude, Phase und Wellenrichtung
- Darstellung der Fouriertransformierten
- Eigenschaften der Fouriertransformierten
- Filterung und Konvolution

Famous Last Question

Was geschieht, wenn man die Frequenzraumrepräsentation vor der Rücktransformation komplex-konjugiert?

Warum?



Famous Last Question

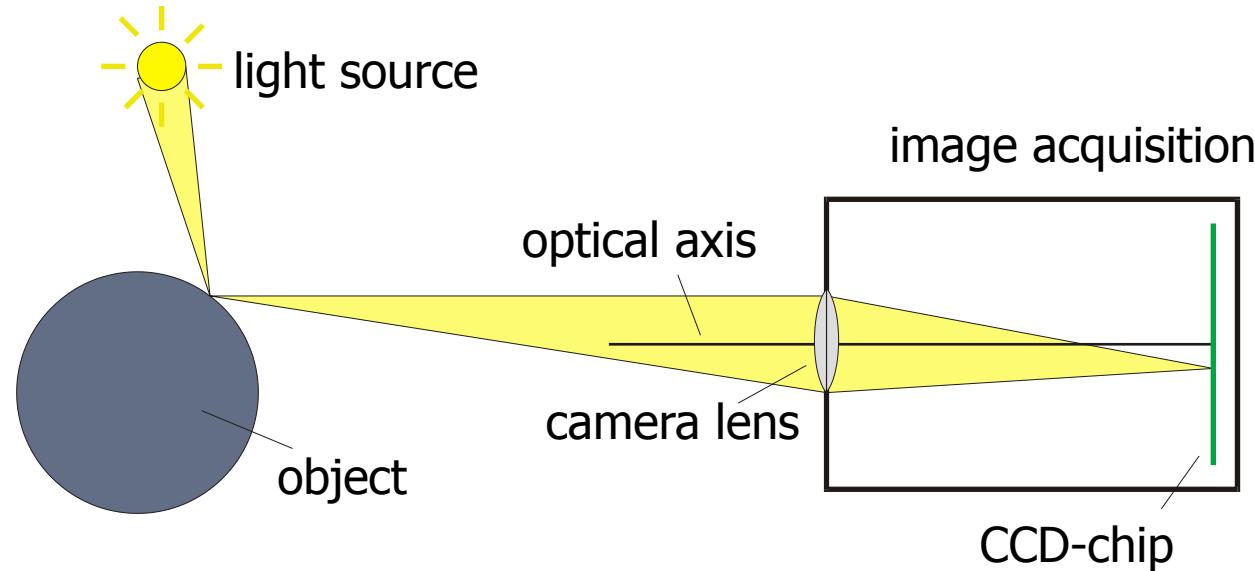
- Was geschieht, wenn man die Frequenzraumrepräsentation vor der Rücktransformation komplex-konjugiert?
- Warum?



Abtastung bei der Bilddaufnahme

- **Informationsverlust:** verloren gegangene Information beziffern und ggf. rekonstruieren (erfordert Modell).
- **Deterministische Informationsveränderung:** Veränderung beschreiben und – falls möglich – invertieren ([Restauration, VL 6](#)).
- **Stochastische Informationsveränderung:** Veränderung beschreiben und näherungsweise rückgängig machen ([Bildverbesserung, VL 8-10](#)).

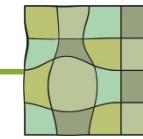
Bildaufnahme



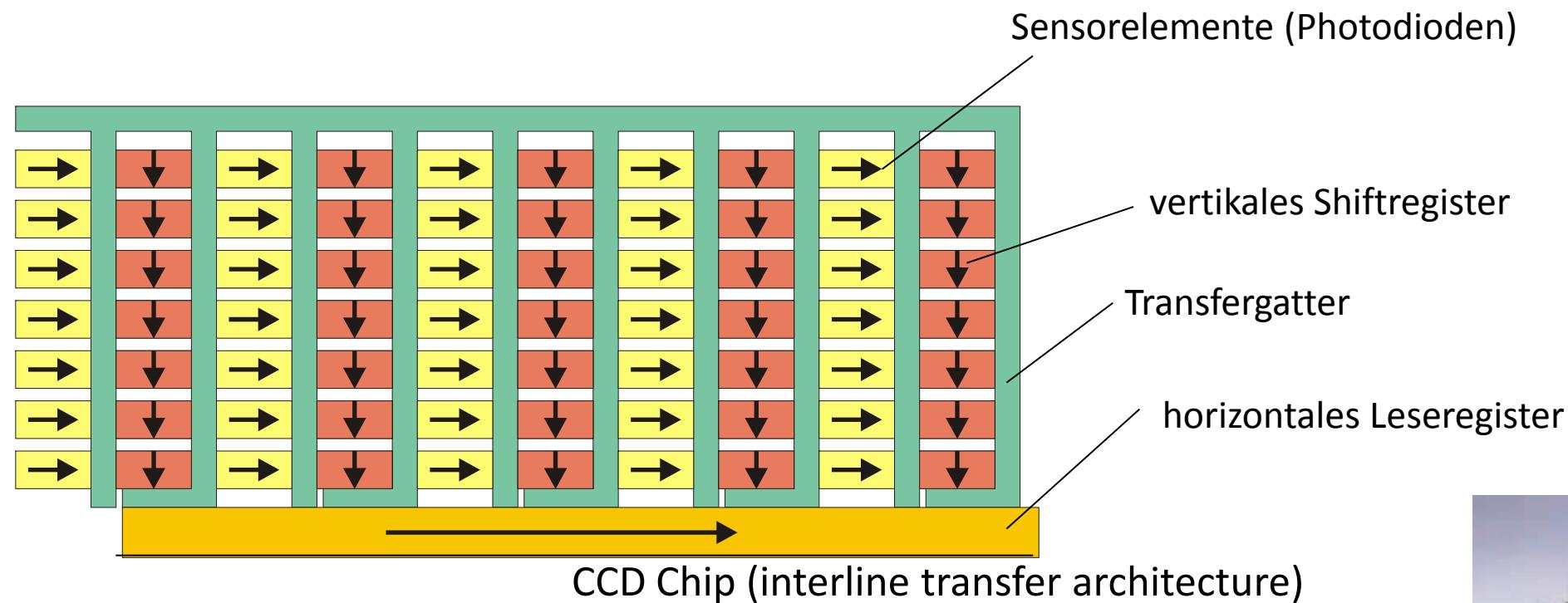
Industrielle CCD Kameras

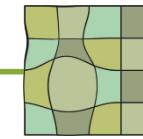


Die **Optische Achse** verläuft durch das Linsenzentrum und den Mittelpunkt des Bildes.



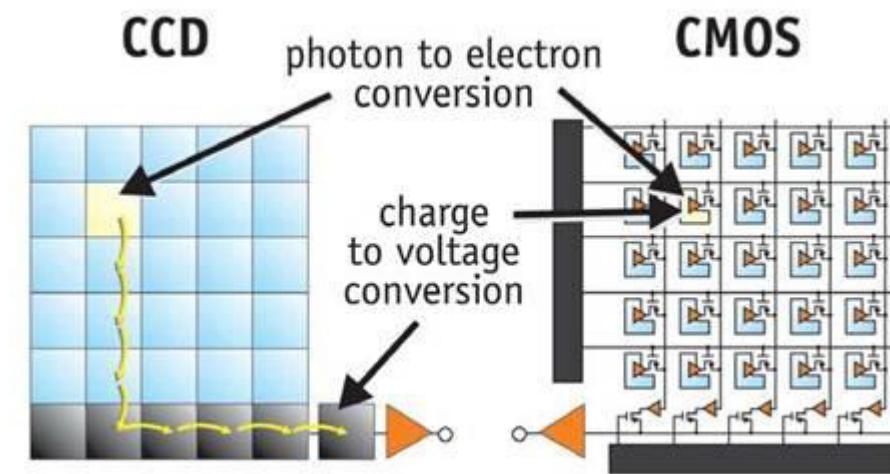
CCD-Kamera





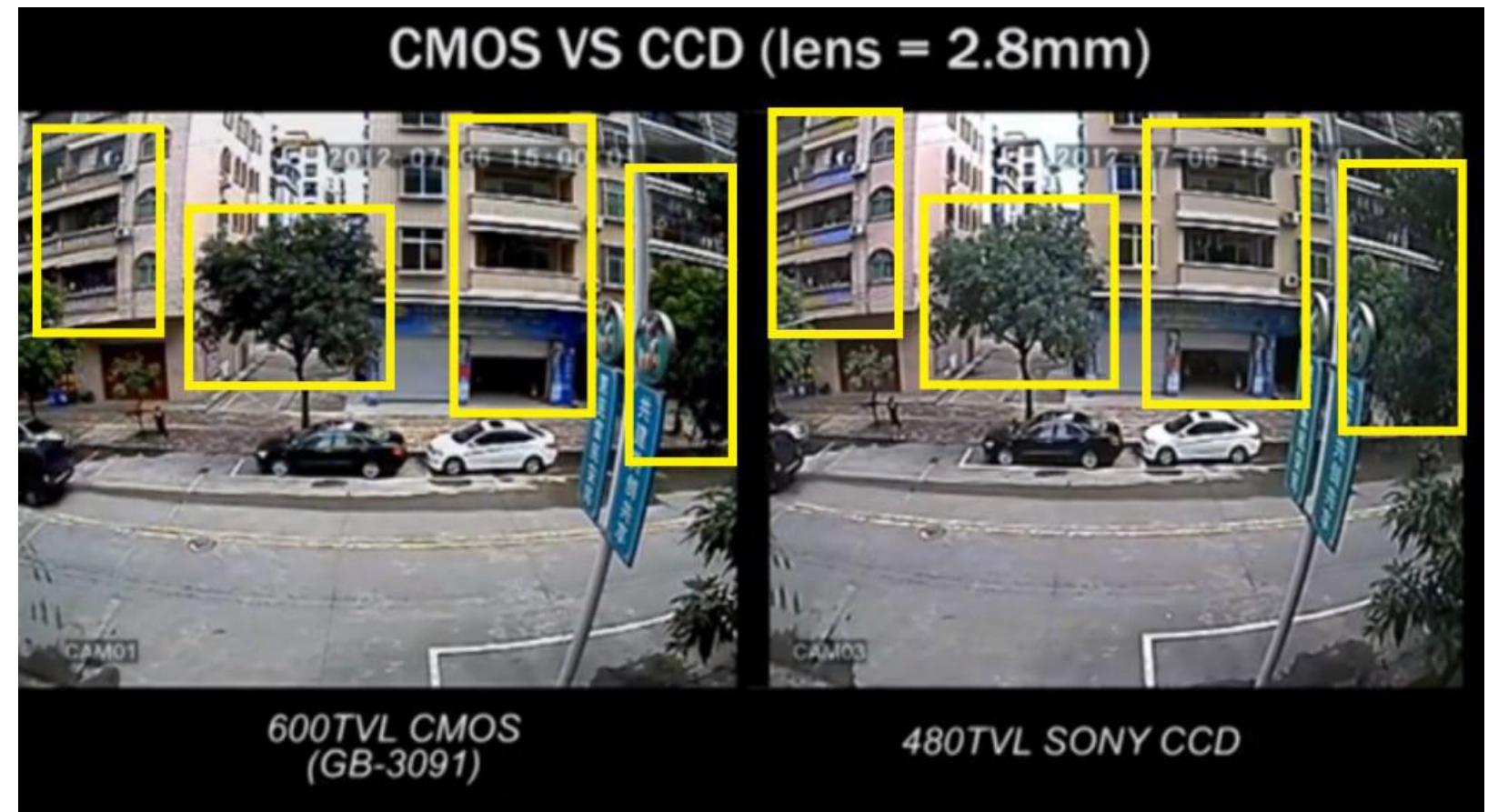
CMOS Sensoren

- Nachteil CCD: Verstärkung des Signals erst nach dem sequentiellen Auslesen
- CMOS-Sensor (Active Sensor): jedes Bildelement wird einzeln verstärkt
- Vorteile
 - Pixel können gleichzeitig ausgelesen werden (hohe Bildraten)
 - Geringer Energieverbrauch
 - preiswerter
- Nachteil
 - Kleinere photosensorische Fläche (geringere Lichtempfindlichkeit)
 - Kleinere Pixel = Höheres Bildrauschen
 - Höherer Dunkelstrom

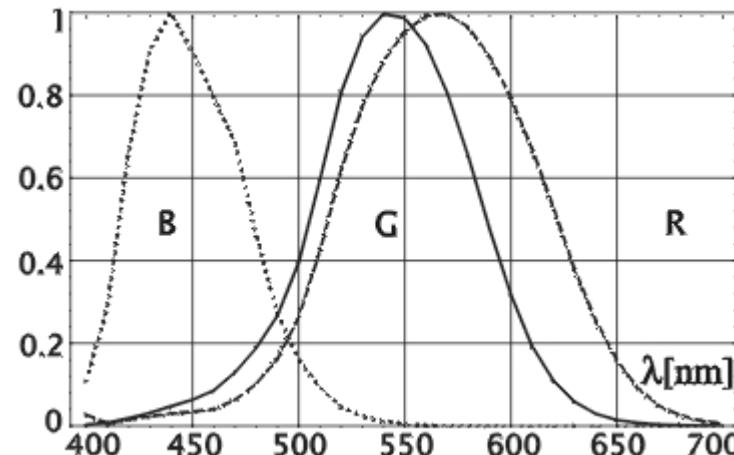


CMOS vs. CCD

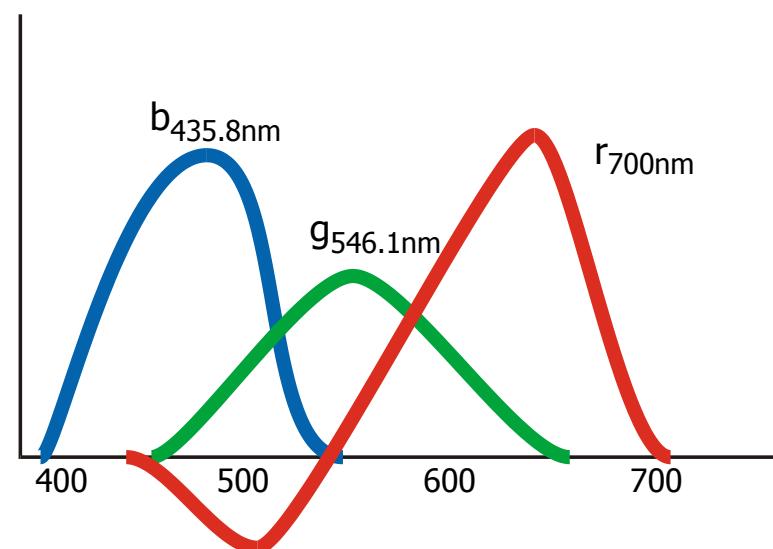
- CCD hat besseres Dynamikverhalten (Abbildung bei großer Helligkeitsvariation)



Farbe

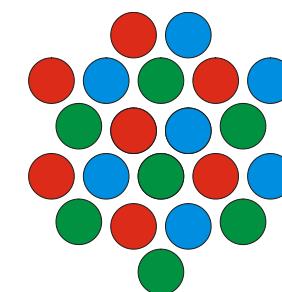


Menschliche visuelle Wahrnehmung erfolgt über drei unterschiedlich für das wahrgenommene Spektrum empfindliche Zelltypen.



Der Wahrnehmung fast aller sichtbaren Wellenlängen kann durch Kombination von drei Wellenlängen angenähert werden.

Farbe kann als additives Rot-Grün-Blau-Signal vermittelt werden.

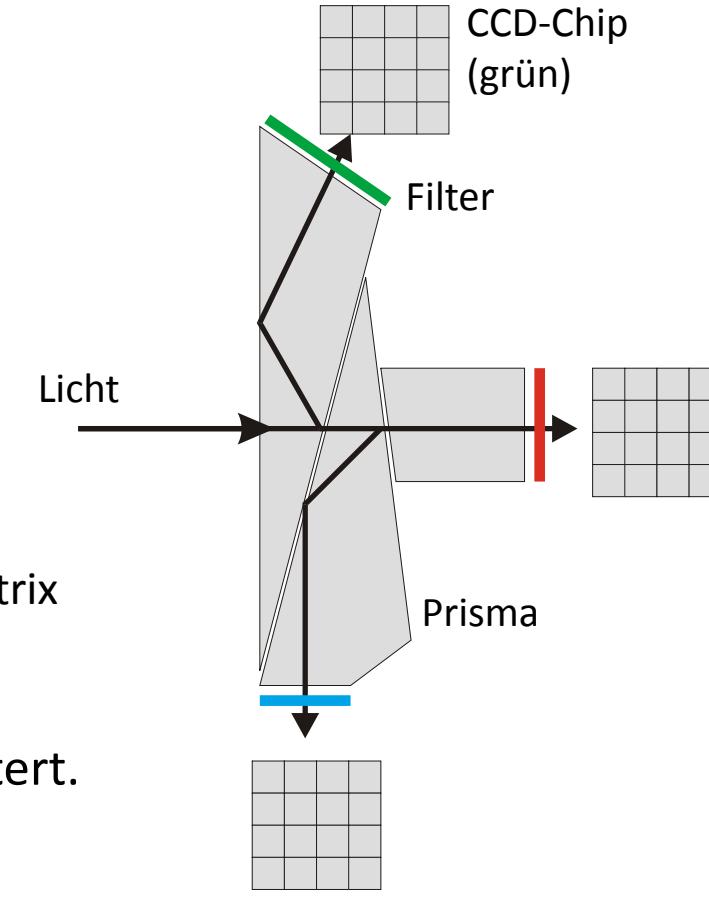
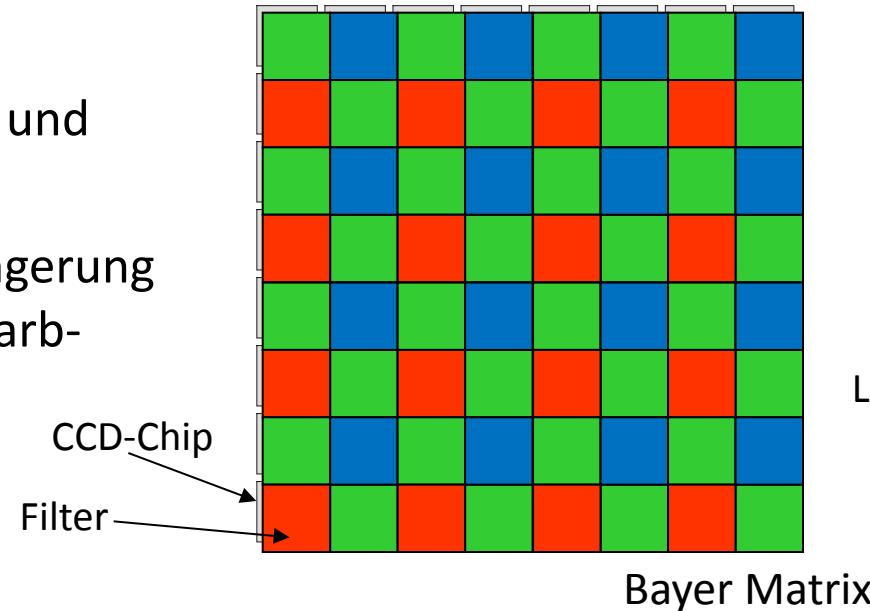


Farbe und CCD-Kameras

1-Chip-Kamera:

Sensorelemente und
RGB-Filter.

Nachteile: Verringerung
der Auflösung, Farb-
aliasing

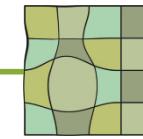


3-Chip-Kamera:

Licht wird durch ein Prisma getrennt und separat gefiltert.

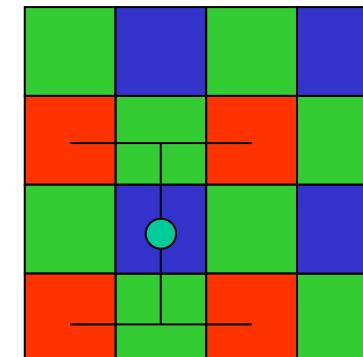
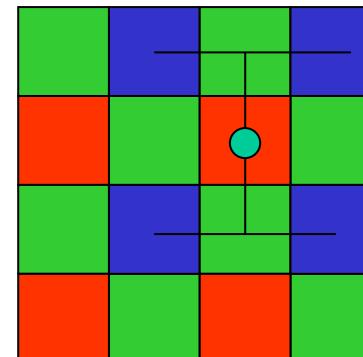
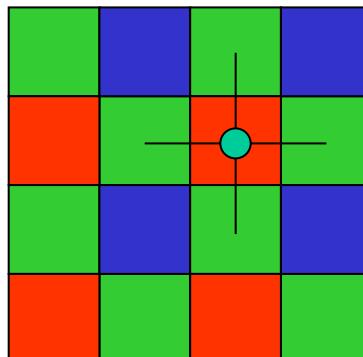
Nachteil:

Aufwändige Konstruktion.



Farb-Interpolation bei 1-Chip Kameras

- Annahme: Farbwerte ändern sich nur langsam
- Interpolation:
 - Fehlende Information (50% bei grün, 75% bei rot und blau) wird interpoliert
 - Einfachste Interpolation: linear aus den Nachbarn
 - Bessere Verfahren: Berücksichtigung von Farbkanten = richtungsabhängige Interpolation



Farbartefakte

- Vor allem an Kanten mit starkem Kontrast
- False Coloring
 - Verursacht durch die örtlich unterschiedlichen Farbkanäle, die interpoliert werden.
- Zippering (zip lock = Reißverschluss)
 - Vor allem an Kanten, die fast parallel zur X- bzw. Y-Achse verlaufen.
- Interpolation unter Berücksichtigung der Kantenorientierung verhindert die Artefakte



Andere Aufnahmesysteme

- Röntgenbilder
- Schichtbildrekonstruktionen
- Reflexionslaufzeitbilder
- Tiefenkarten

Interpretation, Definitions- und Wertebereich dieser Bilder unterscheidet sich von digitalen Fotos

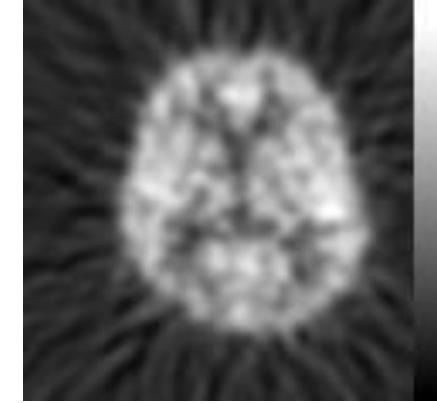
Röntgenbilder

- monochrom.
- Semantik: projizierte Strahlenabsorption.
- Hochauflösend (>4 Megapixel).
- Wertebereich 8-12 Bit.
- Problem: keine Tiefeninformation



Schichtbilder

- monochrom
- Semantik: unterschiedlich
- Geringe Auflösung (0.02-0.25 Megapixel).
- Wertebereich 8-12 Bit
- Probleme: Artefakte und Rauschen.



PET (Metabolismus)



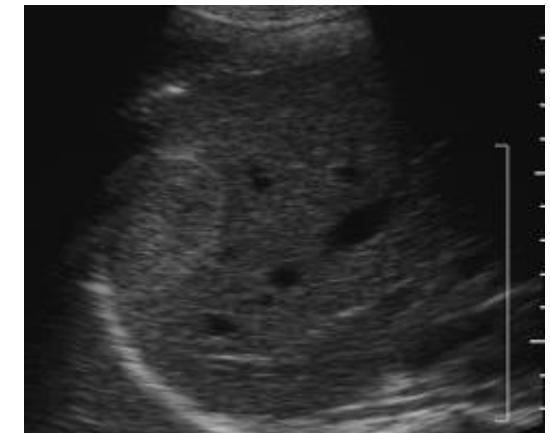
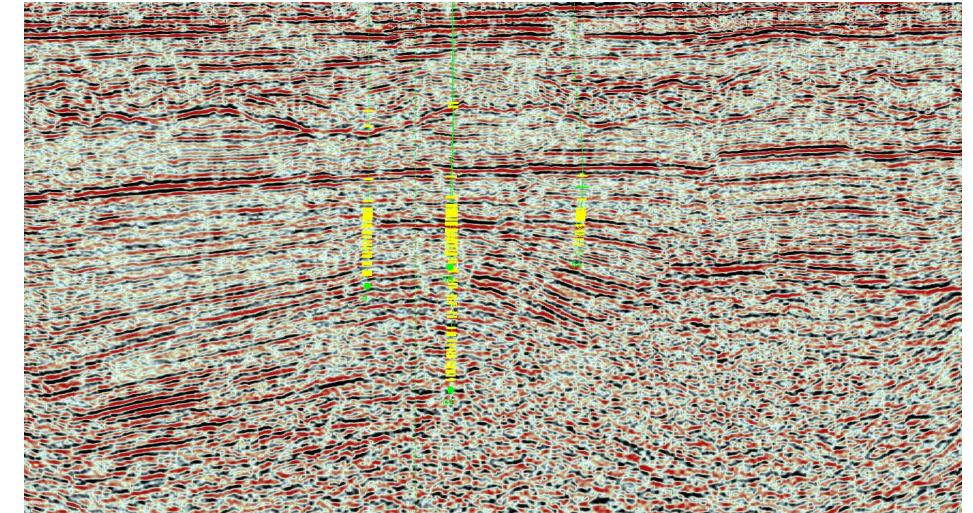
MRT (H_2 und Bindung)



Röntgen-CT (Strahlenabsorption)

Reflexionslaufzeitbilder

- monochrom
- Semantik: Laufzeit und Reflexionsstärke von Schallwellen
- Auflösung unterschiedlich
- Wertebereich 8 Bit
- Problem: Artefakte



Tiefenkarten

- monochrom
- Semantik: Abstand zur Kamera
- Auflösung wie digitale Fotos
- Wertebereich 8-16 Bit
- Problem: häufige „Ausreißer“

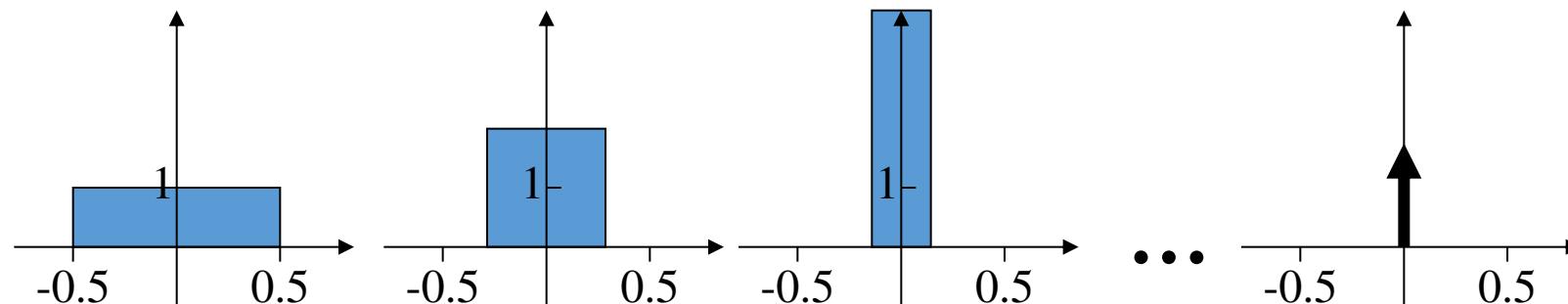


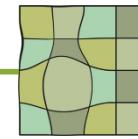
Zusammenfassung

- Semantik eines Bilds wird zum Teil durch das Aufnahmeverfahren bestimmt
- Ein-Kanal- oder Mehr-Kanal-Bilder
- Bilder sind immer eine Abtastung der (bezüglich des Aufnahmesystems sichtbaren) Szene
 - Oberfläche
 - Projektion von transparenten/transluzenten Flächen
 - Schichtbild
- Die Abtastung bildet einen reellen auf einen ganzzahligen Definitionsbereich ab (=Informationsverlust)

Abtastung (Mathematische Beschreibung)

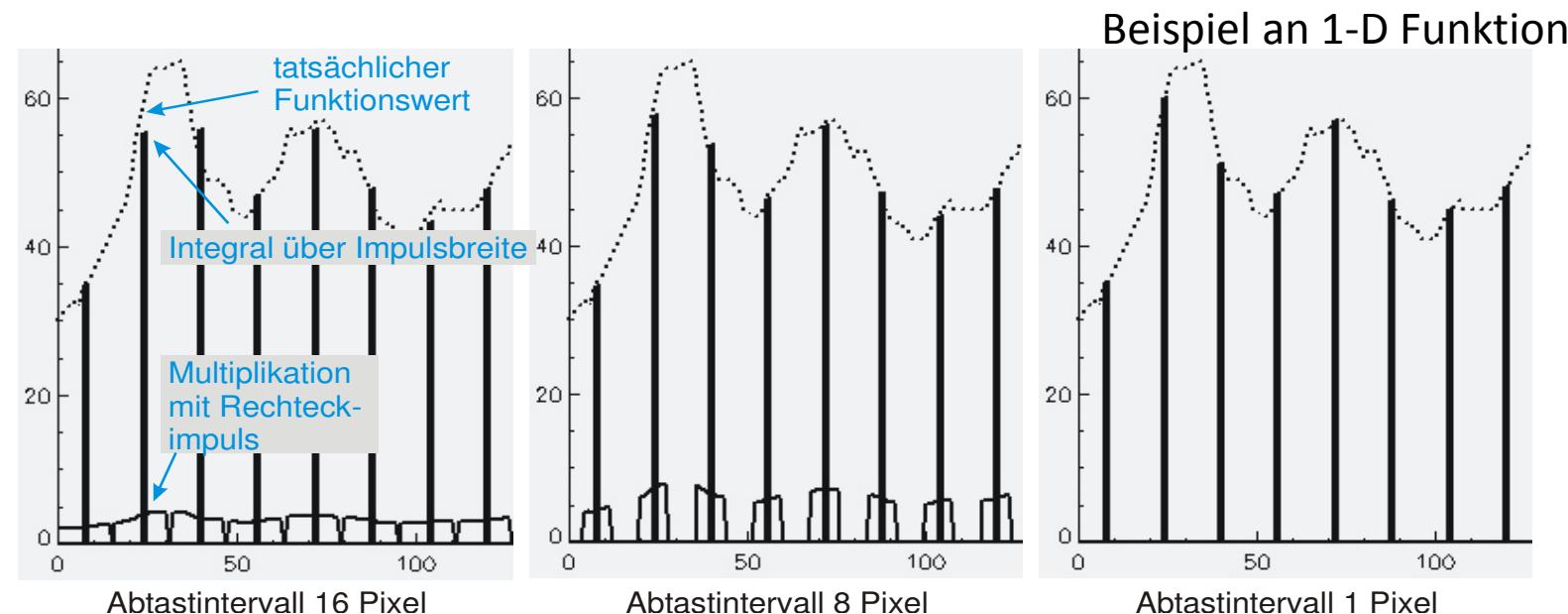
- Multiplikation der Bildfunktion $f(x)$ mit einer Impulsfolge.
- Impulsfolge: Folge von Dirac-Impulsen $\delta(x)$.
- $\delta(x) = 0$ für $x \neq 0$ und $\int \delta(x) = 1$
- Näherungsweise Bestimmung durch Rechteckimpuls:





Abtastung

- Multiplikation mit dem Rechteckimpuls ergibt den Durchschnittswert.
- Je kleiner die Impulsbreite desto mehr nähert sich das Ergebnis dem tatsächlichen Funktionswert am Abtastort an.



Transformation und Abtastung

- Die Transformationen Translation, Rotation und Skalierung sind auf reellen Zahlen definiert:

$$Rot_{\alpha} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \quad Tr_{dx,dy} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + dx \\ y + dy \end{pmatrix}, \quad Sc_s \begin{pmatrix} x \\ y \end{pmatrix} = s \begin{pmatrix} x \\ y \end{pmatrix}.$$

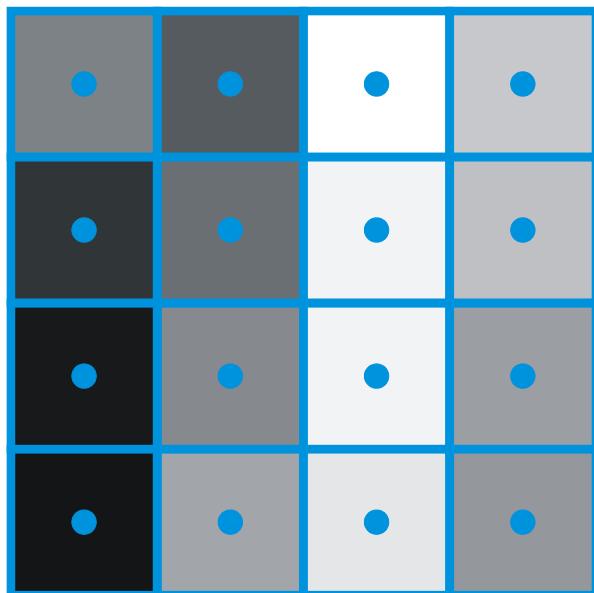
- Digitale Bilder haben einen ganzzahligen Definitionsbereich.
- Nach Transformation eines bereits abgetasteten Bildes ist eine Interpolation notwendig.

Interpolation

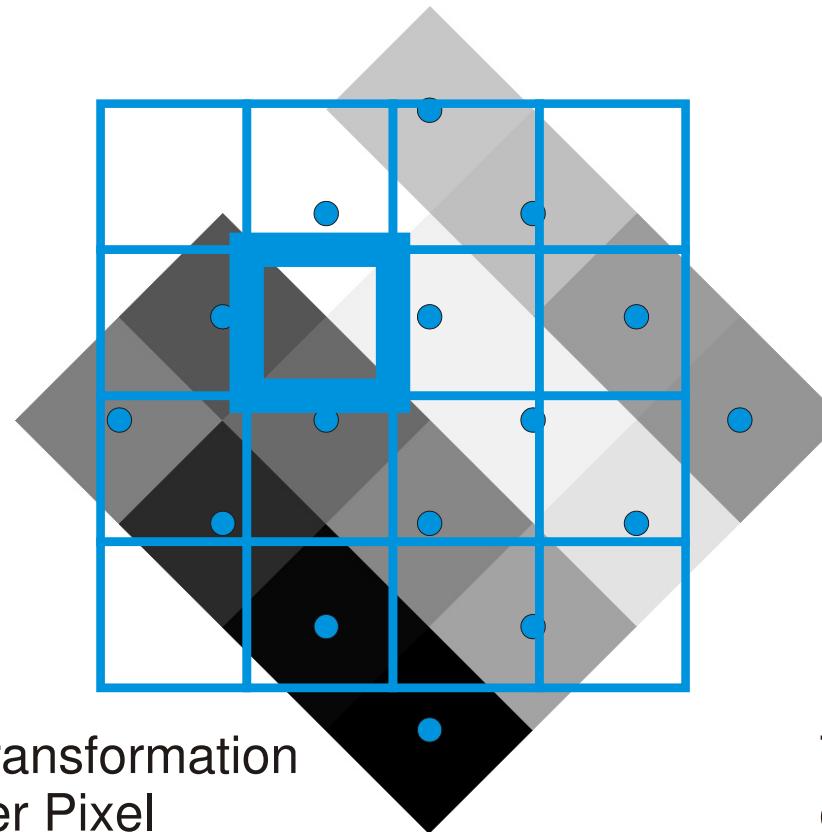
- Konstante Interpolation (Wert des nächsten Nachbarpixels):
- Lineare Interpolation
- Interpolation durch Polynome höheren Grades.
- Interpolation im Frequenzraum.



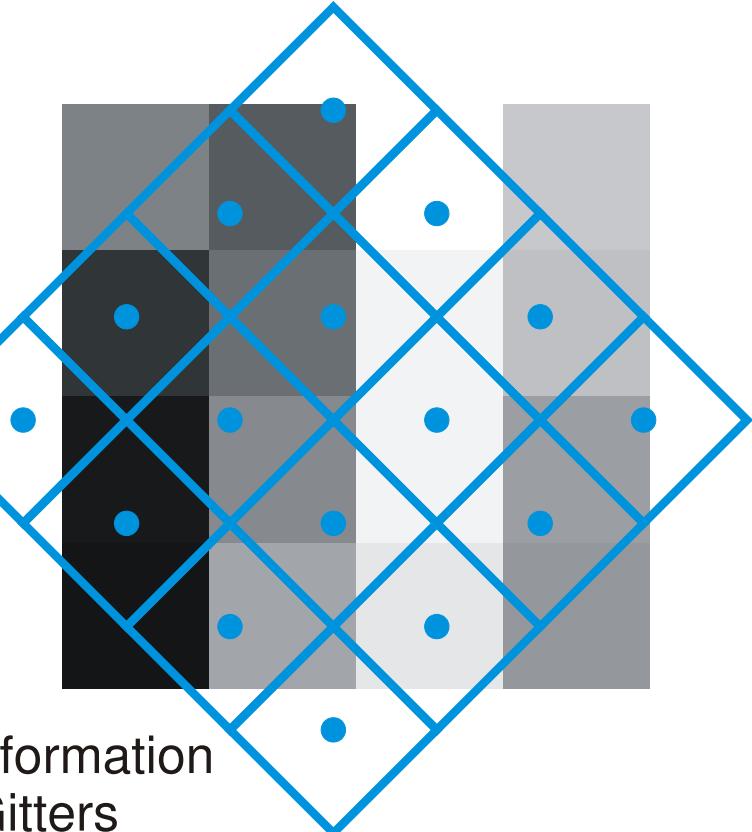
Konstante Interpolation



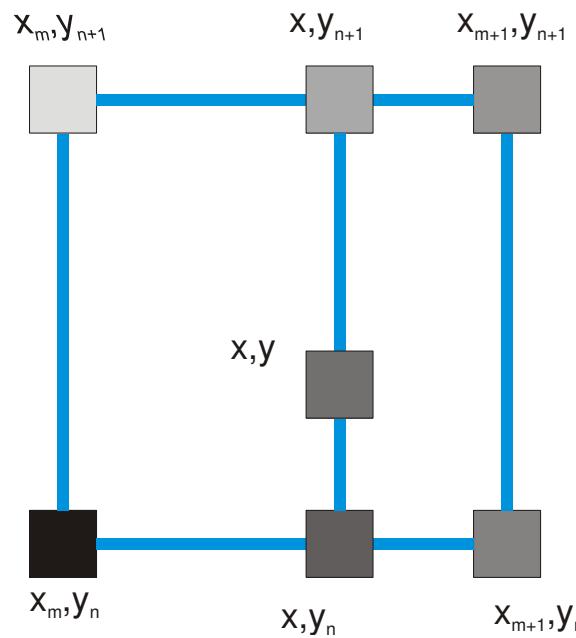
Transformation
der Pixel



Transformation
des Gitters



Bilineare Interpolation



Erster Schritt:

$$g_1(x_m, y) = \frac{y_{n+1} - y}{y_{n+1} - y_n} f(x_m, y_n) + \frac{y - y_n}{y_{n+1} - y_n} f(x_m, y_{n+1}),$$

$$g_1(x_{m+1}, y) = \frac{y_{n+1} - y}{y_{n+1} - y_n} f(x_{m+1}, y_n) + \frac{y - y_n}{y_{n+1} - y_n} f(x_{m+1}, y_{n+1}).$$

Zweiter Schritt:

$$g(x, y) = \frac{x_{m+1} - x}{x_{m+1} - x_m} g_1(x_m, y) + \frac{x - x_m}{x_{m+1} - x_m} g_1(x_{m+1}, y).$$

Polynome höheren Grades

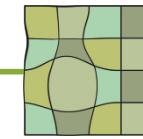
- Die Bildfunktion wird besser angenähert, wenn mehr Terme der Taylor-Approximation berücksichtigt werden.
- Ableitungen für die Taylor-Reihe werden durch Differenzen angenähert.
- Grad des Polynoms ist ein Kompromiss zwischen
 - steigender Anzahl berücksichtigter Terme der Taylor-Reihe.
 - steigender Ungenauigkeit der geschätzten Ableitungen.

Interpolation im Frequenzraum

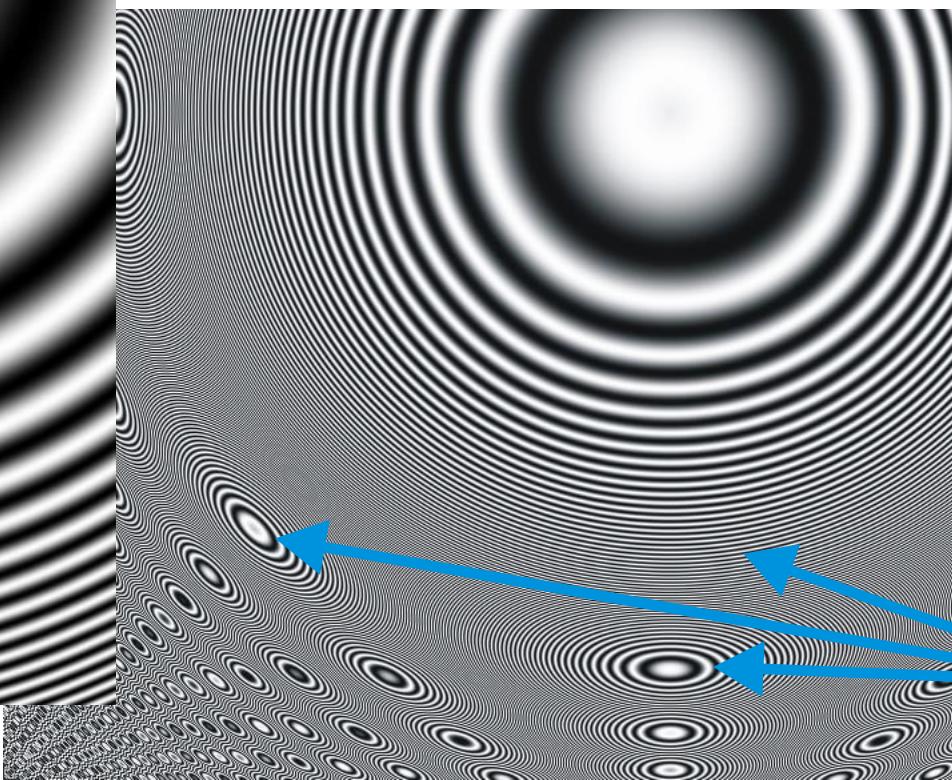
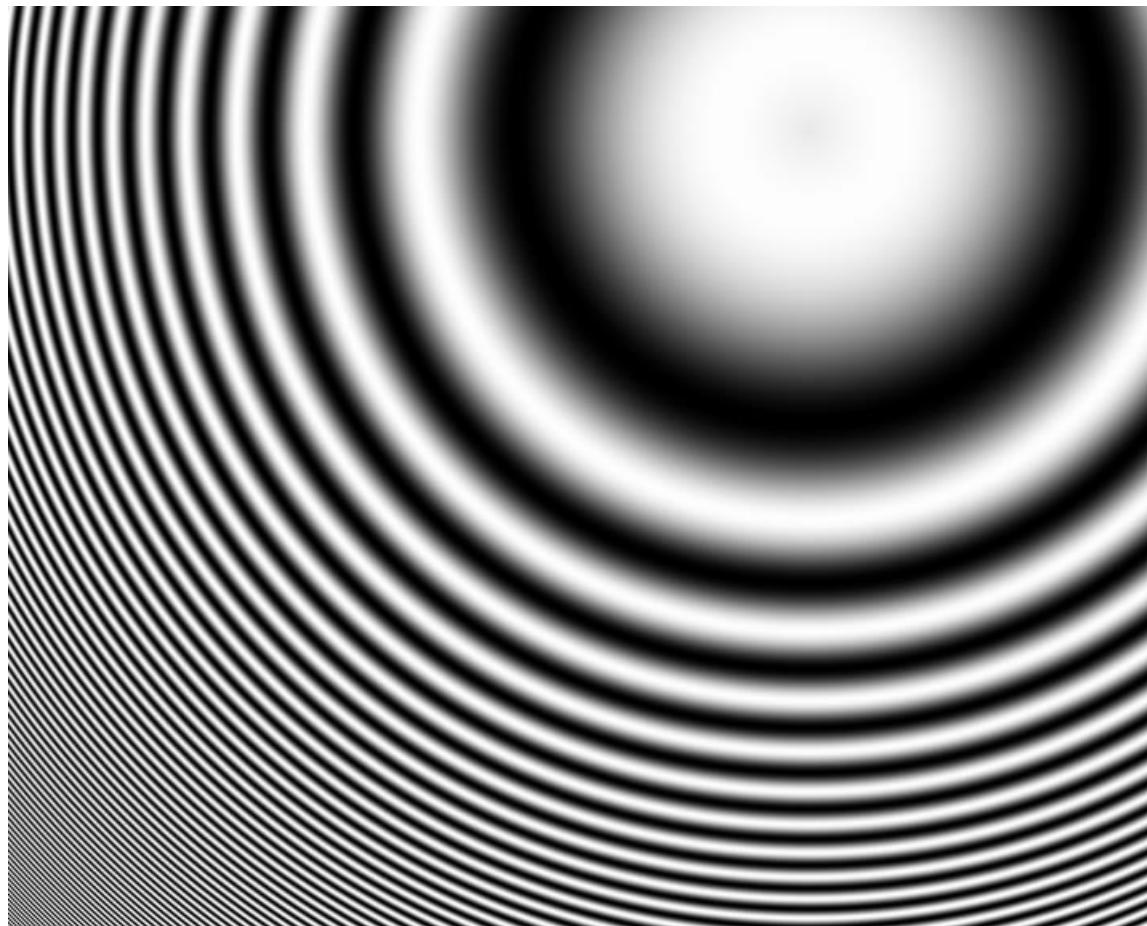
- Die Basisfunktionen der Fouriertransformation haben einen reellen Definitionsbereich.
- Ein Funktionswert kann an beliebiger Stelle (x, y) bestimmt werden durch

$$g_F(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp\left(i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)\right)$$

- Die Interpolation ist exakt, falls die Funktion bandbegrenzt ist (was die meisten Funktionen jedoch nicht sind).



Moiré-Effekt bei Abtastung

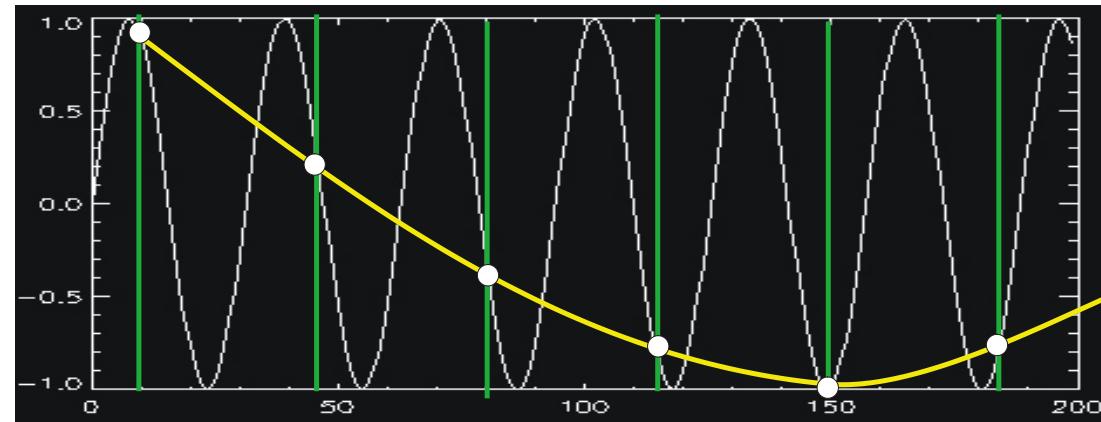


zunehmende
Frequenz



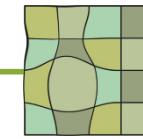
Aliasing-
Artefakte

Aliasing

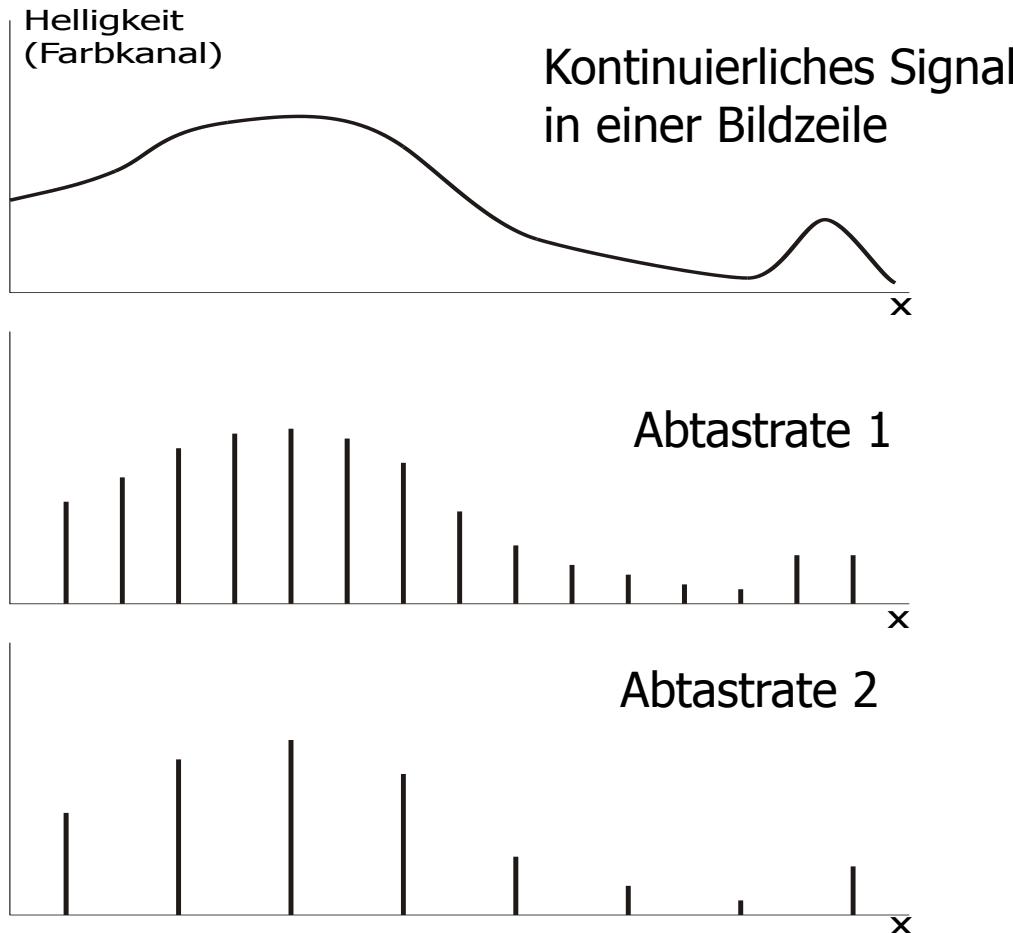


Durch die Abtastung wird eine Frequenz rekonstruiert, die nicht im Original vorhanden ist (Alias-Frequenz).

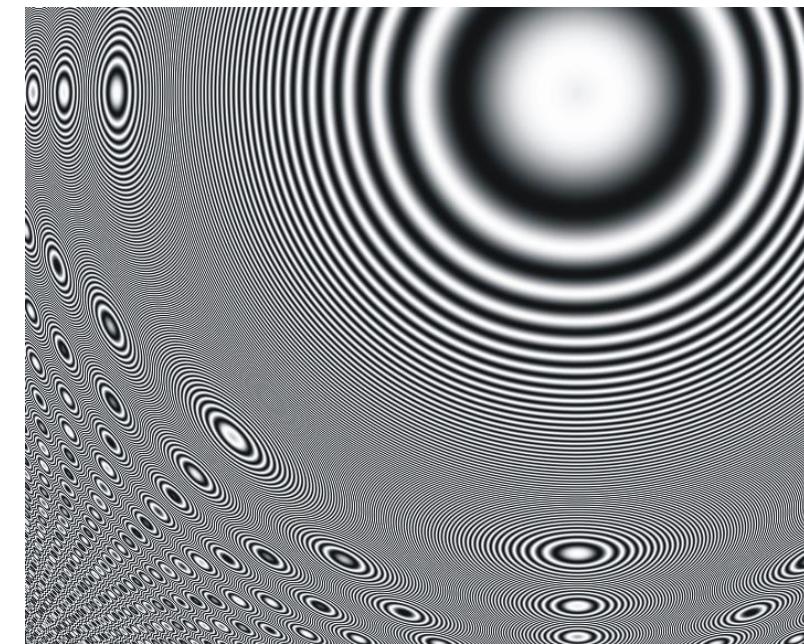
Folgerung: Moiré-Effekt kann mit Hilfe der Repräsentation im Frequenzraum erklärt werden.



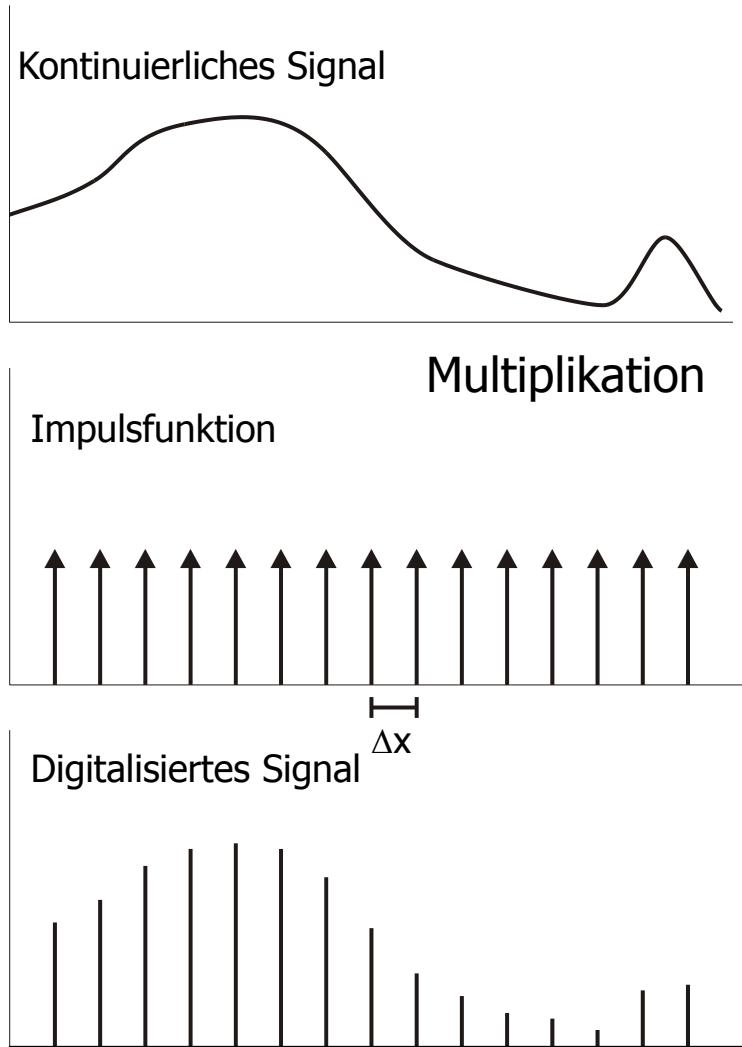
Abtastung im Ortsraum



Abtastung durch CCD-Kamera, als
Punktabtastung.



Was ist geschehen?



Das kontinuierliche Signal $f(x)$ wurde mit der Impulsfunktion $s(x)$ multipliziert.

$s(x) = d(x + n \cdot \Delta x)$ ist die **Abtastfunktion**.

Multiplikation im Ortsraum ist Konvolution im Frequenzraum!

Was geschieht im Frequenzraum?

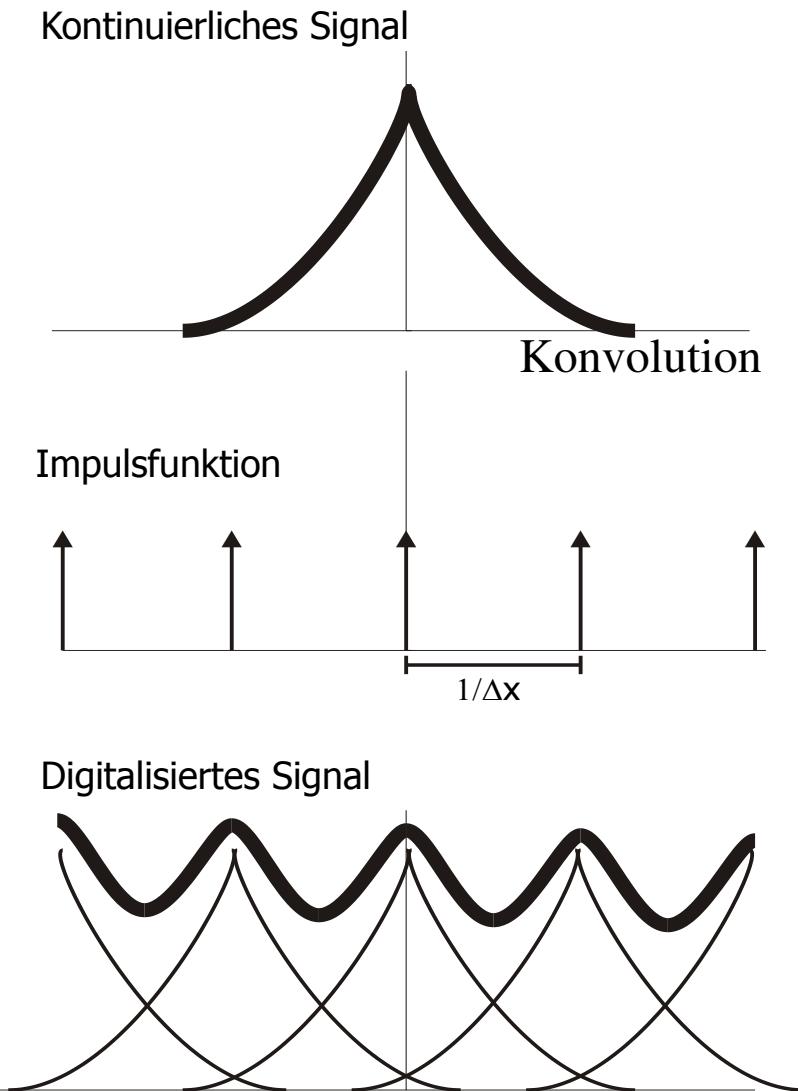
Was ist geschehen?

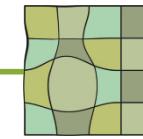
Fouriertransformierte Abtastfunktion:

Impulsfolge mit Abstand $1/\Delta x$ zwischen Impulsen.

Die Konvolution führt zur Überlappung von Anteilen des kontinuierlichen Signals (=Aliasfrequenzen).

Aliasfrequenzen tauchen nur dann nicht auf, wenn die Frequenzanteile nach der Konvolution sich nicht überlappen.



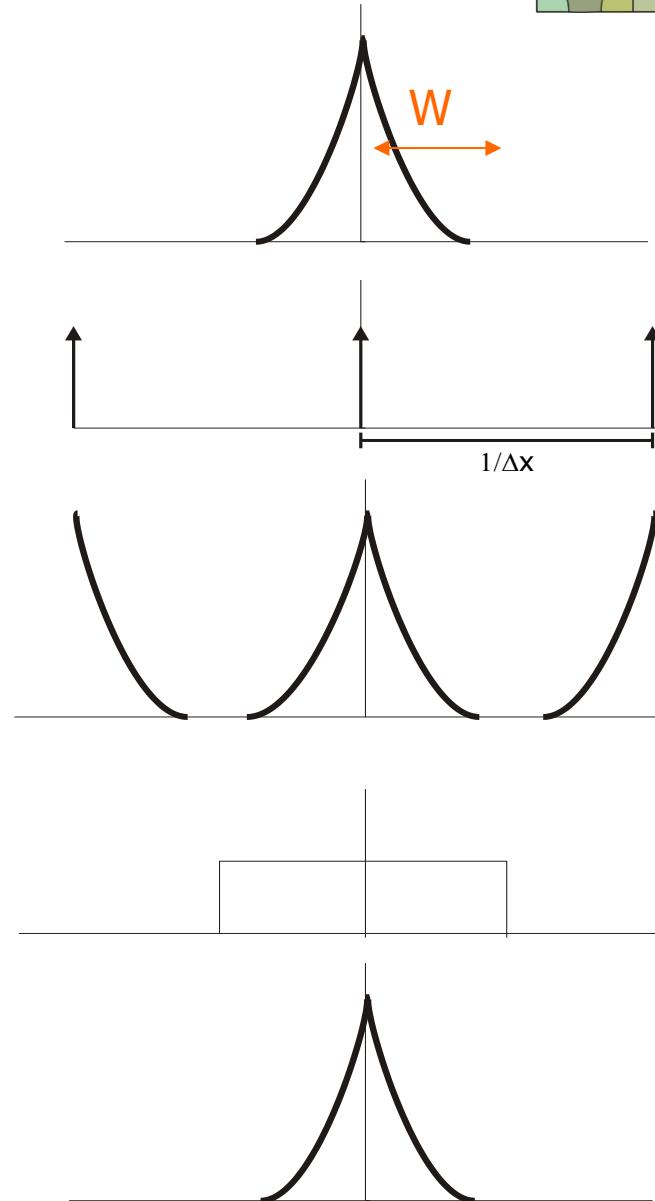


Shannon'sches Abtast-Theorem

Falls die aufgenommene Funktion bandlimitiert auf Frequenzen $< W$ ist und gilt:

$$2W \leq 1/\Delta x \quad \Leftrightarrow \quad \Delta x \leq 1/2W,$$

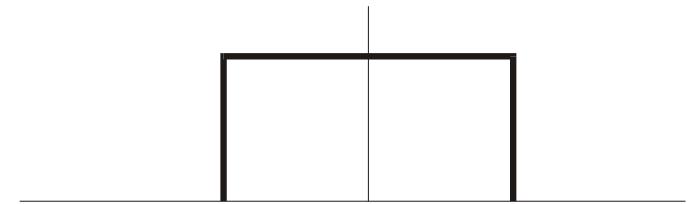
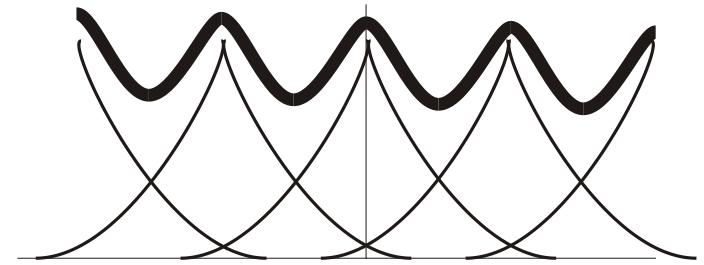
dann kann die ursprüngliche Funktion aus der abgetasteten Funktion komplett rekonstruiert werden.



Anti-Aliasing Techniken

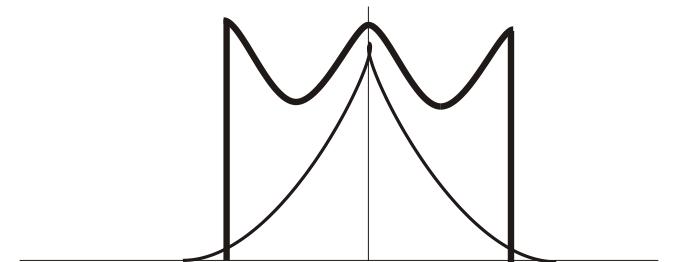
Abzutastende Funktion muss dem Shannon-Theorem genügen

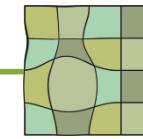
- Abtastrate solange erhöhen, bis $\Delta x \leq 1/(2W)$ gilt
- Filterung durchführen, so dass alle Frequenzen W mit $W \leq 1/(2\Delta x)$ gelöscht werden.



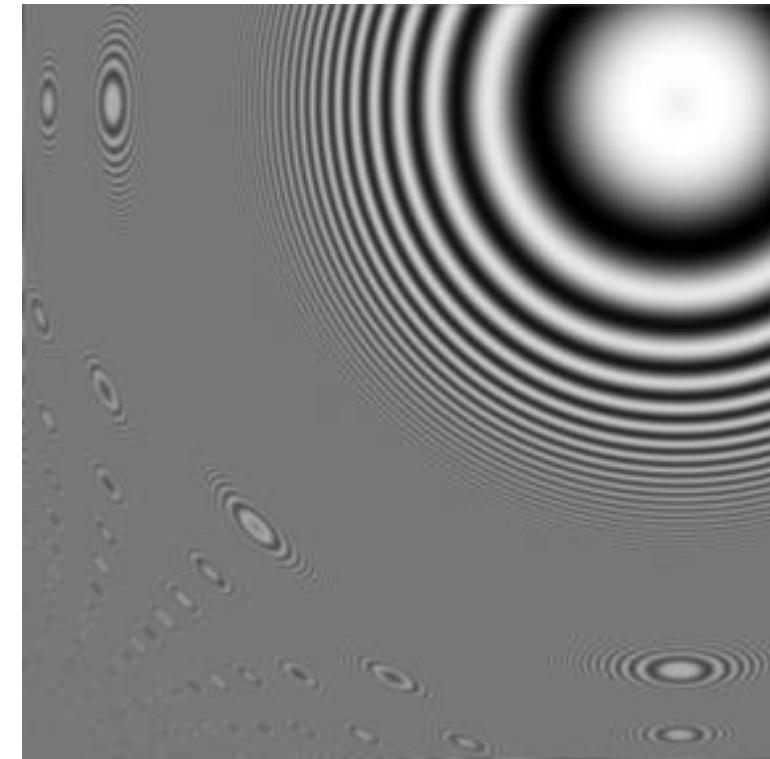
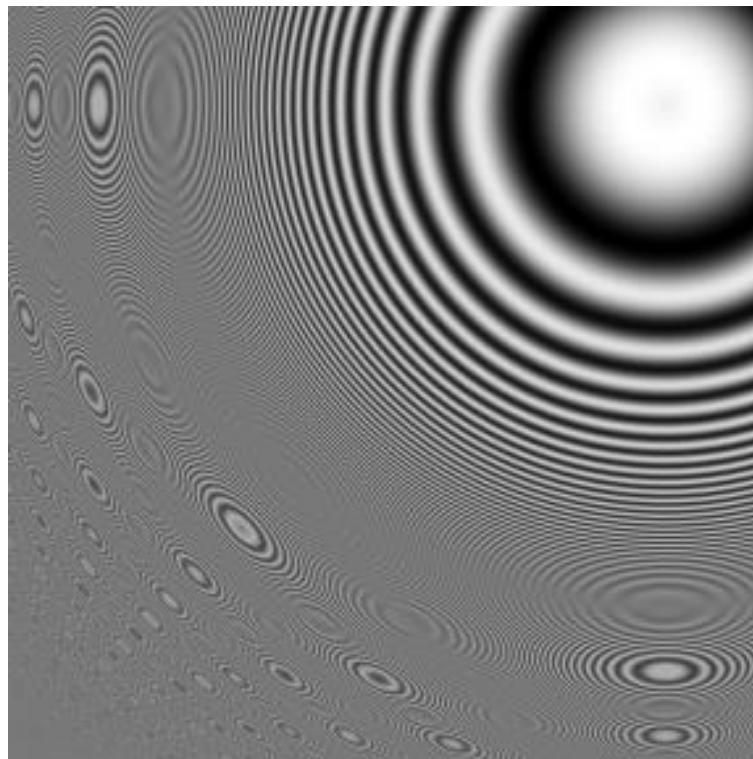
Probleme:

- Funktion ist meist **nicht bandlimitiert**.
- Filterung muss **vor der Abtastung** erfolgen.





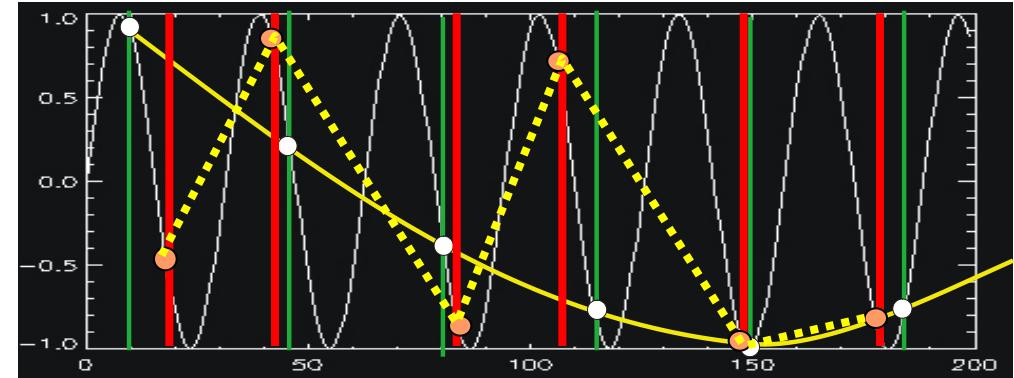
Anti-Aliasing (Beispiel)

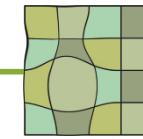


Tiefpassfilterung

Anti-Aliasing (Computergraphik)

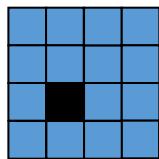
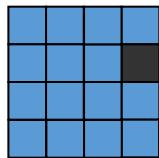
- Menschliche Wahrnehmung ist nicht Aliasing-empfindlich
- Erklärung:
 - Sehzellen sind nicht exakt gleichmäßig verteilt
 - Zerstört das (falsche) Regelmaß des gesehenen Objekts
- Simulation bei der Bildgenerierung (Computergraphik)
 - Zufällige Variation des genauen Abtastorts
 - Übertragen auf die Bildverarbeitung: Neuabtastung mit Reduktion der Ortsauflösung und zufälliger Variation der Abtastorte



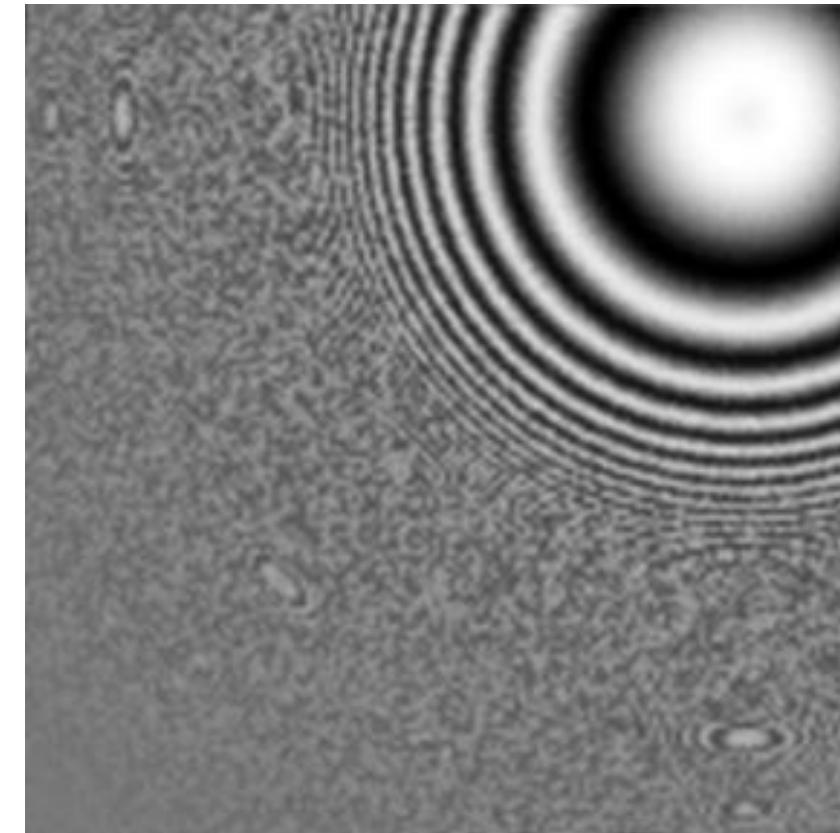
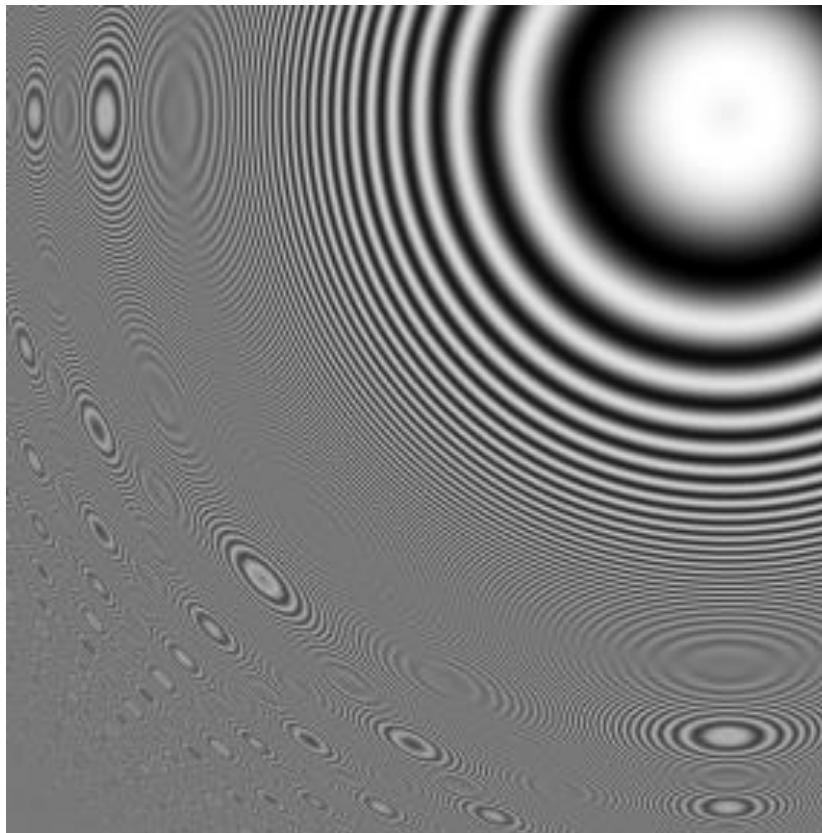


Anti-Aliasing (Beispiel)

Zufällige
Abtastorte



...

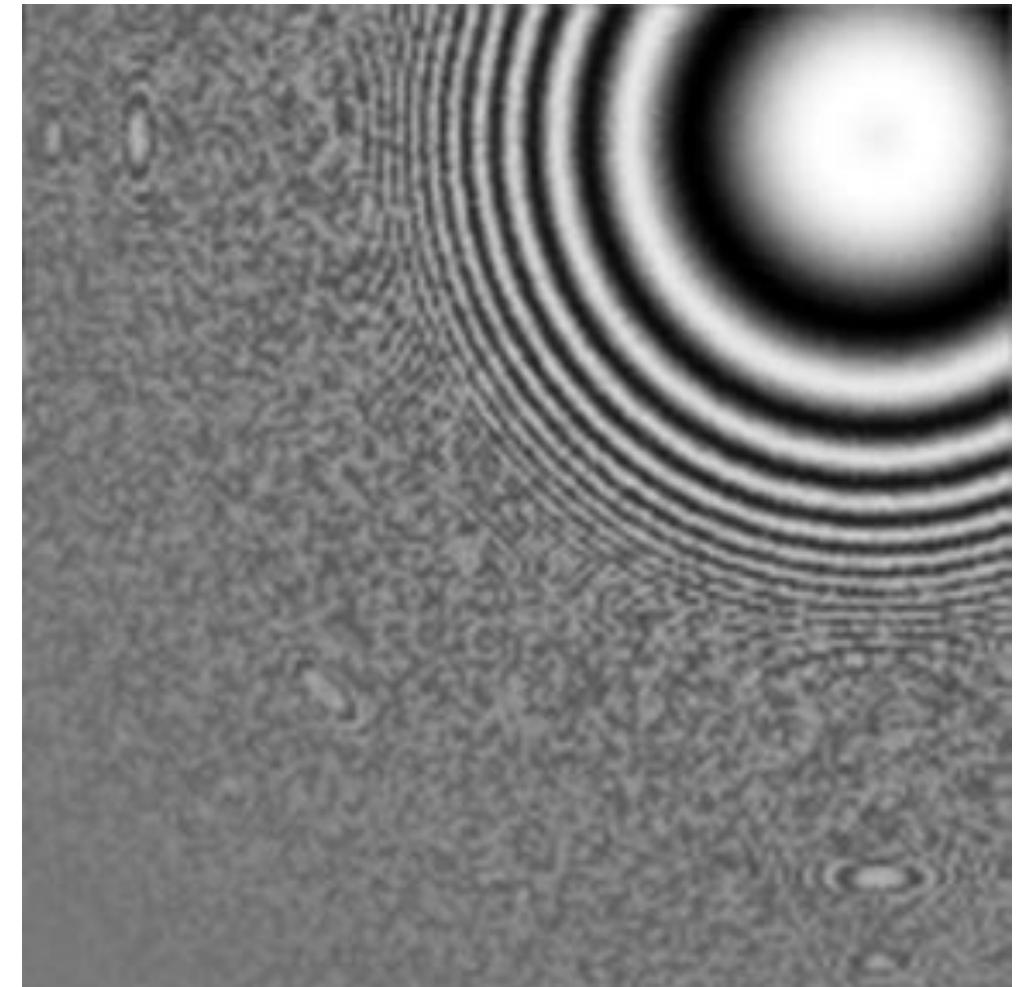


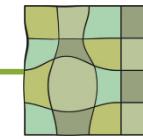
Was sollten Sie heute gelernt haben?

- Informationsverlust durch Abtastung
- Interpolation in Bildern
- Aliasing (Moiré-Effekt)
- Abtasttheorem

Famous Last Question

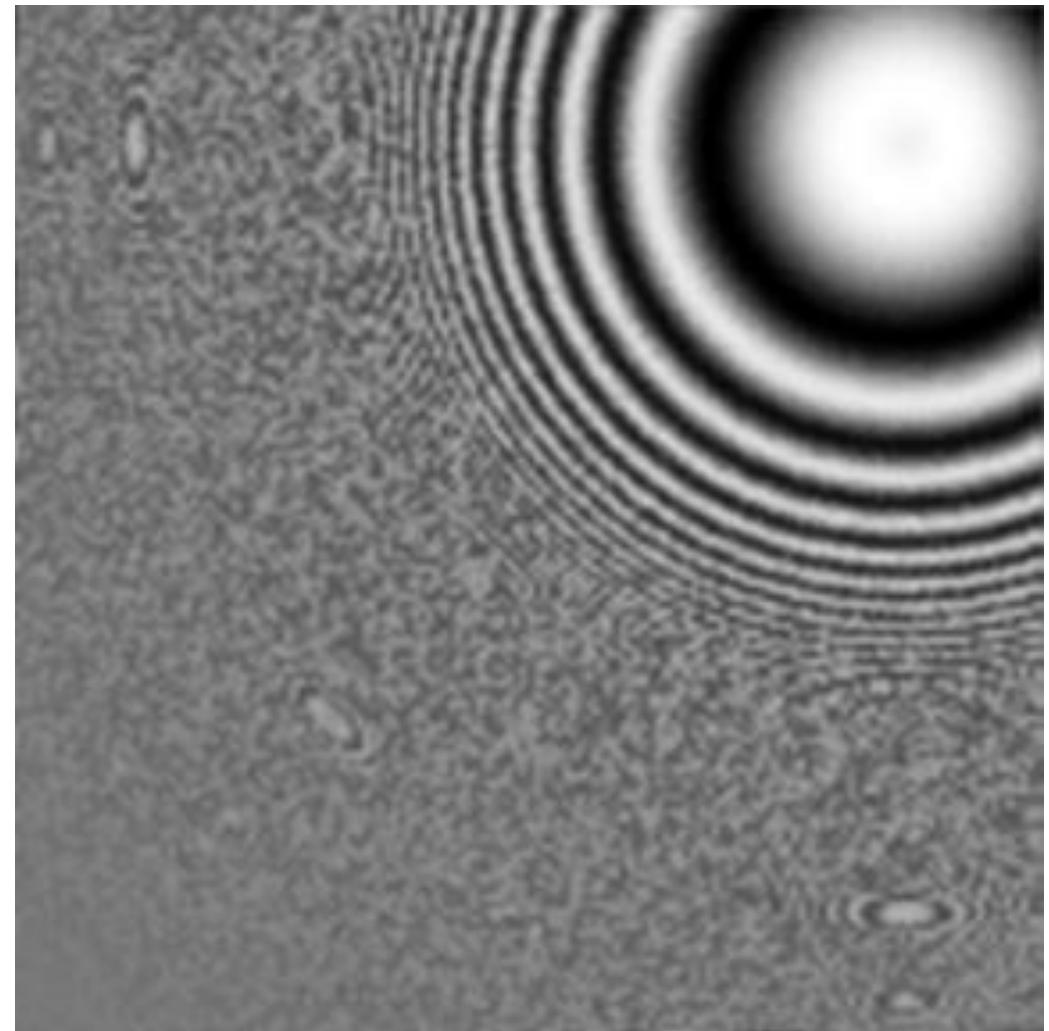
*Wenn die Abtastorte zufällig sind,
wieso sieht man noch die niedrigen
Frequenzen?*





Famous Last Question

Wenn die Abtastorte zufällig sind, wieso sieht man noch die niedrigen Frequenzen?



Bildrestauration

- Wiederherstellung eines ungestörten Bildes aus einem gestörten Bild
- Voraussetzungen
 - Gestörtes Bild ist bekannt
 - Störung kann in einer invertierbaren Form beschrieben werden
- Fragestellungen
 - Geeignete Repräsentation der Störung
 - Bestimmung der Störung
 - Invertierung der Störung



Störoperator

- Störung wird als linearer Operator A beschrieben
 - Modell: $Af = g$
 - Falls Störung A bekannt und invertierbar ist, kann das ungestörte Bild f aus $f = A^{-1}g$ aus dem gestörten Bild g berechnet werden
- Weitere Eigenschaft des Störoperators: Verschiebungsinvarianz
 - leicht regularisierbar und invertierbar
 - Ein verschiebungsinvarianter Operator wirkt im gesamten Bild gleich

$$A \circ f(x+a, y+b) = [A \circ f](x+a, y+b)$$

Verschiebungsinvarianz und Konvolution

Verschiebungsinvarianz bedeutet, dass die Zeilen von \mathbf{A} durch zyklische Verschiebung aus einer Zeile gewonnen werden können, z.B. bei einer Bildzeile (1-D Bild):

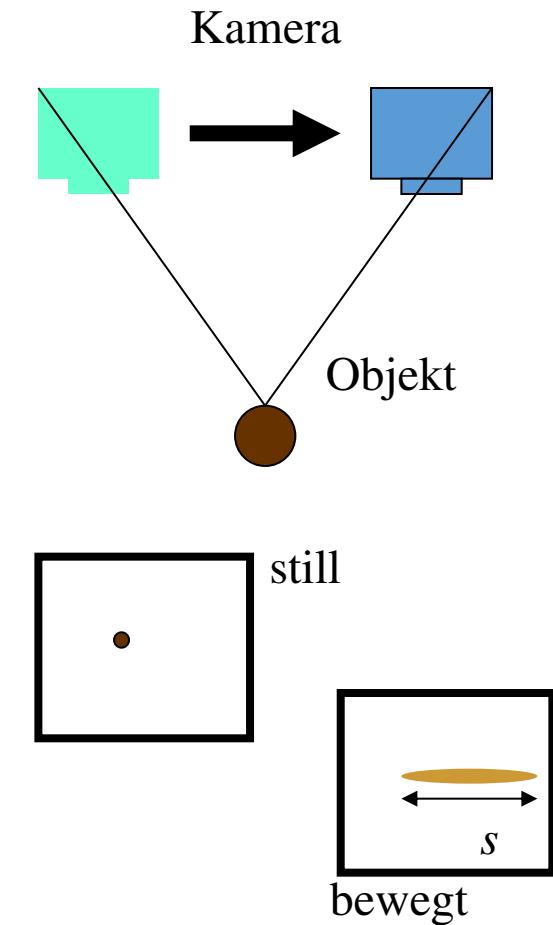
$$\mathbf{A} = \begin{pmatrix} a_1 & a_2 & a_3 & \dots & a_M \\ a_M & a_1 & a_2 & & a_{M-1} \\ a_{M-1} & a_M & a_1 & & \\ \dots & & & \dots & \\ a_2 & & & & a_1 \end{pmatrix}$$

Wegen der zyklischen Verschiebung der Zeile in \mathbf{A} kann die Störung als Faltung des Bildes \mathbf{f} mit einem Störoperator \mathbf{h} ausgeführt werden

$$(h * f)(m, n) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} h(i, j) \cdot f(i - m, j - n)$$

Beispiel I: Bewegungsunschärfe

- Über einen Zeitraum Δt wird ein Objektpunkt p auf immer andere Punkte auf dem CCD-Chip abgebildet.
- Bei unbewegter Kamera sei die Bildhelligkeit des abgebildeten Punkts h .
- ... dann ist sie bei bewegter Kamera $h/\Delta s$, wobei Δs die zurückgelegte Strecke ist.
- Wenn Δs für alle Punkte gleich ist, dann lässt sich die Veränderung durch eine Faltung beschreiben.



Bewegungsunschärfe

- Faltungskern ist eine Funktion w mit $w(t \cdot \cos \alpha, t \cdot \sin \alpha) = \begin{cases} \frac{1}{\Delta s} & \left| \frac{t}{2} \right| < \Delta s \\ 0 & \text{sonst} \end{cases}$
- Der Winkel α gibt die Bewegungsrichtung an.
- Die Strecke Δs gibt die Strecke an, um die sich der Punkt bewegt hat:

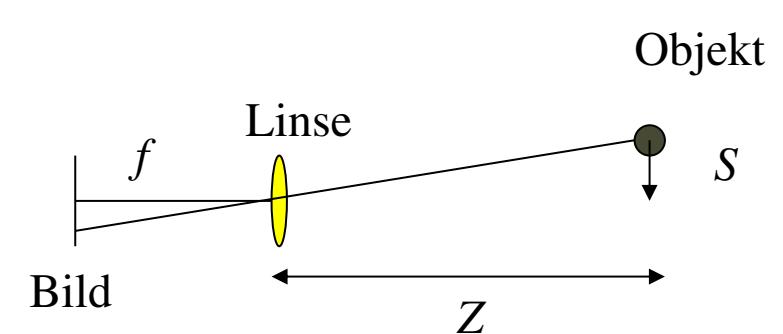
$$\Delta s = \frac{f}{Z} \cdot \frac{\Delta S}{p}$$

f : Brennweite

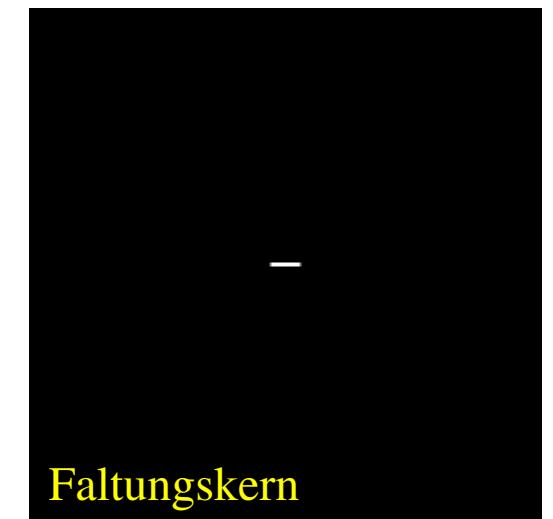
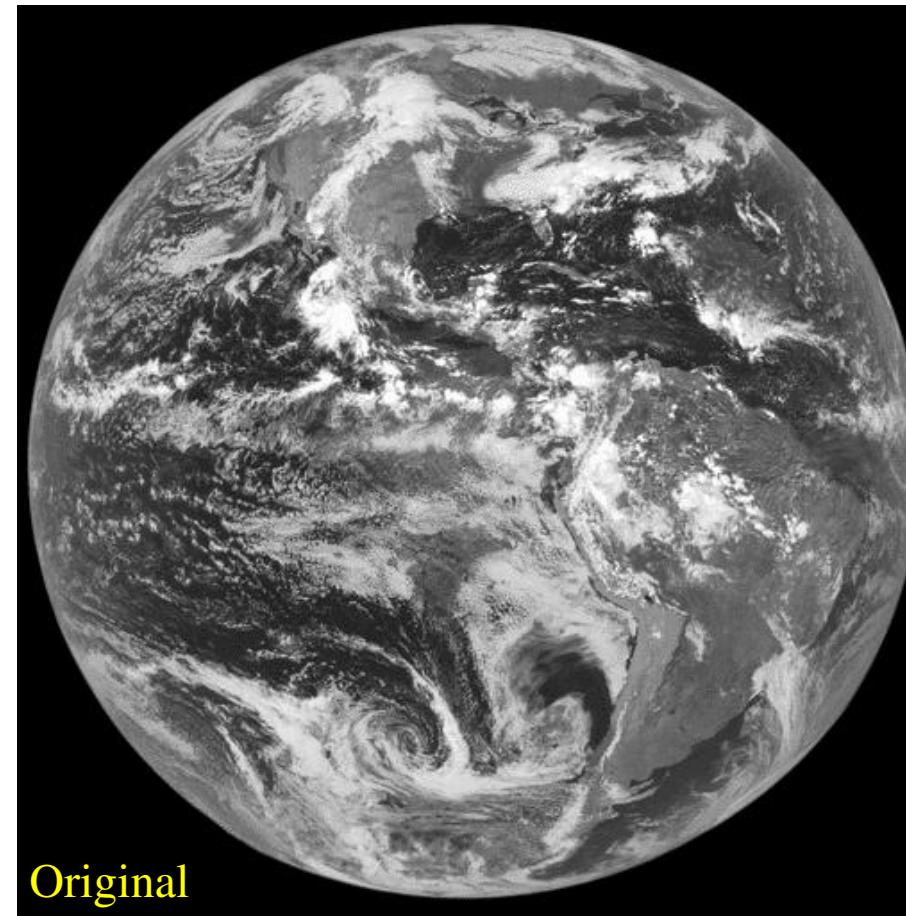
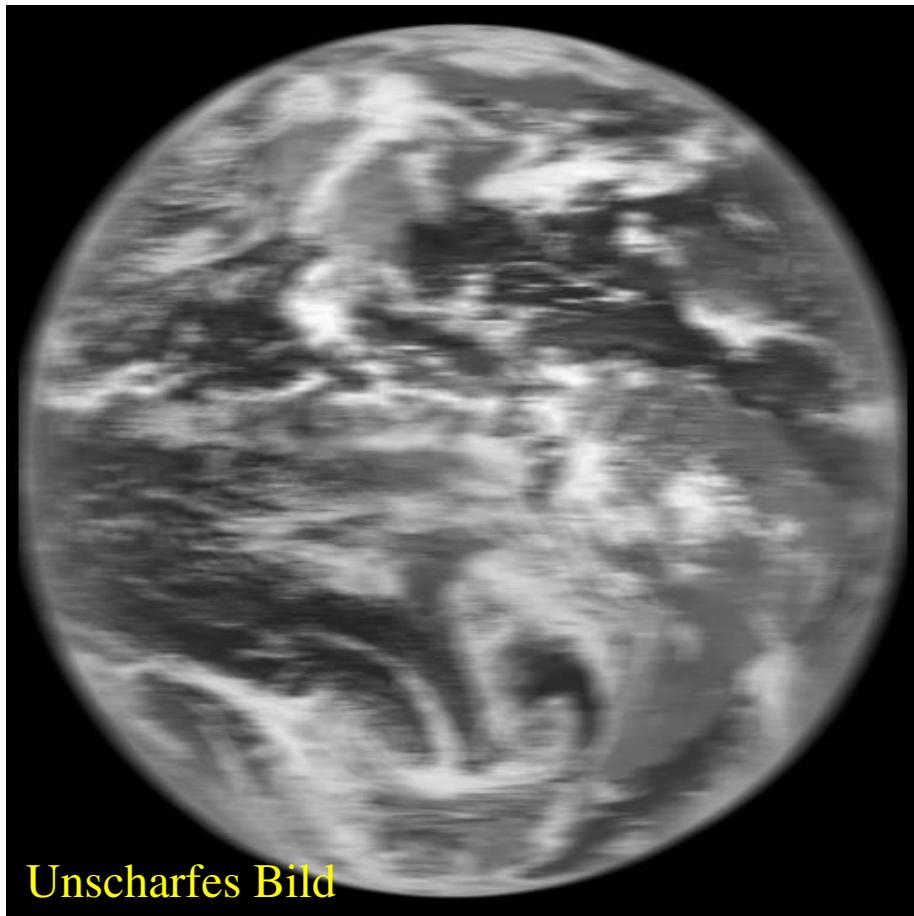
Z : Objektabstand

Δs : Bewegung in der X-Y-Ebene

p : Pixelgröße

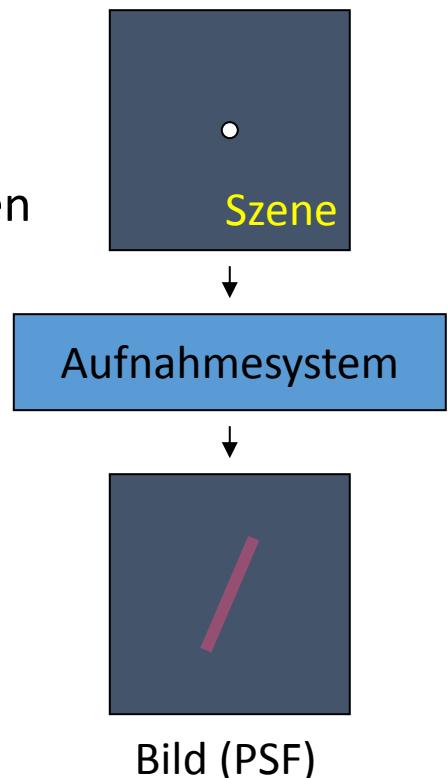


Bewegungsunschärfe



Repräsentation linearer Störungen

- Jede verschiebungsinvariante, lineare Operation wird vollständig durch die Faltungsfunktion beschrieben.
- Die Faltungsfunktion beschreibt die Operation für beliebige Bilder
- Die Faltungsfunktion kann als Resultat der Veränderung eines Punkts erzeugt werden
- Punktantwort = **Point Spread Function (PSF)**



Beispiel II: Fokussierungsunschärfe

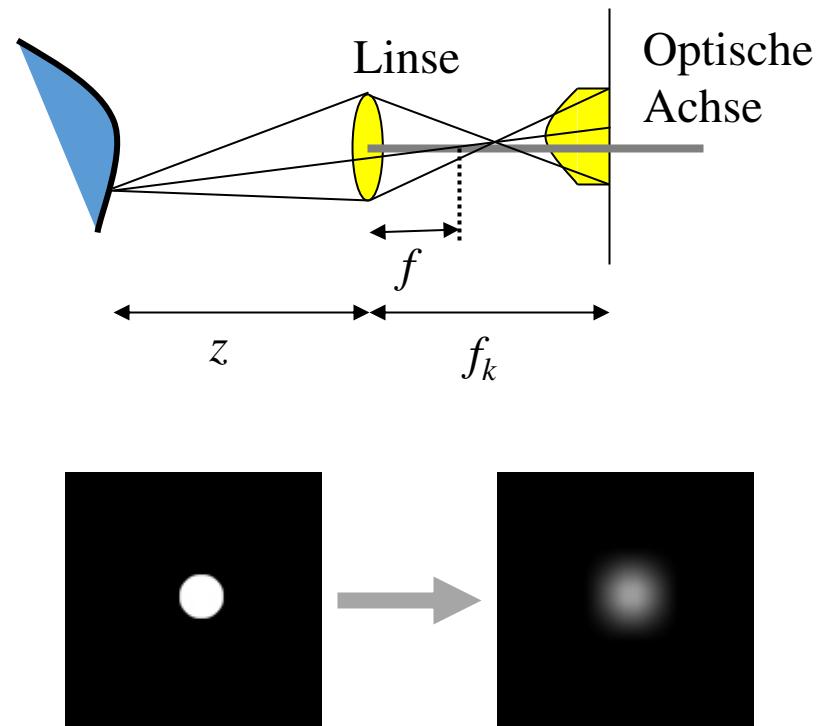
- Maß der Unschärfe hängt vom Punktabstand z , der Brennweite der Linse f und der Kammerkonstante f_k ab.

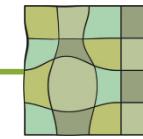
- Linsengesetz $\frac{1}{z} = \frac{1}{f} + \frac{1}{f_k} \Leftrightarrow f_k = \frac{zf}{f-z}$

- Größe des Unschärfekreises:

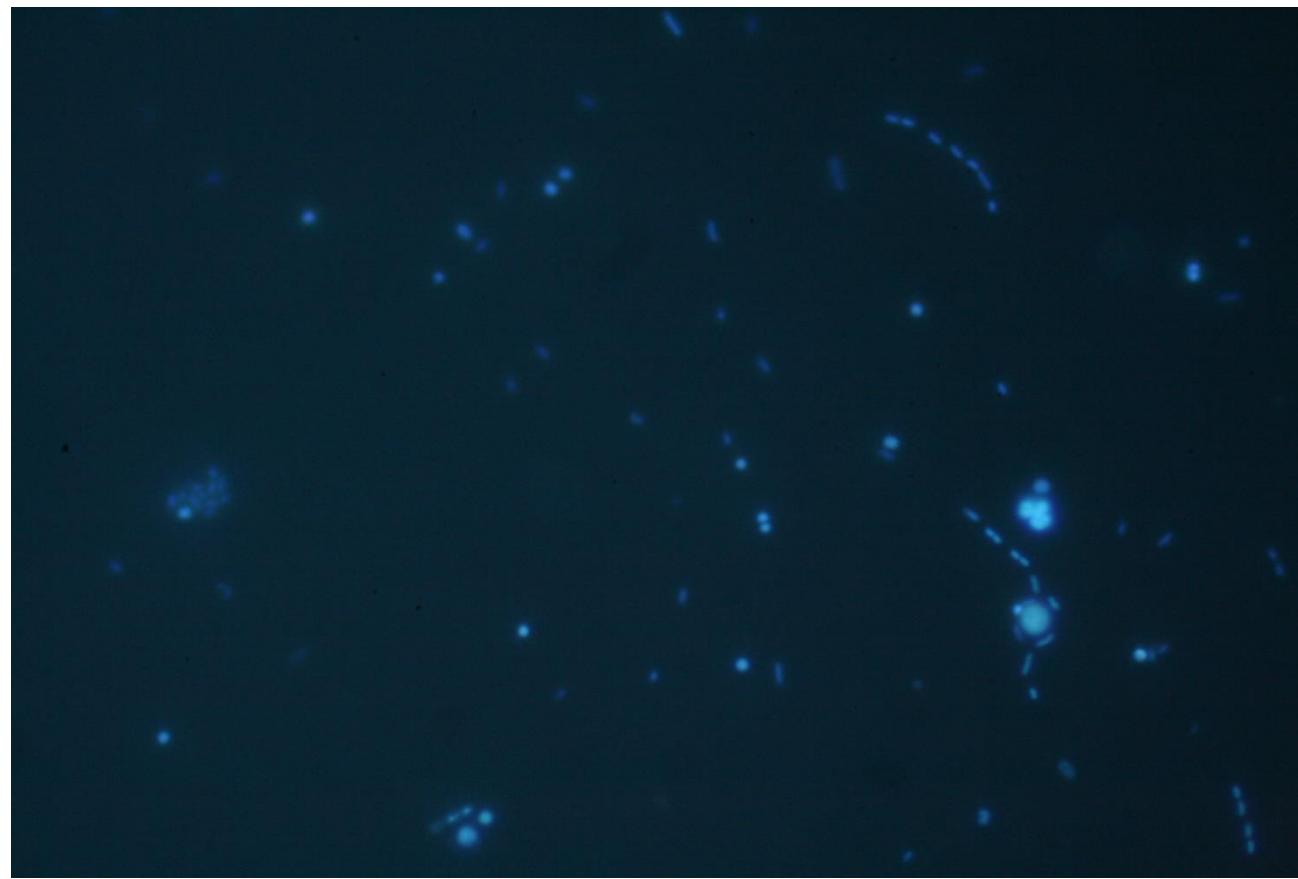
$$\frac{d}{f} = \frac{s}{f_k - f} \Leftrightarrow s = \frac{(f_k - f)d}{f} = \frac{f_k d}{f} - d = \frac{z}{f-z} - d$$

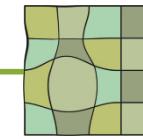
- Unschärfe kann durch Aufnahme eines punktförmigen Testobjekts angenähert werden.





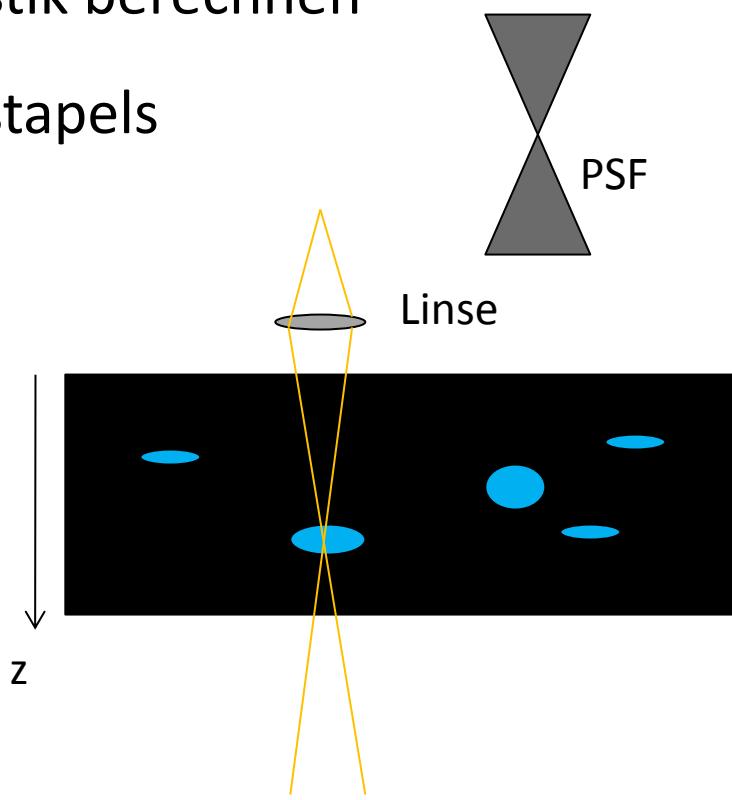
Beispiel – Mikroskopische Aufnahmen

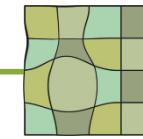




Mikroskopische Aufnahmen

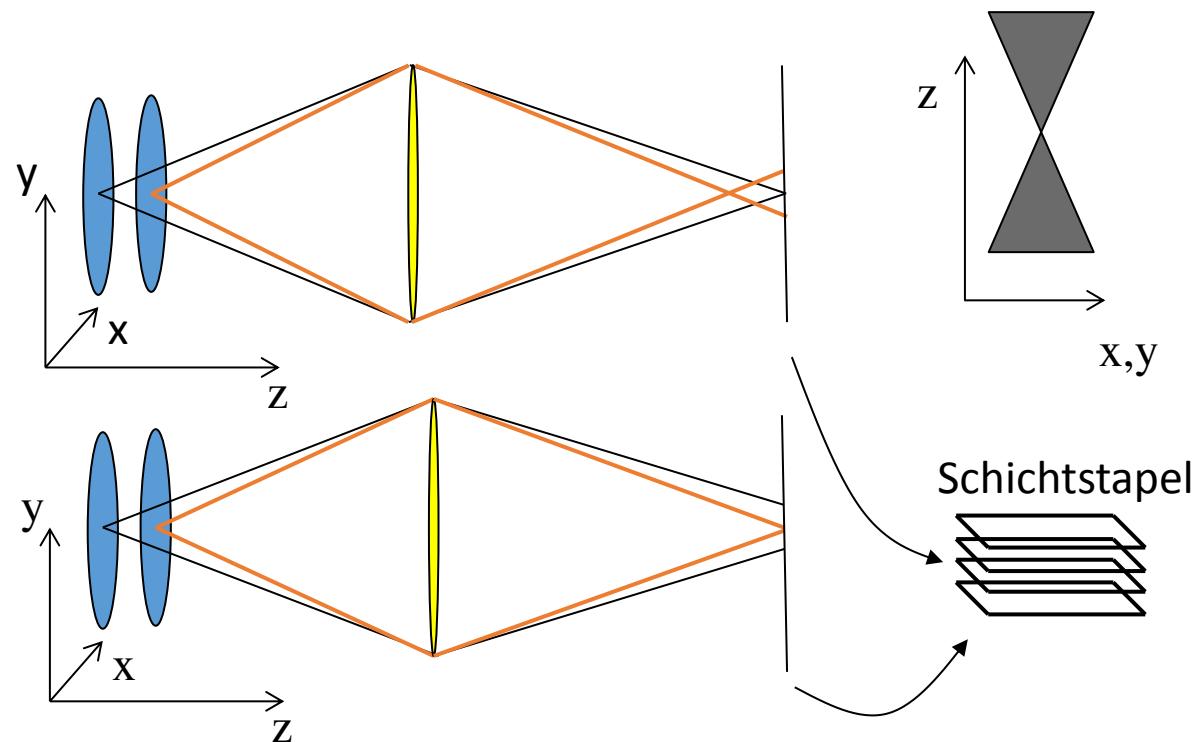
- Aufnahmen mit unterschiedlicher Fokustiefe generieren = Schichtstapel
- PSF aus der Linsencharakteristik berechnen
- Inverse Filterung des Schichtstapels





3d-Rekonstruktion

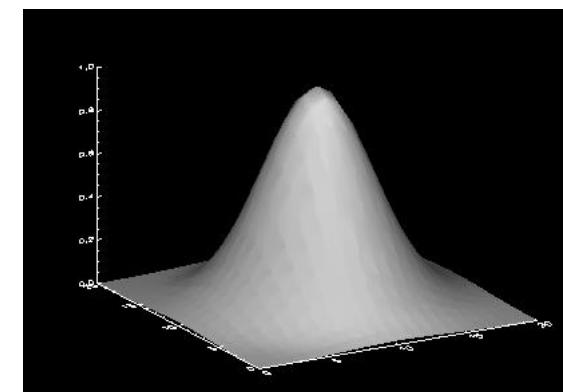
- Einzelbilder mit unterschiedlicher Fokussierung
- Ursprung der PSF wird in z-Richtung verschoben, um die unterschiedlichen Ebenen zu restaurieren



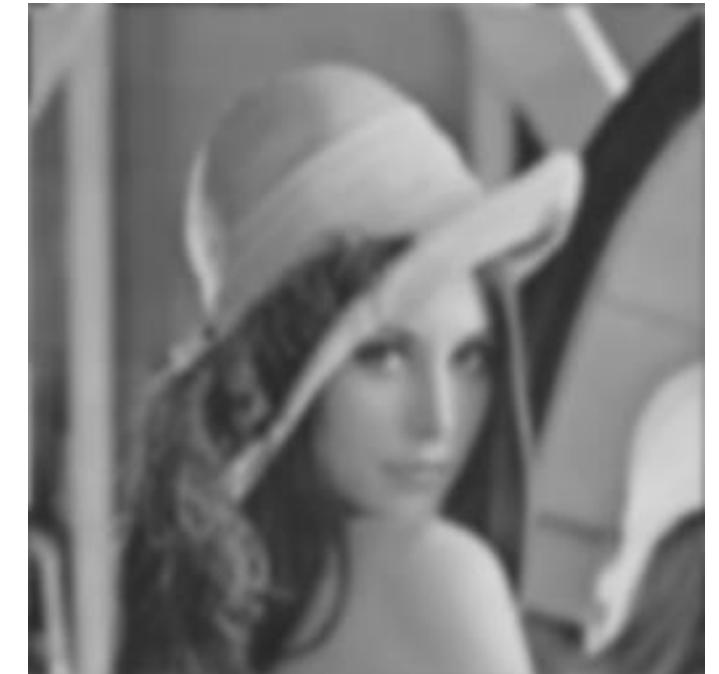
Bestimmung einer unbekannten PSF



Störung H



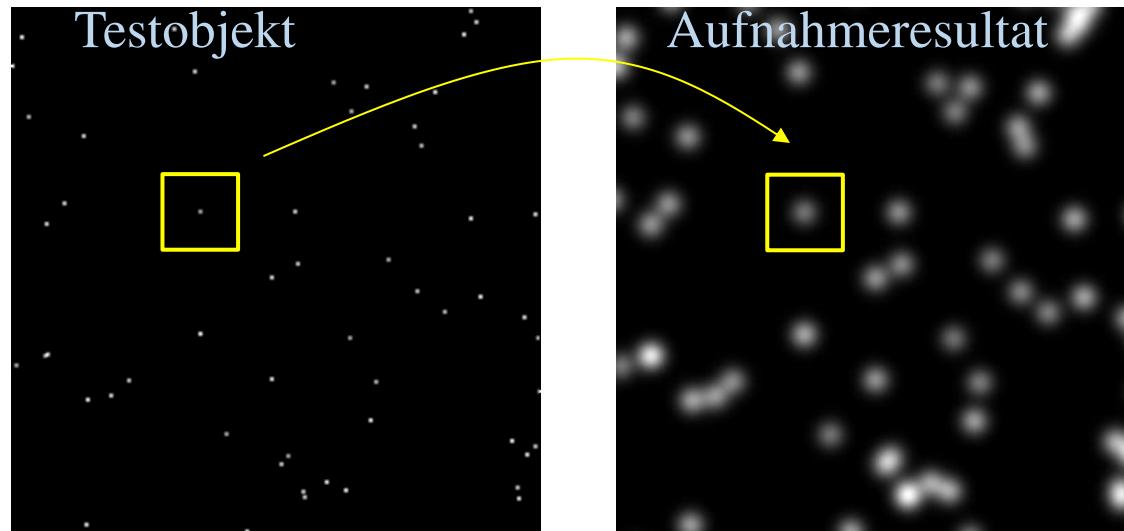
PSF



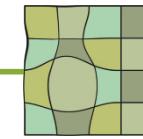
Gesucht:

PSF von Testbildern

Annahme: Störung ist unveränderlich und Testaufnahme ist möglich.

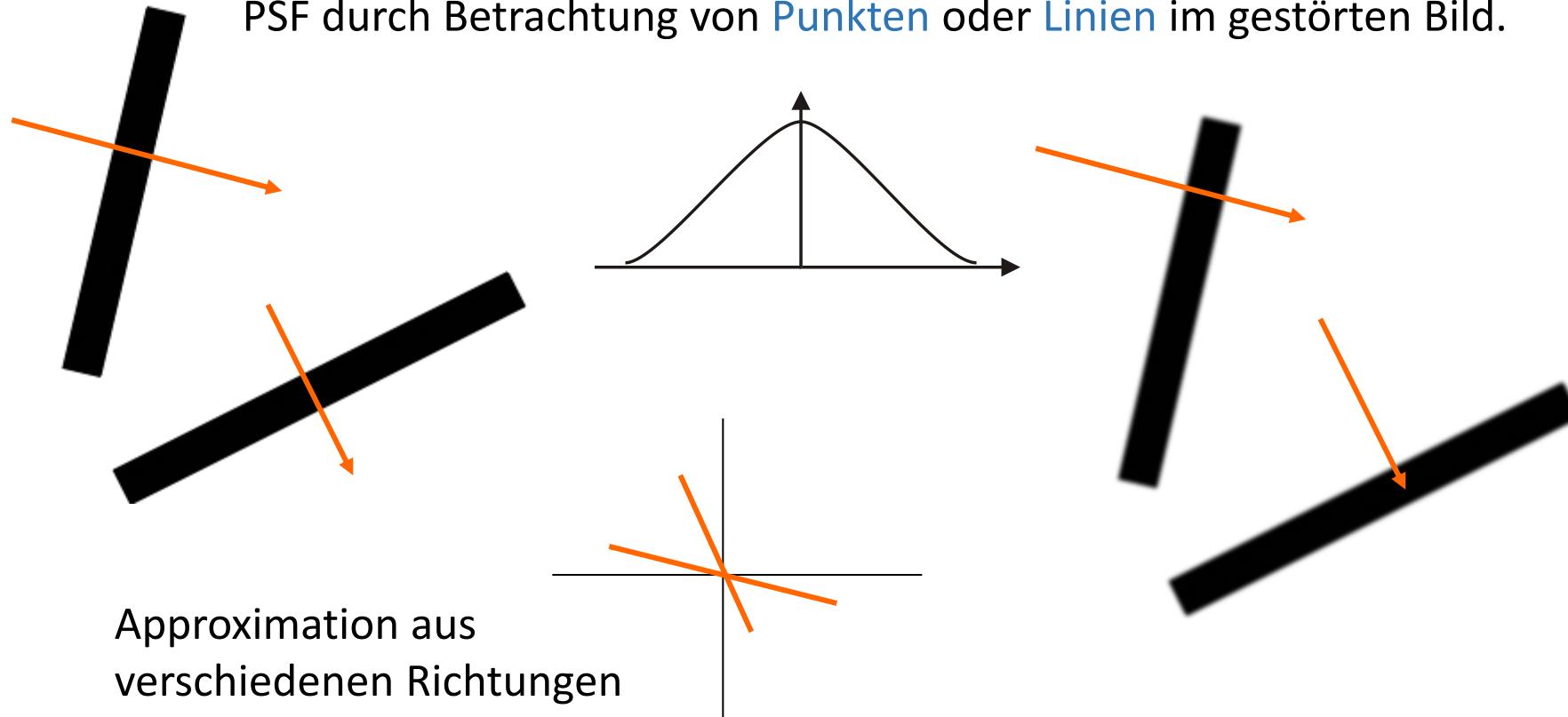


Aufnahme ist eine
Näherung für die PSF.



PSF aus dem aufgenommenen Bild

Testaufnahme ist nicht möglich: Näherungsweise Bestimmung der PSF durch Betrachtung von **Punkten** oder **Linien** im gestörten Bild.



Kanten

Die meisten Bilder weisen wenige Linien oder Punkte auf, aber Kanten können in fast jedem Bild gefunden werden.



Kante

- Was sind Kanten?
- Wie können Kanten zur Bestimmung der PSF benutzt werden?

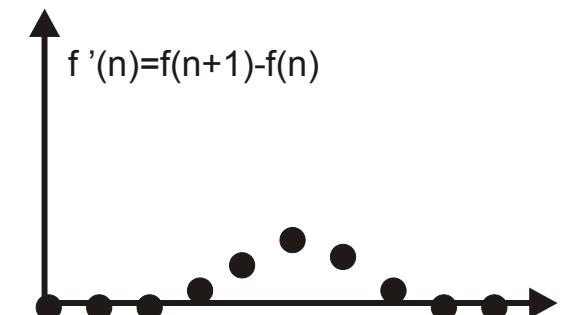
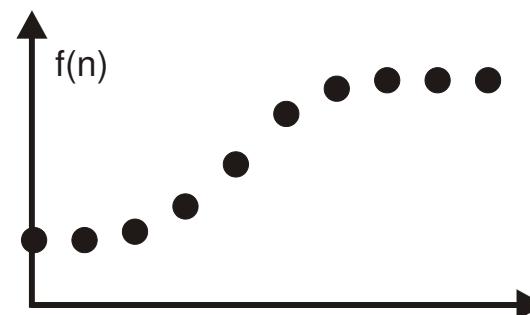
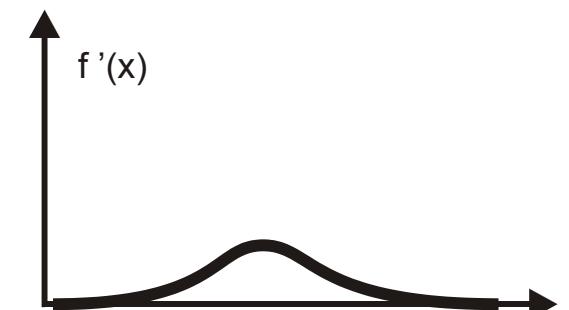
1D-Kanten

Die **Stärke einer Kante** hängt von der Steigung der Funktion ab:

- Betrag der ersten Ableitung bestimmen.

Für **diskrete Funktionen**:

- Ableitung wird durch Differenz angenähert



Approximation der Ableitung

Aus Differentialen werden Differenzen:

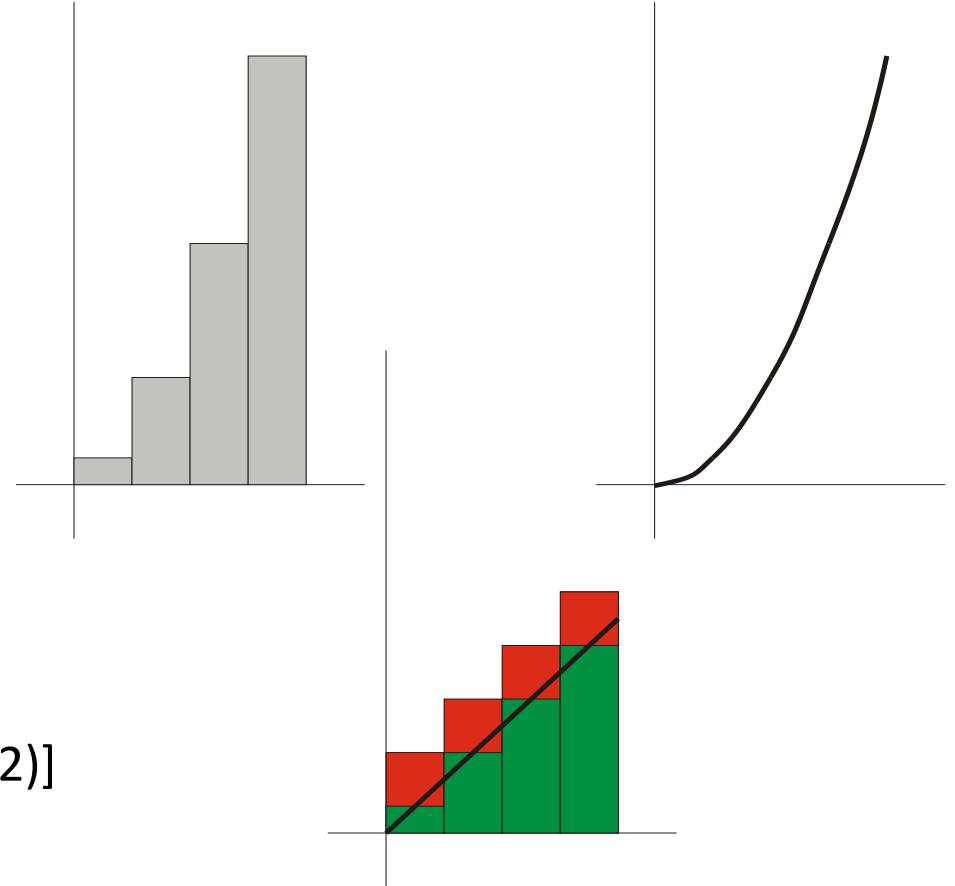
$$\partial f(x)/\partial x \approx [f(x)-f(x-1)] / [x-(x-1)]$$

oder

$$\partial f(x)/\partial x \approx [f(x+1)-f(x)] / [(x+1)-x]$$

Auch andere Näherungsverfahren sind möglich.

$$\text{Bsp.: } \partial f(x)/\partial x \approx [f(x)+f(x+1)-f(x-1)-f(x-2)] / [x+x+1-(x-1)-(x-2)]$$



Kanten im 2-D Raum: Gradienten

Gradient im kontinuierlichen Raum (x,y) : Vektor der partiellen Ableitungen der Bildfunktion in x - und y -Richtung:

$$\vec{G}(f(x, y)) = \begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{pmatrix}$$

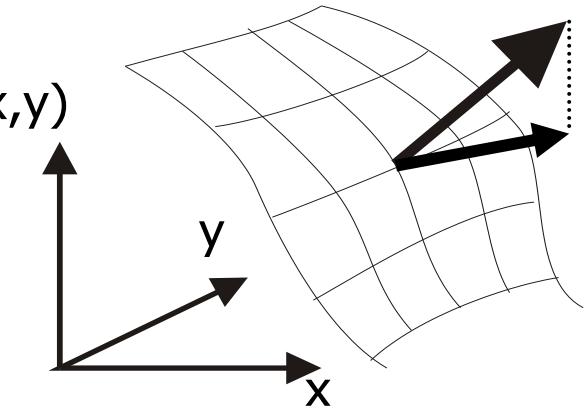
Richtung: Richtung der größten Steigung.

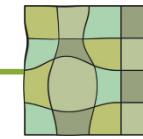
Länge: Stärke der stärksten Steigung.

Approximation des Gradienten: Differential wird durch Differenz approximiert:

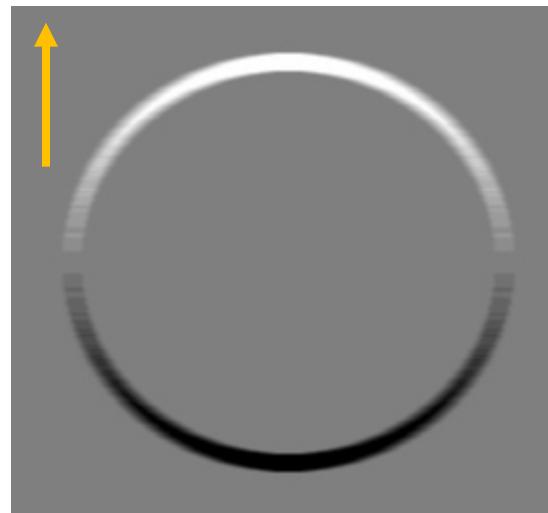
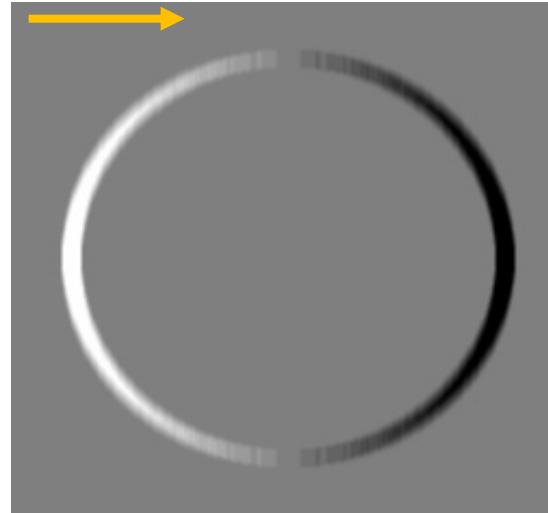
z.B. $\vec{G}(f(m, n)) \approx (G_x(m, n) \quad G_y(m, n)) = (f(m, n) - f(m-1, n) \quad f(m, n) - f(m, n-1))$

Die Länge des Gradienten ist sein Betrag oder näherungsweise $|G_x| + |G_y|$.





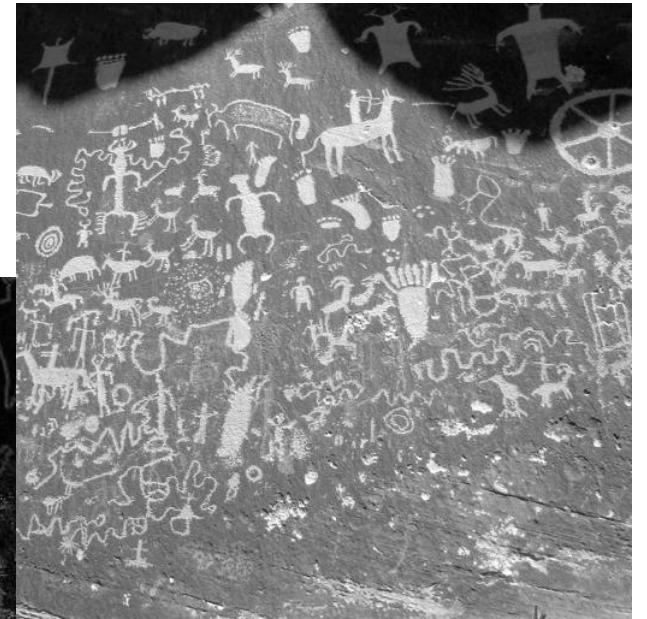
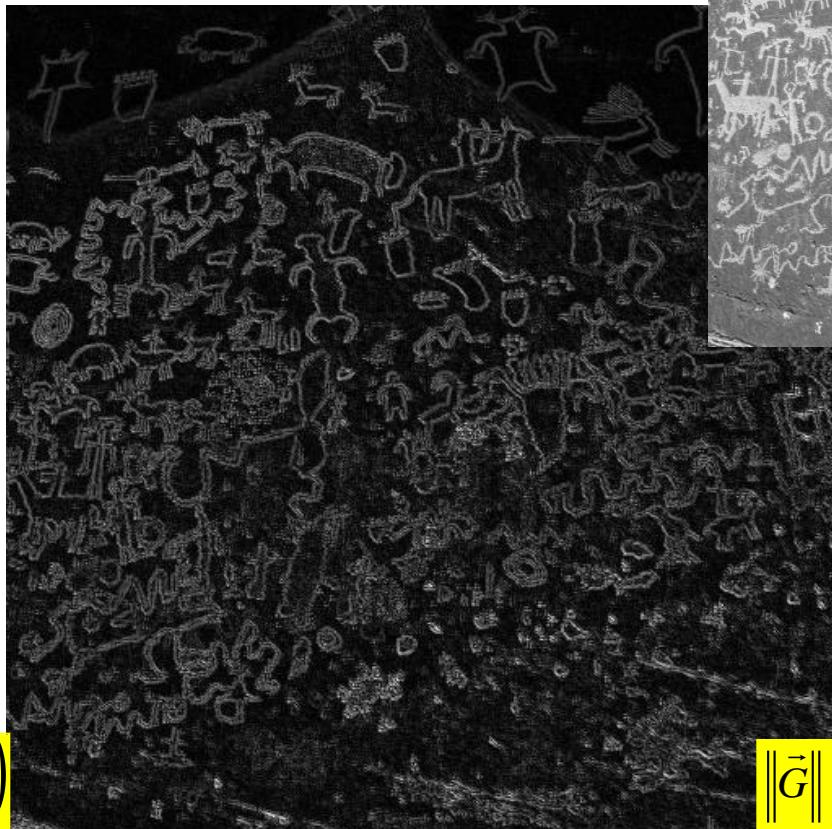
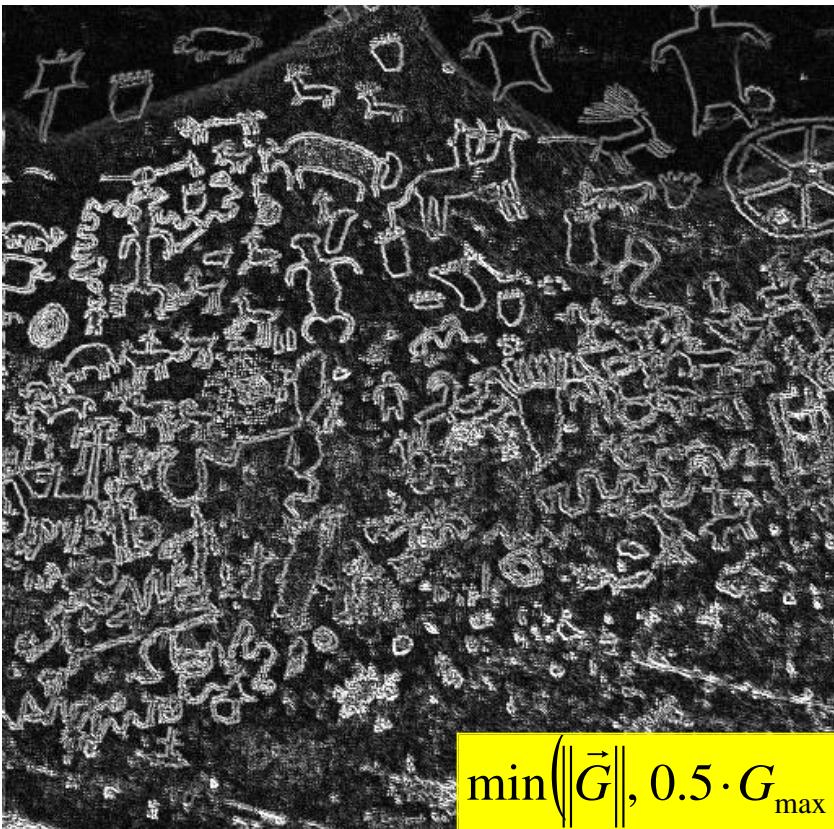
Elemente des Gradienten

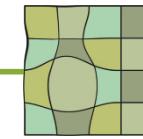


$$\text{Betrag} : \sqrt{(G_x^2 + G_y^2)}$$

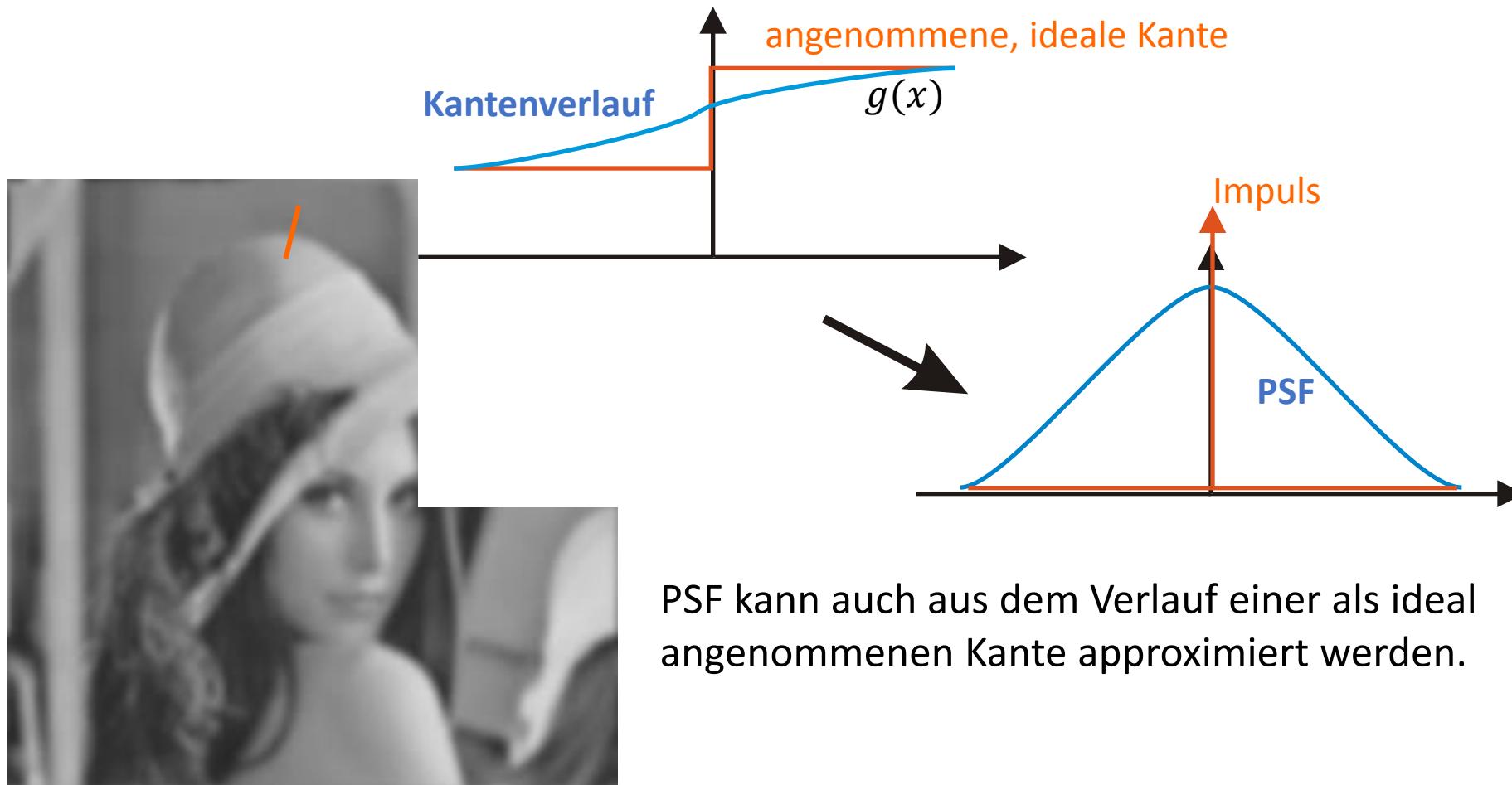
$$\text{Richtung: } \tan^{-1}(G_y/G_x)$$

Gradientenlänge





PSF aus Kanten

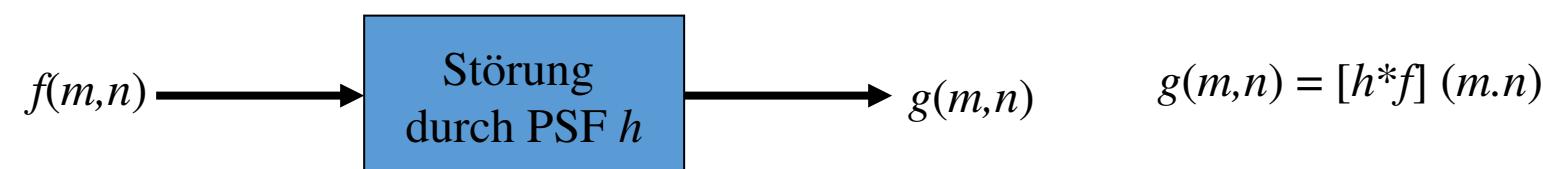


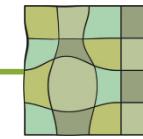
Invertierung der Störung

- Störeinfluss auf ein Bild mit $N \times N$ Pixeln lässt sich als lineares Gleichungssystem mit $N \times N$ Unbekannten repräsentieren.

$$\begin{pmatrix} a_{00} & a_{01} & \dots \\ a_{01} & \dots & \\ \dots & a_{10} & \\ a_{10} & \dots & \\ \dots & & \end{pmatrix} \begin{pmatrix} f_{0,0} \\ f_{0,1} \\ \dots \\ f_{0,N-1} \\ f_{1,0} \\ \dots \\ f_{N-1,N-1} \end{pmatrix} = \begin{pmatrix} g_{0,0} \\ g_{0,1} \\ \dots \\ g_{0,N-1} \\ g_{1,0} \\ \dots \\ g_{N-1,N-1} \end{pmatrix}$$

- Invertierung: Lösung großer Gleichungssysteme
- Besser: Dekonvolution im Frequenzraum





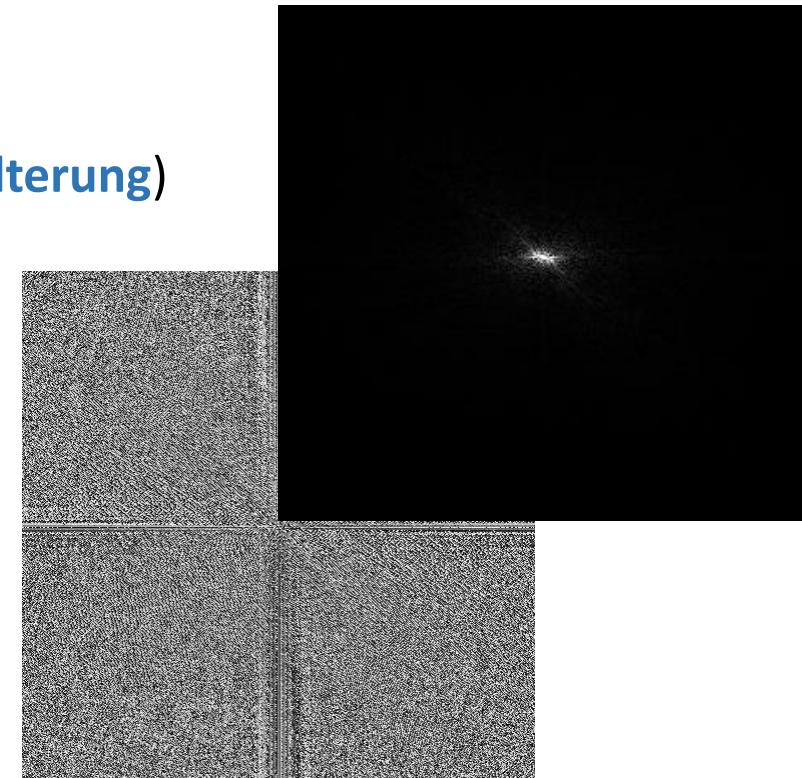
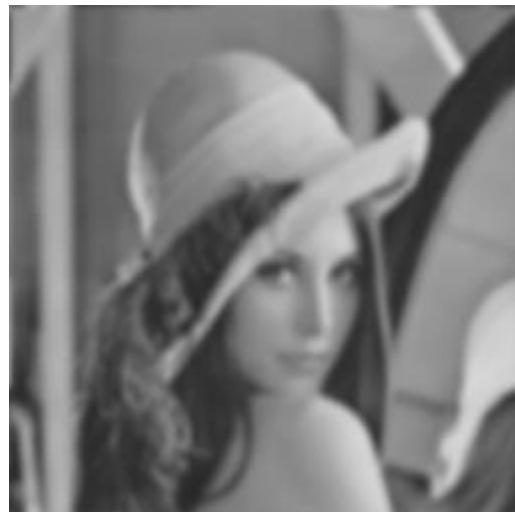
Invertierung der Störung

Überführung der Repräsentation in den Frequenzraum:

$$G(u, v) = \mathbf{FT}[g(m, n)] = \mathbf{FT}[[h * f](m, n)] = H(u, v) \cdot F(u, v)$$

- Invertierung:

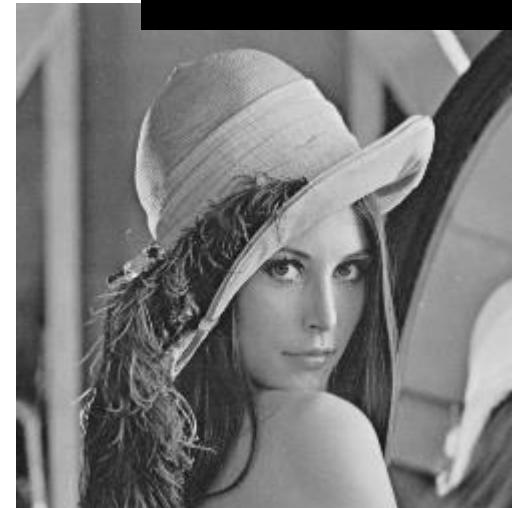
$$f(m, n) = \mathbf{FT}^{-1}[G(u, v)/H(u, v)] \text{ (Inverse Filterung)}$$



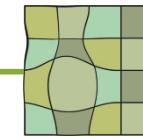
Inverse Filterung

$$\mathbf{FT}^{-1} \left(\begin{array}{c|c} \text{[Image with noise]} & \text{[Filtered image]} \\ \hline & \end{array} \right)$$

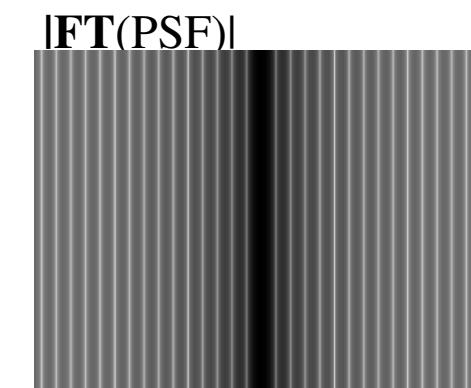
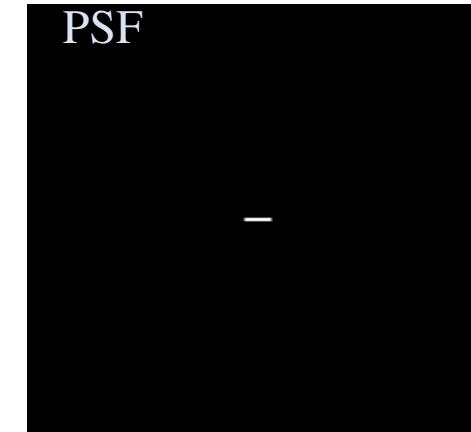
=



Vollständige Rückgewinnung der Information aus den gestörten Daten



Bewegungsunschärfe



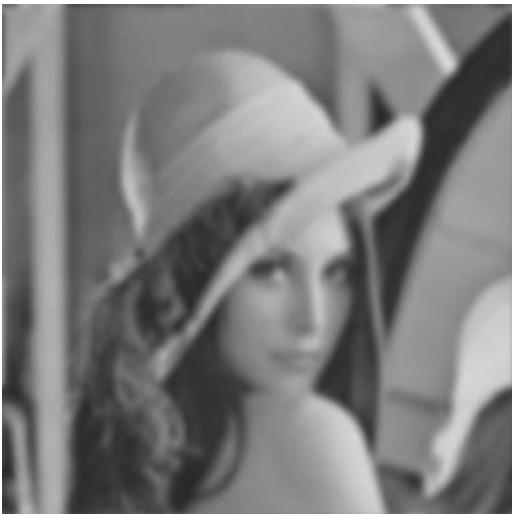
Bewegungsunschärfe



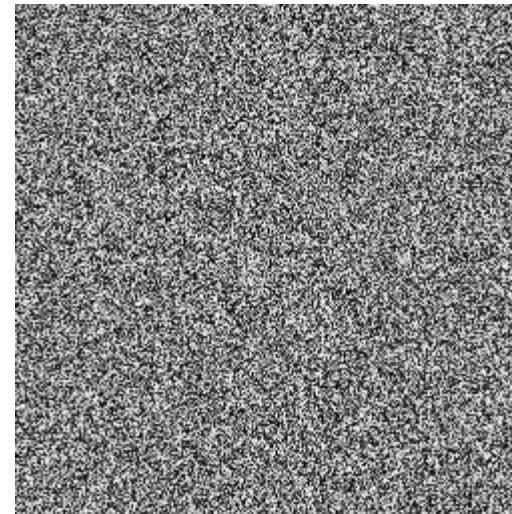
Resultat der Inversen Filterung

$$\mathbf{FT}^{-1} [\mathbf{FT}(g(u, v)) / \mathbf{FT}(PSF(u, v))]$$

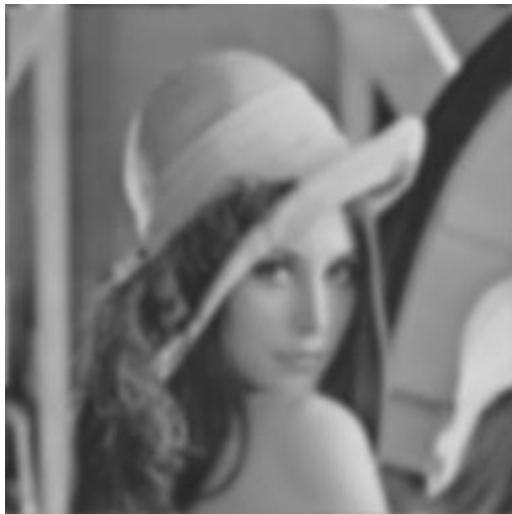
Was ist geschehen?



Inverse Filterung



Rauschen als Störeinfluss



Inverse Filterung

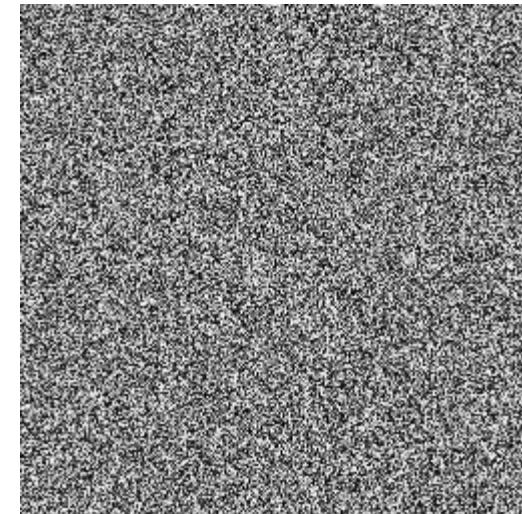


Bild hat ein (geringes) Rauschen n überlagert (SNR in diesem Fall besser als 100:1):

$$g(m, n) = [h * f](m, n) + n(m, n) \quad | \quad G(u, v) = H(u, v) \cdot F(u, v) + N(u, v)$$

Inverse Filterung:

$$\begin{aligned} G_f(u, v) &= G(u, v) / H(u, v) = (H(u, v) \cdot F(u, v) + N(u, v)) / H(u, v) \\ &= F(u, v) + N(u, v) / H(u, v) \end{aligned}$$

Bildrauschen

- nicht-deterministischer (nicht wiederholbarer) Einfluß
- Beschreibbar als Wahrscheinlichkeit, wie ein Pixel gestört ist.
 - Quantenrauschen
 - Impulsrauschen

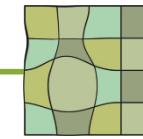


Normalverteiltes Rauschen

- Modell für Quantenrauschen (Wahrscheinlichkeit, dass Lichtquant geradlinig ausbreitend auf den Sensor trifft).
- Gauß'sche Normalverteilung

$$n(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\left(\frac{x-x_0}{\sigma}\right)^2\right)$$

- Erwartungswert x_0 wird mit $x_0 = 0$ angenommen.
- Varianz σ^2 ist der Grad des Rauschens

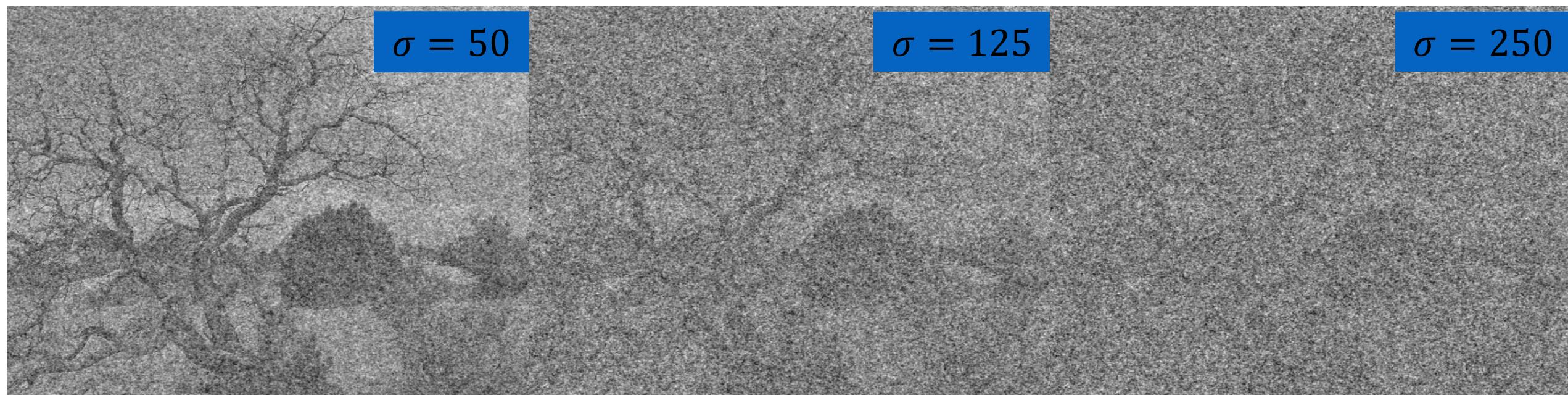


Beispiel

$$\sigma = 50$$

$$\sigma = 125$$

$$\sigma = 250$$

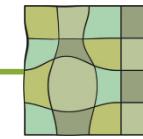


Signal-Rausch-Verhältnis

- Charakterisierung des Grads des Rauschens.
- **Signal:** Unterschied zwischen Objekt und Hintergrund. Falls das Objekt nicht bekannt ist:
 - Maximaler Grauwert
 - Durchschnittlicher Grauwert
- **Rauschen** ist über Standardabweichung σ gegeben
- SNR-Maße z.B.

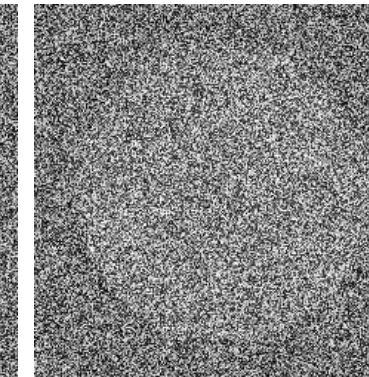
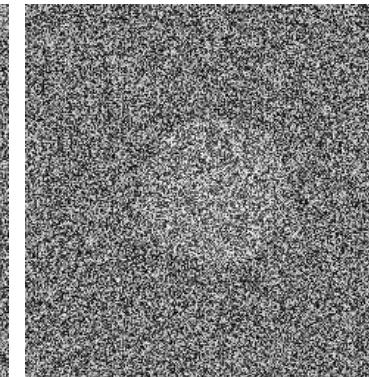
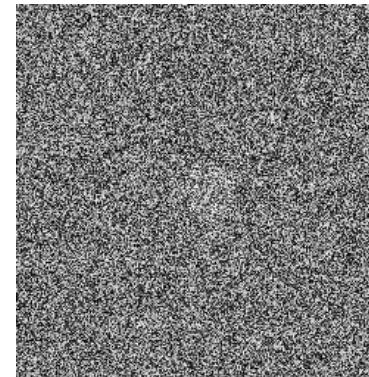
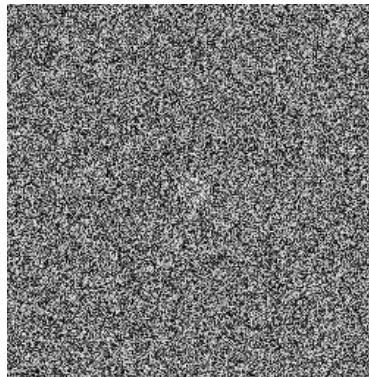
$$SNR_{\max}(f) = \frac{f_{\max}}{\sigma}$$

$$SNR_{avg} = \frac{\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n)}{\sigma}$$

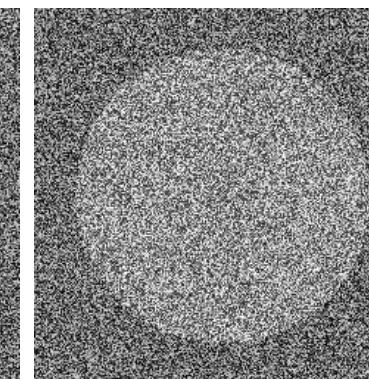
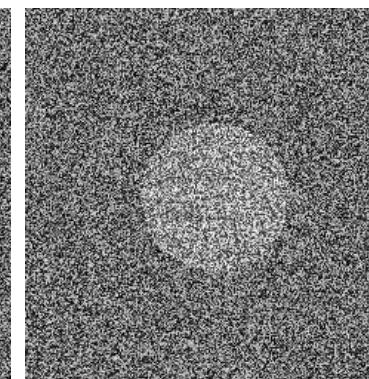
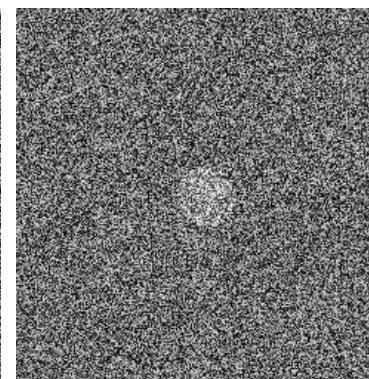
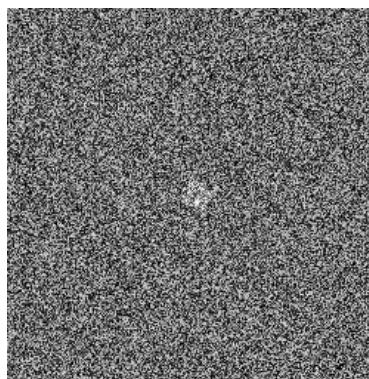


Was taugt das SNR

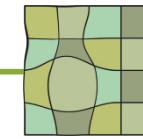
- Signalabstand muss für das erkennende Objekt bestimmt werden.
- Wahrnehmung hängt von Kontrast und Größe eines Objekts ab



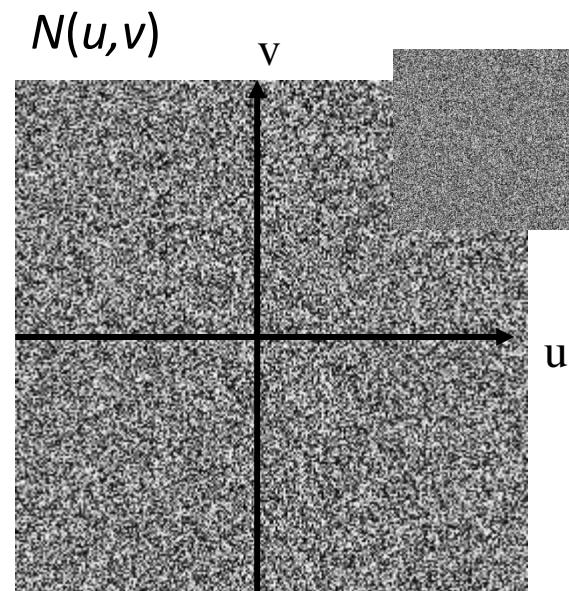
$\text{SNR}_{\max} = 0.3$



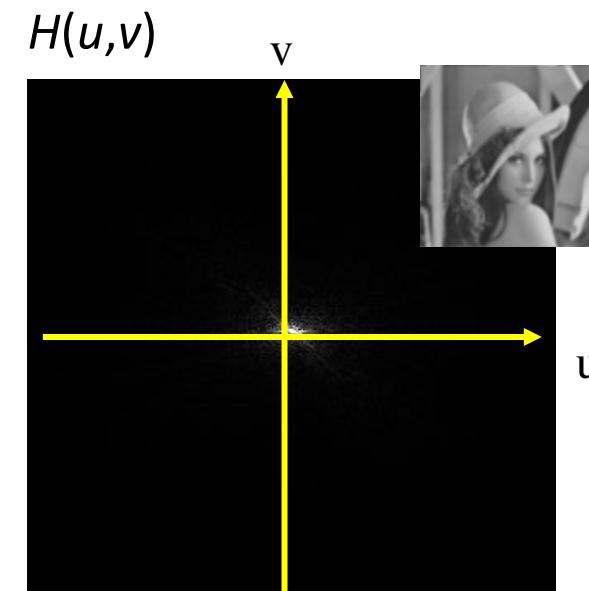
$\text{SNR}_{\max} = 0.6$



„Weißes Rauschen“+Signal

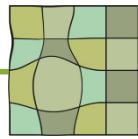


+

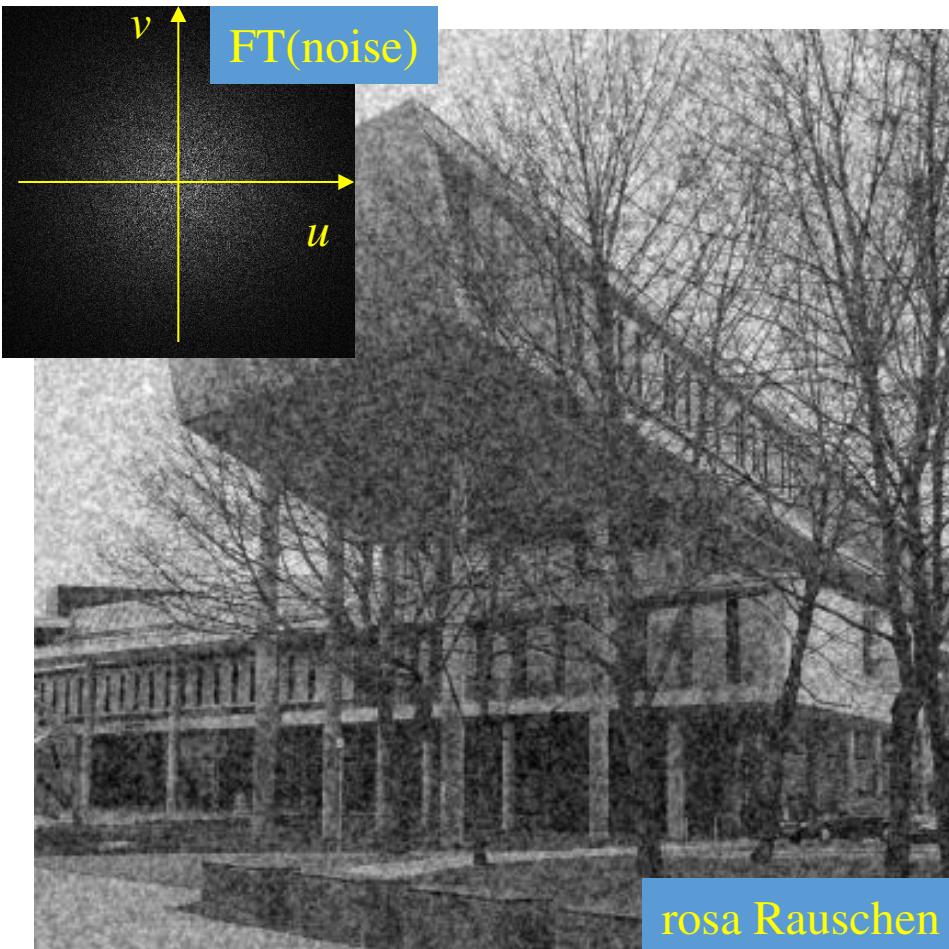
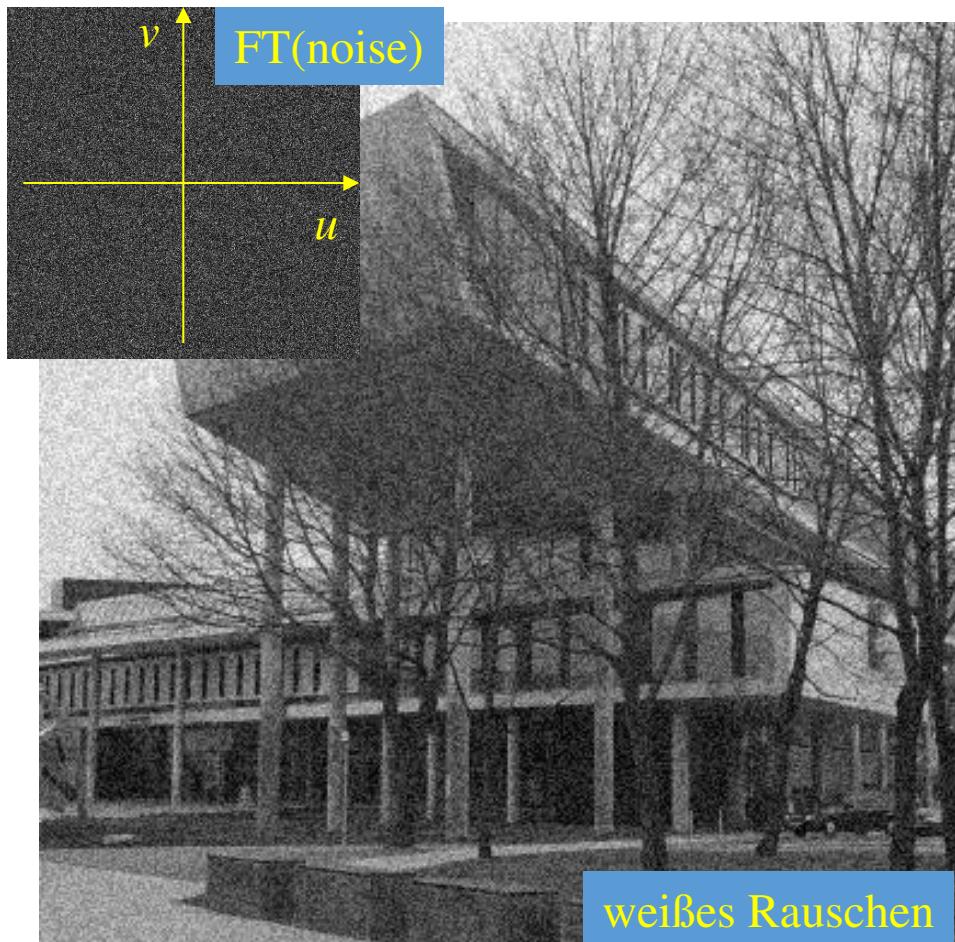


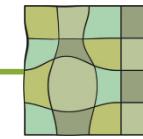
Weißes Rauschen ist
auch im
Frequenzraum
gleichverteilt.

Amplitude des Signals
fällt rasch mit
steigender Frequenz

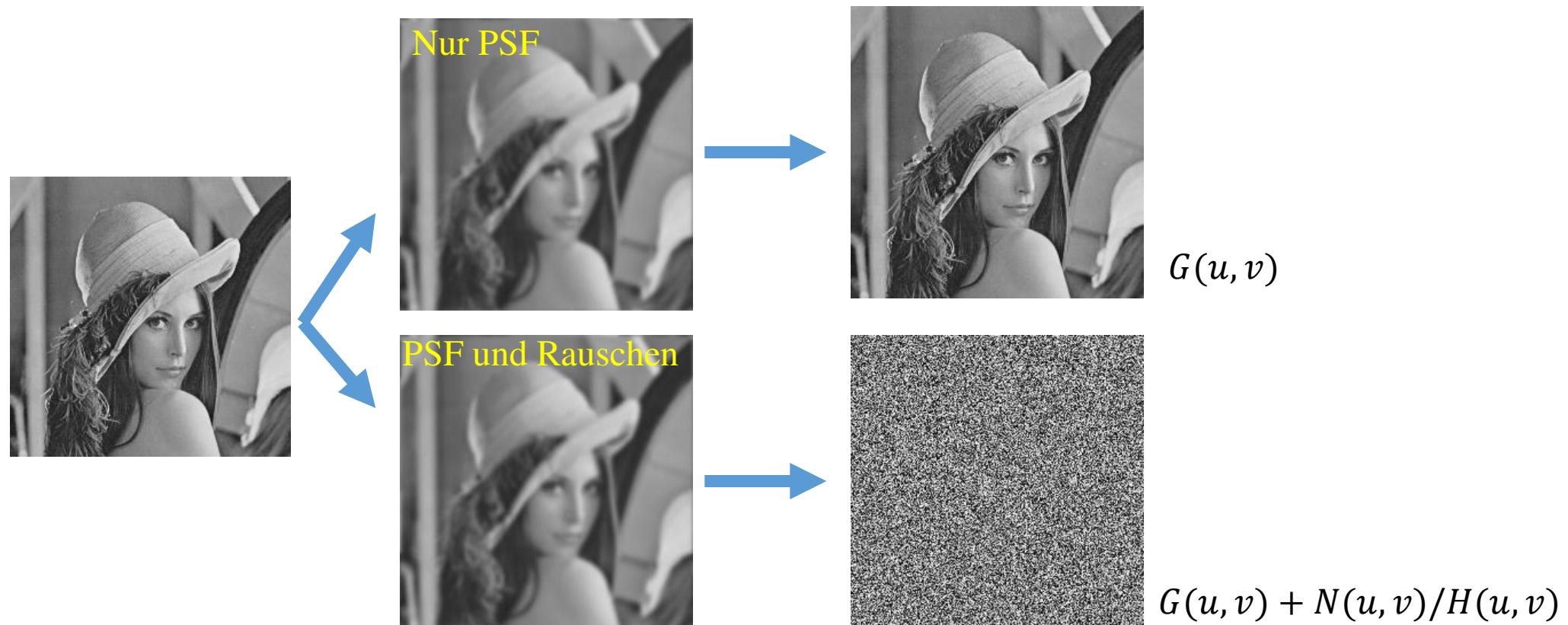


Weißes vs. „farbiges“ Rauschen

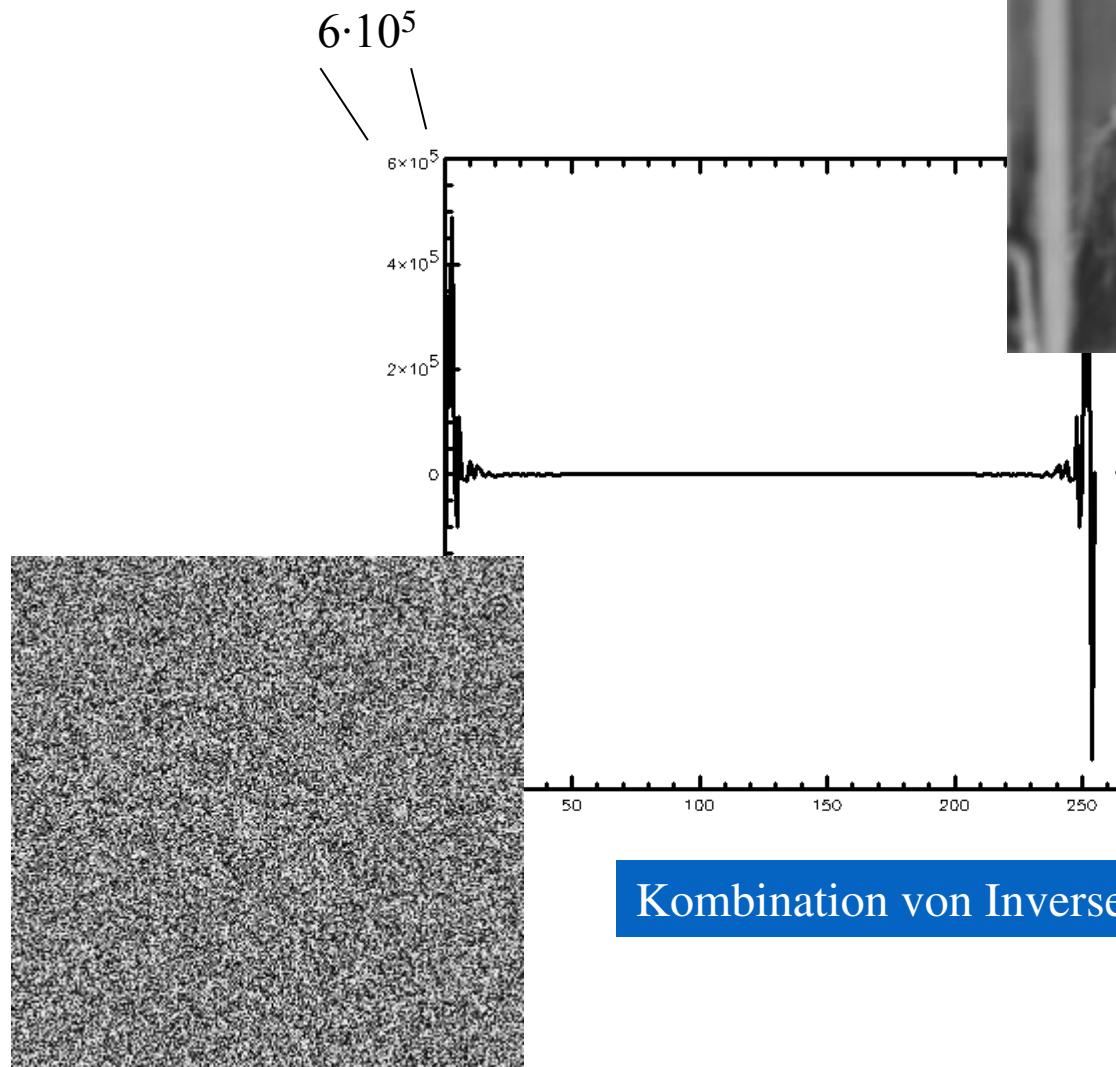




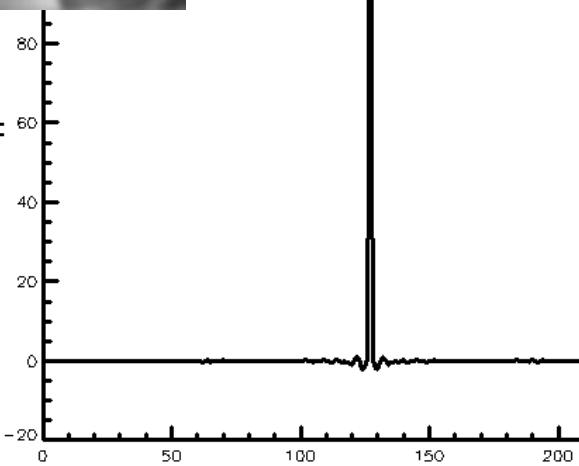
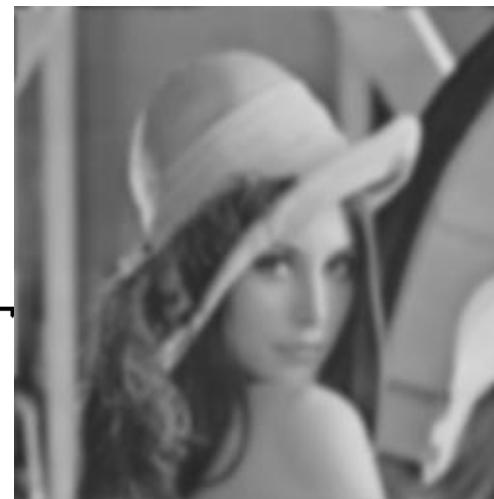
Inverse Filterung: verrauscht vs. unverrauscht



Lösungsansatz



Kombination von Inverser Filterung und Tiefpassfilter:



Wiener Filter

Optimale Rekonstruktion des Erwartungswerts der Bildfunktion

$$H(u,v)^{-1} \frac{|H(u,v)|^2}{|H(u,v)|^2 + \gamma \left(\frac{S_\eta(u,v)}{S_f(u,v)} \right)}$$

mit $S_\eta(u,v)$ - Spektrum des Rauschen

$S_f(u,v)$ - Spektrum des ungestörten und unverrauschten Bildes

γ - Steuerungsparameter für die Dämpfung

($\gamma = 1$: Wiener Filter, $\gamma \neq 1$: parametrisches Wiener Filter)

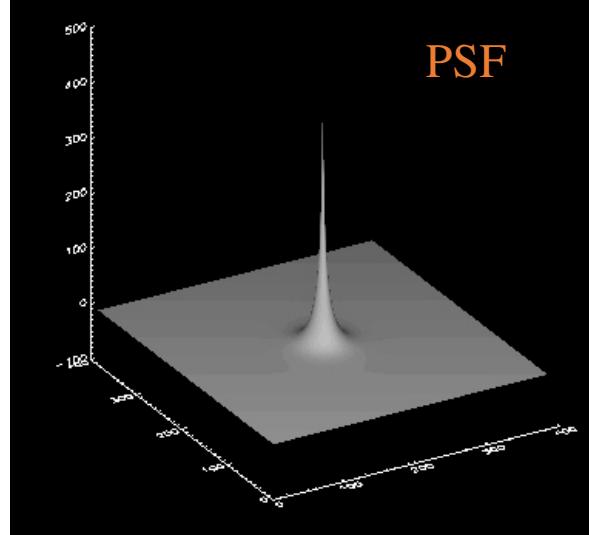
Für $S_\eta(u,v) = 0$ wird aus dem Filter ein Inverses Filter $H(u,v)^{-1}$.

Heuristische Variante:

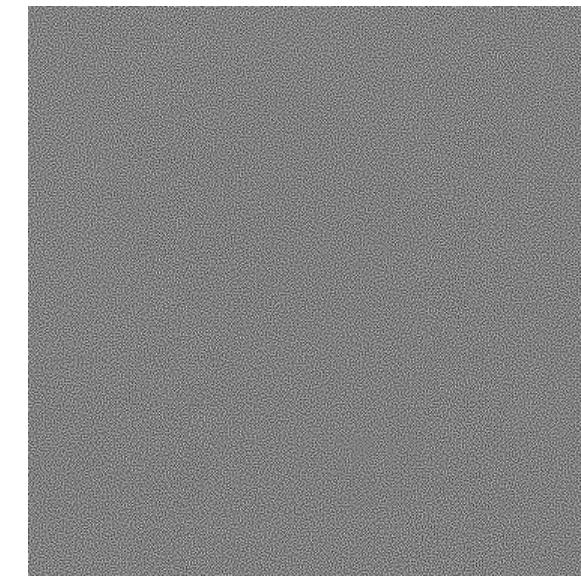
$$H(u,v)^{-1} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K}$$

(falls Spektren von Rauschen und ungestörtem Bildes unbekannt sind)

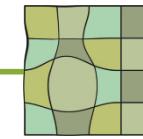
Beispiel



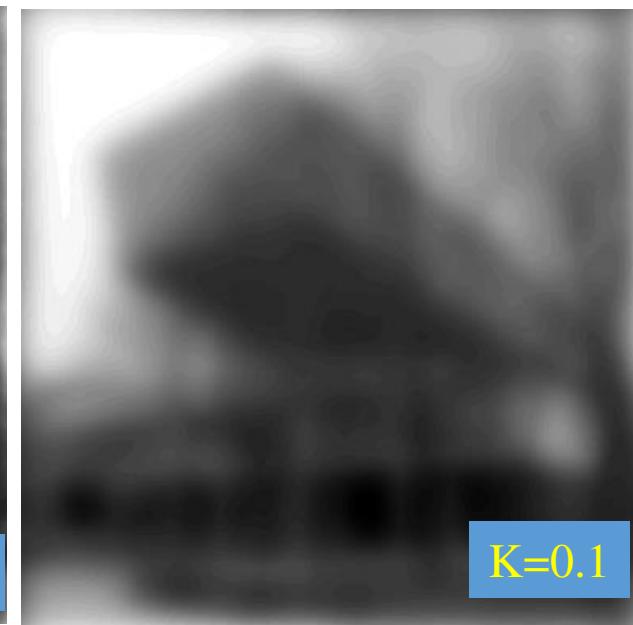
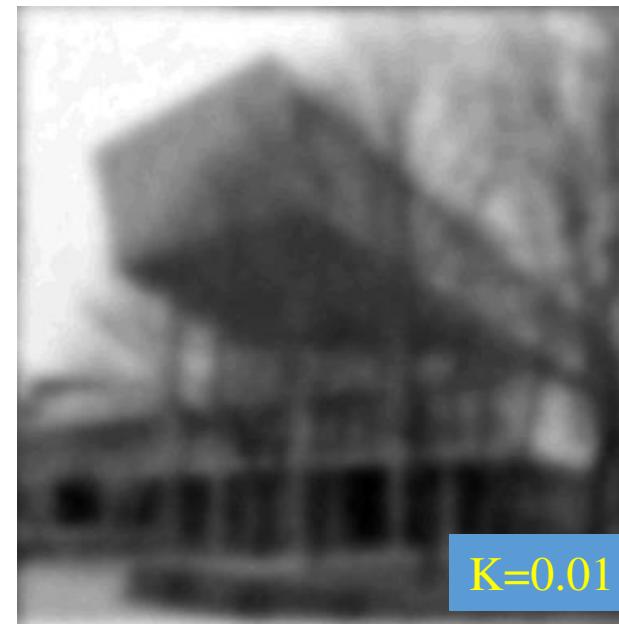
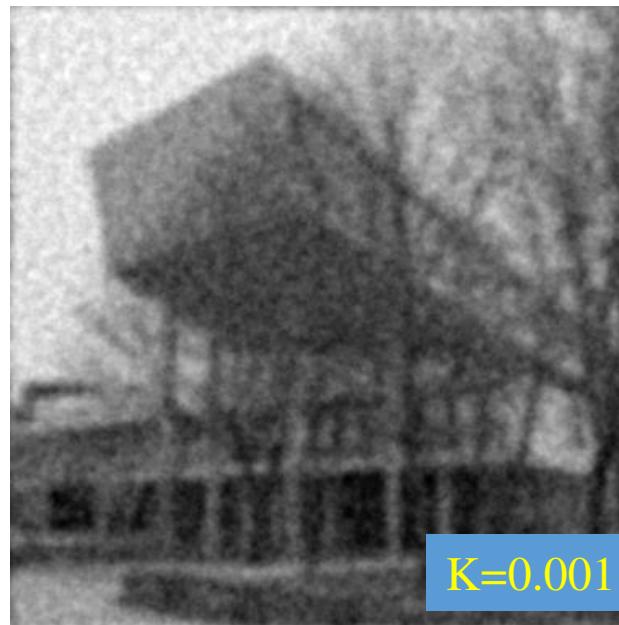
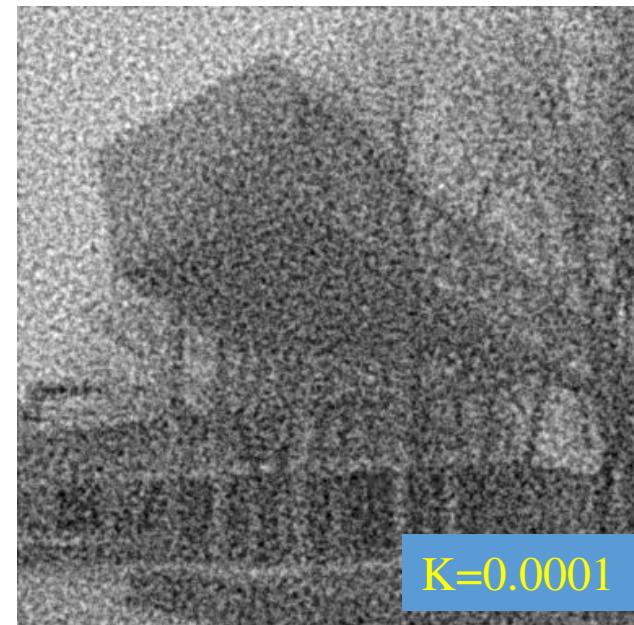
gestörtes Bild



durch Inverse Filterung „restauriert“



Heuristisches Wiener Filter



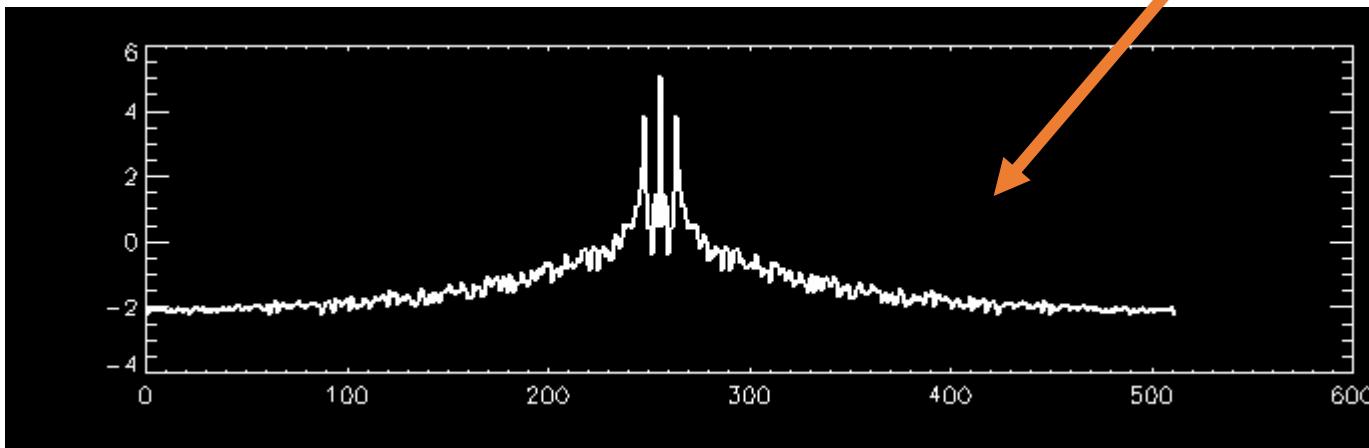
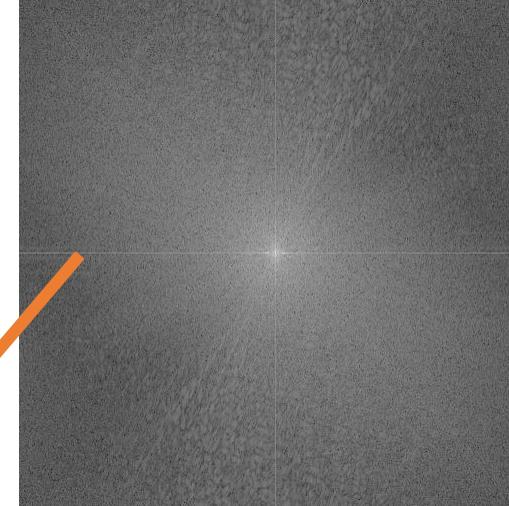
Was sollten Sie heute gelernt haben?

- Lineares Modell der Bildrestauration
- Point Spread Function und dessen Berechnung
- Kanten und Gradienten
- Rauschen: Repräsentation, Messung, Auswirkungen
- Inverse Filterung, Wiener Filter

Famous Last Question



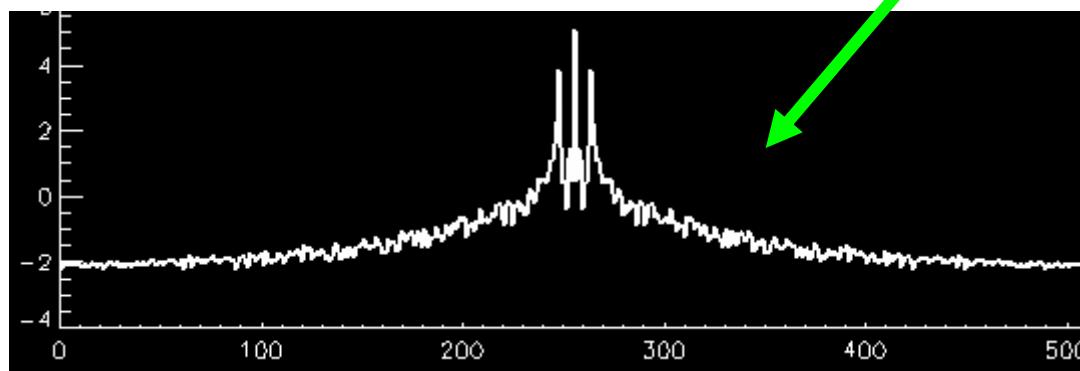
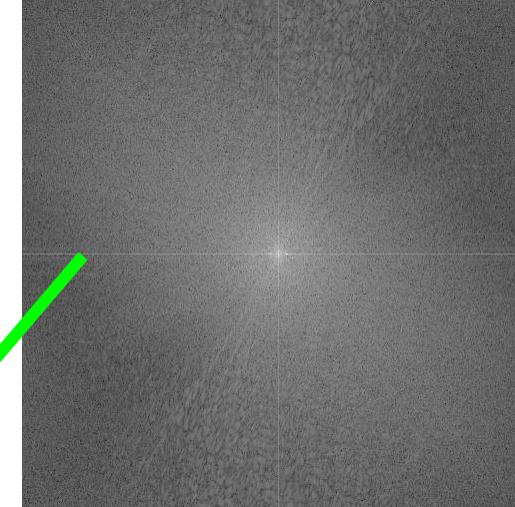
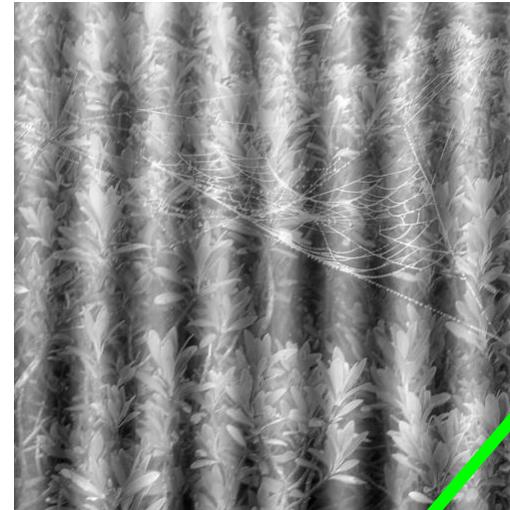
Unbekannte Störung



Was tun??

Famous Last Question

unbekannte
überlagernde Störung



Was tun??

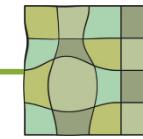
Bildkompression

- Bilddaten sind ein Mittel, um Information zu vermitteln.
- Information ist in der Regel redundant kodiert.
- Reduktion von redundanter Daten
 - Codierungsredundanz (Anzahl der Bits zu Grauwertkodierung).
 - Interpixelredundanz (Anzahl der Datenpunkte pro Pixel).
 - Psychovisuelle Redundanz (zur Erkennung benötigte Bildinformation)
- Kompressionsrate: $C = n_1/n_2$
 - n_1 und n_2 sind die Anzahl der Informationseinheiten, um dieselbe Information zu kodieren.



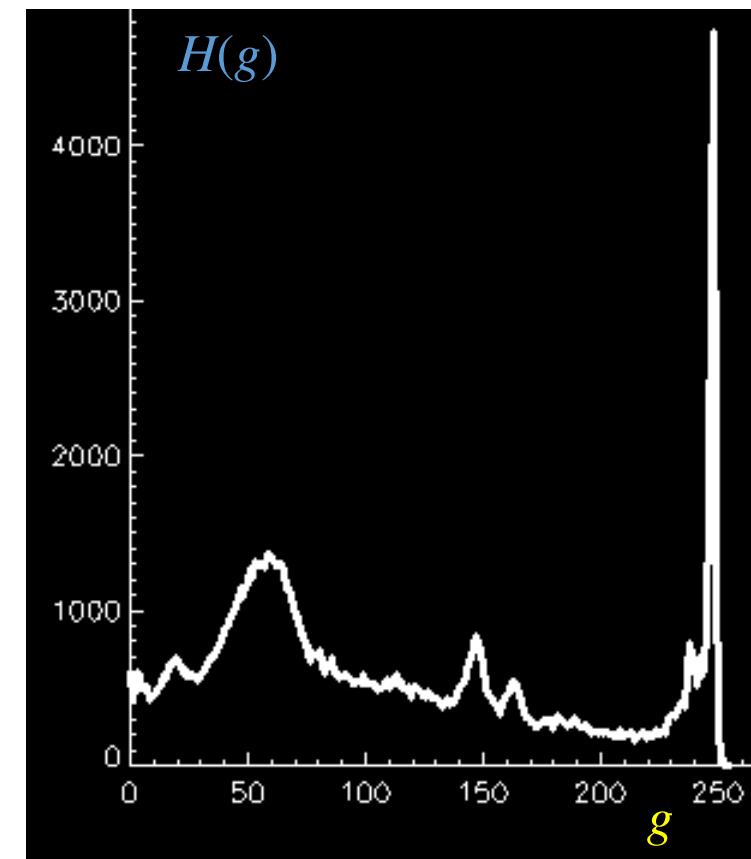
Codierungsredundanz

- Anzahl der verfügbaren Codes ist größer als die der benötigten Codes.
- Beispiel (jedes Pixel = ein Byte)
 - nur Grauwerte 1 bis 100 sind im Bild vorhanden.
 - 90% der Pixel haben den Grauwert 100.
- Lösung: 1-Bit-Code für 100, 9-Bit-Code für alle anderen
 - (100): 0, (1): 100000001, (2): 100000010, (3): 100000011, (4): 100000100, ...
 - (100) (100) (100) (17) (15) (100)...: 0001000010011000011110...
 - Kompression: $8/(0.9*1+0.1*9)=8/1.8\approx4.4$
- Ziele
 - Berechnung der Redundanz (Berechnung des Informationsgehalts)
 - Reduzierung der Codelänge.



Histogramm

Häufigkeit $H(g)$ der Grauwerte $g=\{0,1,\dots,N-1\}$ in einem Bild.

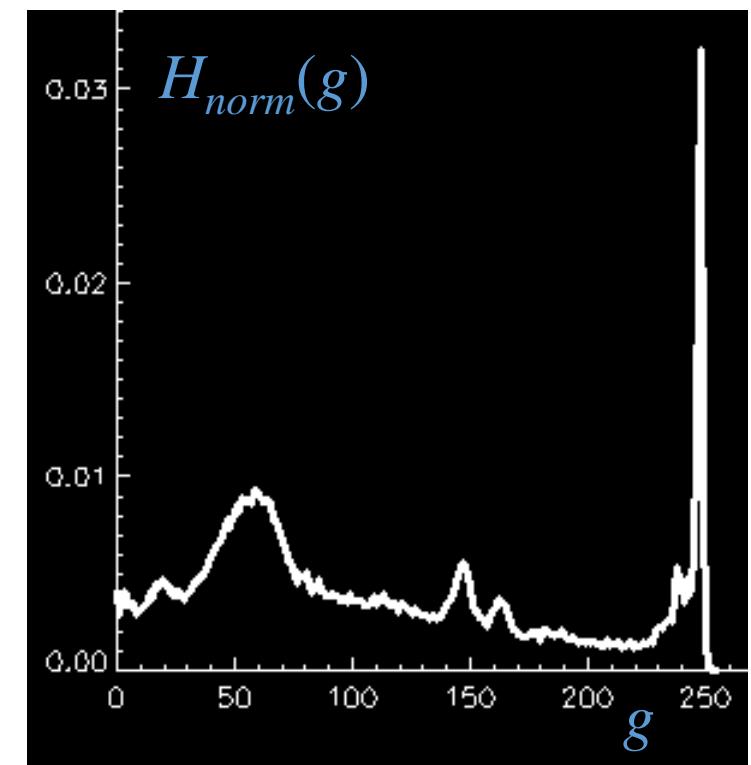


Normiertes Histogramm

- Normierung nach Anzahl der Pixel eines Bildes (Größe $M \times N$):

$$H_{norm}(g) = H(g) / (M \cdot N)$$

- Ein normiertes Histogramm gibt für jeden Grauwert g die Wahrscheinlichkeit an, dass ein beliebiges Pixel diesen Grauwert hat.



Informationsgehalt

Messbare Einheit von Information mit intuitiver Bedeutung.

Ein erster Ansatz:

Informationsgehalt $IG(E)$ eines Grauwerts E ist umso höher, je größer die Gesamtanzahl M der verwendeten Grauwerte ist:

- $IG_M(E) = M$.
- Informationsgehalt ist unabhängig davon, welcher Grauwert aus der Liste $E = \{E_0, E_1, \dots, E_{M-1}\}$ übermittelt wurde.
- Informationseinheiten $I_M(E) = \text{Anzahl der benötigten } n\text{-wertigen Symbole, für die Speicherung des Informationsgehalts,}$ also $I_M(E) = \log_n IG_M(E)$.
- Beispiel:
 - Anzahl der Grauwerte: 256
 - Informationsgehalt jedes Grauwerts: 256
 - Symbol: Bit (2-wertig)
 - Benötigte Informationseinheiten: $\log_2 256 = 8$

Informationsgehalt

- **Nachteil:** Informationsgehalt eines häufig vorkommenden Grauwerts ist genauso groß wie die eines selten vorkommenden Werts.
- Informationsgehalt $IG(E)$ eines Pixelwerts E unter *Berücksichtigung der Häufigkeit* von E :
 - Umgekehrt proportional zur Wahrscheinlichkeit $P(E)$ des Eintreffens $IG(E) = 1/P(E)$.
 - Anzahl der benötigten Informationseinheiten ist dann $I(E) = \log_n 1/P(E) = -\log_n P(E)$
- Zur Repräsentation der Information $IG(E)$ werden $I(E)$ Informationseinheiten benötigt.
- Beispiel für Schwarzweißbilder mit gleicher Anzahl schwarzer und weißer Pixel:
 - Wahrscheinlichkeit für Eintreffen von E ist 0.5
 - Informationsgehalt ist dann $I(E) = -\log_2 0.5 = \log_2 2 = 1$.

Information in einer Pixelfolge

- Grauwertbereich $\{g_0, g_1, \dots, g_{N-1}\}$ mit Wahrscheinlichkeiten des Auftretens $\{P(g_0), \dots, P(g_{N-1})\}$
- Information einer Pixelfolge der Länge k – Wahrscheinlichkeit des Auftretens gewichtet mit der Anzahl der benötigten Informationseinheiten:

$$-k \cdot P(g_0) \cdot \log_2 P(g_0) - k \cdot P(g_1) \cdot \log_2 P(g_1) - k \cdot P(g_2) \cdot \log_2 P(g_2) \dots = -k \sum_{i=0}^{N-1} P(g_i) \cdot \log_2 P(g_i)$$

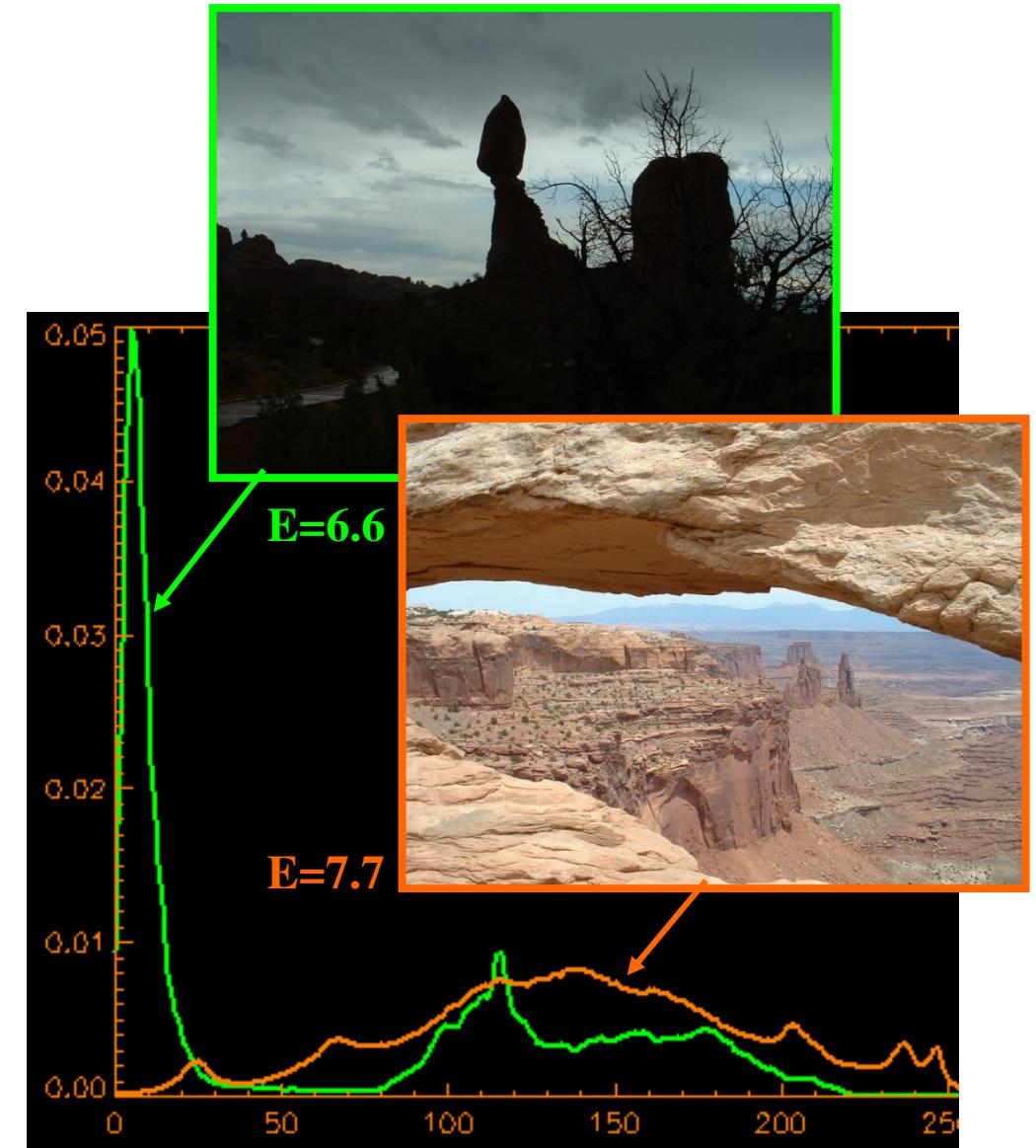
- Durchschnittlicher Informationsgehalt in Informationseinheiten = *Entropie*:

$$\text{Entropie}(P) = - \sum_{i=0}^{N-1} P(g_i) \cdot \log_2 P(g_i)$$

- Das normierte Histogramm kann als Schätzung für P verwendet werden.

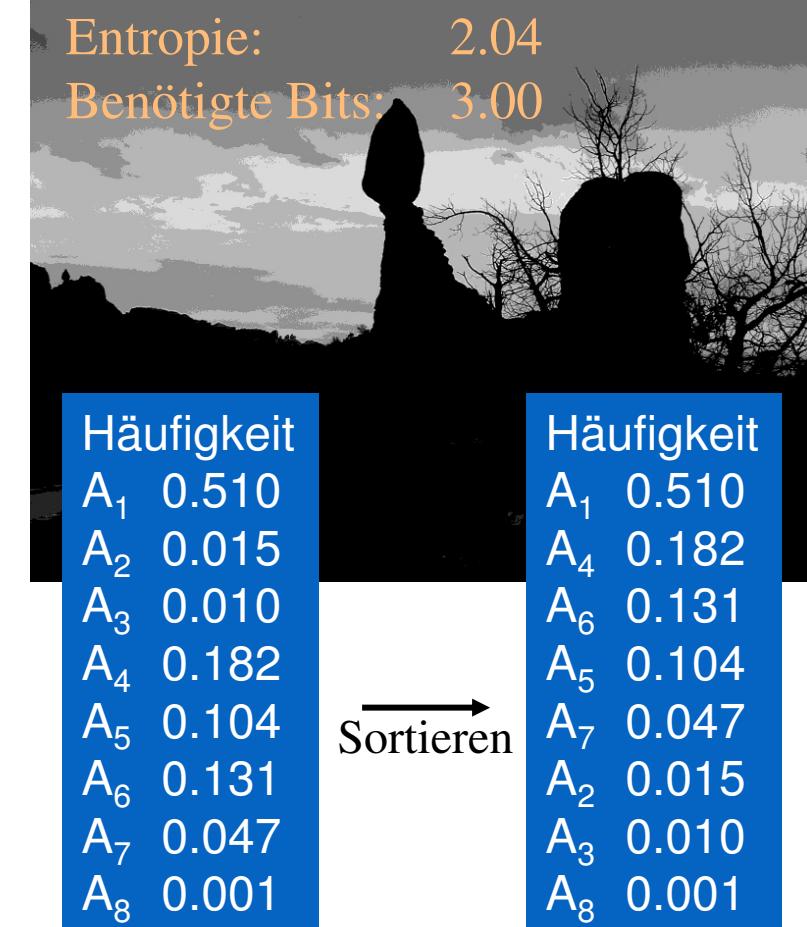
Codierungs-Redundanz

- Konstante Code-Längen sind nur bei gleichverteilten Grauerthäufigkeiten optimal.
- Code-Länge sollte an Häufigkeit angepasst werden.
- Maximale erreichbarer Kompressionsfaktor:
#Bits/Entropie

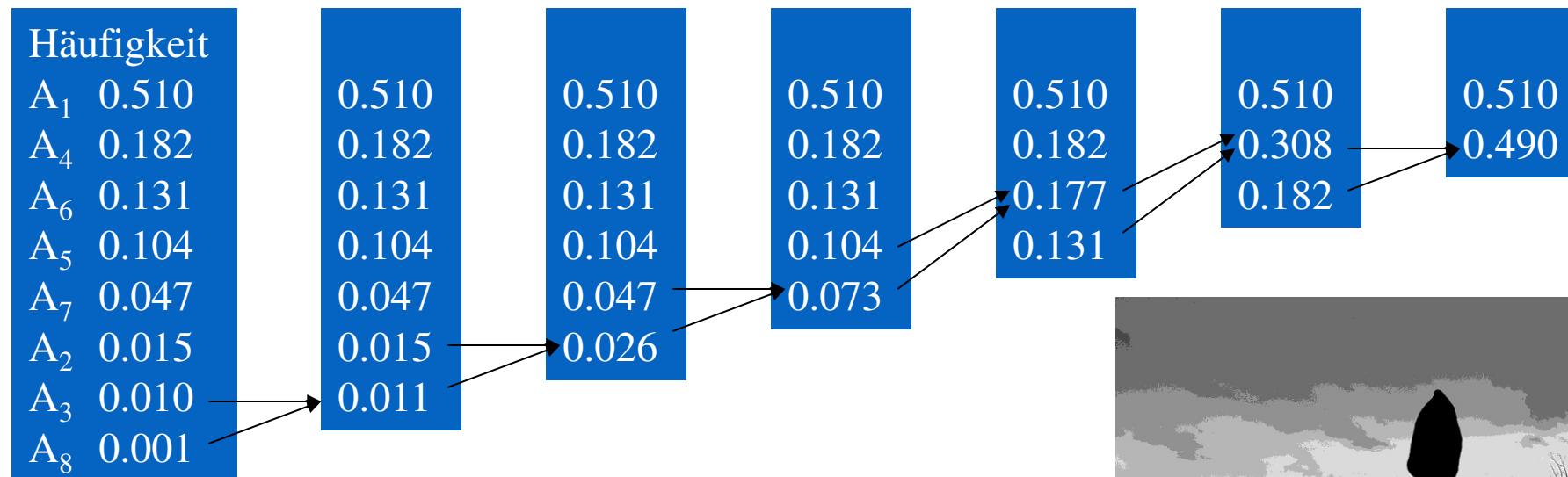


Huffman Kodierung

- Berechne Häufigkeitsverteilung (normiertes Histogramm)
- Eingangssymbole nach Häufigkeit ordnen
- Die zwei seltensten Symbole zu einem kombinierten Symbol zusammensetzen.
- Den Prozess solange fortsetzen, bis nur zwei Symbole existieren.

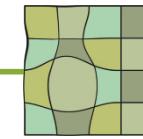


Reduktionsschritt



Nächster Schritt:
Code-Verteilung nach Häufigkeit





Code-Zuteilung

Häufigkeit	A ₁ 0.510	A ₄ 0.182	A ₆ 0.131	A ₅ 0.104	A ₇ 0.047	A ₂ 0.015	A ₃ 0.010	A ₈ 0.001
	1 00	00	010	0111	01101	011001	0110001	0110000
	0.510	0.182	0.131	0.104	0.047	0.015	0.010	0.001

0.510 1	0.182 00	0.131 010	0.104 0111	0.047 01101	0.015 011001	0.011 011000	0.026 01100	0.073 0110
0.510	0.182	0.131	0.104	0.047	0.015	0.011	0.026	0.073
1	00	00	0111	01101	011001	0110001	0110000	0110

0.510 1	0.182 00	0.131 010	0.104 0111	0.047 01101	0.026 01100	0.073 0110	0.047 01101	0.026 01100
0.510	0.182	0.131	0.104	0.047	0.026	0.073	0.047	0.026
1	00	00	0111	01101	01100	0110	01101	01100

0.510 1	0.182 00	0.177 011	0.131 010	0.073 0110	0.026 01100	0.047 01101	0.026 01100	0.073 0110
0.510	0.182	0.177	0.131	0.073	0.026	0.047	0.026	0.073
1	00	011	010	0110	01100	01101	01100	0110

0.510 1	0.308 01	0.490 0
0.510	0.308	0.490
1	01	0

Code-Verteilung nach Häufigkeit

Benötigter Speicherplatz

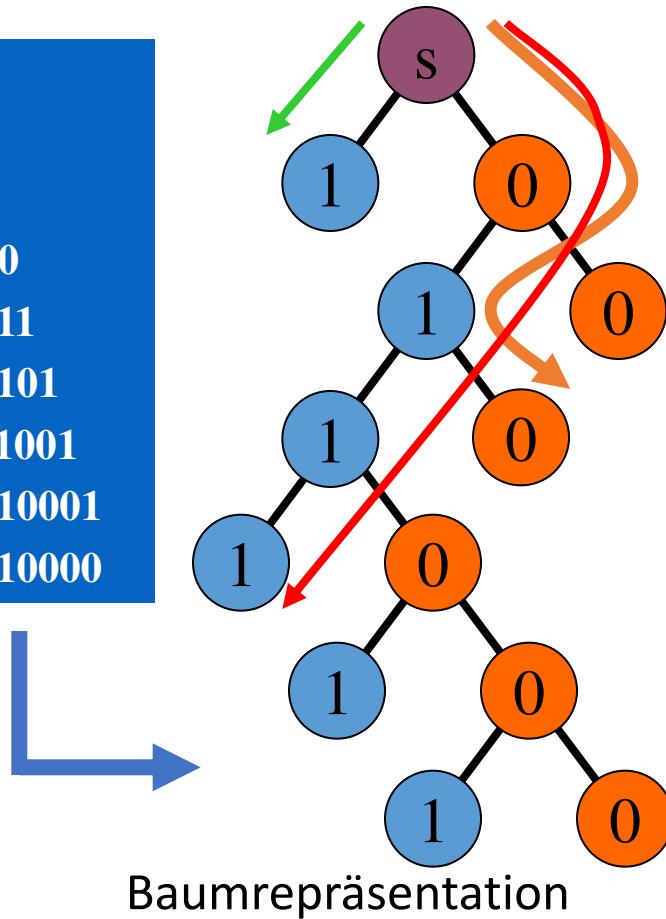
$$1 \cdot 0.510 + 2 \cdot 0.182 + 3 \cdot 0.131 + 4 \cdot 0.104 + 5 \cdot 0.047 + 6 \cdot 0.015 + 7 \cdot 0.010 + 7 \cdot 0.001 = 2.079$$

Bei gleicher Code-Länge: = 3.000

Entropie: = 2.041

Dekodierung

Häufigkeit
A ₁ 0.510 1
A ₄ 0.182 00
A ₆ 0.131 010
A ₅ 0.104 0111
A ₇ 0.047 01101
A ₂ 0.015 011001
A ₃ 0.010 0110001
A ₈ 0.001 0110000



1 0 1 0 | 0 1 1 1 | 0 1 0 1 ...

Während des Dekodierens wird der Kodierungsbaum anhand der Bitfolge abgearbeitet.

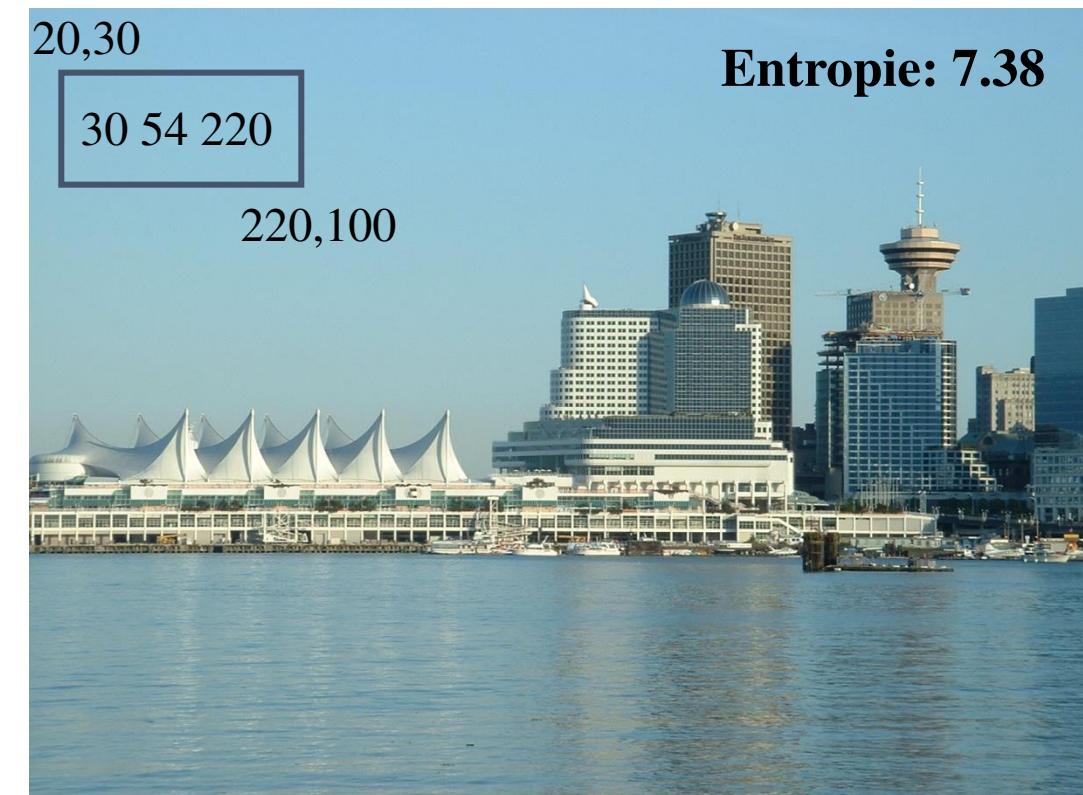
Ein Zeichen ist gefunden, wenn ein Blatt erreicht wurde.

Aufwand $O(N)$, wenn N die Anzahl der Bits ist.

Interpixel-Redundanz

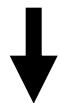
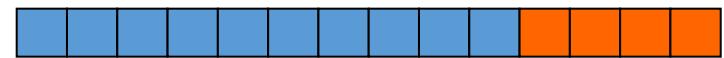
- Minderung der Kodierungsredundanz berücksichtigt die *Homogenität* innerhalb des Bilds *nicht*.
- Interpixelredundanz: Repräsentation durch
 - Blockspezifikation (z.B. [20,30,220,100])
 - Blockintensität (z.B. [30 54 220])
- Bekannteste Methode:

Run-Length-Encoding (RLE), bzw.
Lauflängenkodierung

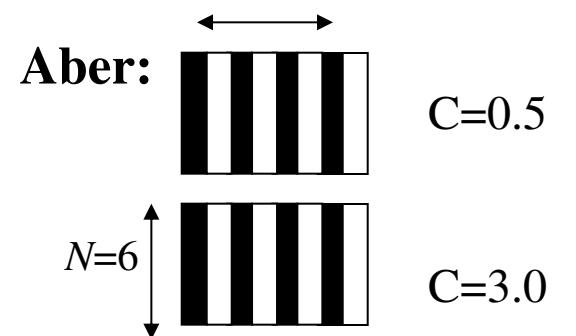


Run-Length-Encoding

- Bild wird zeilenweise codiert.
- Eine Zeile mit Grauwertfolge $g(0), g(2), \dots, g(n)$ wird zerlegt in Zeilenstücke mit gleichem Grauwert (die *runs*):
 $(g(0), \dots, g(r_1)), (g(r_1 + 1) \dots g(r_1 + 1 + r_2)), \dots$
- Die Zeilenstücke werden durch Angabe von Länge und Grauwert codiert:
 $(r_1, g(r_1)), (r_2, g(r_2)), \dots$
- Kompressionsrate für jede Zeile ist $n/(2 \cdot rd)$, wobei r_d die durchschnittliche Länge des *runs* ist.

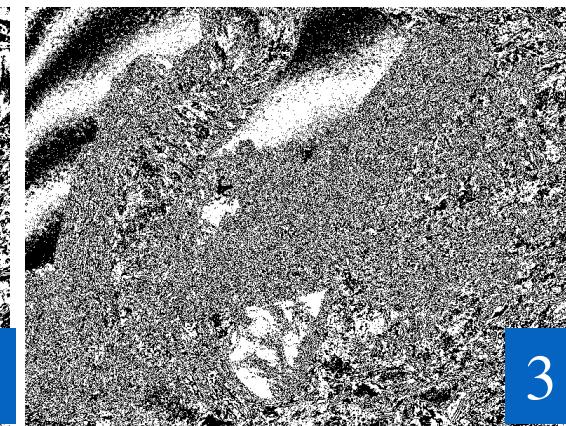
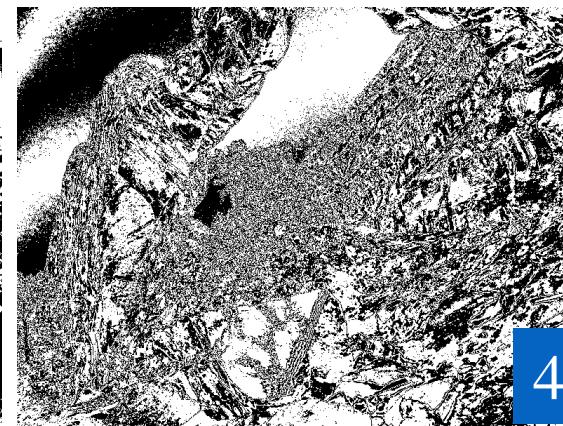
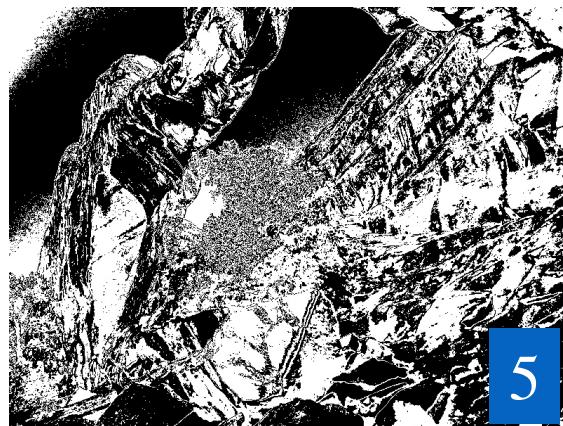


$(10, \blacksquare), (4, \blacksquare), (2, \blacksquare)$



RLE auf Bit-Ebenen

- RLE ist ineffizient bei verrauschten Bildern (sehr viele kleine Grauwertänderungen = kurze Runs).
- Lösung:
 - Zerlegung in Bit-Ebenen
 - Runs separat auf jeder Bit-Ebene
- Problem: Niedrige Bit-Ebenen



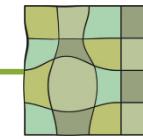
Gray-Code

- Code, bei dem sich zwei aufeinander folgenden Zahlen um höchstens 1 Bit unterscheiden.
- Generierung für Bitfolge b_1, b_2, \dots, b_n (höchstwertiges Bit hat niedrigsten Index):
 - $k=n, \dots, 2$: falls $g_{k-1}=1$, dann $g_k=1-b_k$
sonst $g_k=b_k$
 - $k=1$: $g_1=b_1$

Beispiel:

b : 000 001 010 011 100 101 110 111

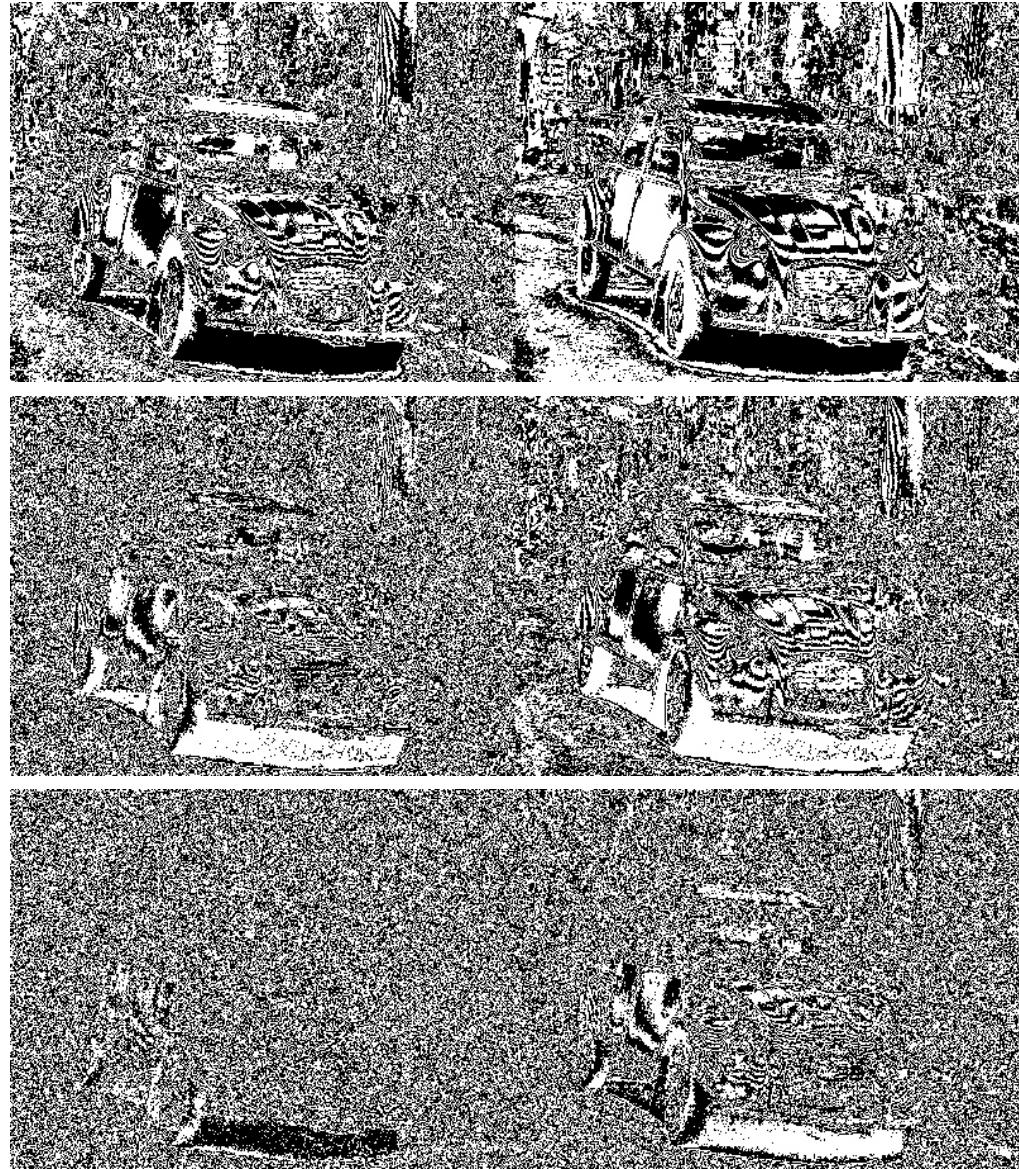
g : 000 001 011 010 110 111 101 100



Beispiel



Normal und gray-kodierte untere
drei Bit-Ebenen des Rotkanals.



Psychovisuelle Redundanz

- Nichtredundante Daten tragen redundante Information.
- Reduktion der psycho-visuellen Redundanz ist verlustbehaftet (lossy) bezogen auf die Daten.
- Erhalt des Informationsgehalts ist zielabhängig.



Reduktion der Kontrastauflösung

- Unterscheidbare Helligkeitsstufen (30-100) ist geringer, als die repräsentierbare Anzahl.
- Bildinformation ist auch bei noch kleinerer Anzahl wahrnehmbar.
- Machbandeffekt kann durch Addition von Rauschen vermieden werden.



Reduktion hochfrequenter Anteile

Transformation zerlegt in einzelne Frequenzanteile (z.B., Fouriertransformation, Diskrete Kosinustransformation, Wavelet Transformation).

Sehr viel der Bildinformation ist in den niedrigen Frequenzen kodiert.

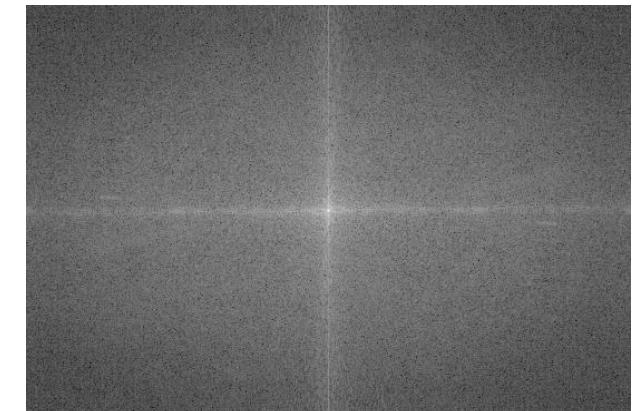
Kompression:

- Transformation
- Sortierung der Koeffizienten nach Größe
- Quantisierung, d.h. Abbildung auf ganze Zahlen:

$$q(e) = Q \cdot \text{sign}(e) \cdot (|e| - e_{\min}) / (e_{\max} - e_{\min}).$$

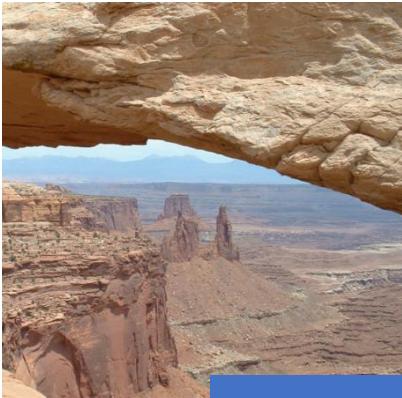
- Kompression der quantisierten Werte durch verlustfreies Verfahren.

Dekodierung durch Rücktransformation.

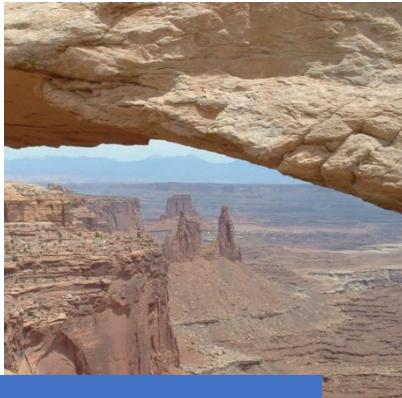


Reduktion hochfrequenter Anteile

1:1



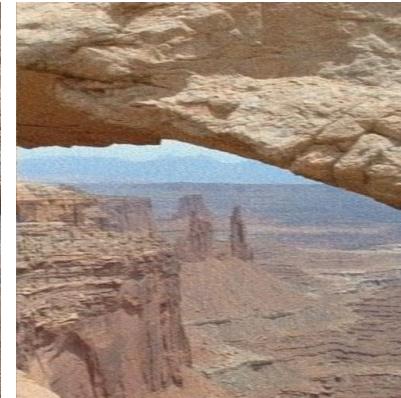
1:4



1:10



1:30



1:300



Fouriertransformation



Diskrete Kosinustransformation (DCT)

- Zerlegung in Wellen unterschiedlicher Frequenz.
 - Sehr ähnlich zur Fouriertransformation.
 - Basisfunktionen sind reell.
 - Wird vor allem bei Kompressionsverfahren verwendet (JPEG).
-
- ***Wie lassen sich genügend reellwertige Basisfunktionen finden?***

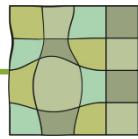
Basisfunktionen

- Basisfunktionen sind alle halben und alle vollen Frequenzen der Cosinustransformation

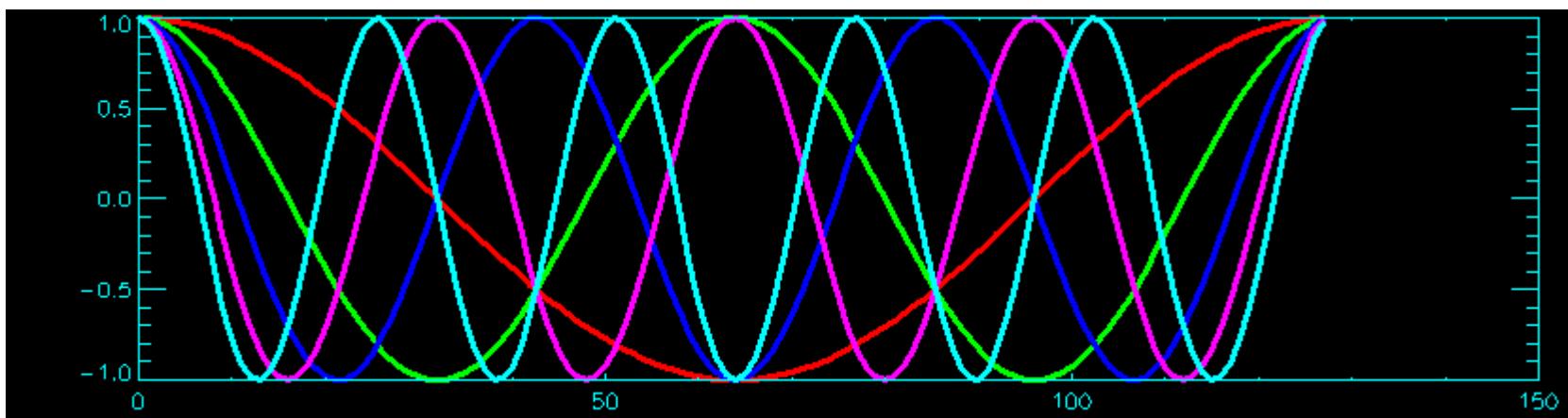
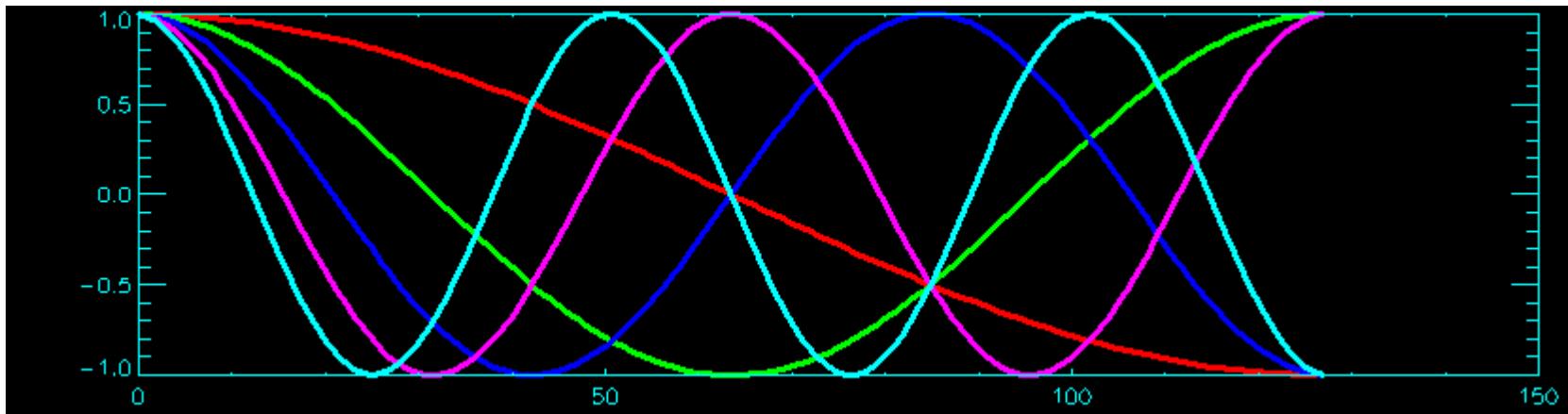
$$C(u) = \sum_{m=1}^{M-1} f(m) \cos\left(\frac{mu\pi}{M}\right)$$

- Meist (z.B. auch bei der jpeg-Transformation) werden die Transformationswerte aus den Funktionswerten zwischen den Abtastpunkten gemittelt, d.h., m wird um 0.5 verschoben

$$C(u) = \sum_{m=1}^{M-1} f(m) \cos\left(\frac{(m + 0.5)u\pi}{M}\right)$$



Reelle Basisfunktionen DCT / DFT



Basisfunktionen der 2-d DCT

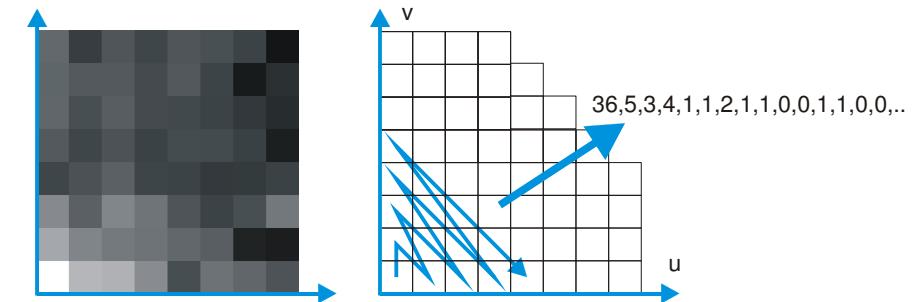
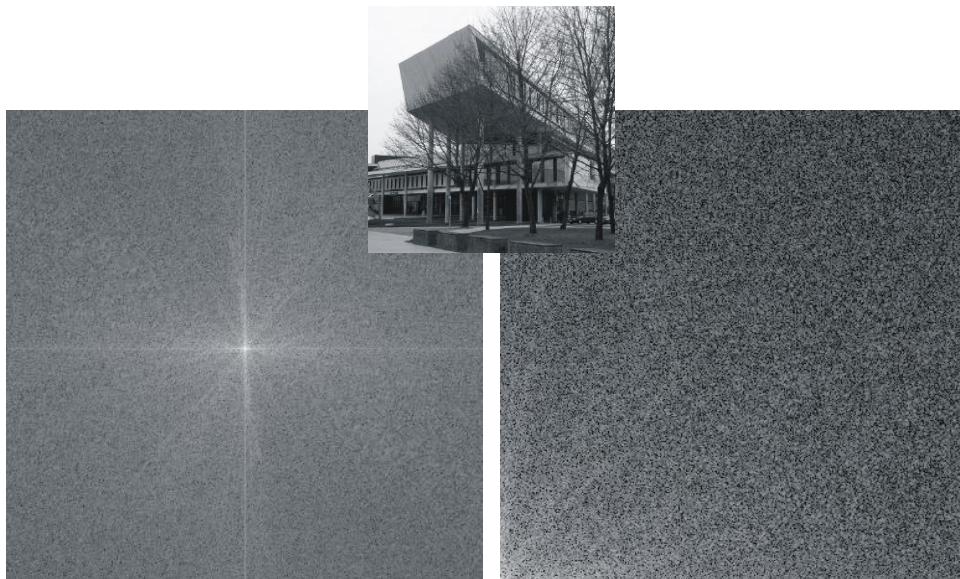
$$C(u, v) = \alpha(u)\alpha(v) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cos\left(\frac{(m+0.5)u\pi}{M}\right) \cos\left(\frac{(n+0.5)v\pi}{N}\right)$$

$$f(m, n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v) C(u, v) \cos\left(\frac{(u+0.5)m\pi}{M}\right) \cos\left(\frac{(v+0.5)n\pi}{N}\right).$$

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{M}} & , \text{für } u = 0 \\ \sqrt{\frac{2}{M}} & , \text{für } u \neq 0 \end{cases} \quad \text{und} \quad \alpha(v) = \begin{cases} \sqrt{\frac{1}{N}} & , \text{für } v = 0 \\ \sqrt{\frac{2}{N}} & , \text{für } v \neq 0 \end{cases}$$

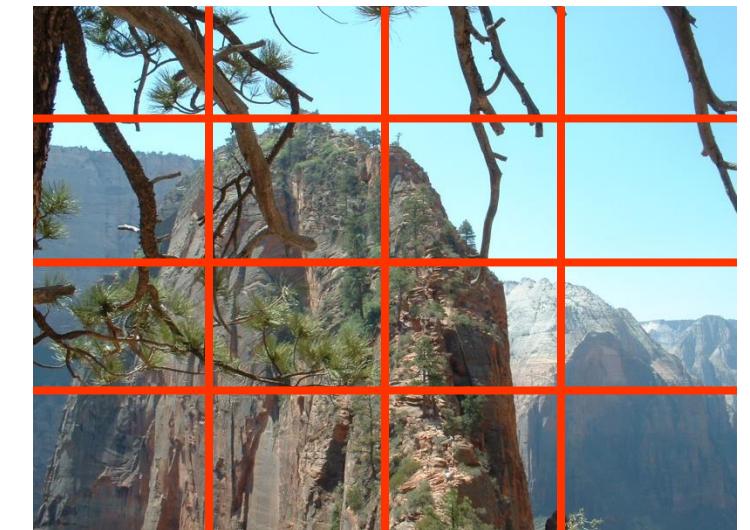
DCT vs DFT

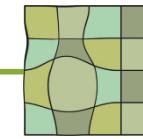
- DCT hat nur reelle Komponenten, deren Wert mit Abstand vom Ursprung abnimmt.
- Kompression durch Quantisierung:
 - Werte entlang der Diagonale auslesen,
 - Quantisieren,
 - nur bis zum letzten von Null verschiedenen Wert speichern.



Zerlegung in Teilbilder

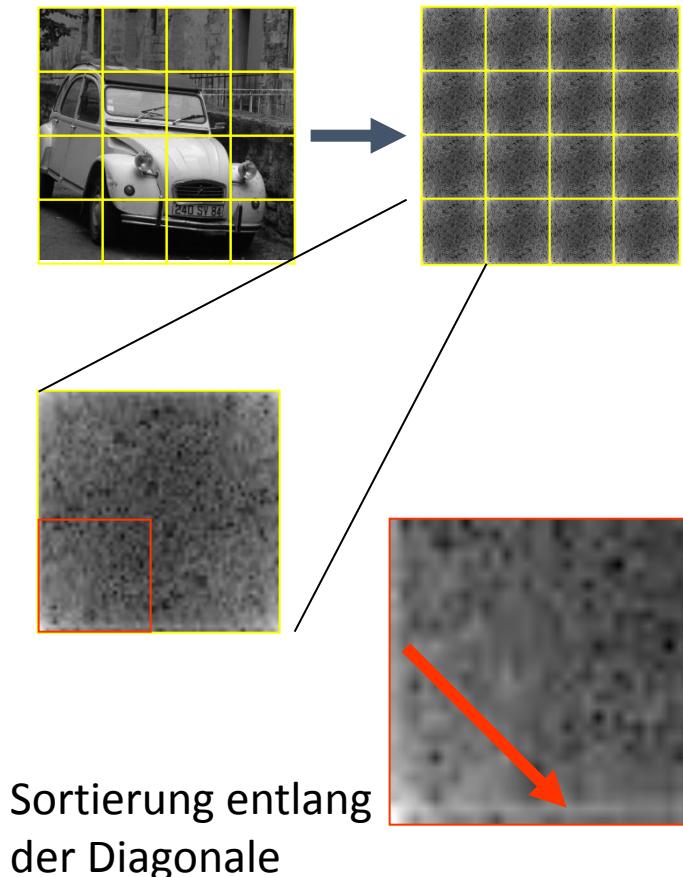
- Anzahl der Information tragenden Koeffizienten ist geringer, je geringer die Anzahl der Kanten ist.
- Zerlegung in Teilbilder
 - Anzahl der Funktionswerte im Frequenzraum entspricht der Anzahl der Pixel des Teilbilds.
 - Wahrscheinlichkeit von Kanten in Teilbildern sinkt mit deren Größe.
 - Teilbildgröße: Kompromiss zwischen durchschnittlicher Anzahl der Information tragenden Frequenzen und Gesamtanzahl der Teilbilder.





JPEG Kompression von Teilbildern

- Farbraumtransformation von RGB nach YCbCr
- Tiefpassfilterung und Unter-abtastung der Farbabweichungs-signale Cb und Cr.
- Einteilung in 8×8 -Blöcke und DCT auf dem Y-Kanal
- Quantisierung der DCT-Koeffizienten.
- Umsortierung und Entropiekodierung (Huffman).



Quantisierungseffekte

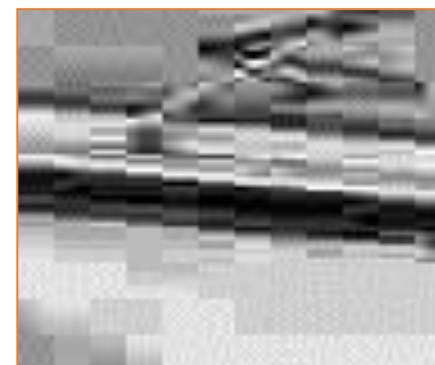


Zerlegung in 8x8-Blöcke, Quantisierung und Rücktransformation

1:30



1:45

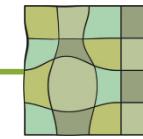


1:75



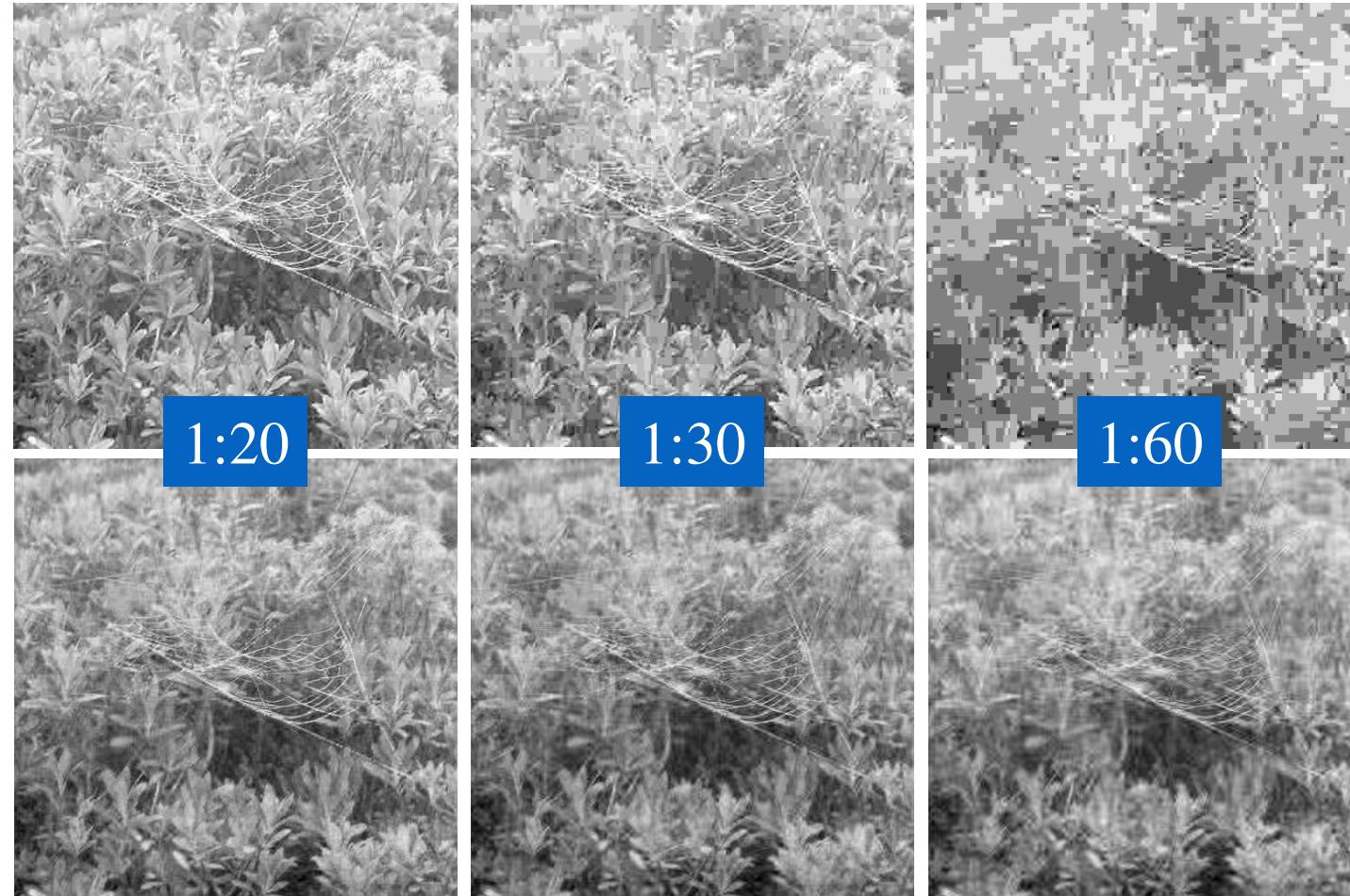
JPEG2000: Wavelet-Koeffizienten

- Wavelet-Transformation zerlegt das Bild in lokal wirkende Wellen unterschiedlicher Frequenz.
- Keine explizite Blockung notwendig.
- Kompression wird umso besser sein, umso kompakter die Wavelets in Orts- und Frequenzraum sind.
- Wavelet-Kompression durch Daubechies-Wavelets höherer Ordnung (Wavelet-Kompression ist Teil des JPEG2000-Standards).



Vergleich Wavelet vs DCT

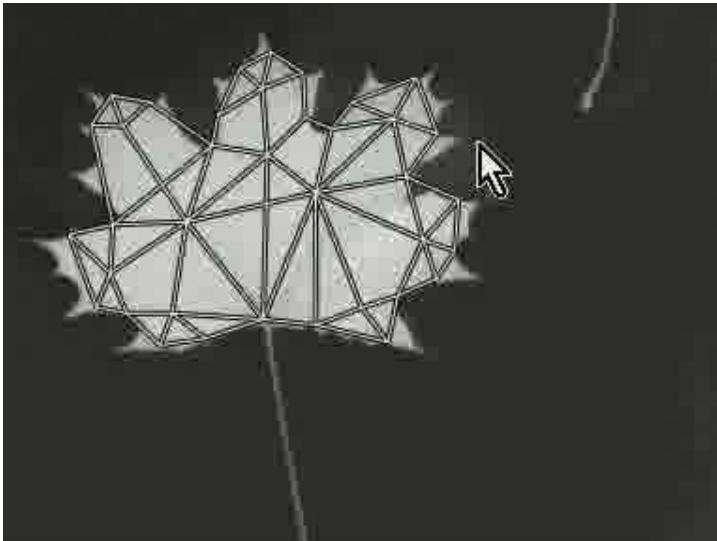
DCT-basierte
Kompression



Was sollten Sie heute gelernt haben?

- Kompression entfernt Redundanz aus Bildern
- Redundanz: Kodierungs-, Interpixel-, Psychovisuelle Redundanz.
- Kodierungs- und Interpixelredundanz ist Datenredundanz (=verlustfreie Kompression).
- Psychovisuelle Redundanz ist Informationsredundanz (=verlustbehaftete Kompression).

Famous Last Question



Wie könnte man eine Sequenz von Filmbildern besonders gut *verlustfrei* komprimieren?

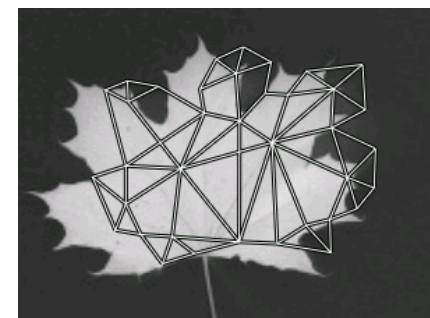


Bild 1

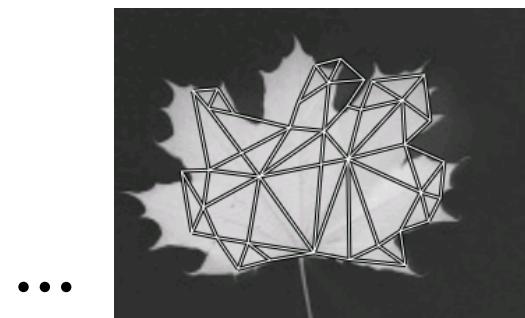


Bild 41

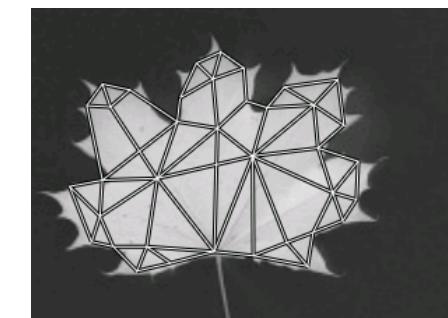
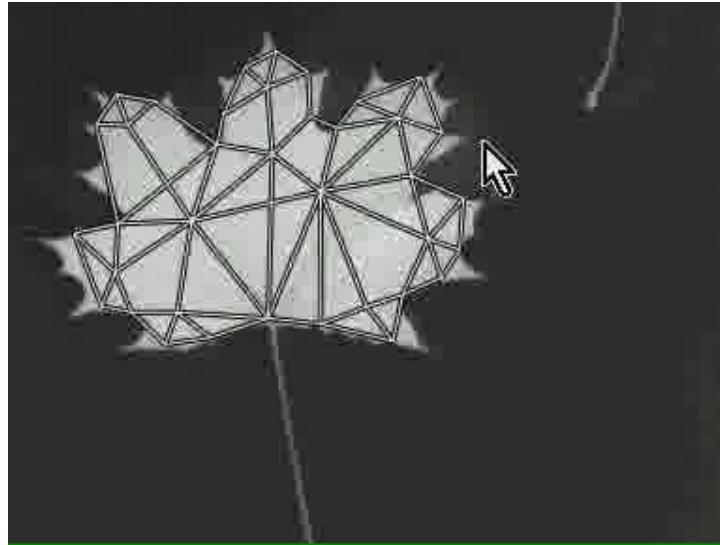


Bild 81

...

Famous Last Question



Wie könnte man eine Sequenz von Filmbildern besonders gut verlustfrei komprimieren?

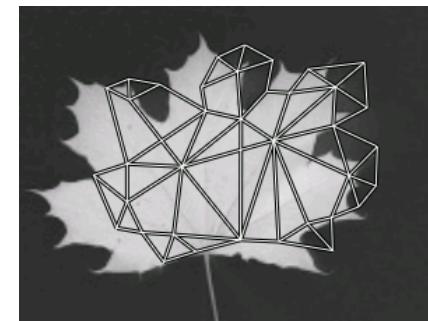


Bild 1



Bild 41



Bild 81

...

...

...

Bildverbesserung

- Verbesserung von Bildeigenschaften zur *besseren Wahrnehmbarkeit* oder zur *Vorbereitung von Analyseschritten*.
- Bildeigenschaften:
 - Signal-Rausch-Verhältnis
 - Kontrast
 - Informationsgehalt
- Punktbasierte Methoden
- Flächenbasierte Methoden



Welches Bild ist besser?
Warum?
Wie ist das messbar?



Pixelbasierte Bildverbesserung

- Abbildung der Grau-/Farbwerte unabhängig von ihrem Ort oder ihrer Zuordnung

- innerhalb der Grau- bzw. Farbwerte:

$$g_{neu} = f(g) \text{ oder } [r_{neu}, g_{neu}, b_{neu}] = [f(r), f(g), f(b)]$$

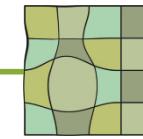
- von Grauwerten in Farbwerte (Falschfarbdarstellung):

$$[r_{neu}, g_{neu}, b_{neu}] = [f_{rot}(g), f_{grün}(g), f_{blau}(g)]$$

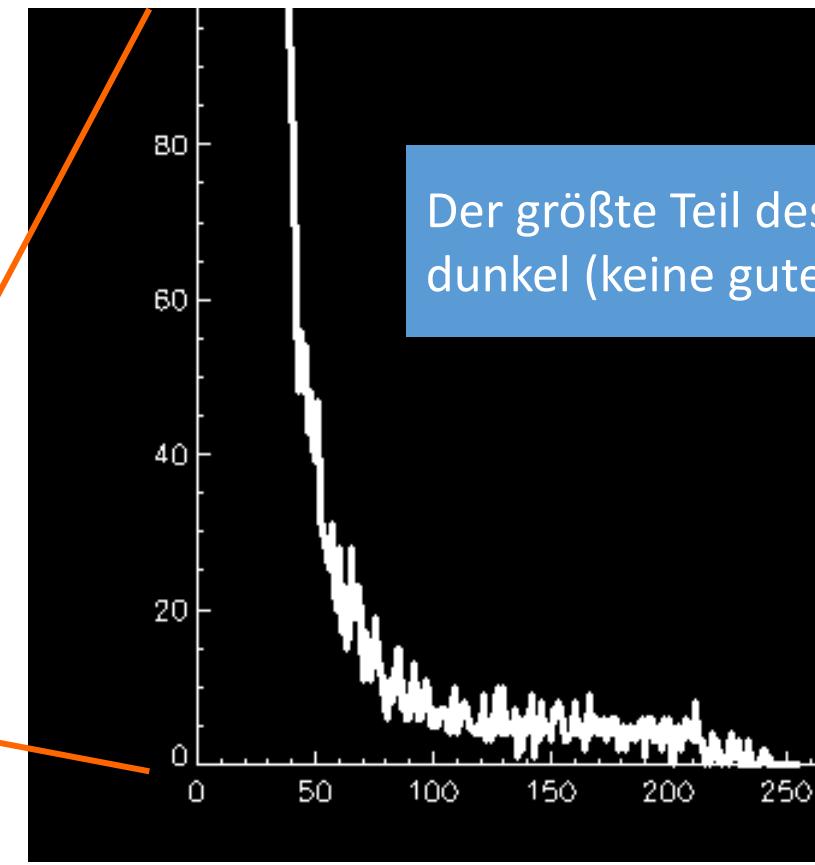
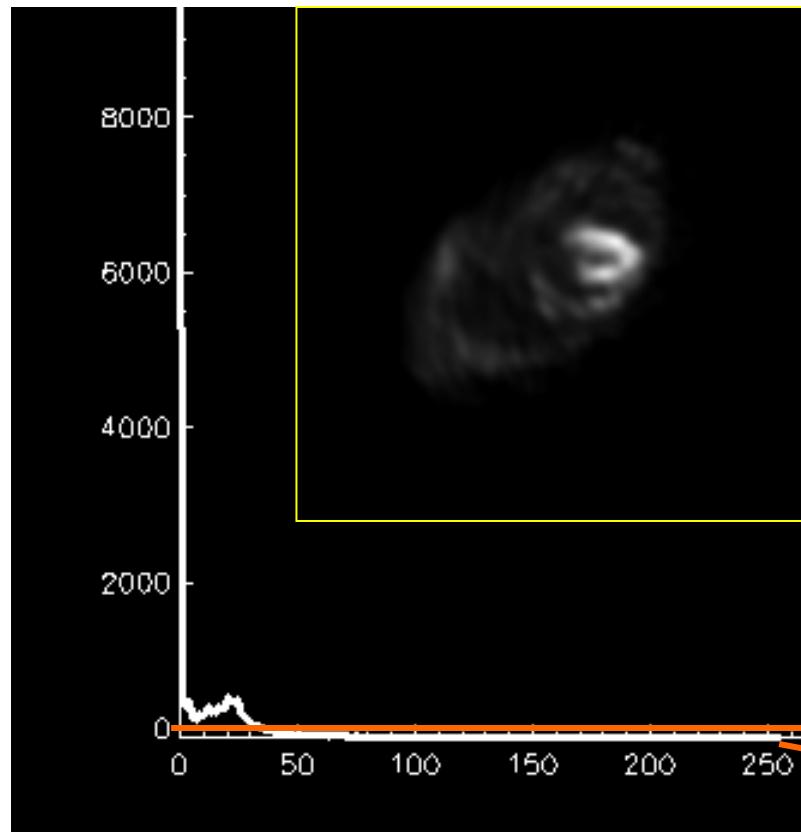
- Figure of Merit (Qualitätsmerkmal): globaler/lokaler Kontrast, Entropie

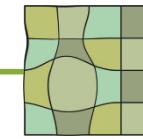
- Methoden

- Monotone Abbildung der Grauwerte
 - Nicht monotone Grauwertabbildung
 - Falschfarbdarstellung

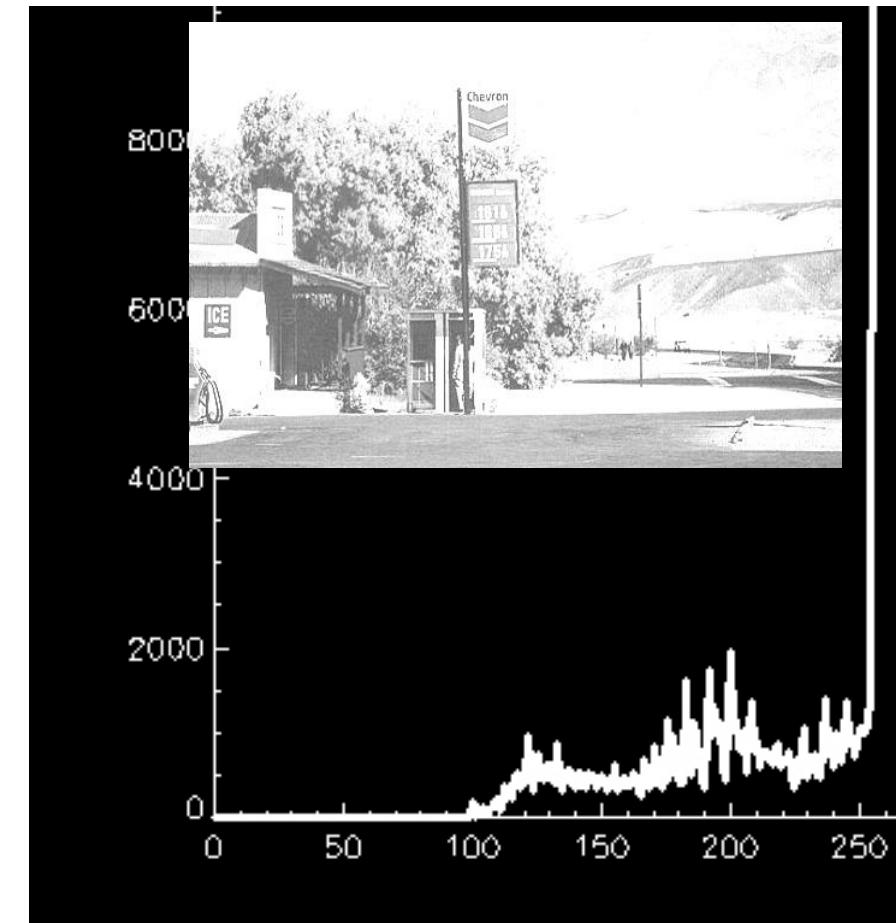
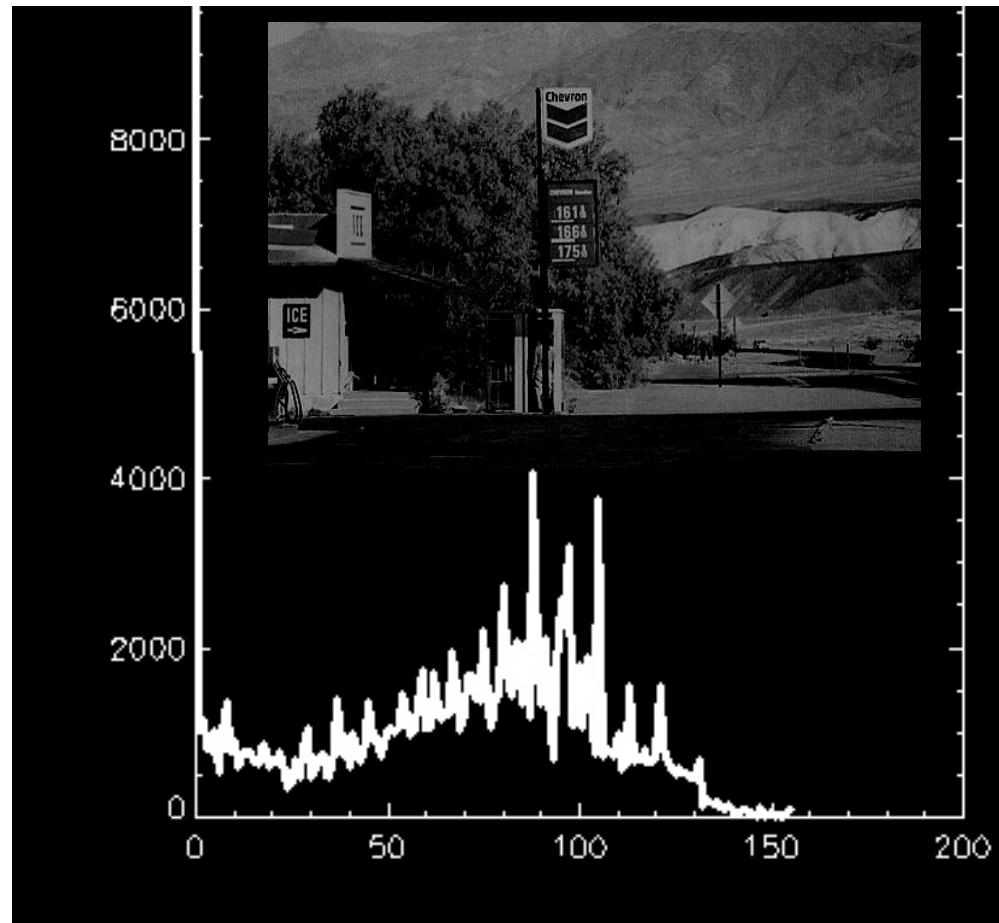


Nutzung des Grauwertspektrums





Unter-/Überbelichtung

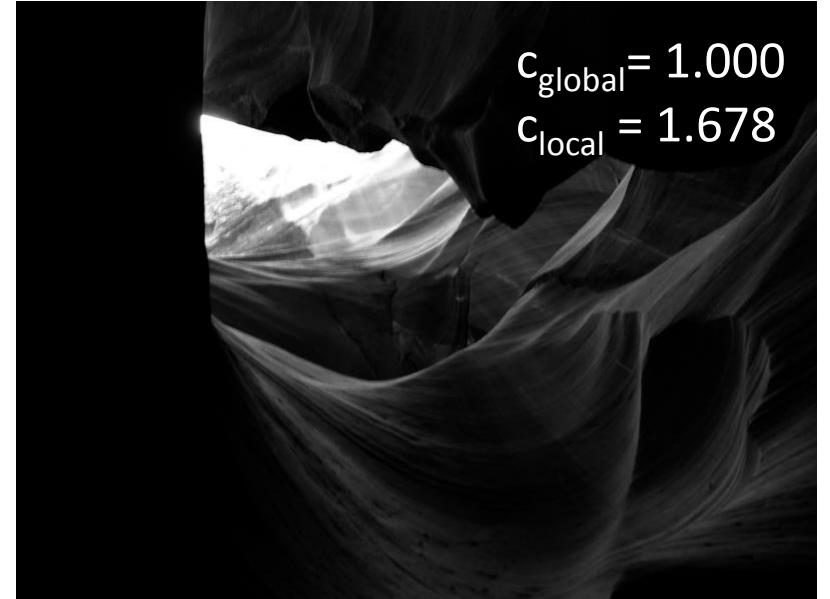


Kontrast

- *Globaler Kontrast*: Größter Grauwertunterschied im Bild

$$c_{global}(f) = \frac{[max_{m,n}(f(m,n)) - min_{m,n}(f(m,n))]}{g_{range}},$$

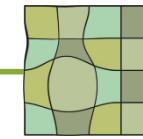
mit g_{range} – Grauwertbereich



- *Lokaler Kontrast*: z.B. durchschnittlicher Grauwertunterschied zwischen benachbarten Pixeln

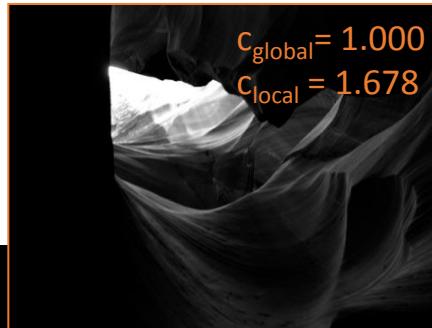
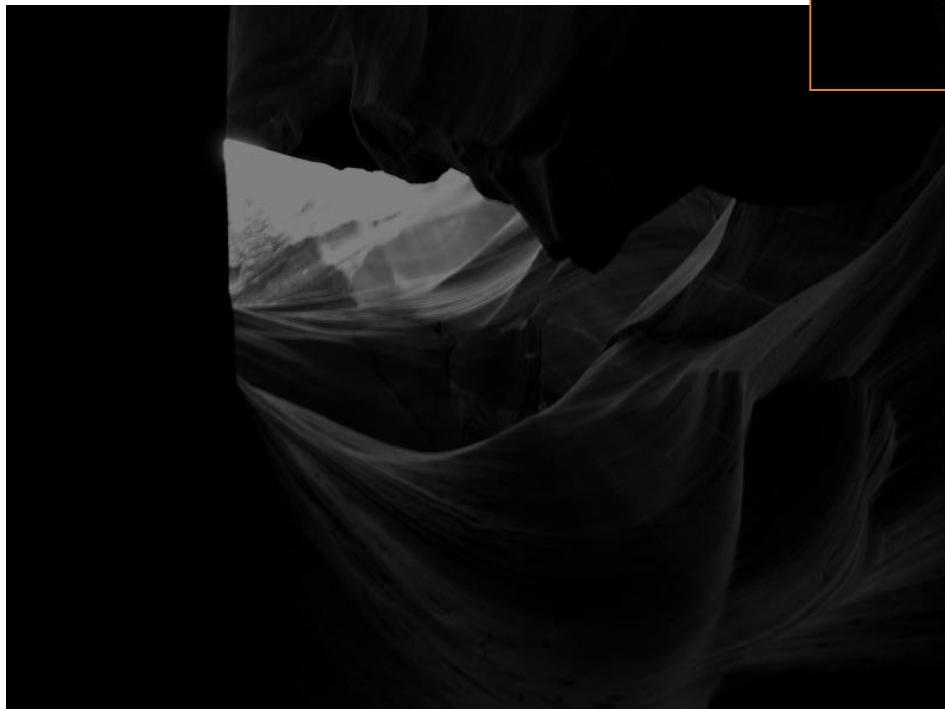
$$c_{local}(f) = 1/MN \sum_m \sum_n |f(m, n) - f_{nb}(m, n)|,$$

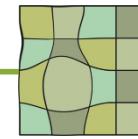
mit $f_{nb}(m, n)$ – durchschnittlicher Grauwert in der Umgebung von (m, n) .



Globaler / Lokaler Kontrast

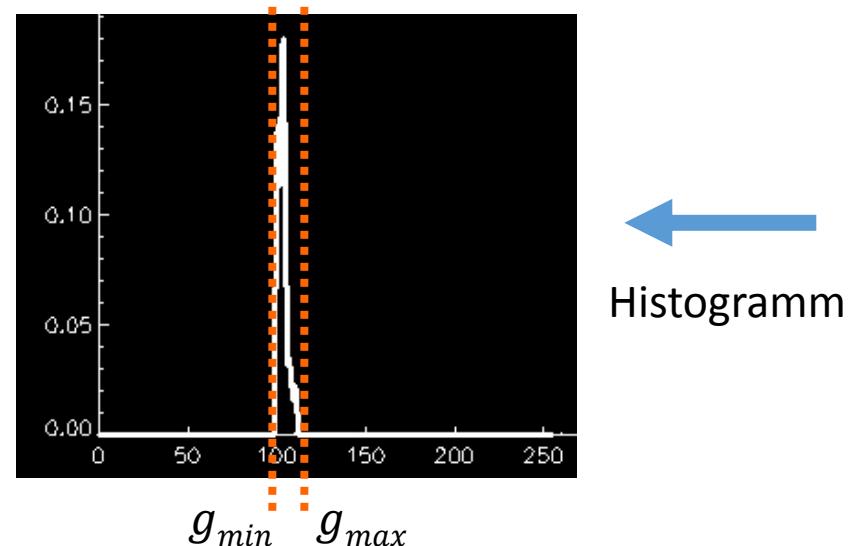
$$c_{\text{global}} = 0.500$$
$$c_{\text{local}} = 0.839$$





Maximierung des globalen Kontrasts

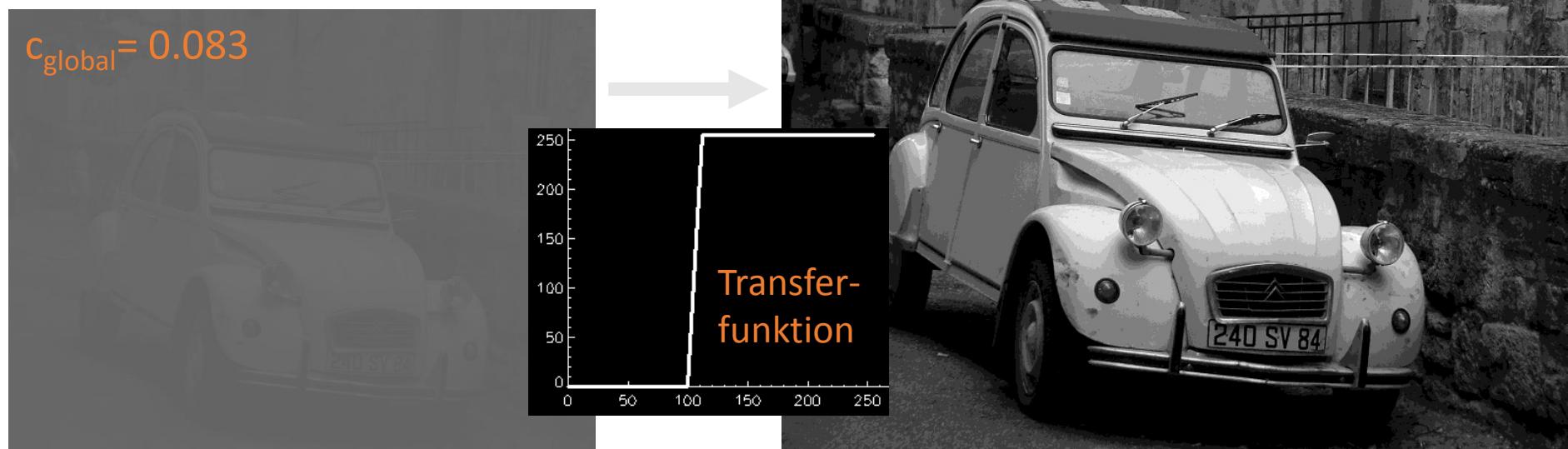
- Kontrastumfang g_{min} bis g_{max} im Verhältnis zum maximalen Wertebereich w_{min} bis w_{max} (z.B. 0...255) ist der Skalierungsfaktor.
- Transferfunktion:
$$g'(g) = (g - g_{min}) \cdot \frac{w_{max} - w_{min}}{g_{max} - g_{min}}$$



Maximierung des globalen Kontrasts

$$g'(g) = (g - g_{\min}) \cdot \frac{w_{\max} - w_{\min}}{g_{\max} - g_{\min}}, \quad g_{\min} = 100, g_{\max} = 112, w_{\min} = 0, w_{\max} = 255$$

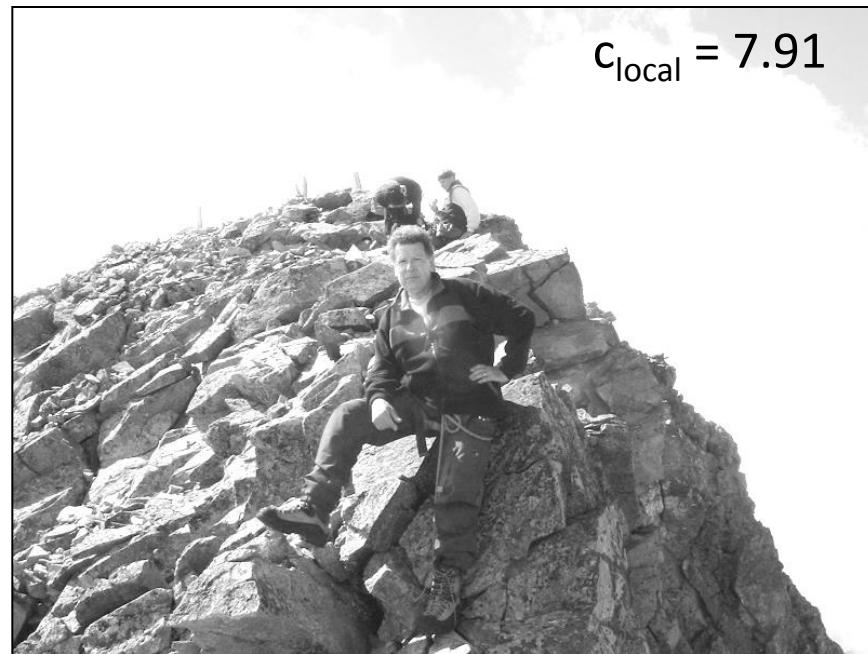
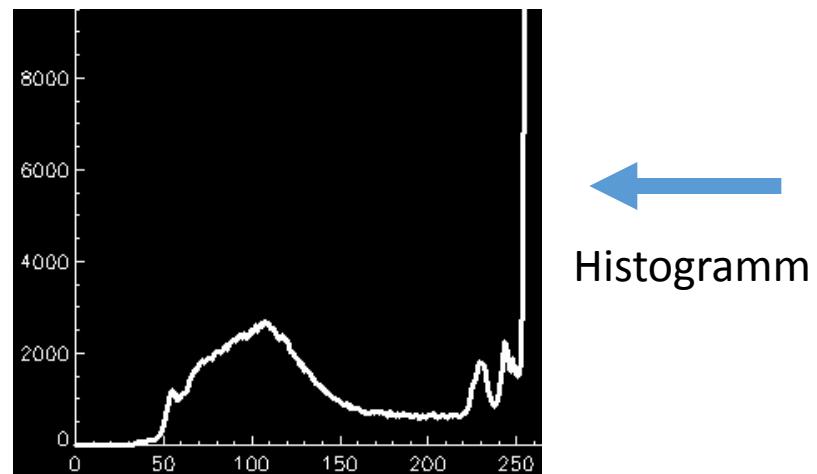
$$g'(g) = (g - 100) \cdot \frac{255}{12}$$



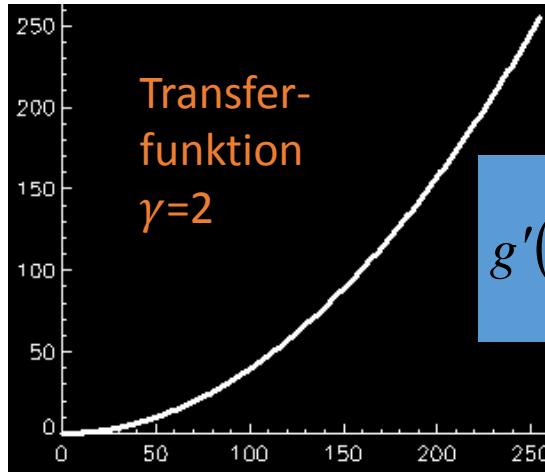
Verbesserung des lokalen Kontrasts

- Bild ist zu hell (zu dunkel), aber Grauwertbereich ist nahezu ausgenutzt.
- Nichtlineare, monotone Transferfunktion, z.B. Gammakorrektur:

$$g'(g) = w_{\max} \cdot \left(\frac{g}{w_{\max}} \right)^{\gamma}$$

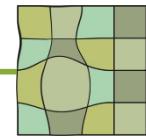


Verbesserung des lokalen Kontrasts



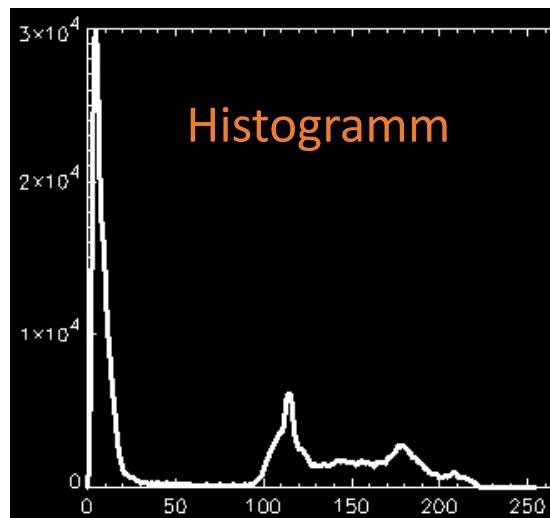
$$g'(g) = w_{\max} \cdot \left(\frac{g}{w_{\max}} \right)^{\gamma}$$

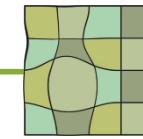




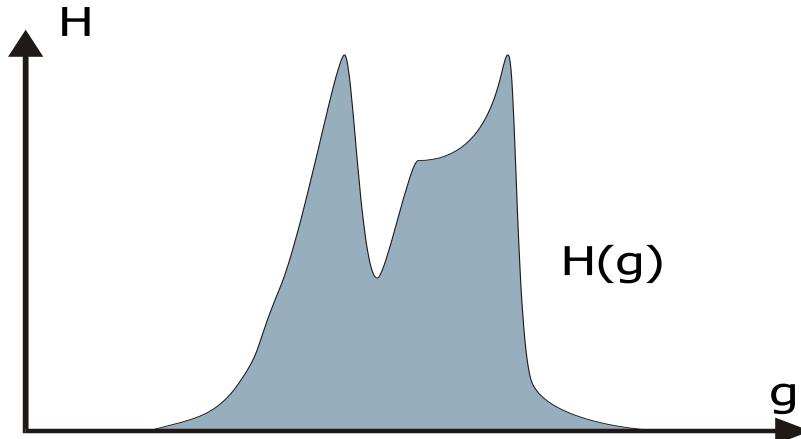
Maximierung des Informationsgehalts

- Gibt es eine „optimale“ Korrektur?
- Optimal = maximaler Informationsgehalt

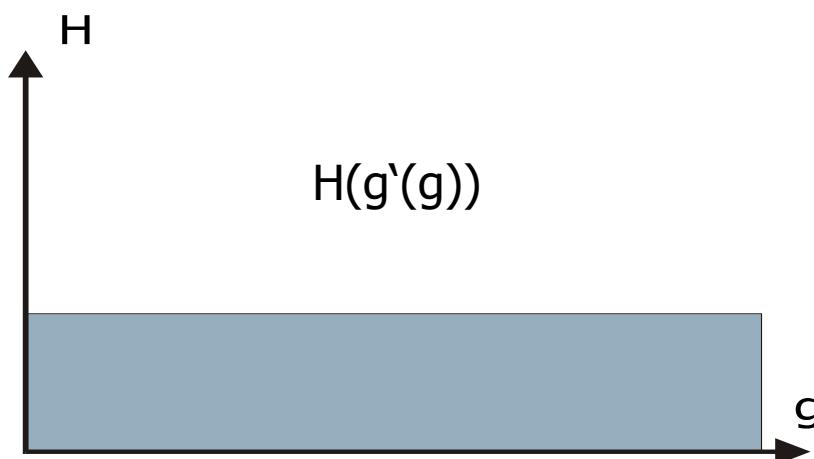




Maximaler Informationsgehalt



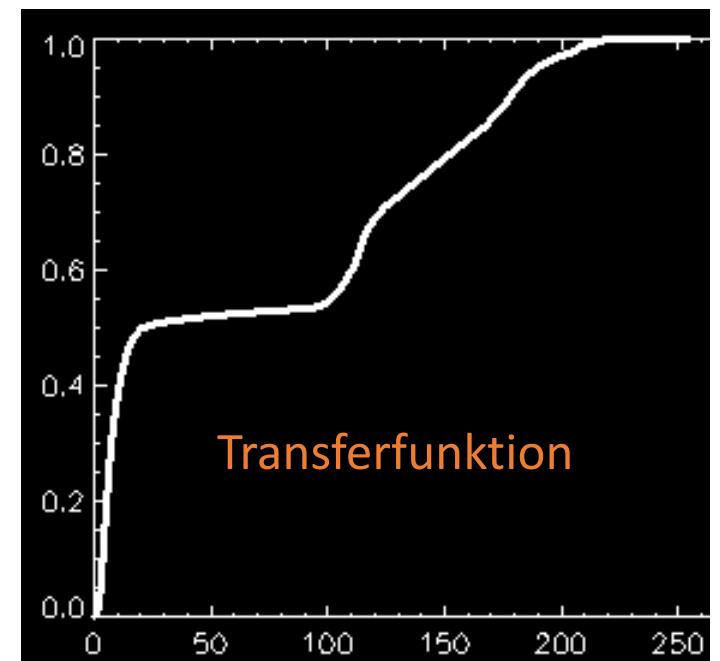
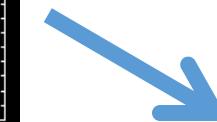
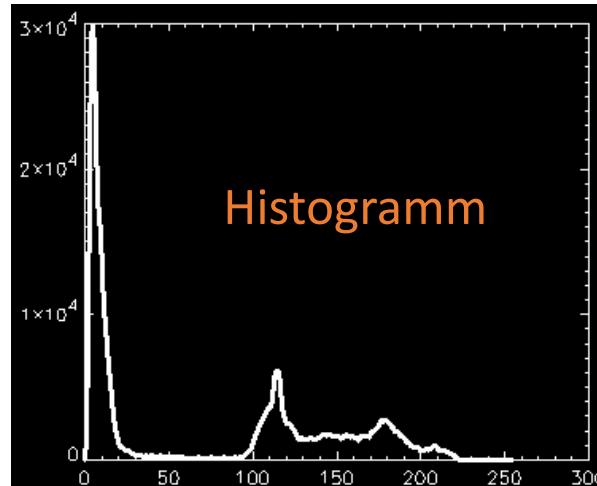
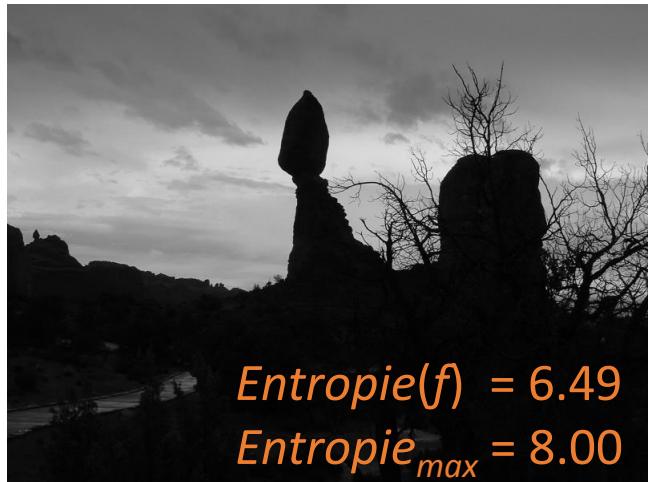
Entropie ist maximal, falls $P(g_i) = \text{const}$ für $i = 0, N - 1$
gesucht: Histogrammtransformation $g'(g)$ zur
Maximierung der Entropie



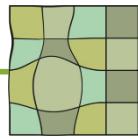
Annahme:
 $H(g)$ ist normiert und kontinuierlich, d.h., $\int H(g) = 1$.
Dann existiert die folgende Transferfunktion g' :

$$g'(g) = \int_0^g H(w) dw$$

Beispiel



Aber: was ist, falls $g'(g) \cdot (N - 1)$ keine ganze Zahl ist?



Histogrammlinearisation

- Transferfunktion für ein diskretes Histogramm:

$$E[H(g)] = \lceil N_g \sum_{w=0, g-1} H(w) \rceil - 1, \text{ mit: } N_g - \text{Anzahl der Grauwerte.}$$

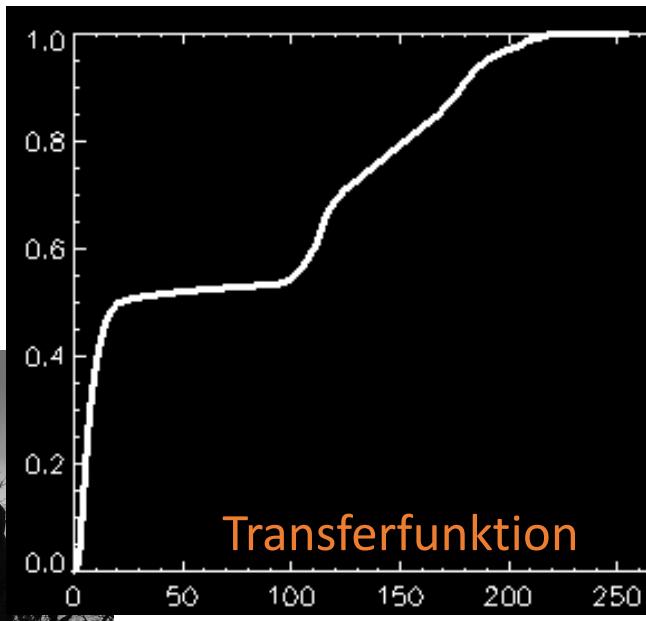
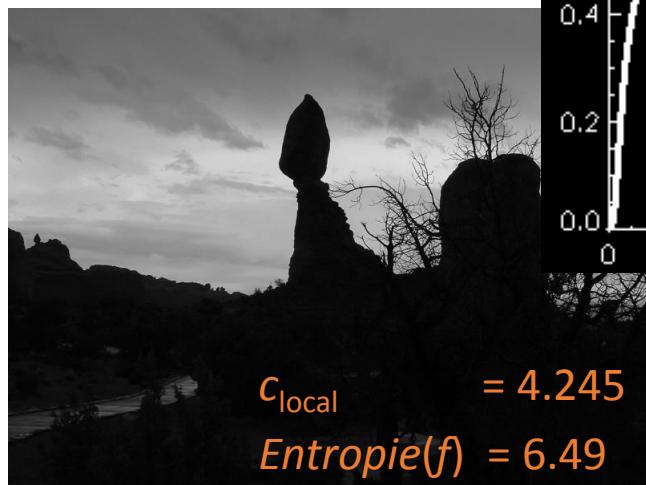
- Beispiel:

Grauwert	0	1	2	3	4	5	6	7
Häufigkeit	50	150	350	250	100	60	30	10
H(g)	0.05	0.15	0.35	0.25	0.10	0.06	0.03	0.01
kumulativ	0.05	0.20	0.55	0.80	0.90	0.96	0.99	1.00
Grauwert	0.4	1.6	4.4	6.4	7.2	7.68	7.92	8.00
aufgerundet	1	2	5	7	8	8	8	8
$\lceil \cdot \rceil - 1$	0	1	4	6	7	7	7	7

- Keine *Linearisierung*, sondern von der Häufigkeit abhängige *Spreizung*.

Beispiel

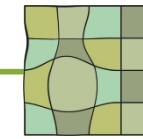
- Warum wurde die Entropie **kleiner**? ► FLQ



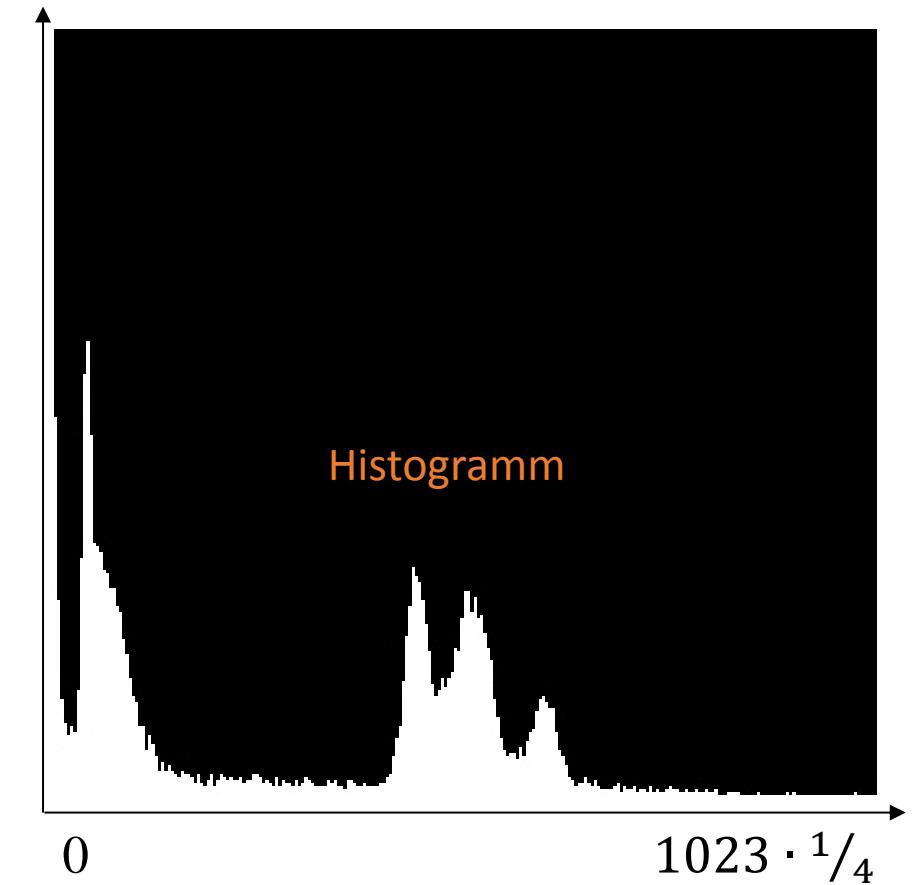
Beispiel

- Das geht auch in Farbe:
 - Farbkanäle werden unabhängig voneinander behandelt.
 - *(Ist das eine gute Idee? Was wäre eine bessere?)*



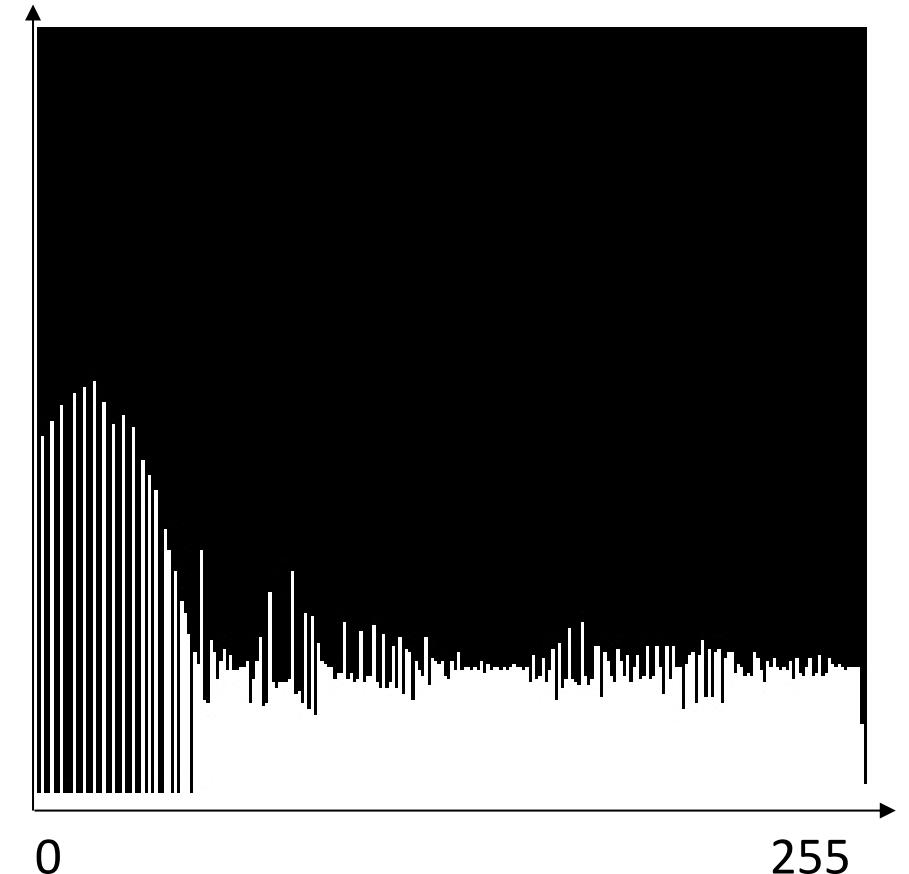
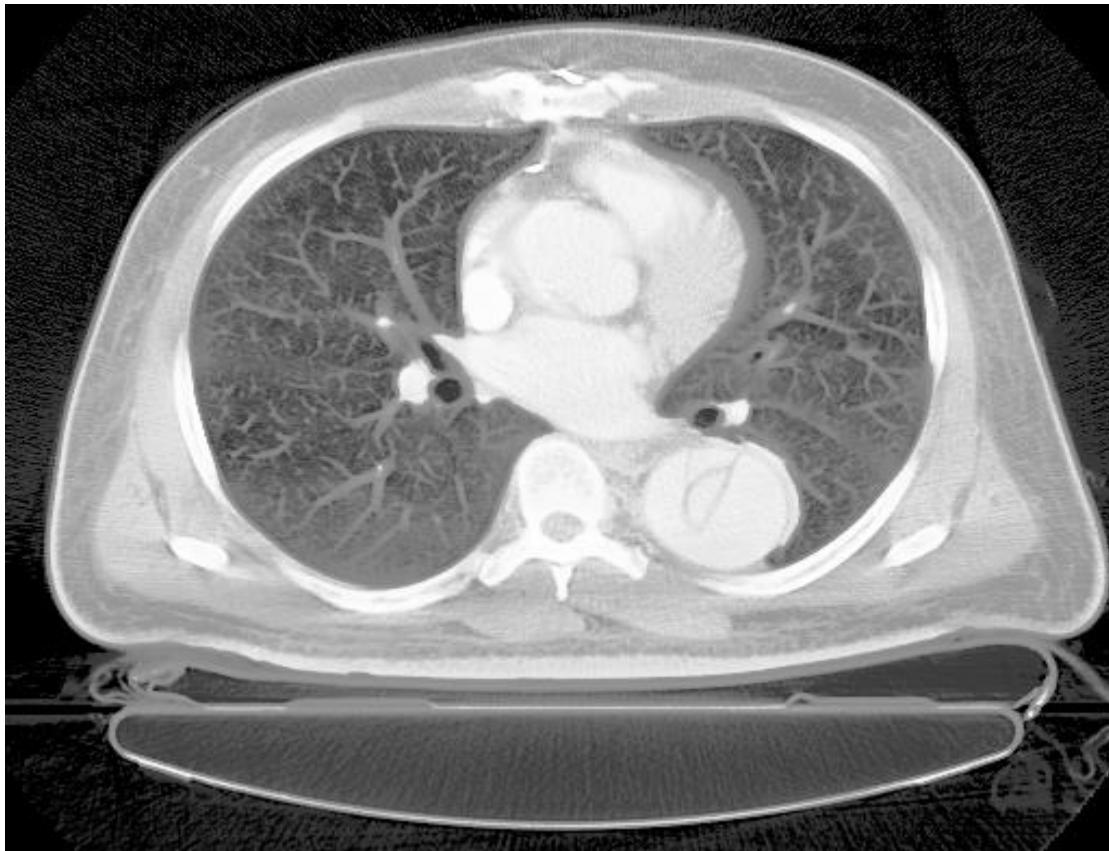


Histogrammlinearisierung

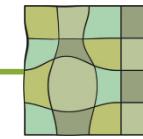


(in $\frac{1}{4}$ -Schritten)

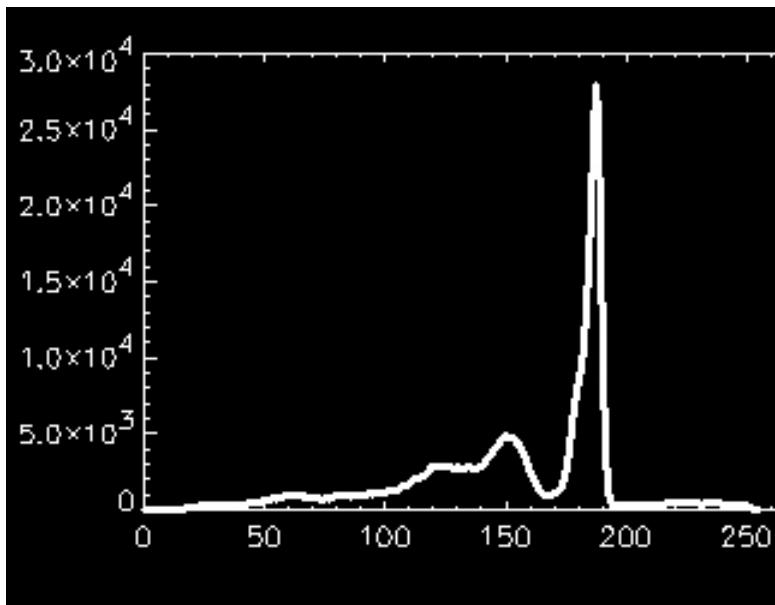
Histogrammlinearisation



(in vollen Schritten)



Problem

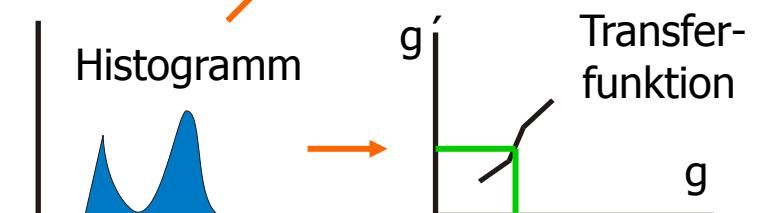
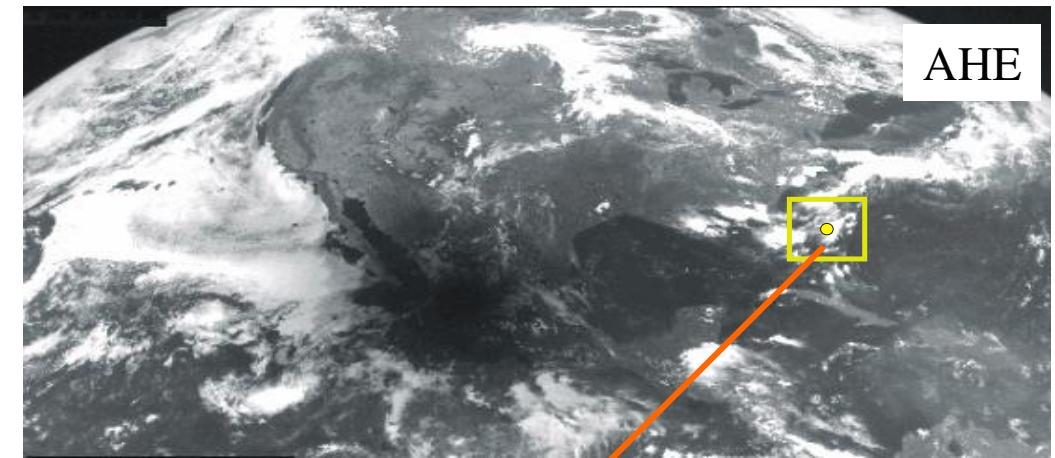


Problem



Histogrammlinearisierung: Varianten

- *Adaptive Histogram Equalisation (AHE)*
 - Histogramm wird für jeden Punkt für eine vorgegebene Umgebung erstellt.
 - Linearisierung nach diesem Histogramm
 - Nur der Grauwert des betreffenden Punkts wird modifiziert
- *Contrast Limited Adaptive Histogram Equalisation (CLAHE):*
 - wie AHE, aber Kontrastverstärkung nur bis zu einem gewissen Maximum.
 - verhindert die bei AHE vorkommende Kontrastverstärkung im Bildhintergrund.

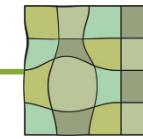


Adaptive Histogramlinearisierung



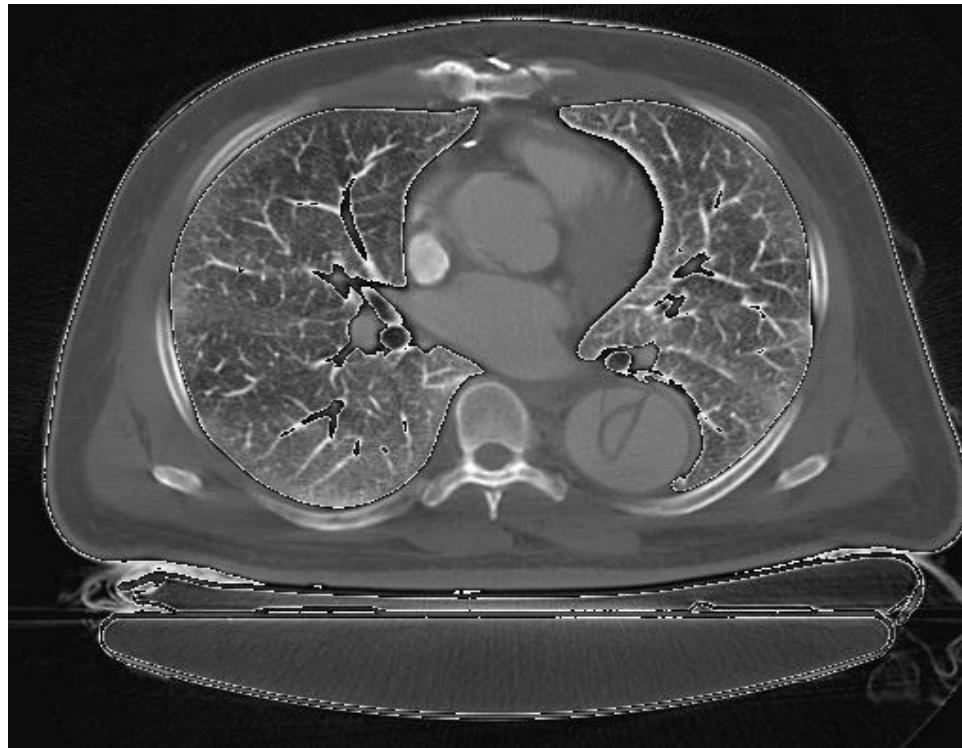
Kontrastlimitiertes AHE



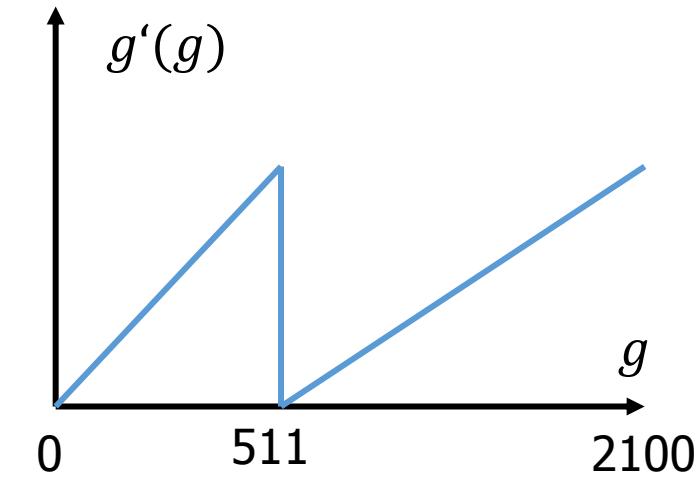


Nichtmonotone Grauwertabbildung

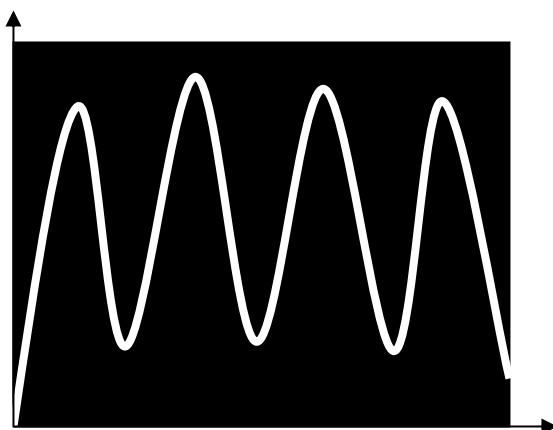
- Erzeugt künstliche Kanten.
- Grenzen von Maxima der Transferfunktion nicht immer erkennbar.



Zwei Grauwertfenster in einem Bild.

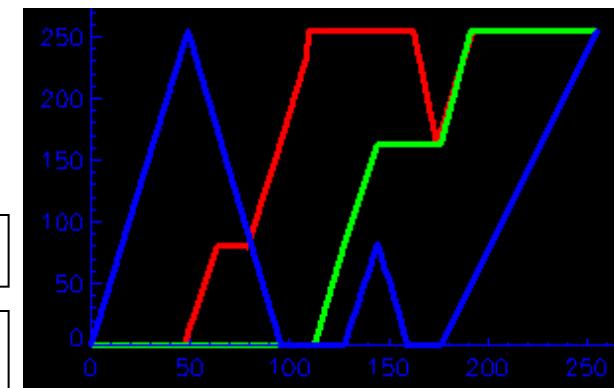
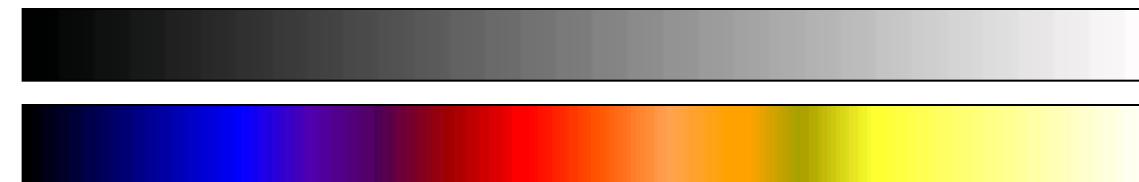
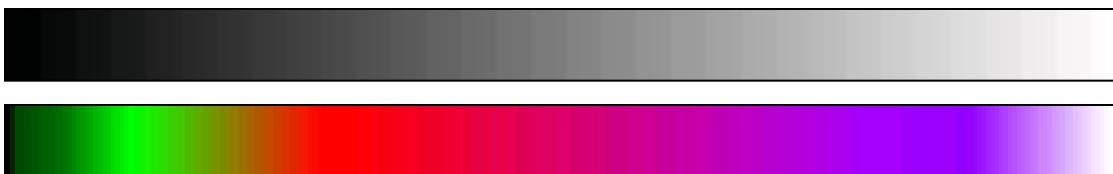
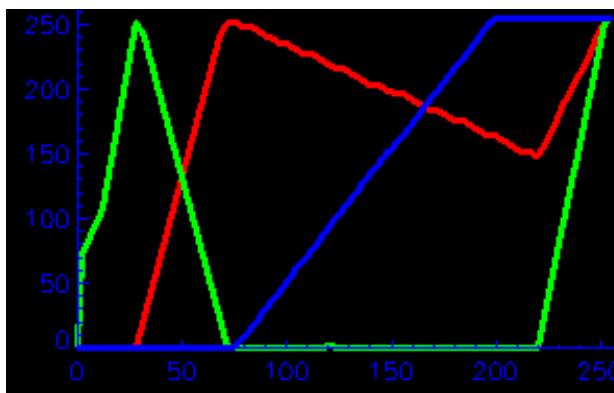


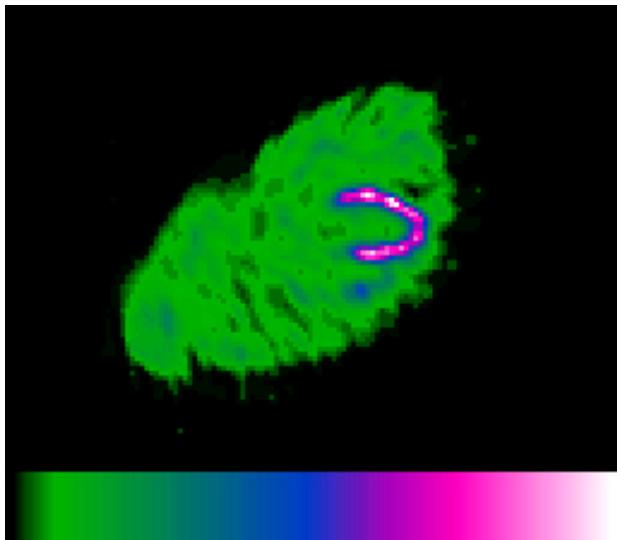
Beispiel



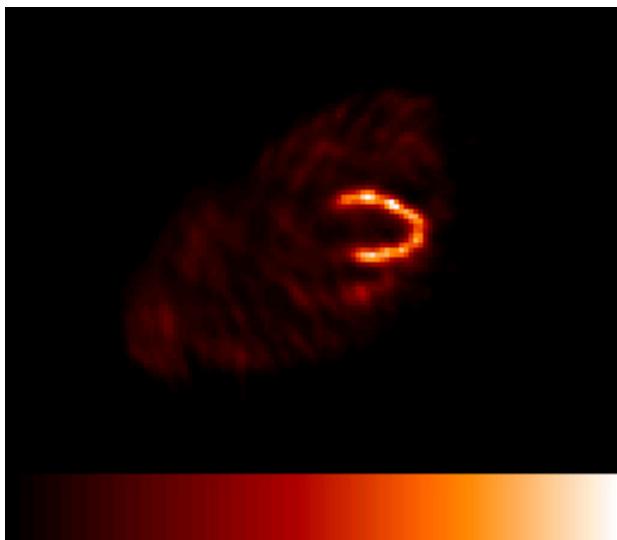
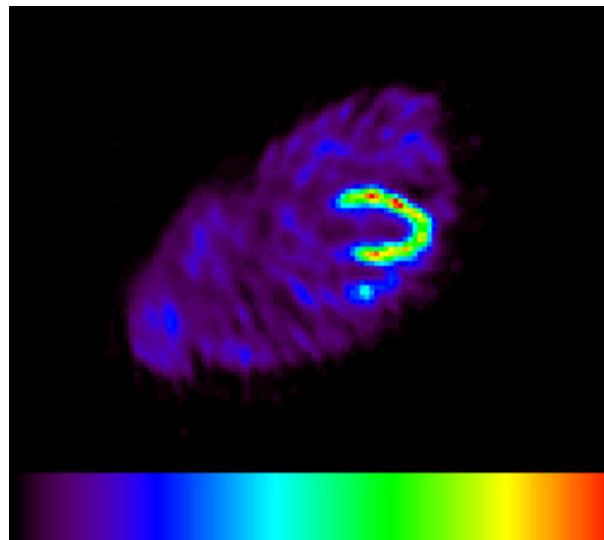
Farbe zur Kontrastverstärkung

- Es können wesentlich mehr Farb- als Grauwerte unterschieden werden.
- Kontrastverstärkung durch drei nicht-lineare, nicht-monotone Abbildungsfunktionen der Grauwerte: $f_{rot}(g)$, $f_{grün}(g)$, $f_{blau}(g)$

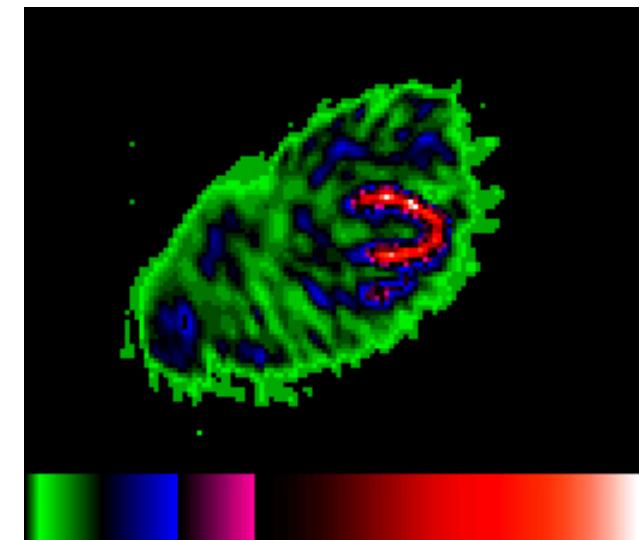
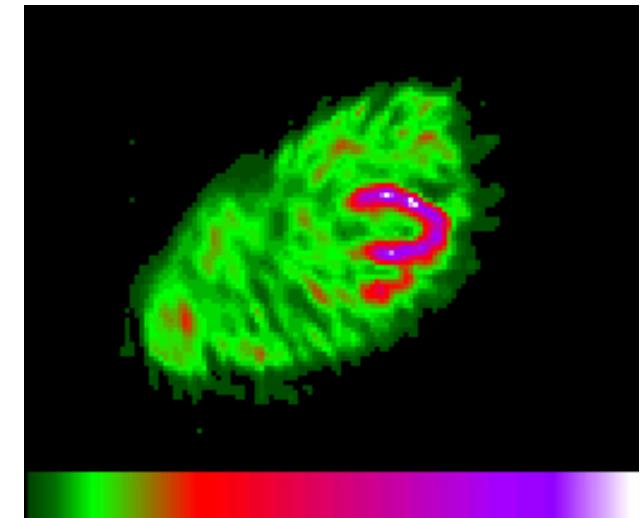




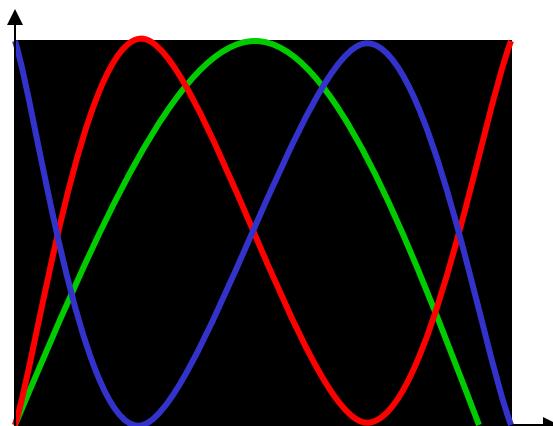
Beispiel



Achtung: Nichtlineare Transformationen erzeugen *künstliche Kanten*.



Beispiel



Ein Zusammenhang zwischen Farbe und Grauwert ist nicht mehr erkennbar.

Was sollten Sie heute gelernt haben?

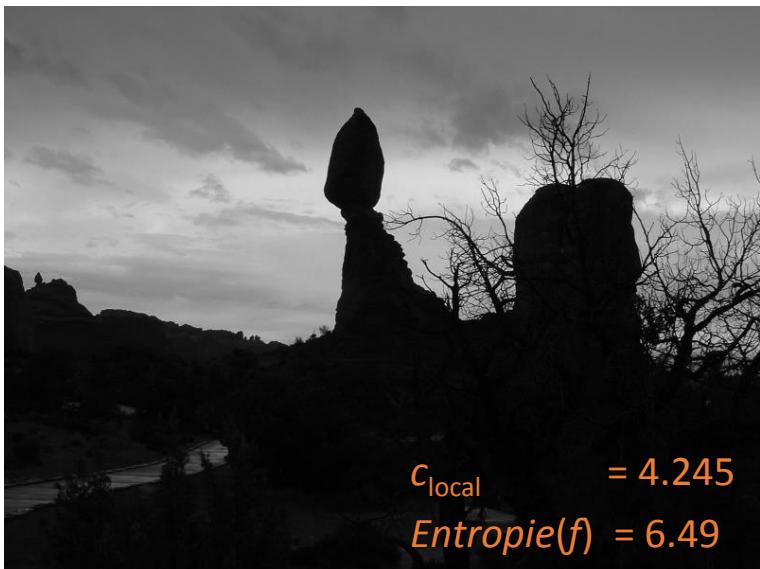
- Punktbasierte Verfahren werden über eine Transferfunktion zwischen Grauwerten (Farbwerten) definiert.
- Grauwerttransformationen
 - monoton: linear, γ -Korrektur, Histogrammlinearialisierung
 - Nicht monoton: Stufentransformation, Falschfarbdarstellung.
- Erfolg kann an kontrastbasierten Maßzahlen ermittelt werden.
- Objektabhängige Bildverbesserung erfordert Zusatzwissen.

Famous Last Question

Warum wird bei der Histogrammlinearisierung die Entropie kleiner?

Hat das damit zu tun, dass einfache HE das Rauschen verstärkt?

Und wieso sieht das Bild trotz schlechterer Entropie besser aus?

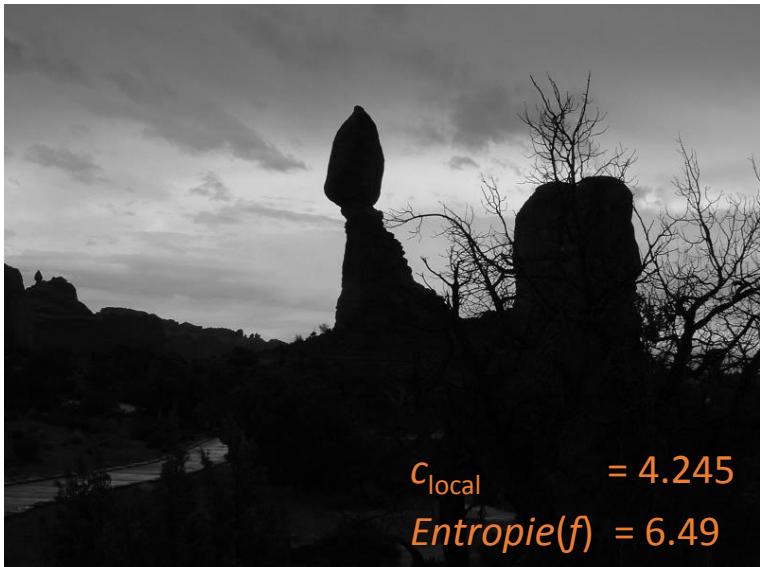


Famous Last Question

Warum wird bei der Histogrammlinearisierung die Entropie kleiner?

Hat das damit zu tun, dass einfache HE das Rauschen verstärkt?

Und wieso sieht das Bild trotz schlechterer Entropie besser aus?



Flächenbasierte Bildverbesserung - Rauschunterdrückung

- Rauschunterdrückung durch lineare Operatoren
 - Lineare Operatoren im Orts- und Frequenzbereich
 - Eigenschaften von Rauschunterdrückungsfiltern
 - Unsharp Masking
- Kanteninformation aus linearen Operatoren

Rauschen

- Bildinformation
 - Homogene Regionen mit gleicher Bedeutung
 - Kanten innerhalb von nicht-glatten Objekten
 - Kanten zwischen Objekten
- Was unterscheidet Rauschen von der Bildinformation
 - Rauschen ist ein stochastischer Prozess
 - Benachbarte, ungestörte Bildpixel haben wahrscheinlicher den gleichen Grauwert
 - Der Kantenverlauf benachbarte Kantenpixel ist ähnlich
- Rauschunterdrückung
 - Erhalt von Bildinformation bei gleichzeitiger Reduktion des Rauschens



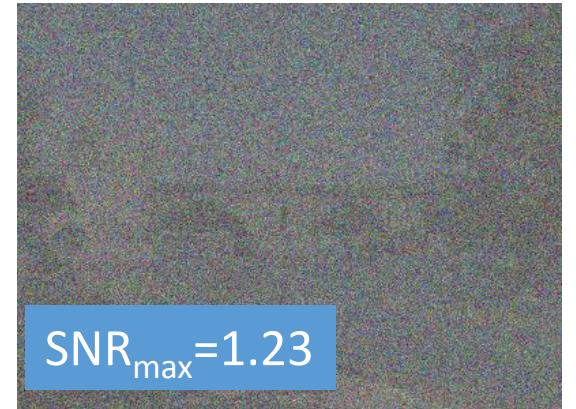
Rauschen als stochastischer Prozess: Zeitliche Folge

- Annahmen
 - Aufnahme mehrerer Bilder $g_i, i = 1, \dots, I$ über einen gegebenen Zeitraum.
 - Bild verändert sich über den Zeitraum nicht (keine Bewegung, keine Beleuchtungsänderung).
 - Erwartungswert E des Rauschens n ist 0.
- Näherung an die unverrauschte Funktion f :
 - $$\begin{aligned} E\{g(m, n)\} &= E\{f(m, n)\} + E\{n(m, n)\} \\ &= E\{f(m, n)\} + 0 = f(m, n) \end{aligned}$$
 - Abschätzung von $E\{g(m, n)\}$ durch Integration über die Bilder.



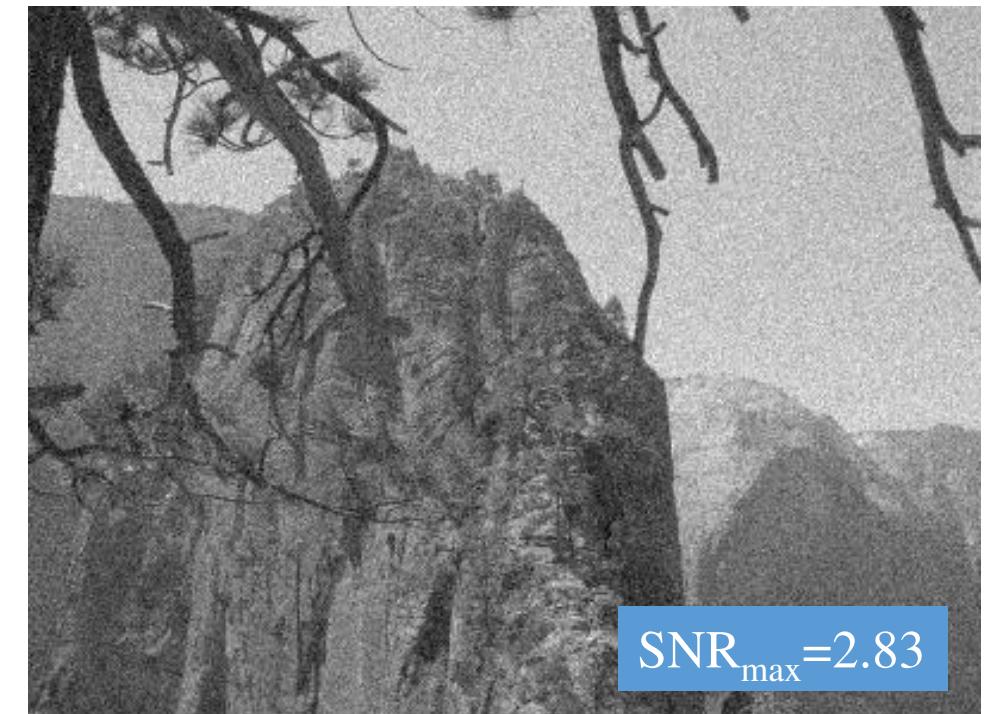
Beispiel

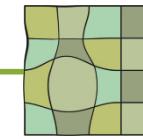
- Einzelne Aufnahme mit normalverteiltem Rauschen ($\text{SNR} \approx 1.2$)
- Addition von 10 bzw. 50 Aufnahmen



Integration über die Fläche

- Falls für Punkte (p_0, \dots, p_n) $f(p_i) = \text{const.}$ gilt, dann kann Rauschen n mit $E\{n\} = 0$ durch Mittelung der gemessenen Grauwerte $g(p_i)$ reduziert werden.
- Annahmen:
 - Bild besteht aus homogenen Bereichen.
 - Benachbarte Punkte haben den gleichen Grauwert.
- Rauschunterdrückung:
 - Mittelwertbildung über vorgegebene Nachbarschaft.

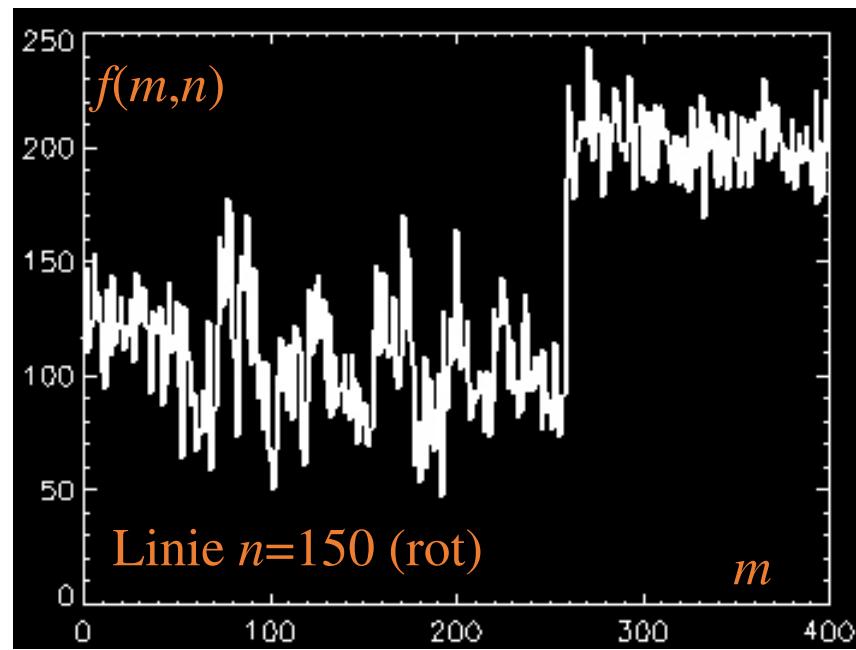




Mittelwertbildung durch Konvolution

Konvolutionskern:

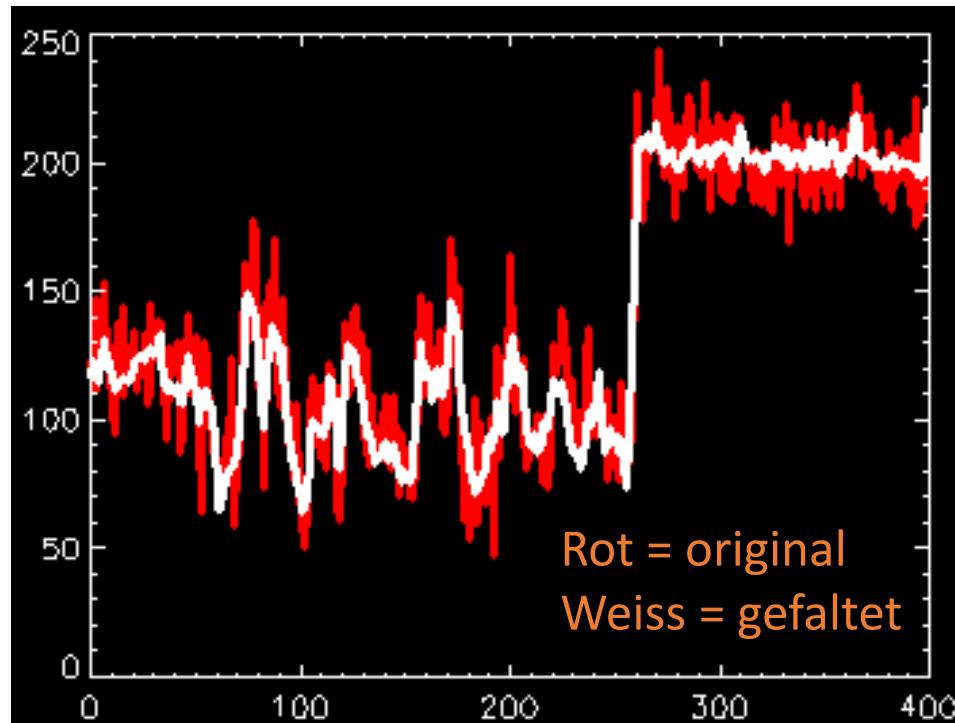
Gleichmäßige Gewichtung der Pixel in einer gegebenen Nachbarschaft



3x3 Mittelwert-Filter

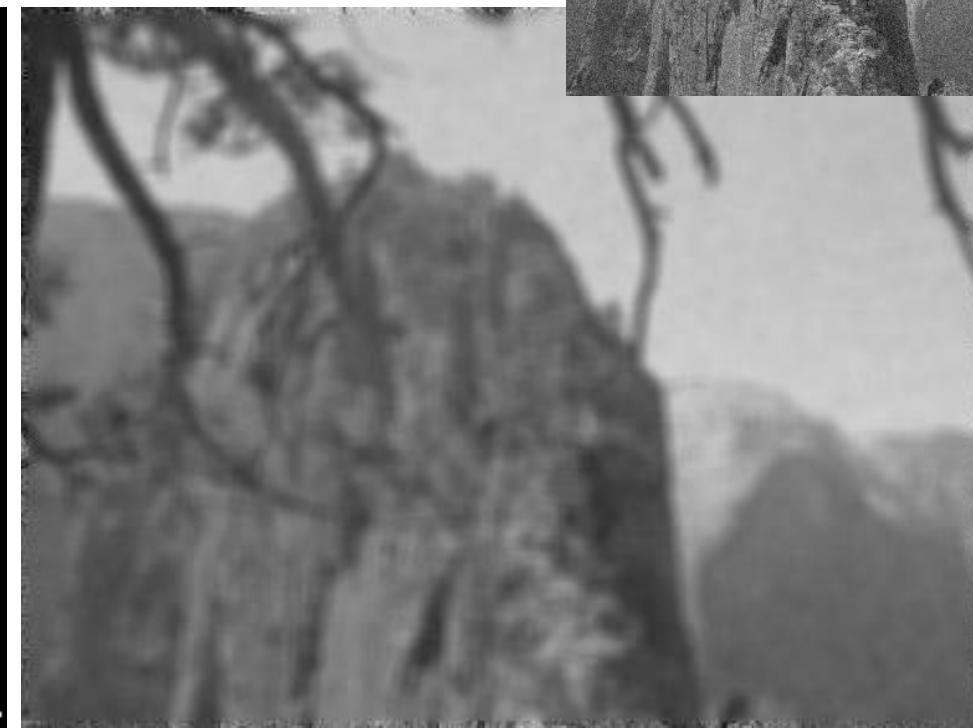
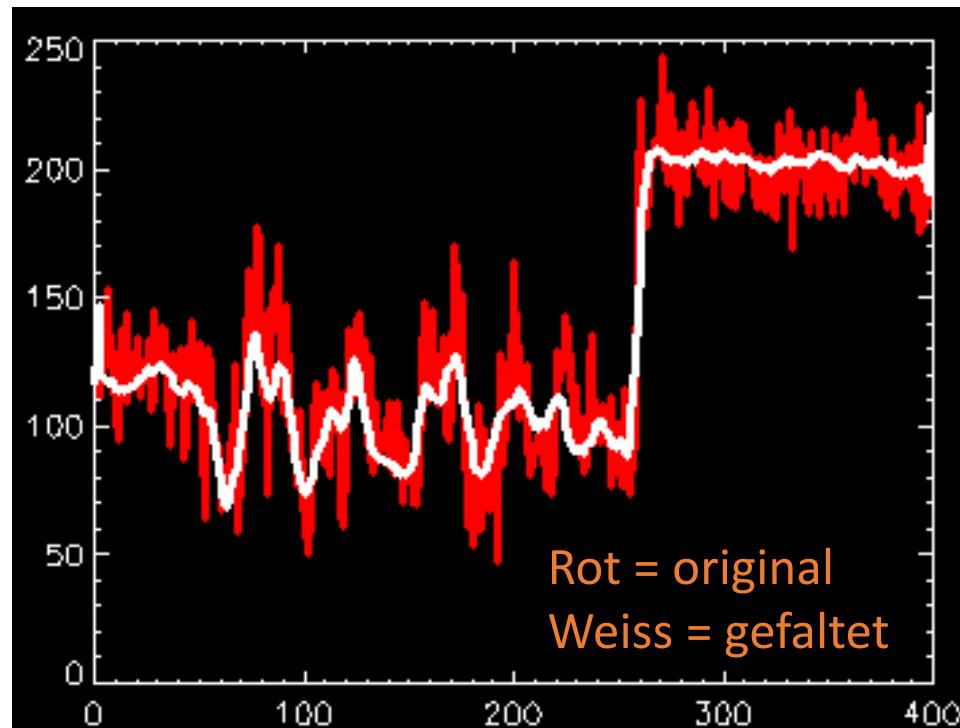
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Filterkern



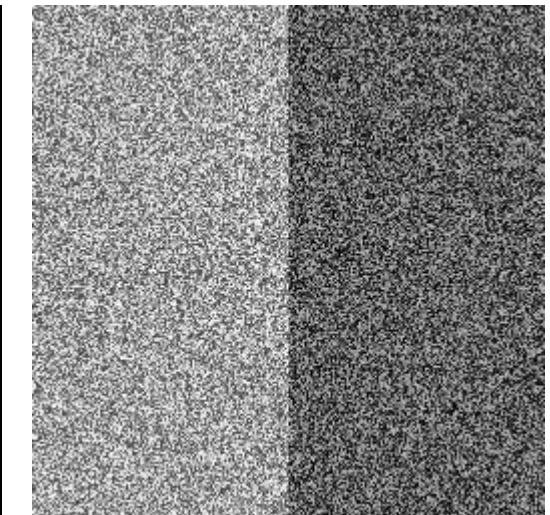
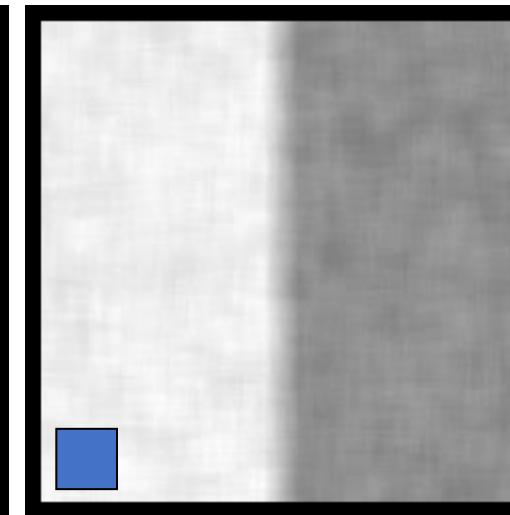
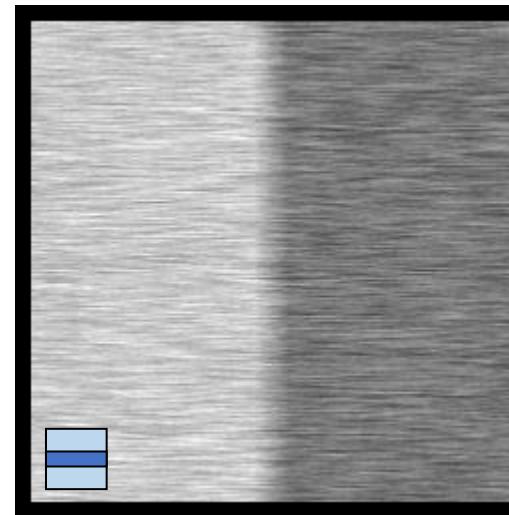
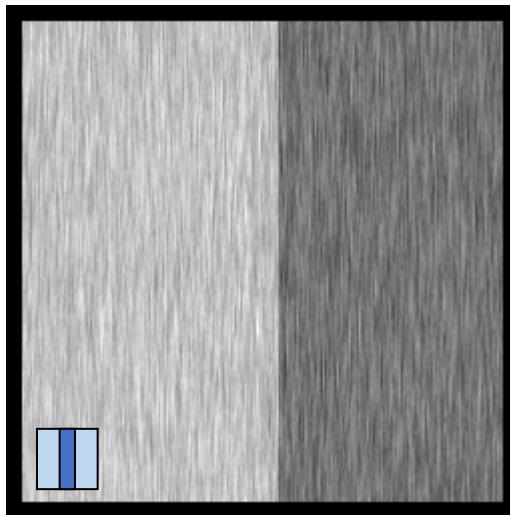
7x7 Mittelwert-Filter

- **Beobachtung:** Kanten werden degradiert.
- **Grund:** Annahme konstanter Funktionswerte ist nicht wahr.

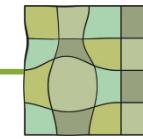


Eigenschaften aller Rauschunterdrückungsfilter

- Die Summe aller Elemente des Faltungskerns ist immer 1.
 - Gesamthelligkeit bleibt gleich.
- Der Faltungskern ist punktsymmetrisch zum Koordinatenursprung der Faltungsfunktion
 - Bild wird nicht verschoben und keine Richtung wird bevorzugt

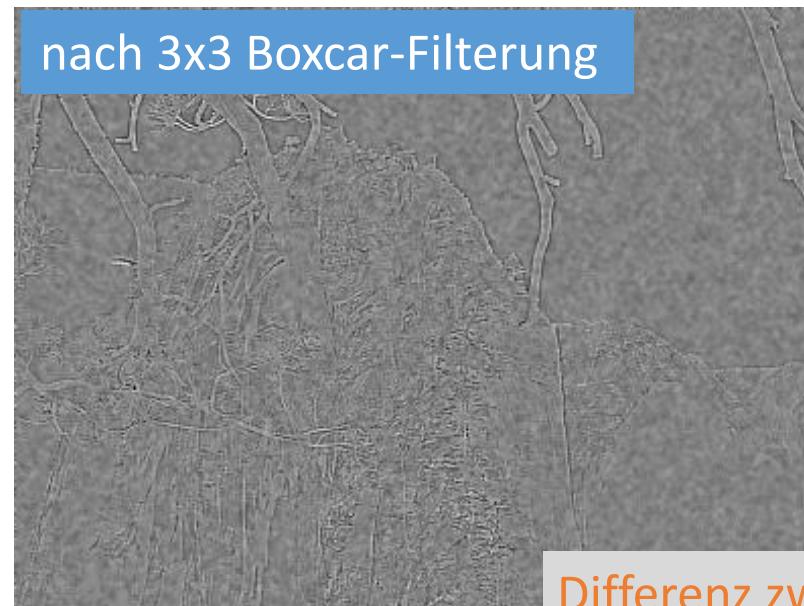
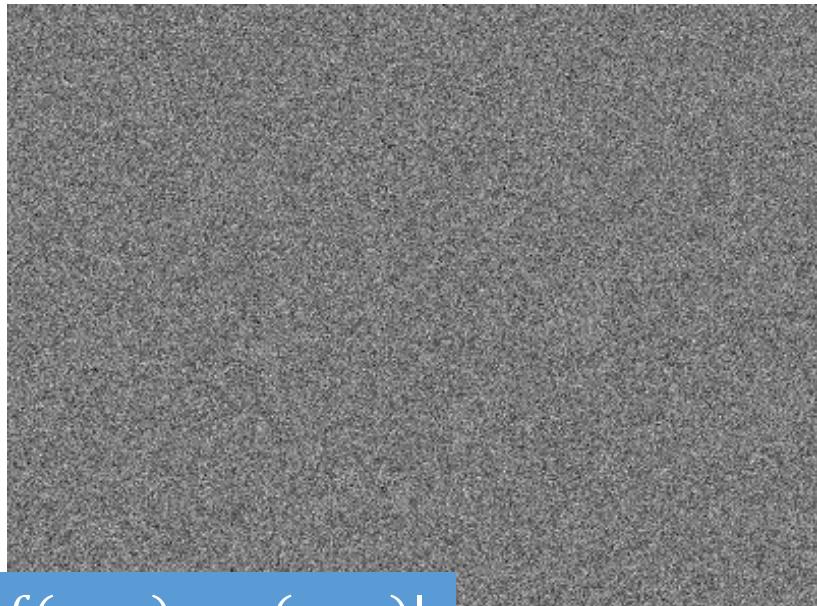


original



Verhalten an Kanten

- Was wird heraus gefiltert?



$$|f(m, n) - g(m, n)|$$

Differenz zwischen gefiltertem
und Originalbild

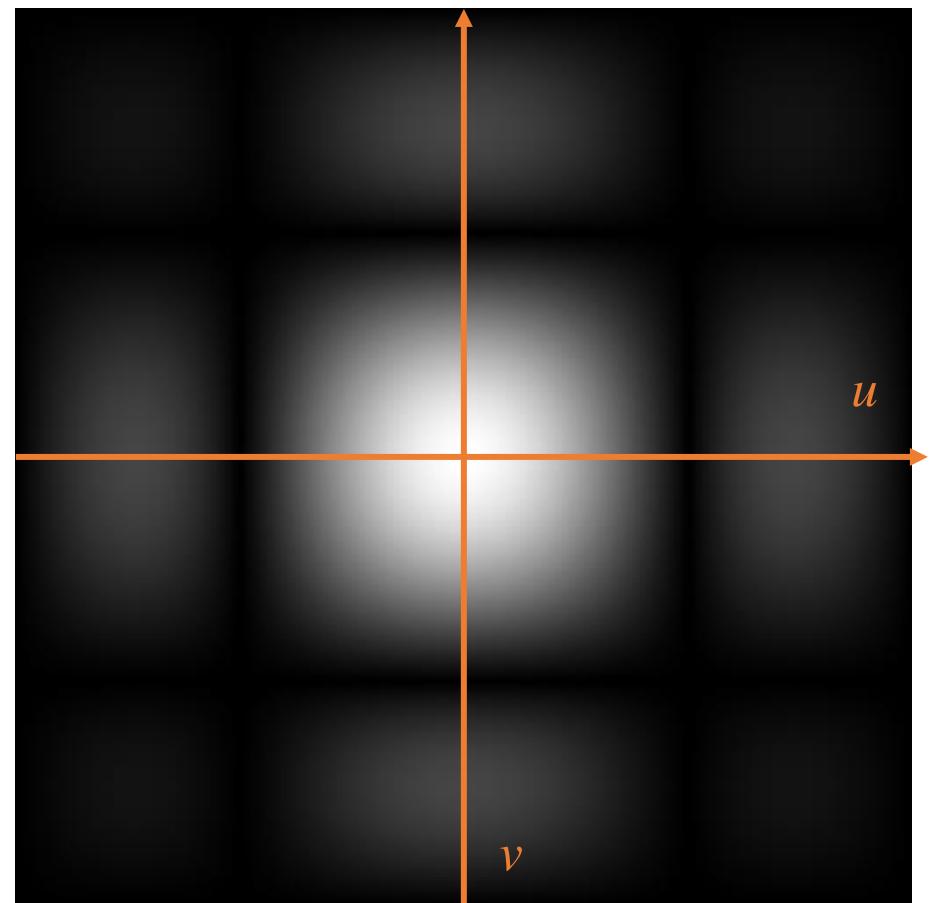
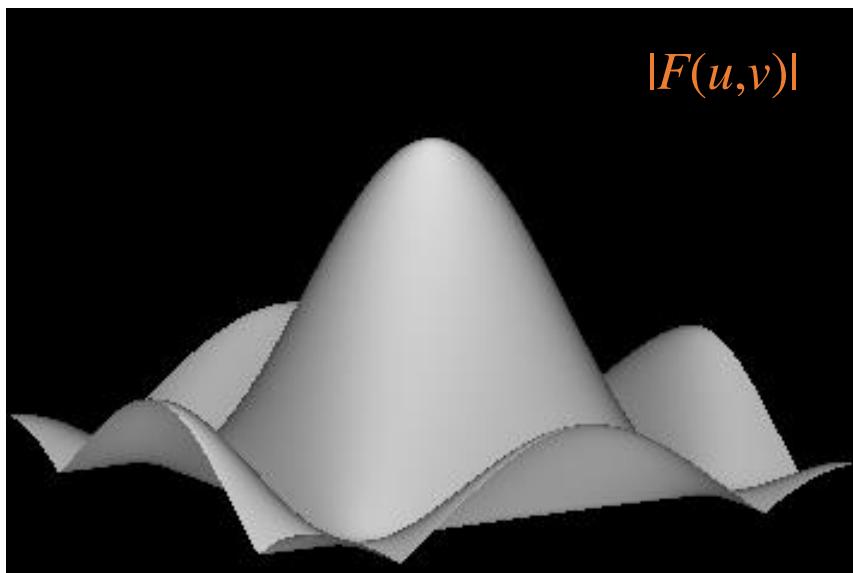
Verhalten an Kanten

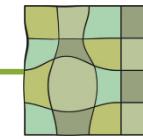
An Kanten ist die Bildinformation nicht lokal konstant

- Erwartungswert der ungestörten Funktion ergibt sich als Mittel der überdeckten Gebiete
- Größere Faltungskerne vergrößern den Bereich der Unschärfe

Richtungsabhängigkeit des Mittelwertfilters

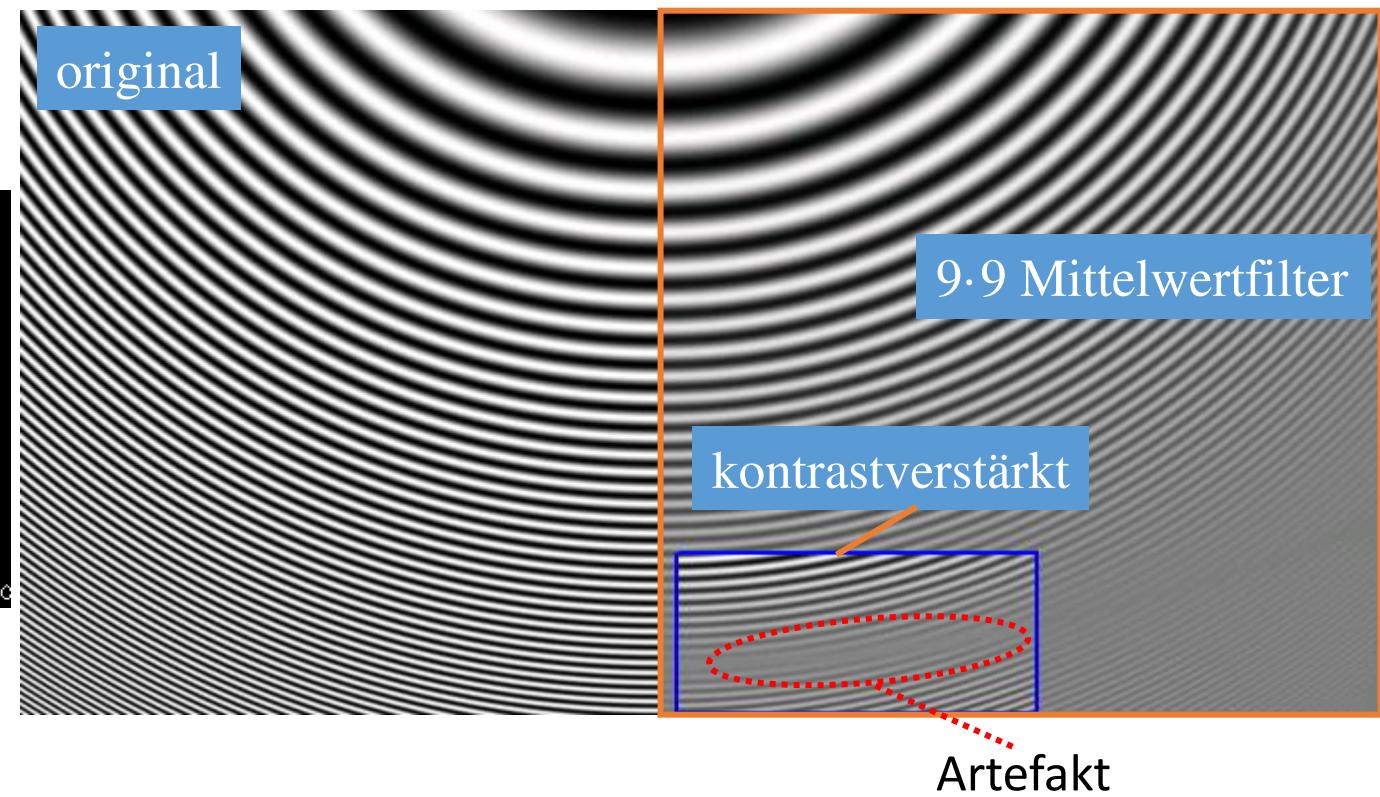
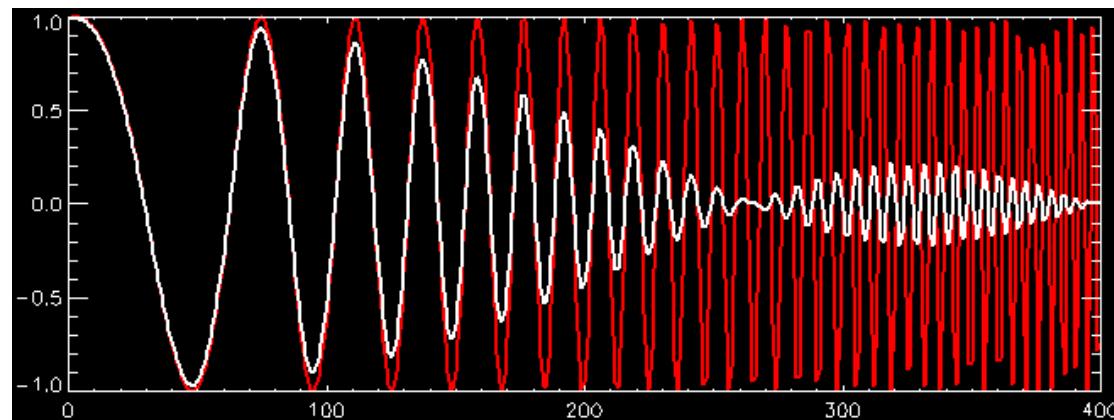
Transferfunktion (Repräsentation im Frequenzraum).





Auswirkungen

- Bildzeile
 - **rot:** vor der Filterung
 - **weiß:** nach Filterung



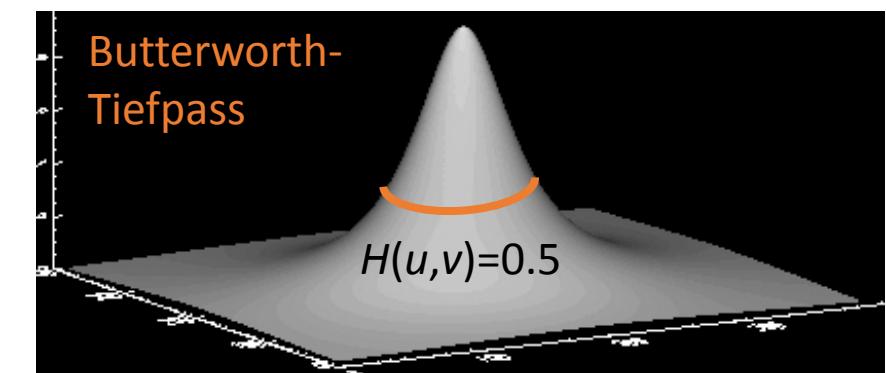
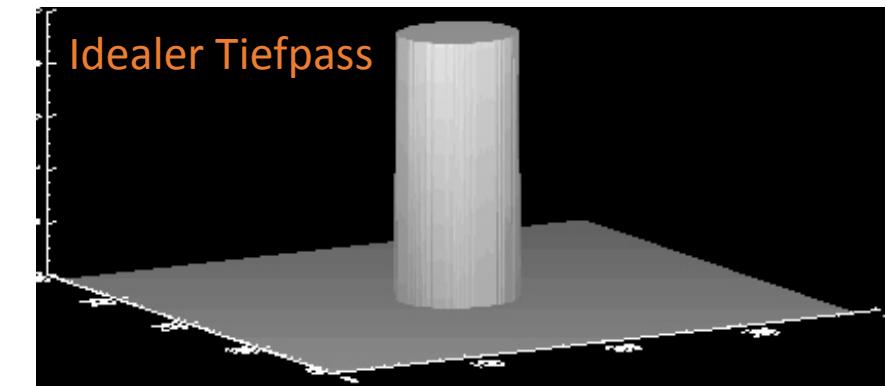
Erinnerung: Butterworth-Tiefpassfilter

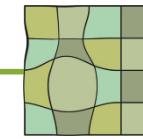
- Frequenzen werden im Frequenzraum gleichmäßig mit zunehmender Frequenz abgeschwächt.
- Butterworth-Tiefpass-Filter:

$$H(u, v) = \frac{1}{1 + (D(u, v)/D_0)^{2n}}$$

D_0 : Cutoff-Frequenz,

$D(u, v)$: Frequenz, d.h. Abstand
vom Ursprung





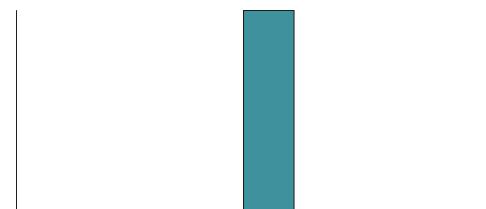
Butterworth vs. Einfacher Tiefpass



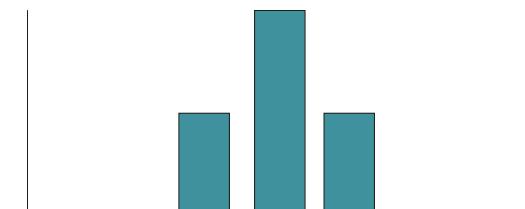
Ortsraum-Alternative: Binomialfilter

Eindimensionaler Binomialfilter $B^p = [1 \ 1] * [1 \ 1] * \dots * [1 \ 1]$ (p -mal):

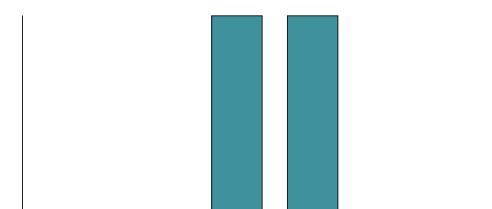
$$B^0 = 1^{-1} \cdot [1]$$



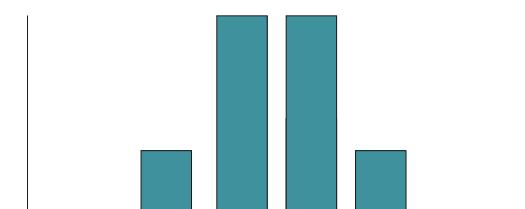
$$B^1 = 2^{-1} \cdot [1 \ 1]$$



$$B^2 = 4^{-1} \cdot [1 \ 2 \ 1]$$



$$B^3 = 8^{-1} \cdot [1 \ 3 \ 3 \ 1]$$

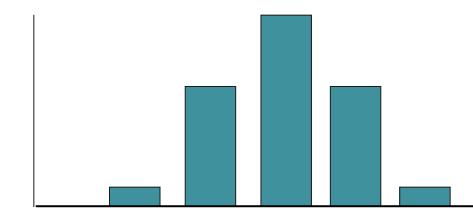


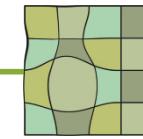
$$B^4 = 16^{-1} \cdot [1 \ 4 \ 6 \ 4 \ 1]$$

...

Zweidimensionaler Binomialfilter $\mathbf{B}^p = B^p * (B^p)^T$:

$$\mathbf{B}^2 = 4^{-1} \cdot [1 \ 2 \ 1]^T \cdot 4^{-1} \cdot [1 \ 2 \ 1] = 16^{-1} \cdot \begin{Bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{Bmatrix}$$

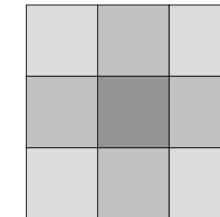




Zweidimensionale Binomialfilter

$$\mathbf{B}^2 = 1/16 \cdot [1 \ 2 \ 1]^T \cdot [1 \ 2 \ 1] = 1/16 \cdot$$

1	2	1
2	4	2
1	2	1

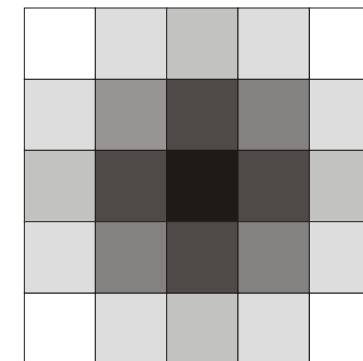


$$\mathbf{B}^3 = 1/64 \cdot [1 \ 3 \ 3 \ 1]^T \cdot [1 \ 3 \ 3 \ 1] = 1/64 \cdot$$

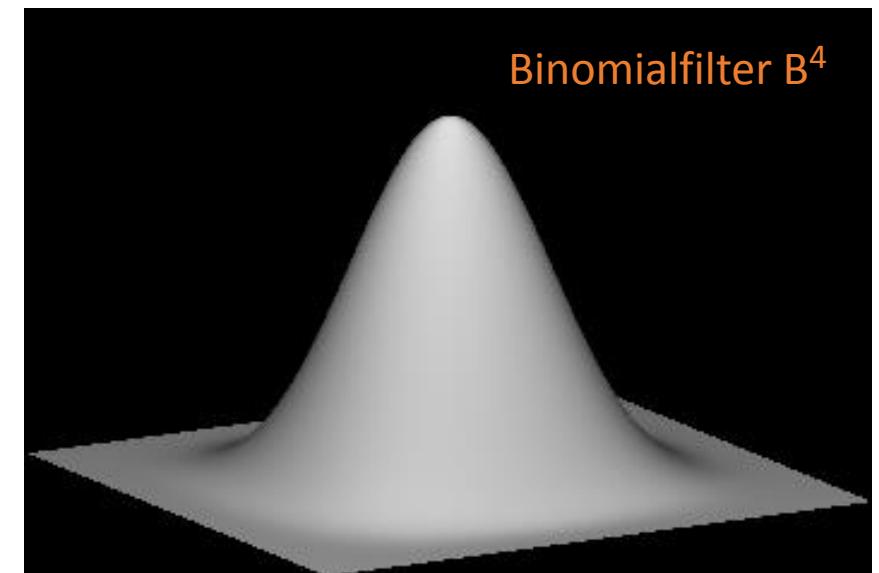
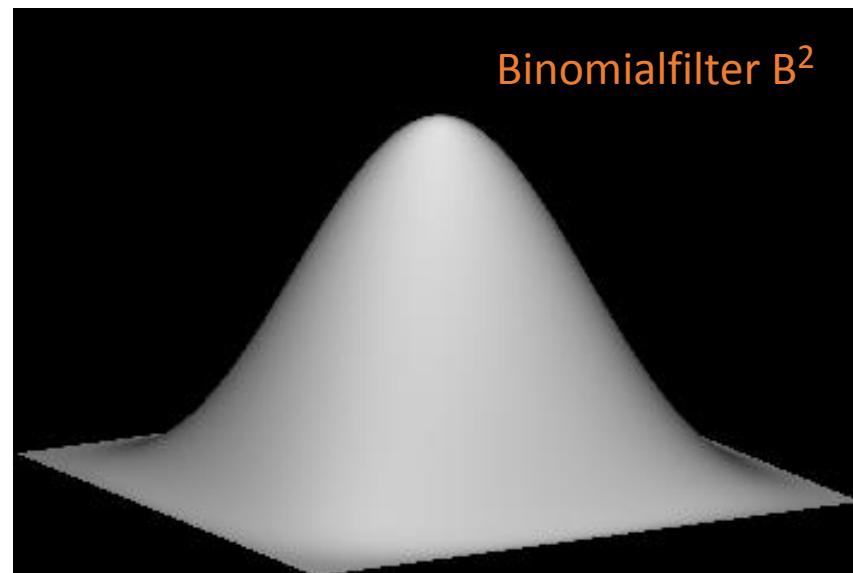
1	3	3	1
3	9	9	3
3	9	9	3
1	3	3	1

$$\mathbf{B}^4 = 1/256 \cdot$$

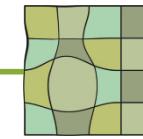
1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1



Transferfunktion des Binomialfilters



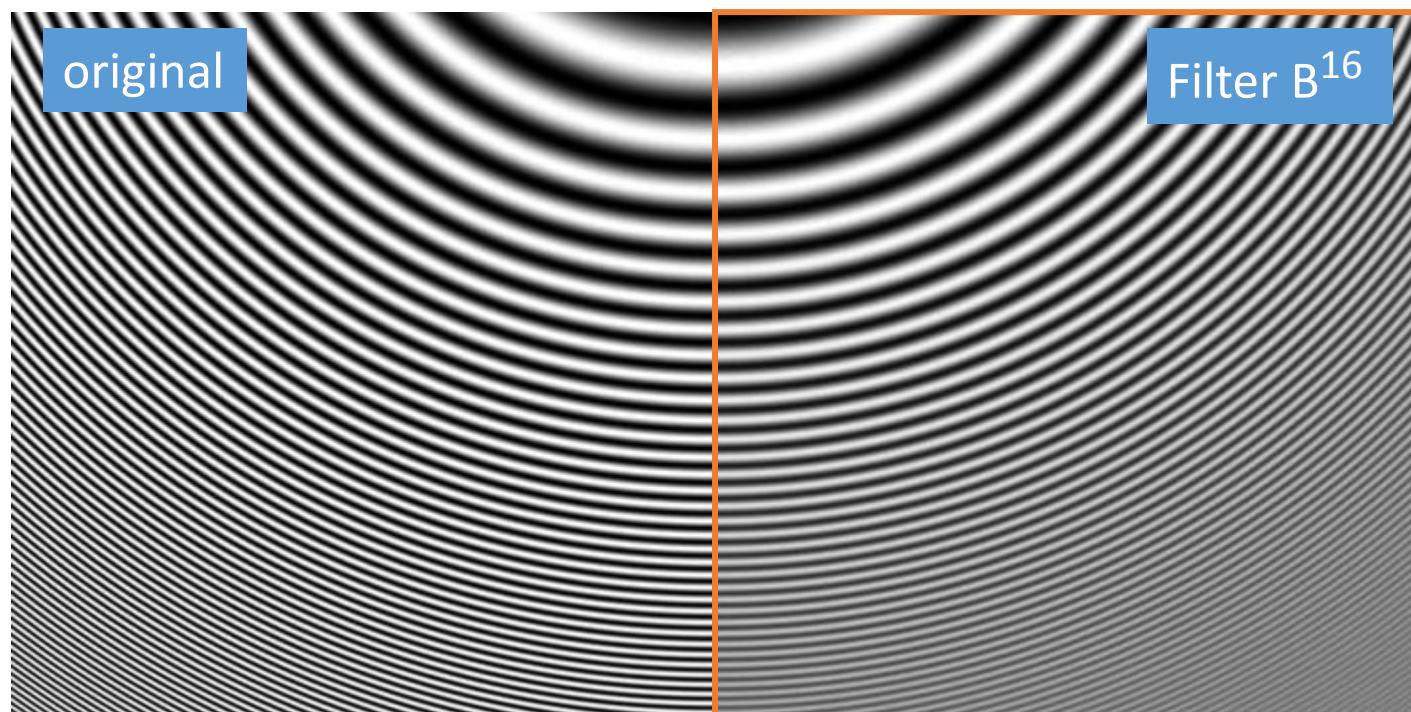
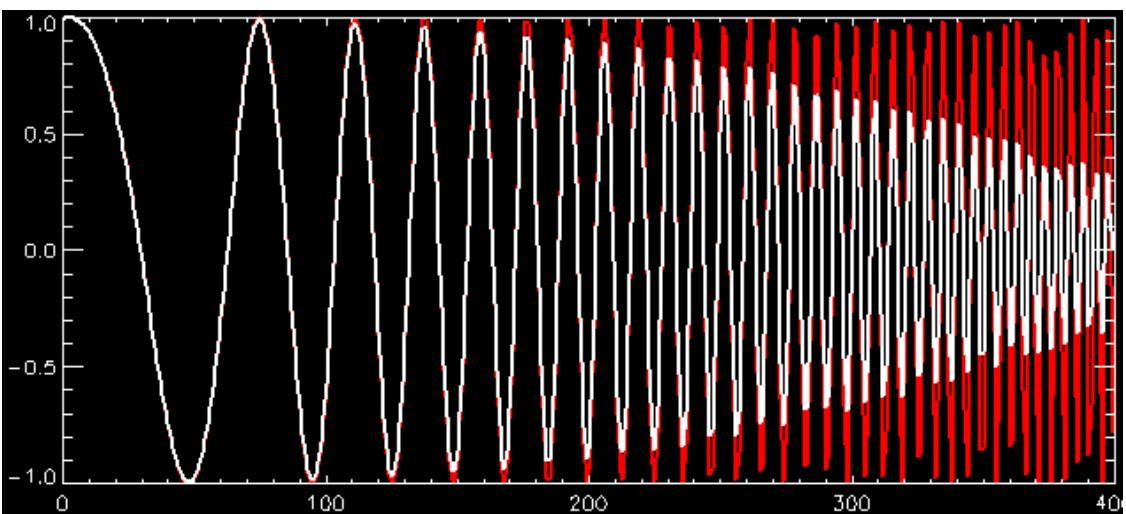
Weniger Artefakte an Kanten sind zu erwarten.



Filterresultate des Binomialfilters

Bildzeile

- **rot:** vor der Filterung
- **weiß:** nach Filterung



Butterworth-Filter vs. Binomialfilter

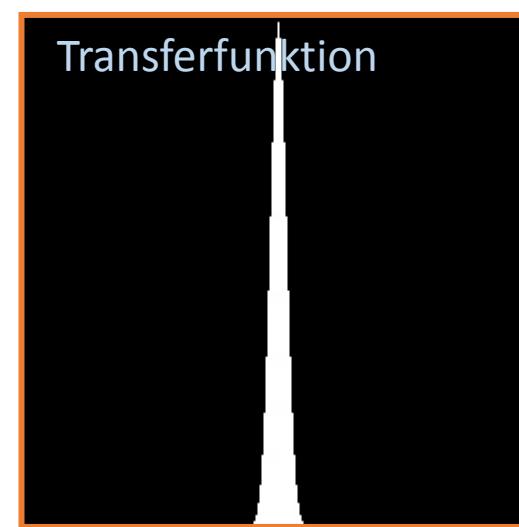
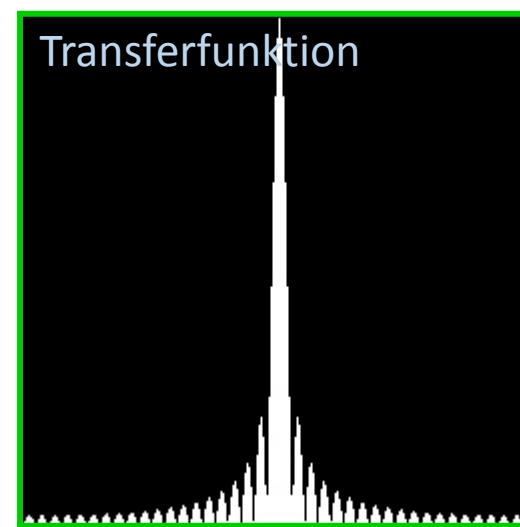
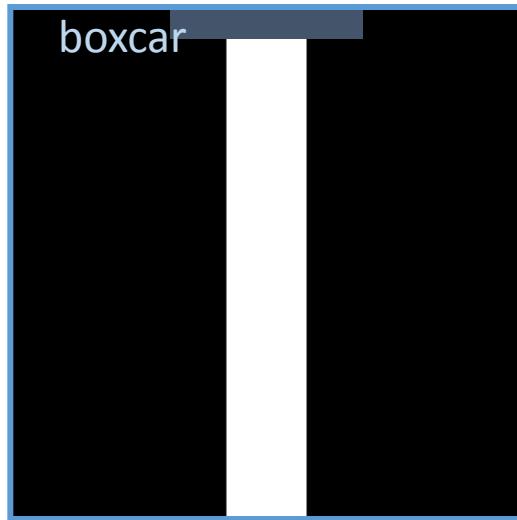
- Ideales Tiefpassfilter: kompakter Träger im Frequenzraum, aber artefakt-verursachende Ortsraumrepräsentation
- Butterworth-Filter: kontrolliert monoton fallende Funktion im Frequenzraum, deren Ortsraumrepräsentation ebenfalls monoton fällt.

- Mittelwertfilter: kompakter Träger im Ortsraum, aber artefakt-verursachende Frequenzraumrepräsentation
- Binomial-Filter: monoton fallende Funktion mit kompaktem Träger im Ortsraum, deren Frequenzraumrepräsentation monoton fällt.

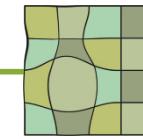
Binomialfilter und Gaußfunktion

Für größere Filterkerne nähert sich das Binomialfilter der [Gauß'schen Glockenkurve](#) an.

Der Betrag der [Transferfunktion](#) einer solchen Funktion ist wieder eine [Gauß'sche Glockenkurve](#).



$$\text{gauss}(x,y) = [\sigma \cdot \sqrt{2\pi}]^{-1} \cdot \exp[-(x^2+y^2)/(2\sigma^2)]$$



Filterung mit 2D Gaußfilter



=1.5



=3.0



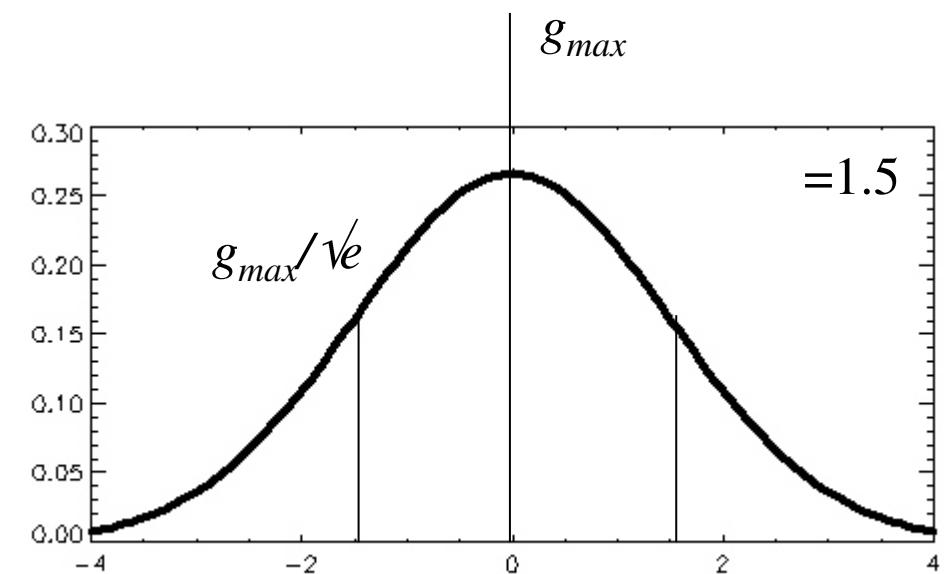
Gaußkurve und Gaussfilter

Funktion wird nie Null

- Filterkern endlicher Größe „schneidet“ die Funktion ab
- Kerngröße ca. $2 \times 3s + 1$
- ausgewählter Filterkern muss normiert werden

Separabilität macht Filterung effizient:

$$g(x, y) = g(x) * g(y)$$



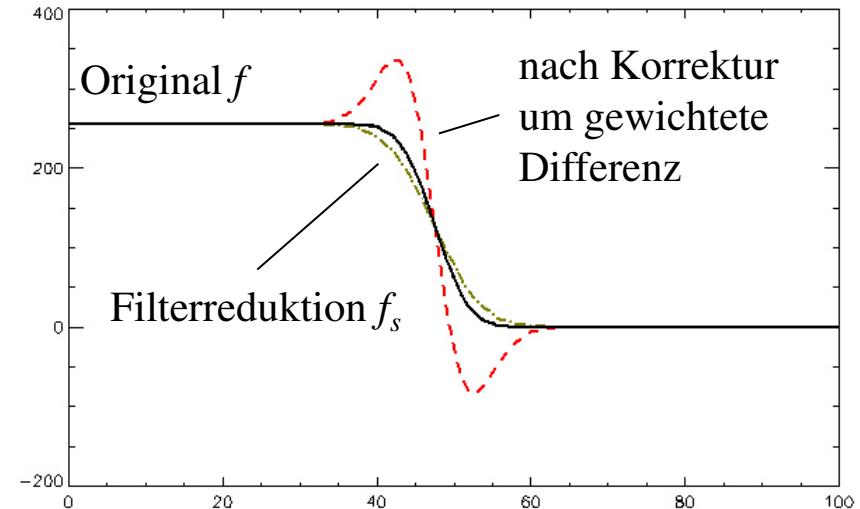
Glätten für Kanten: Unsharp Masking

- Gaußfilter verwenden, um Unschärfe abzuziehen.
- Idee:
 - Berechne Bild f_s durch Filterung von f mit Gaußfunktion mit Standardabweichung s
 - Addiere mit p gewichtete Differenz $f-f_s$ auf das Originalbild

$$f_{USM} = f + p(f - f_s)$$

(p wird oft in Prozent angegeben, d.h. 50% entspricht $p=0.5$)

- f_{USM} wird für f eingesetzt, falls der Unterschied zwischen f und f_s größer ist, als eine Schwelle t .

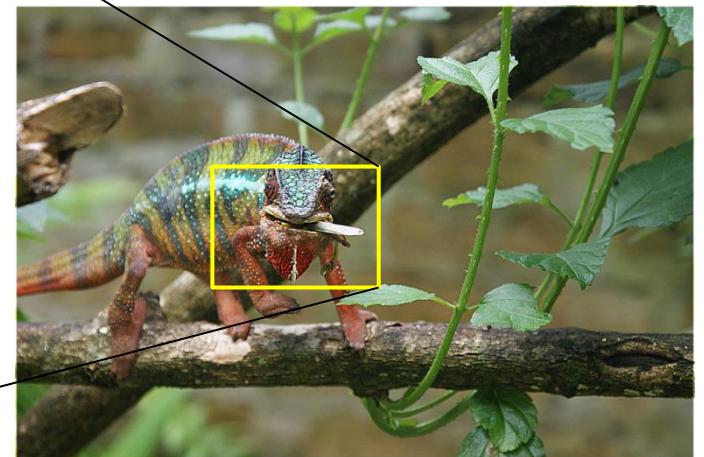


Unsharp Masking - Parameter

- **r (radius)**
 - je größer der Radius, desto breiter ist der verstärkende Rand an Kanten
 - Der Wert ist die Standardabweichung der Gaußfunktion, d.h. die Breite ist ca. 6-mal größer
- **p (amount)**
 - Je größer der Wert ist, desto höher ist die Verstärkung an den Rändern
- **t (threshold)**
 - Je höher der Wert ist, desto stärker muss die Kante sein, damit überhaupt ein Sharpening stattfindet
 - Jeder Wert $t>0$ führt dazu, dass das Filter nicht mehr linear ist (Artefakte sind möglich!)

Unsharp Masking ist keine Bildrestauration

- was unscharf war, bleibt auch unscharf
- nur bei Unschärfe nahe der Auflösungsgrenze führt der künstliche Machbandeffekt zu einem schärfer wahrgenommenen Bildeindruck



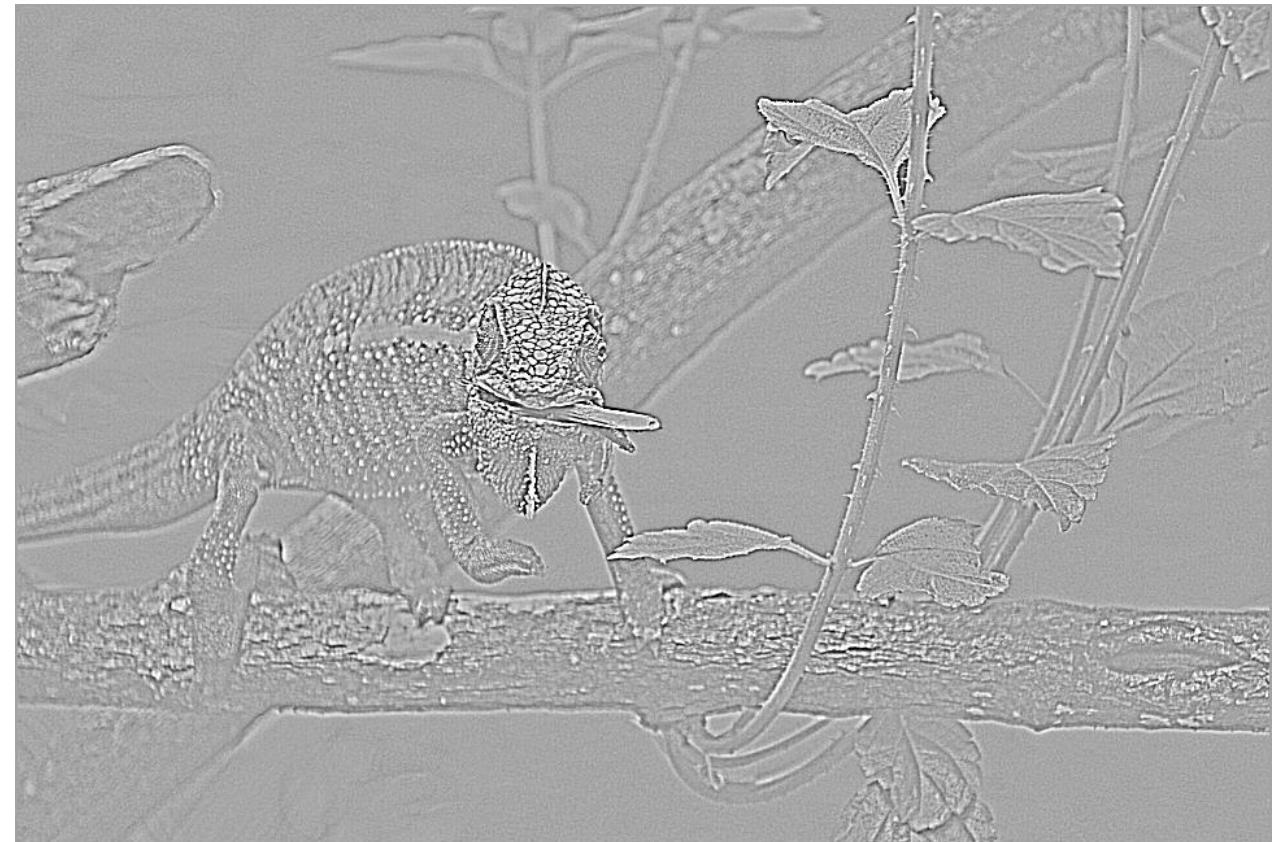
Kanten

- Rauschunterdrückung versagt an Kanten
- Kanten sind wesentliche Bestandteile der abgebildeten Information
- Kantenfilter
 - heben Kanteninformation gegenüber restlicher Bildinformation hervor
 - produzieren Kantenpixel, aber noch nicht Kantenzüge
 - verorten die Kantenpunkte im hochfrequenten Bereich des Frequenzspektrums



Kanten und USM

- USM erhöht die Bildschärfe durch Addition von $p(f-fs)$ \rightarrow Kanteninformation ist durch $f-fs$ gegeben.
- Resultat $f-fs$



Subtraktionsbild

Geglättetes Bild ist das Ergebnis einer Faltung des Originalbildes mit einem Tiefpassfilter s , also ist

...und im Frequenzraum

$$f - f_s = f - f * s_\sigma$$

...also ist das „kantenartige“ durch einen Hochpassfilter gegeben

$$F - F_s = F - F \cdot S_\sigma = F(1 - S_\sigma)$$

Erinnerung: Butterworth-Hochpassfilter



$$H(u, v) = \frac{1}{1 + (D_0 / D(u, v))^{2n}}$$

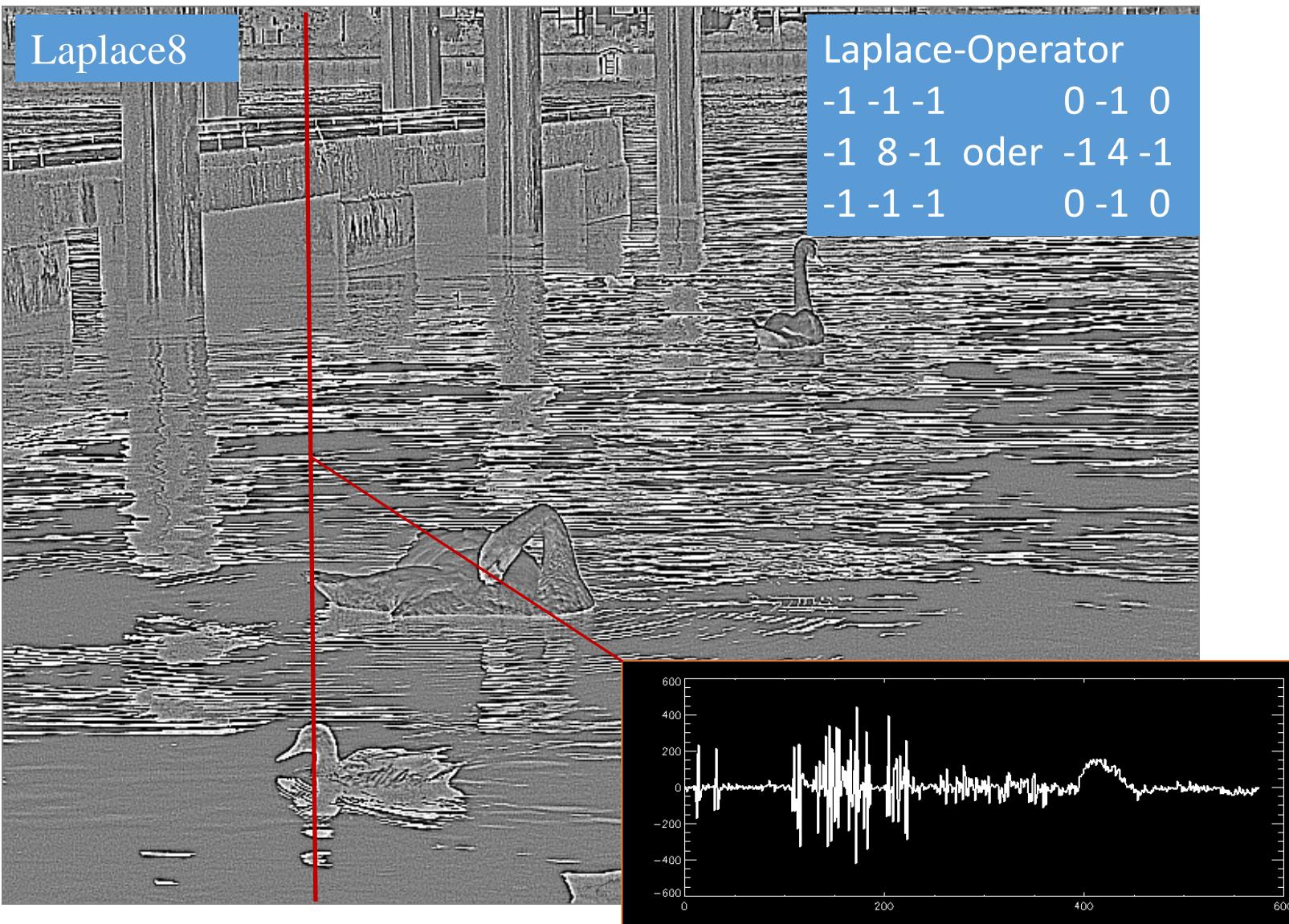


Hochpassfilter = Kantenfilter?

- nicht die ganze Wahrheit
 - Hochpassfilter entsprechen näherungsweise einem Laplace-Filter
 - Laplace-Filter: Summe der partiellen zweiten Ableitungen in x- und in y-Richtung
- Laplace-Filter und Kanten
 - Definieren den Ort einer Kante (durch Nulldurchgänge)
 - geben nicht Stärke und Richtung einer Kante an (so, wie Gradienten)

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 Laplacefunktion = zusammenhängende Kurven
- Approximation durch Kombination einer doppelten Differenzbildung in x- und y-Richtung,
 - z.B. Faltung eines Differenzoperators $[-1 \ 1]$ mit sich selbst: $[-1 \ 2 \ -1]$
 - ergibt für $[-1 \ 1]*[-1 \ 1] + [-1 \ 1]^T * [-1 \ 1]^T$:
$$\begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$
- Summe **aller** partiellen Ableitungen: $\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} + \frac{\partial^2 f(x, y)}{\partial x \partial y} + \frac{\partial^2 f(x, y)}{\partial y \partial x}$:
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



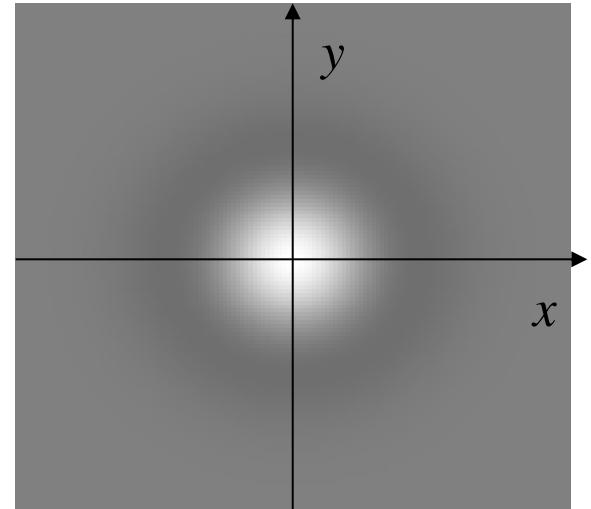
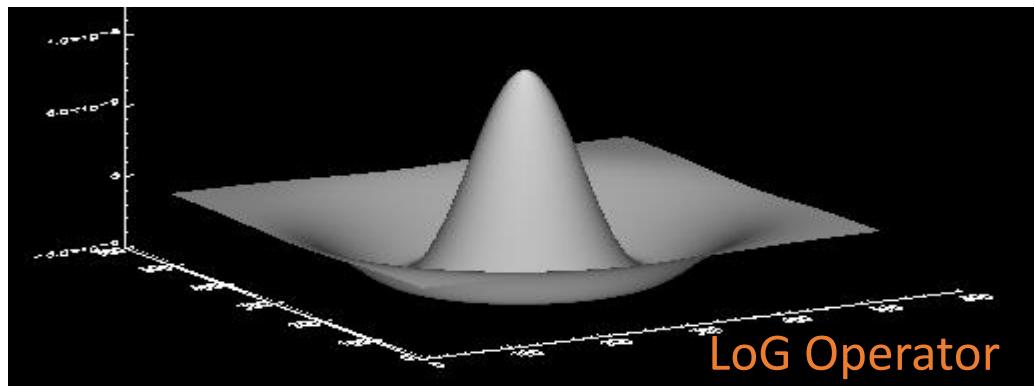
Nulldurchgänge



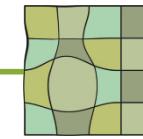
Marr-Hildreth-Filter = LoG-Filter

- Laplacian-of-Gaussian, d.h. **Laplaceoperator** auf die **Gaußfunktion** angewendet.

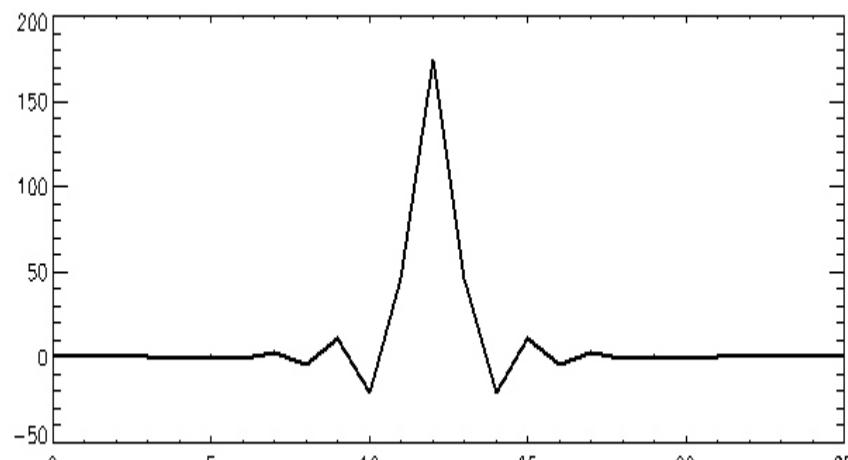
$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \text{ mit } f(x, y) = \frac{1}{\sqrt{2\pi}\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



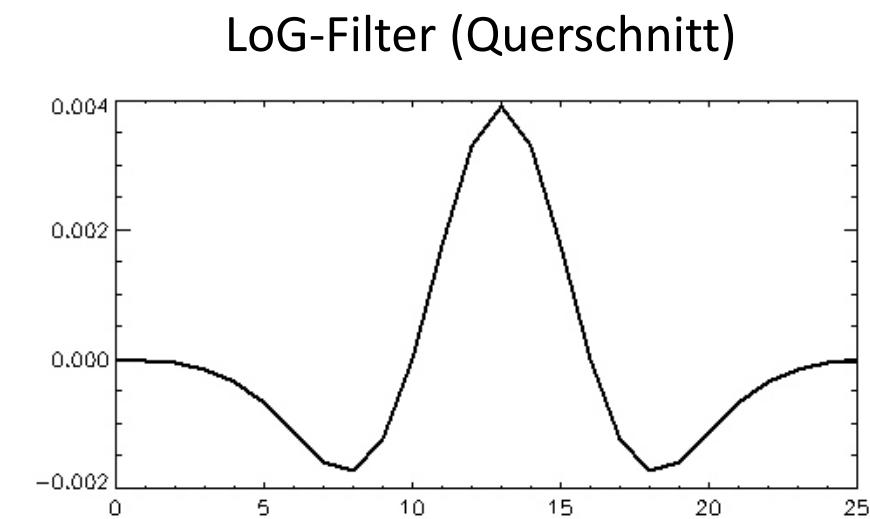
$$LoG(x, y) = \frac{1}{\pi\sigma^4} \left(1 - \frac{x^2 + y^2}{2\sigma^2} \right) \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$



Vergleich Butterworth-Hochpass und LoG im Ortsraum



Butterworth-Hochpassfilter (Querschnitt)



LoG-Filter (Querschnitt)

Was sollten Sie heute gelernt haben

- Rauschunterdrückung durch Lineare Filterung
- Artefakte bei der Filterung
- Standardfilter zur Rauschfilterung in Orts- und Frequenzraum
- Kanten: Unsharp Masking und Laplace-Operator

Famous Last Question



Wie könnte man linearen Filter benutzen, um (nahezu) randparallele Ecken (90° oder kleiner) zu finden?

Famous Last Question



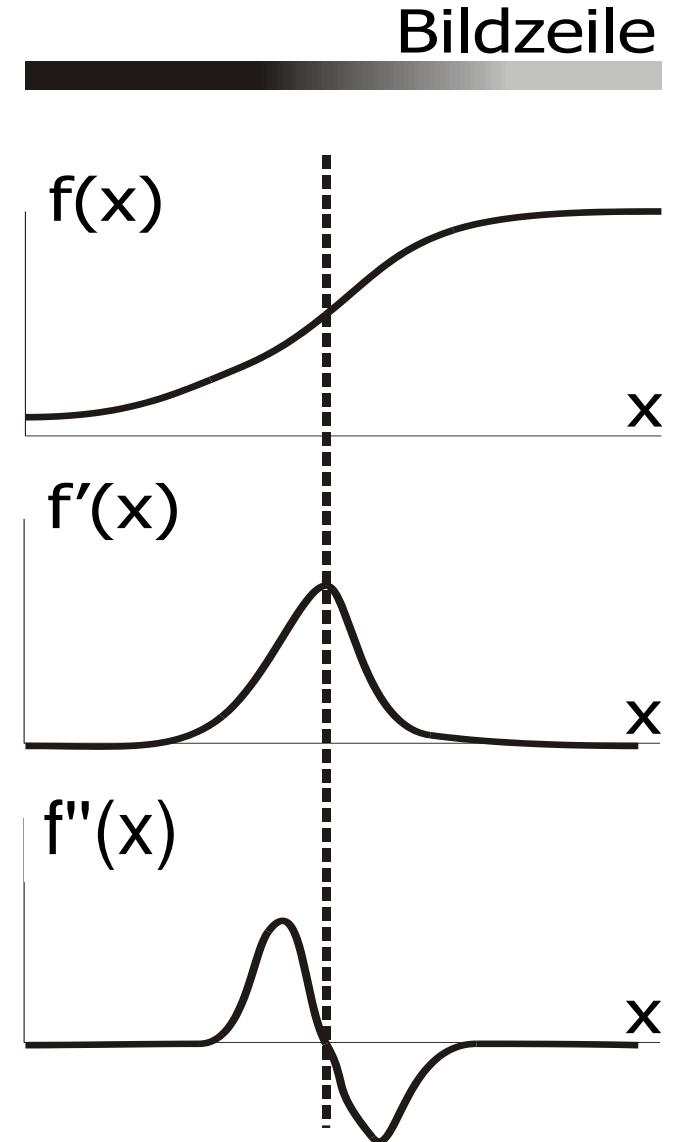
Wie könnte man linearen Filter benutzen, um (nahezu) randparallele Ecken (90° oder kleiner) zu finden?

Kanten und Rauschen

- Bestimmung von Kantenstärke und Kantenrichtung
 - Gradienten und Gradientenoperatoren
- Kantenerhaltende Rauschunterdrückung
 - Trennung von Rauschen und Kanten
 - Medianfilter
 - Varianten des Medianfilters

Kanten und Nulldurchgänge

- Vorzeichenwechsel ist leichter zu erkennen, als ein Minimum oder Maximum.
 - *Gradient* (Länge) als Kennzeichen für die Wichtigkeit einer Kante
 - zweite Ableitung für den Ort der Kante (Nulldurchgang)

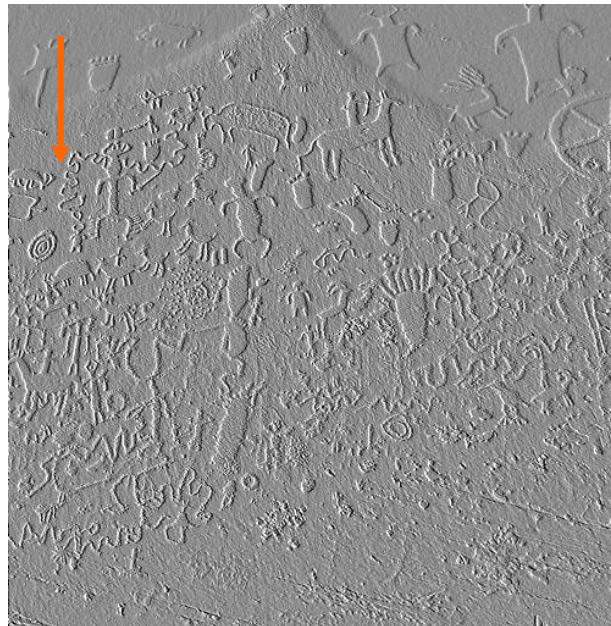


Erinnerung: Gradienten

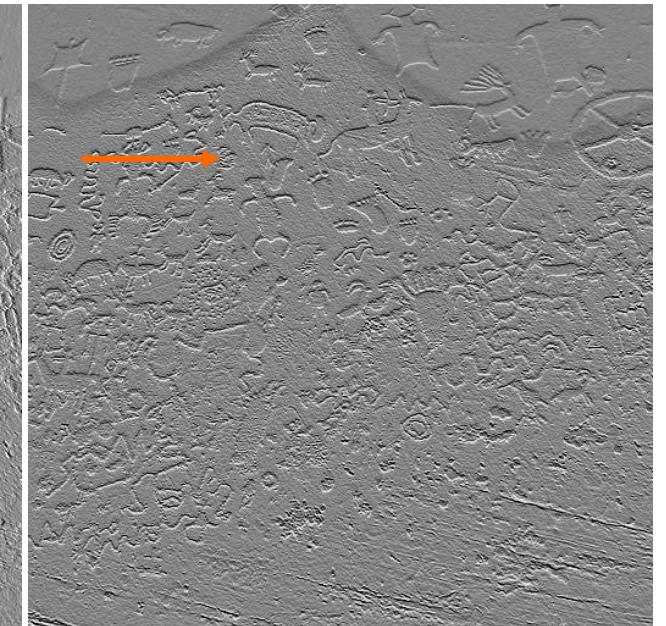


Pseudo-
3D
Eindruck:


Differenzbildung in n -Richtung



Differenzbildung in m -Richtung



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:

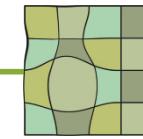
$$\vec{G}(m, n) = \begin{pmatrix} G_x(m, n) \\ G_y(m, n) \end{pmatrix}, \quad G_x(m, n) \approx [f * g_x](m, n), \quad G_y(m, n) \approx [f * g_y](m, n).$$

$$g_x = \begin{bmatrix} -1 & 1 \end{bmatrix} \quad g_y = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \quad g_{R1} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad g_{R2} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

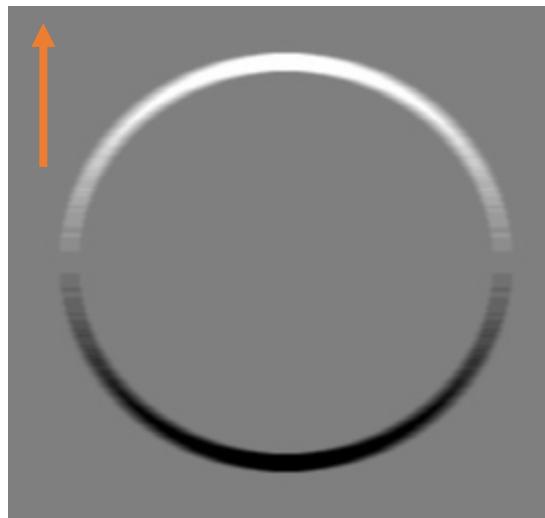
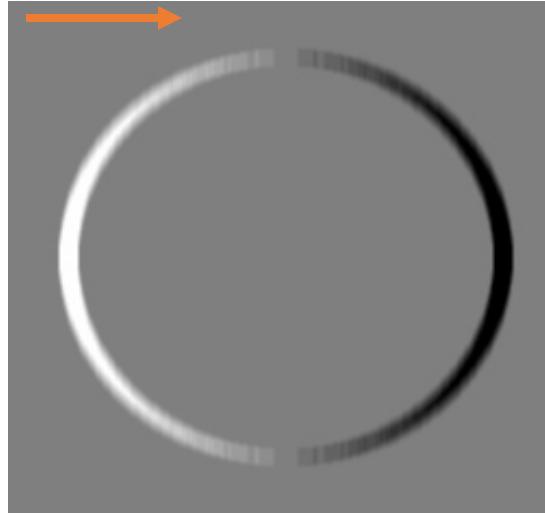
(Robert's Gradient)

$$g_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \quad g_y = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \quad (\text{symmetrischer Konvolutionskern})$$

Wieso kann man den
Gradienten entlang der
Diagonale bestimmen?



Elemente des Gradienten



$$\text{Betrag} : \sqrt{G_x^2 + G_y^2}$$

$$\text{Richtung: } \tan^{-1}(G_y/G_x)$$

Rauschen und Kanten



Hintergrundartefakte (Rauschen) wird verstärkt.

Rauschen und Kanten

Wie kommt es zur Verstärkung des Rauschens?

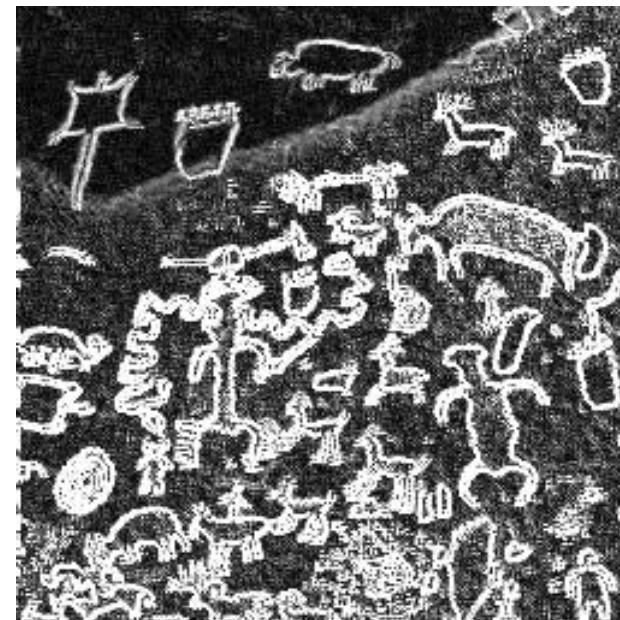
- Rauschen ist additiver Zufallsprozess mit Erwartungswert 0 und Standardabweichung σ
- Stärke des Rauschens ist gegeben durch die Größe der Standardabweichung
(Erinnerung: Definition des SNR)
- Subtraktion zweier Zufallsfunktionen f_1 und f_2 mit $E()=0$ und Standardabweichungen σ_1 und σ_2 ergibt eine neue Funktion f_{diff} mit $E(f_{diff})=0$ und $\sigma(f_{diff})=(\sigma_1 + \sigma_2)/2$
- Signal von f_{diff} ist in der Regel geringer als das von f_1 und f_2
- Abhilfe: Reduktion des Rauschens durch Glättung

Sobel Operator

- Glättung orthogonal zur Differenzierungsrichtung
- Glättung durch gewichtete Mittelung (ähnlich wie beim Binomialfilter)
- Faltungskerne:

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

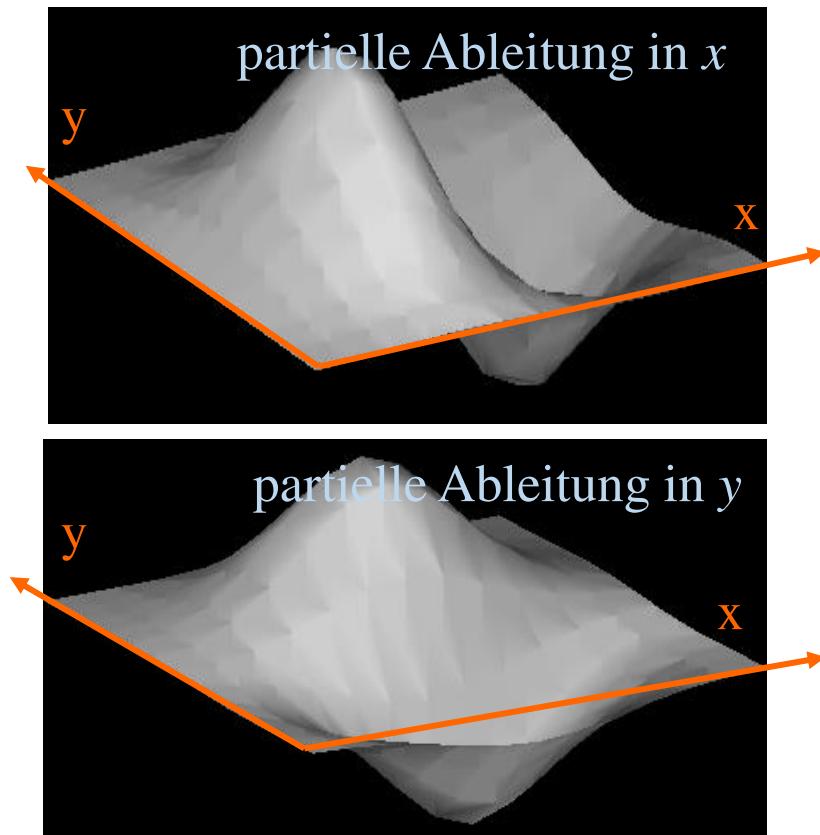
Einfache Differenz



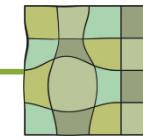
Sobel-Operator



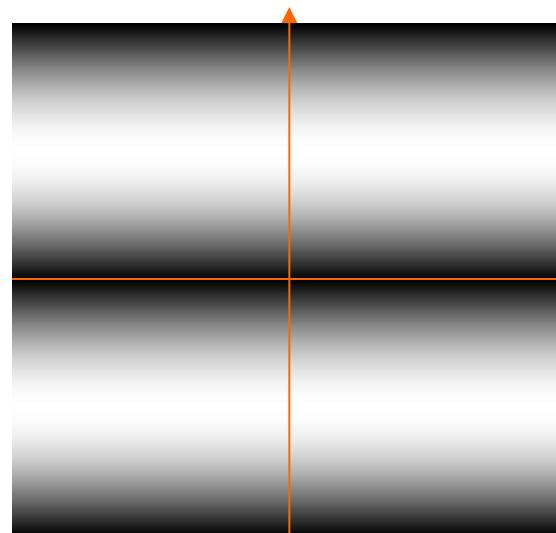
Ableitungen der Gaußfunktion



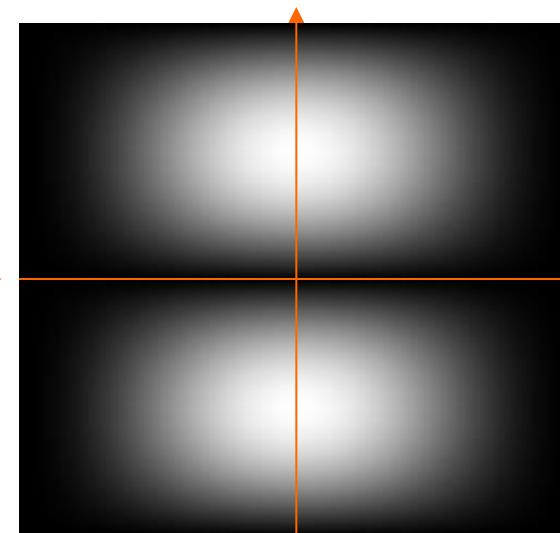
Wesentlich größerer Faltungskern
Größe (und Maß der Glättung) ist
abhängig von σ



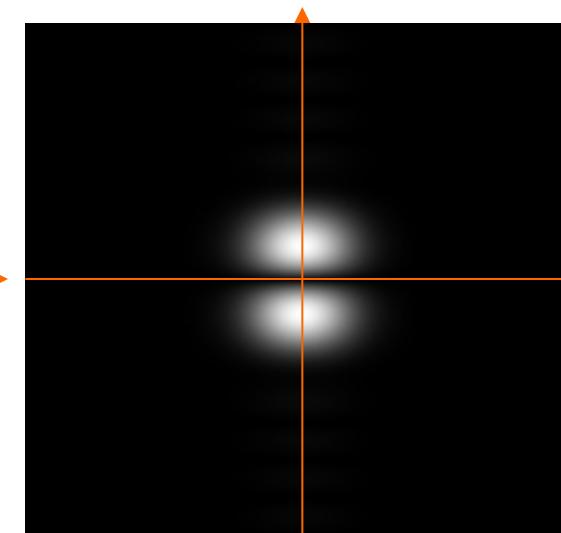
Vergleich der Differenzfilter im Frequenzbereich



Einfache Differenz



Sobelfilter



Approximation der
ersten Ableitung der Gaußfunktion

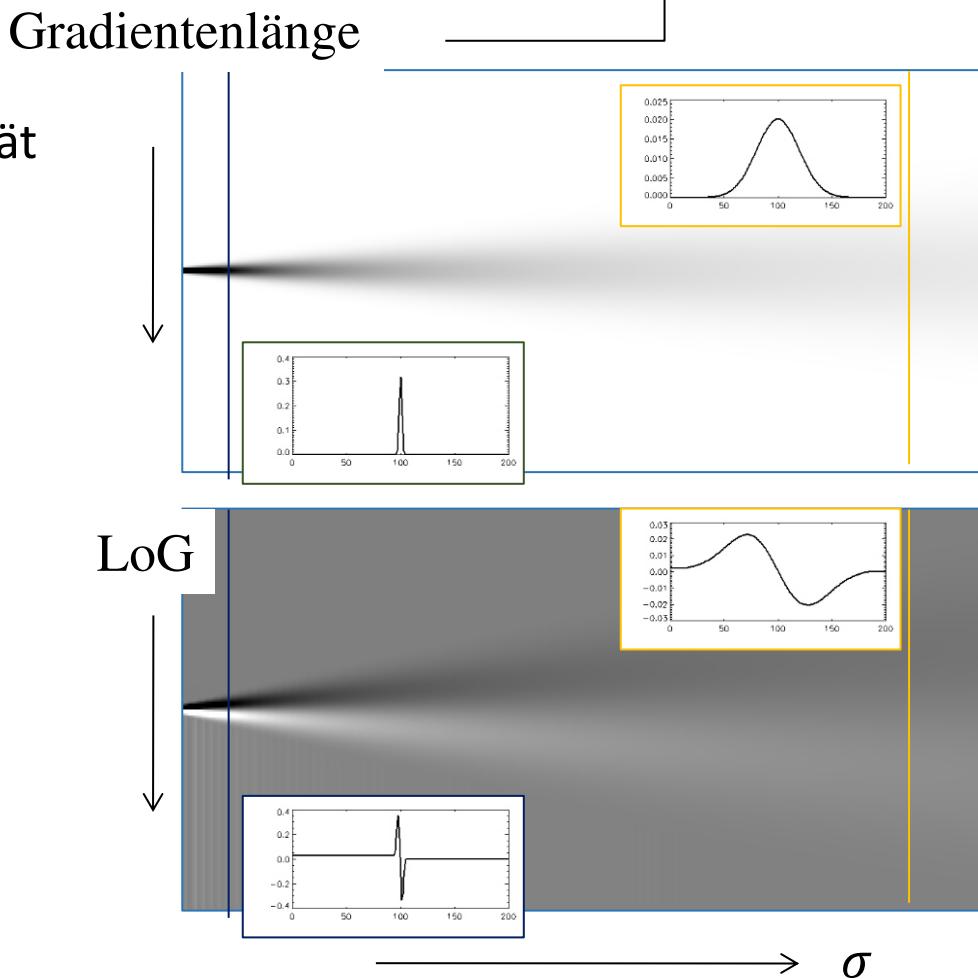
Gradient und Laplace-Operator

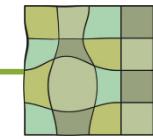
- Gradient: Stärke und Richtung einer Kante
- Laplace-Operator: Kantenorte an den Nulldurchgängen
- Kombination
 - Berechnung von Gradient und Laplace-Faltung
 - Gradienteninformation nur an den Nulldurchgängen erhalten (eine Art von Non-Maximum-Supression)



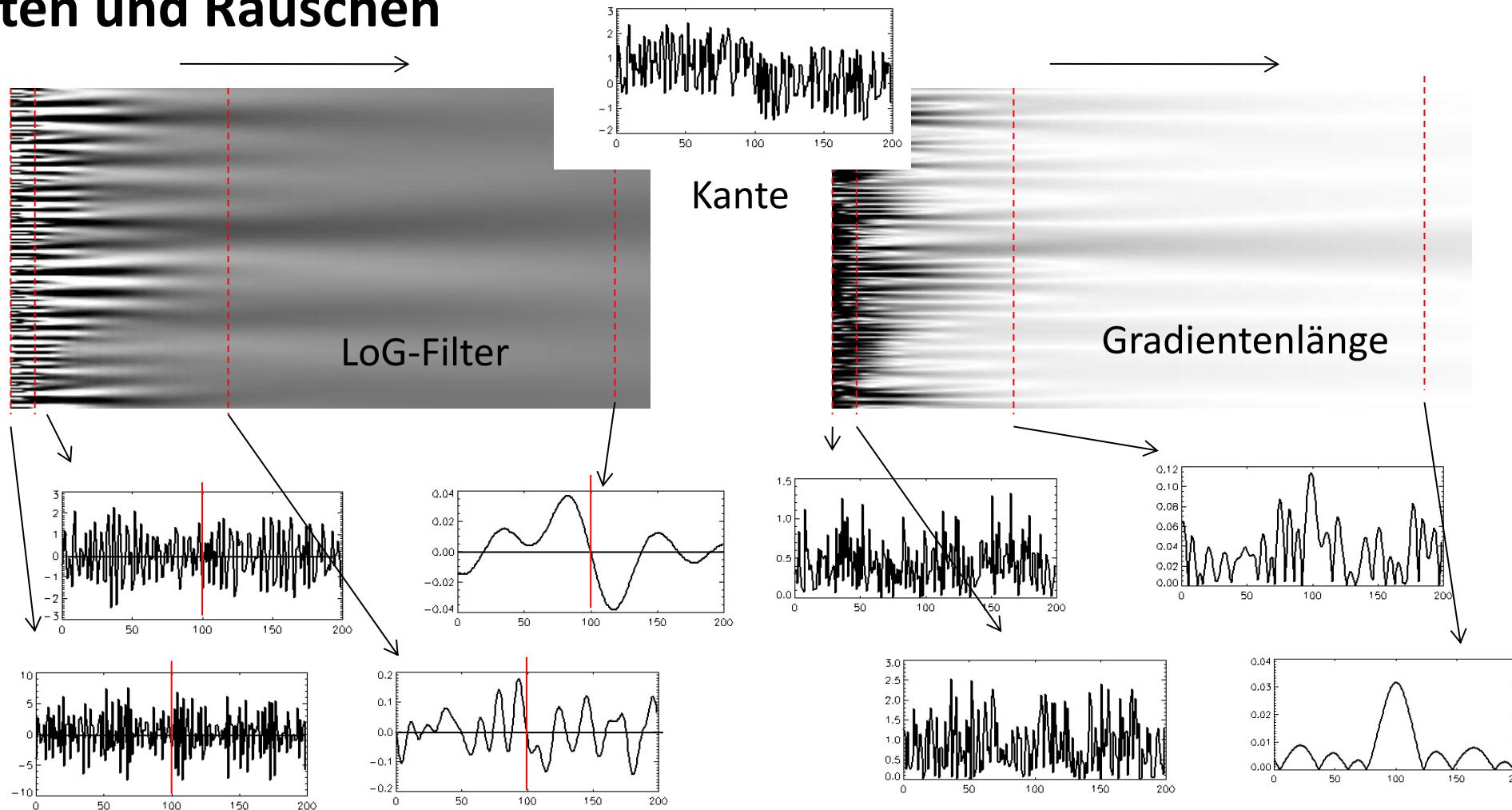
Kantendetektion

- Annahmen
 - Nachbarschaft um die Kante hat lokal konstante Intensität
 - Rauschen hat Erwartungswert 0
- dann
 - Glättung so, dass Kantensignal das Rauschen überwiegt
 - Kantenselektion durch Laplace-Operator
 - Unschärfe stört bei der Kantenbestimmung nicht



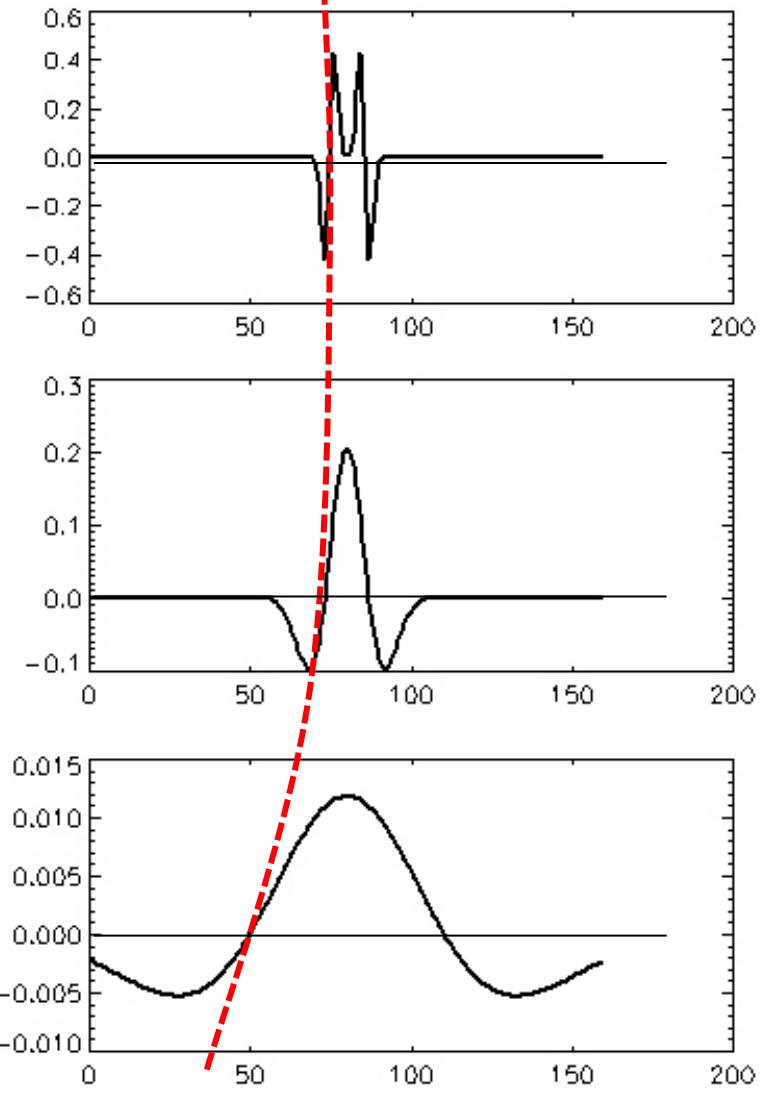
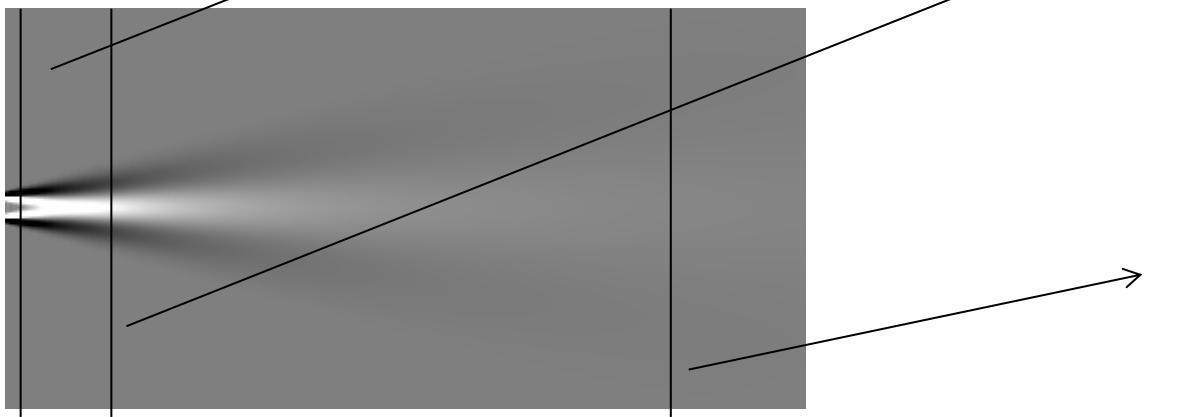


Kanten und Rauschen



Mehr als eine Kante

- Ungenaue Kantenlokalisierung bei Kantenabstand, der geringer als die Größe des Faltungskerns ist.
- Umfang der Glättung ist ein *Kompromiss* zwischen Rauschunterdrückung und Detailierungsgrad



Kanten und Rauschunterdrückung

- Grenzen einer Kantenbestimmung durch Glättung + Differenzoperatoren
 - Zu verrausches und/oder zu detailreiches Bild: Es gibt keinen Kompromiss für die Größe des Faltungskerns
 - Kanten sollen im Bild verschärft werden (ähnlich wie Unsharp Masking): Unschärfe wirkt auf den Betrachter unangenehmer als das Bildrauschen
- Alternative
 - Kantenerhaltende Rauschunterdrückung
 - Setzt voraus, dass Unterschiede zwischen Merkmalen von Kanten und von Rauschen herausgearbeitet werden

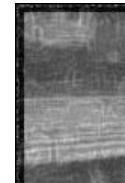
Strategie I: Kombination von Linearen Filtern

- Rauschunterdrückung, gefolgt von Kantenverstärkung
 - Rauschunterdrückung reduziert Amplituden in hohem Frequenzbereich
 - Kantenverstärkung verstärkt mittleren Frequenzbereich
 - Niedriger Frequenzbereich bleibt unverändert
- Beispiel: Unsharp Masking
- Resultat ist heuristisch und produziert „künstlerische“ Effekte vom Originalbilds

Beispiel



Glättung mit Gaussfilter



Glättung mit Gaussfilter
+ Gradientenlänge



Strategie II: Charakteristik der Grauwertverteilung

- Rauschen und Kanten haben im Frequenzbereich ähnliche Attribute.
- Ist ein nichtlineares Filter denkbar, dass für Rauschen und Kanteneigenschaften unterschiedlich sensitiv ist?
 - Rauschen sind räumlich gleichverteilte Grauwertvariationen.
 - Grauwertvariationen an Kanten sind nicht räumlich gleichverteilt.
 - Filter muss diesen Unterschied berücksichtigen.

Rangordnungsfilter

Vorgehen:

- Sortierung der Elemente in einer Filtermaske
- Auswahl des an einer bestimmten Stelle eingesortierten Werts
- Eintragung des ausgewählten Werts in die zentrale Position

Eigenschaften:

- Es entstehen keine neuen Werte
- Filter ist nichtlinear, nicht kommutativ, nicht assoziativ

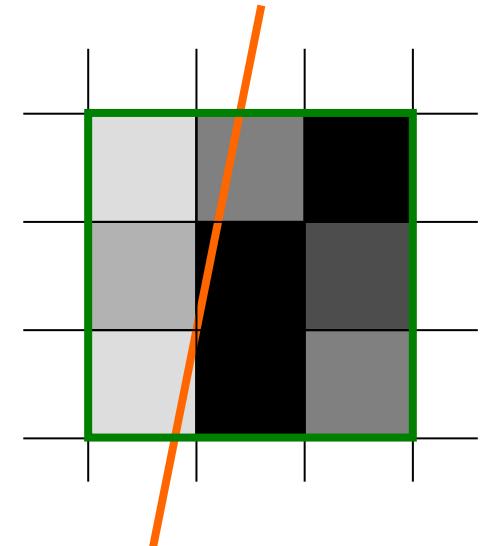
Gebräuchlichstes Rangordnungsfilter
ist das Medianfilter

26 3.	132 8.	112 5.
25 2.	102 4.	142 9.
17 1.	122 7.	117 6.

- erster Rang
- mittlerer Rang (Median)
- letzter Rang

Medianfilter

- Annahmen:
 - Medianfilter ist quadratisch mit ungerader Seitenlänge.
 - Kantensignal ist größer als das Rauschsignal
 - Kante im Filterbereich verläuft in diesem Bereich (nahezu) gerade.
 - Grauwert ist (nahezu) konstant in einer Umgebung von Größe des Filters.
- Kante verläuft durch Filterbereich:
 - Der Mittelpunkt des Filters liegt auf der gleichen Seite der Kante wie die Mehrzahl der Pixel (z.B. rechts von der Kante).
 - Pixel von dieser Seite (z.B. rechts) wird selektiert ([kantenerhaltend](#)).
- Keine Kante im Filterbereich:
 - Median nähert sich dem Erwartungswert mit Anzahl der Stichproben ([rauschunterdrückend](#)).



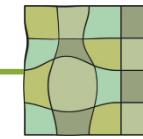
Medianfilter und Impulsrauschen

- Impulsrauschen:
 - Wenige Pixel sind gestört
 - Gestörte Pixel sind entweder maximal positiv oder negativ
 - Wird auch „Salt-and-Pepper-Noise“ genannt
- Glättung von Impulsrauschen
 - Zu große Nachbarschaft notwendig, um Erwartungswert zu schätzen



Medianfilter zur Entfernung von Salt & Pepper Rauschen (5 x 5)





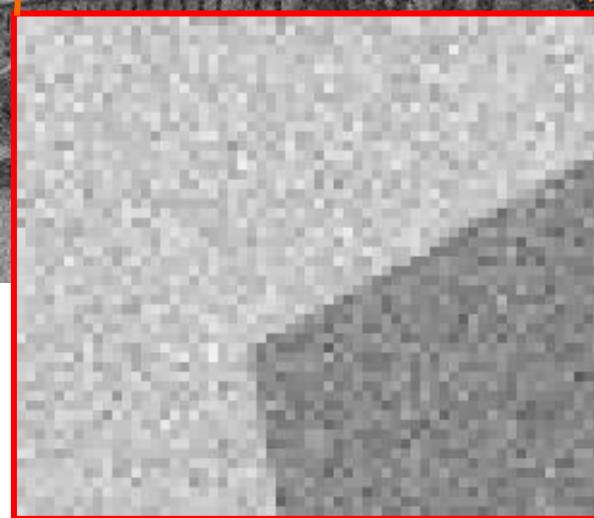
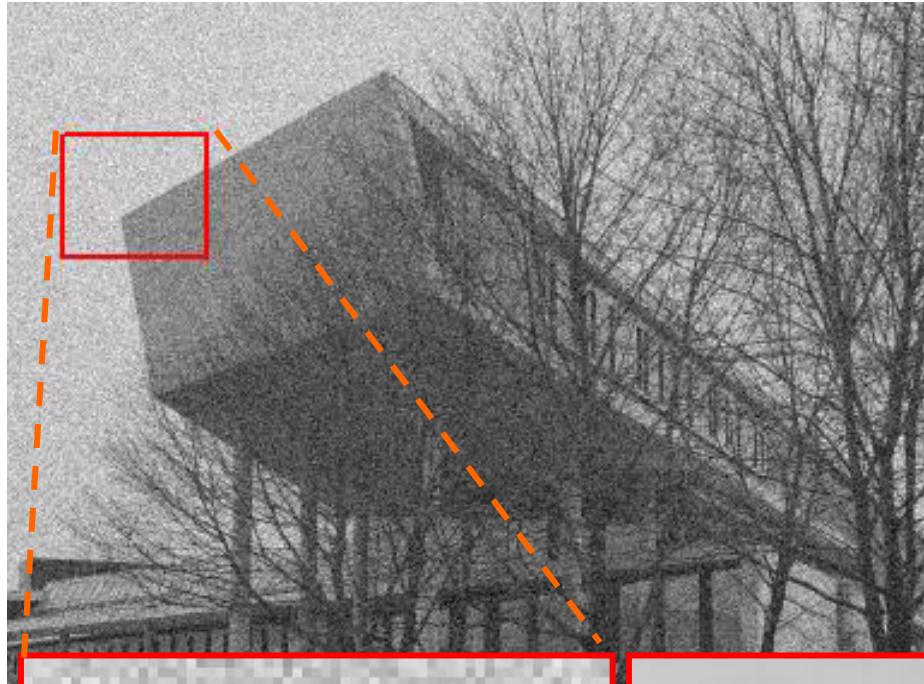
Median- vs. Mittelwertfilter



Differenz zum Original



Medianfilter in verrauschten Bildern



Mittelwert 7x7



Median 7x7

Kanten bleiben
scharf



Artefakt

Varianten: Gewichtetes Median Filter

- Isotrope, mit Abstand zum Zentrum abnehmende Gewichtung
- (Indirekte) Berücksichtigung des Umstands, dass Korrelation zwischen Pixeln mit der Entfernung abnimmt.
- Beispiel

1	2	1
2	3	2
1	2	1

Gewichtung

1	3	1
1	6	6
1	6	7

Grauwerte

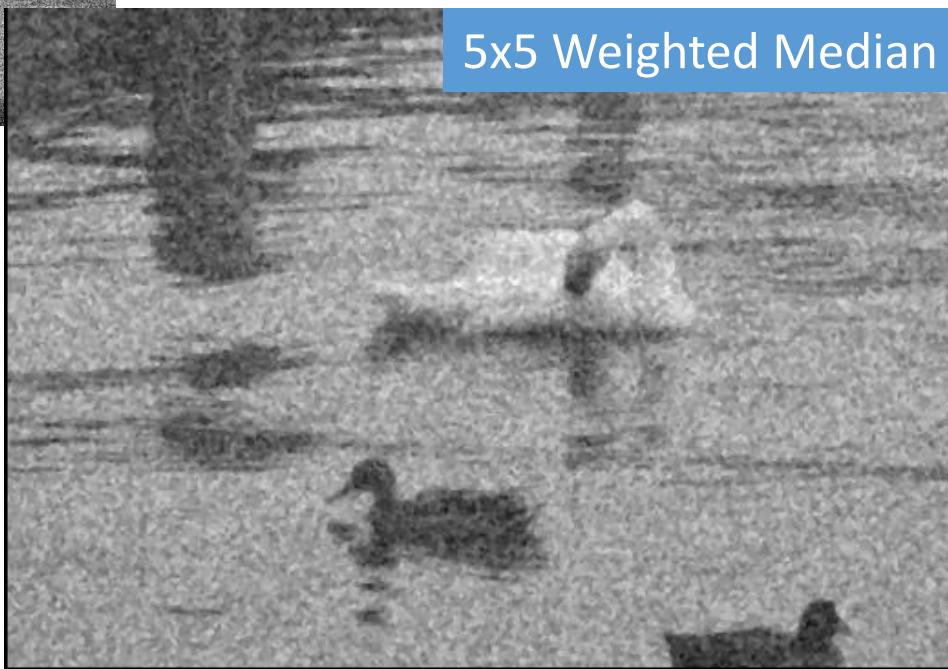
ohne Gewichtung

1,1,1,1,3,6,6,6,7

mit Gewichtung

1,1,1,1,1,1,3,3,6,6,6,6,6,6,7

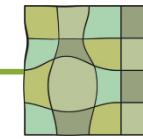
Beispiel



5x5 Weighted Median

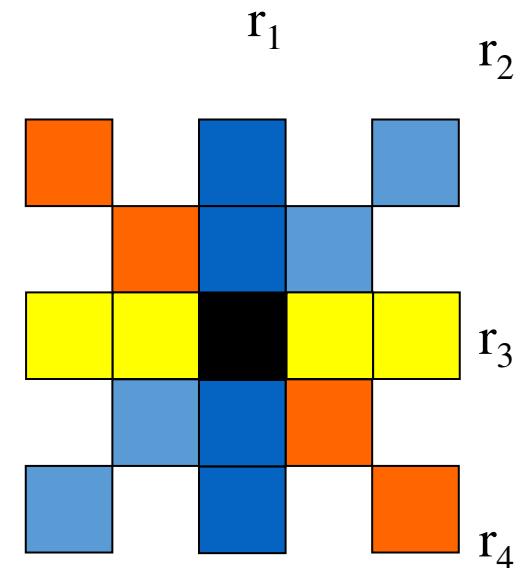


5x5 Median



Varianten: Anisotropes Medianfilter

- Zerlegung in verschiedene Richtungen
- Medianfilterung auf allen Richtungen
- Berechnung des Medians aller Mediane
 - $m_i = \text{Median}(r_i)$
 - $M = \text{Median}(m_1, m_2, m_3, m_4)$
- Besserer Erhalt von Bilddetails nach Medianfilterung



Beispiel



Median 11x11

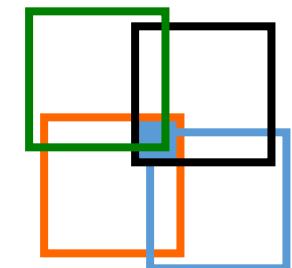


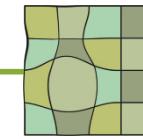
Anisotroper Median 11x11



Kombination der Medianfilterstrategie mit Rauschunterdrückung

- Für jedes Pixel
 - Bestimme eine Reihe von Regionen, die das Pixel enthalten
 - Bestimme Durchschnittswert und Varianz jeder Region
 - Ersetze den Pixelwert durch den Durchschnittswert derjenigen Region, für die die Varianz am geringsten ist
- Einfaches und schnelles Verfahren aus der Frühzeit der Bildverarbeitung
 - setzt voraus, dass die Regionen kleiner sind als die Bilddetails
 - Annahme: Kantenregionen haben höhere Varianz



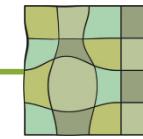


Beispiel



Kleine Subregionen





Beispiel



Zu große Subregionen



Fortgeschritte Methoden

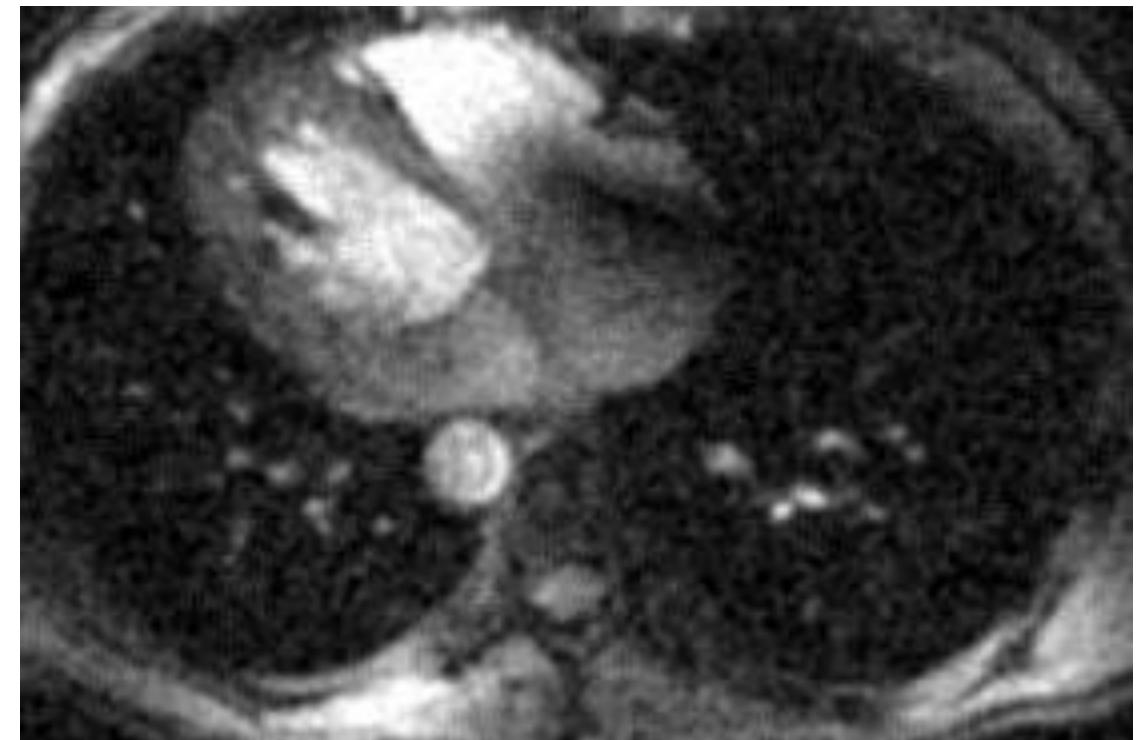
- Mean Shift Smoothing
 - Jedes Pixel wird durch Ort (x,y) und Grauwert in 3-d Raum repräsentiert
 - Pixel wird mit Grauwert näher zum Clusterzentrum geschoben
- Anisotrope Diffusion
 - Bildhelligkeit wird als Konzentration einer diffundierenden Flüssigkeit betrachtet
 - Diffusion an Kanten wird eingeschränkt
 - Filterresultat ergibt sich aus Simulation der Diffusion über die Zeit
- Bayesian Smoothing
 - Bildhelligkeit wird als Resultat eines stochastischen Prozesses betrachtet, bei dem die Bildhelligkeit zufällig von der unbekannten, tatsächlichen Helligkeit variiert
 - A-Priori-Wissen: benachbarte Pixel haben wahrscheinlich ähnliche Grauwerte
 - Suche nach der wahrscheinlichsten Ursache des wahrgenommenen Bildes

Was sollten Sie heute gelernt haben?

- Gradientenberechnung durch Faltung
 - Filtervarianten
 - Kombination mit Laplacefilter
- Kantenerhaltende Rauschfilterung:
 - Medianfilter
 - Varianten des Medianfilters

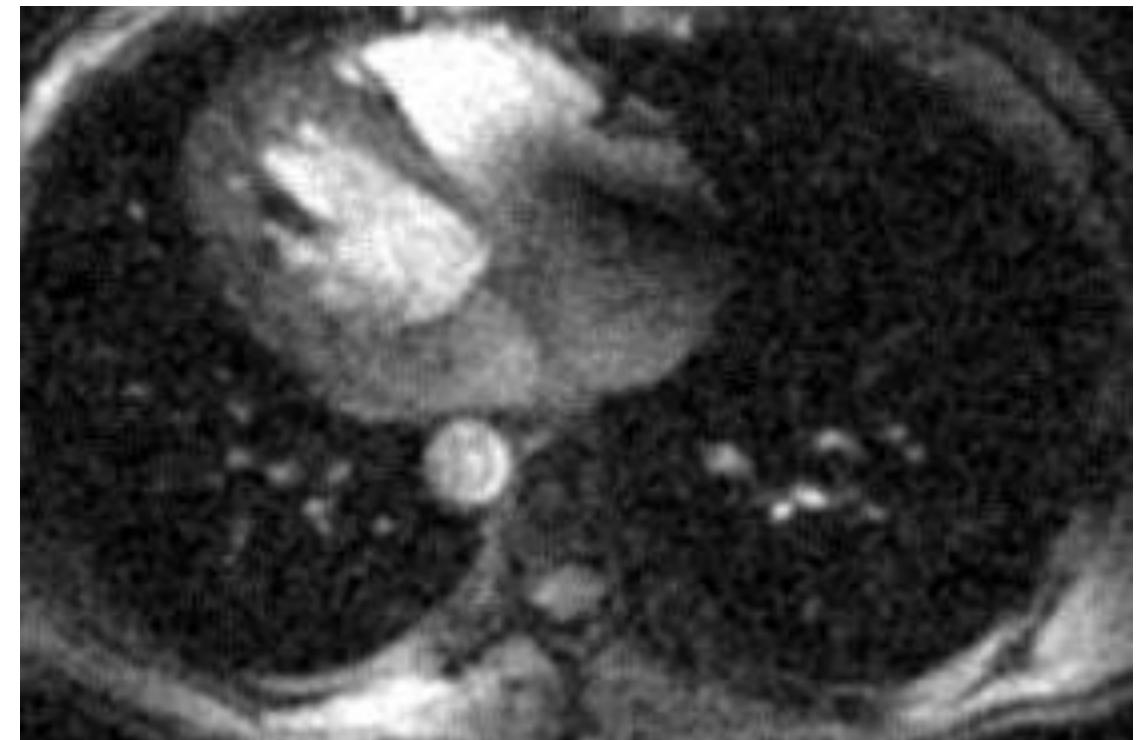
Famous Last Question

Wie könnte eine kantenerhaltene Rauschunterdrückung aussehen, die ausnutzt, dass Kanten ihre Richtung nur langsam ändern?



Famous Last Question

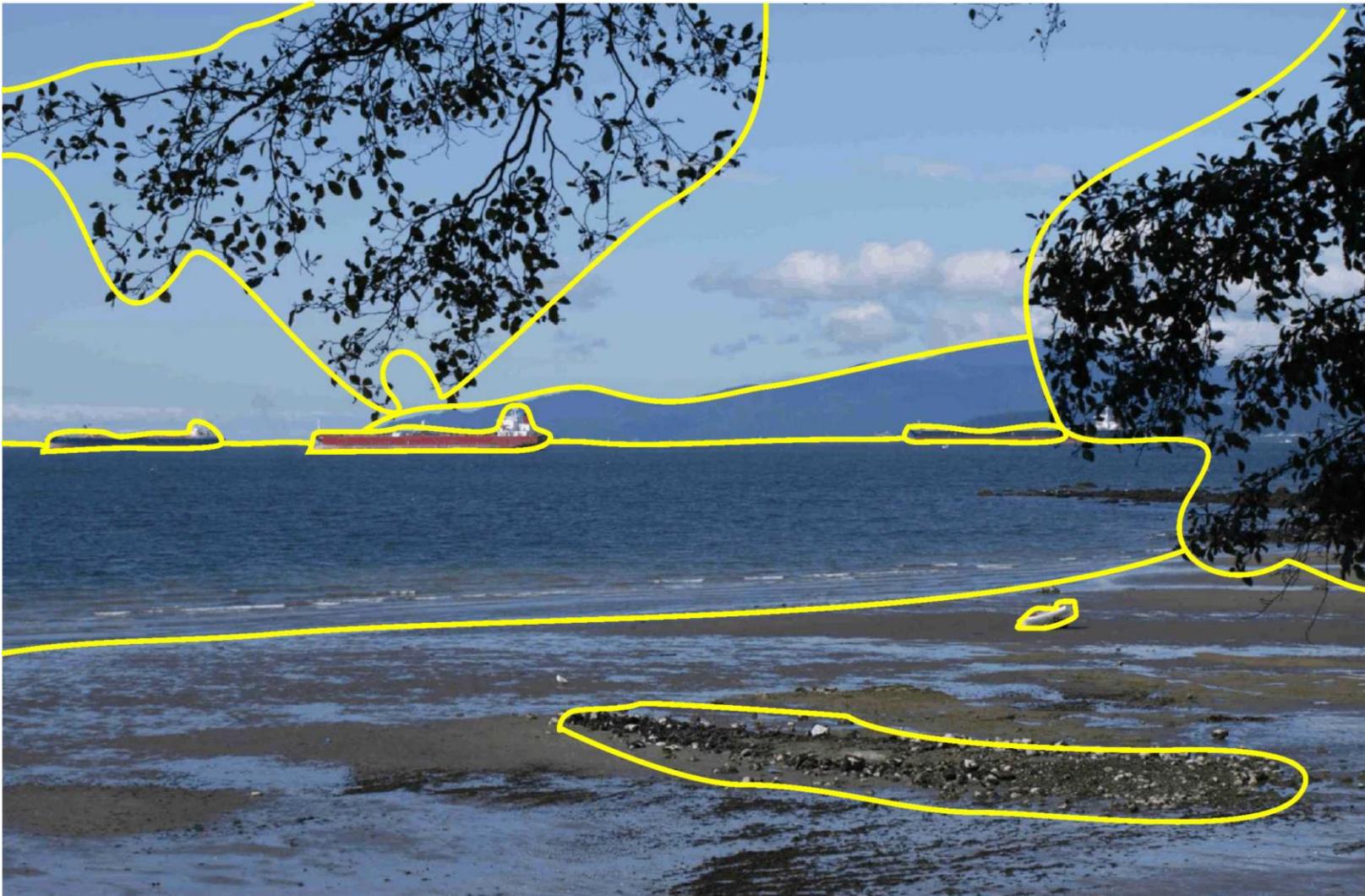
Wie könnte eine kantenerhaltene Rauschunterdrückung aussehen, die ausnutzt, dass Kanten ihre Richtung nur langsam ändern?



Segmentierung



Segmentierung

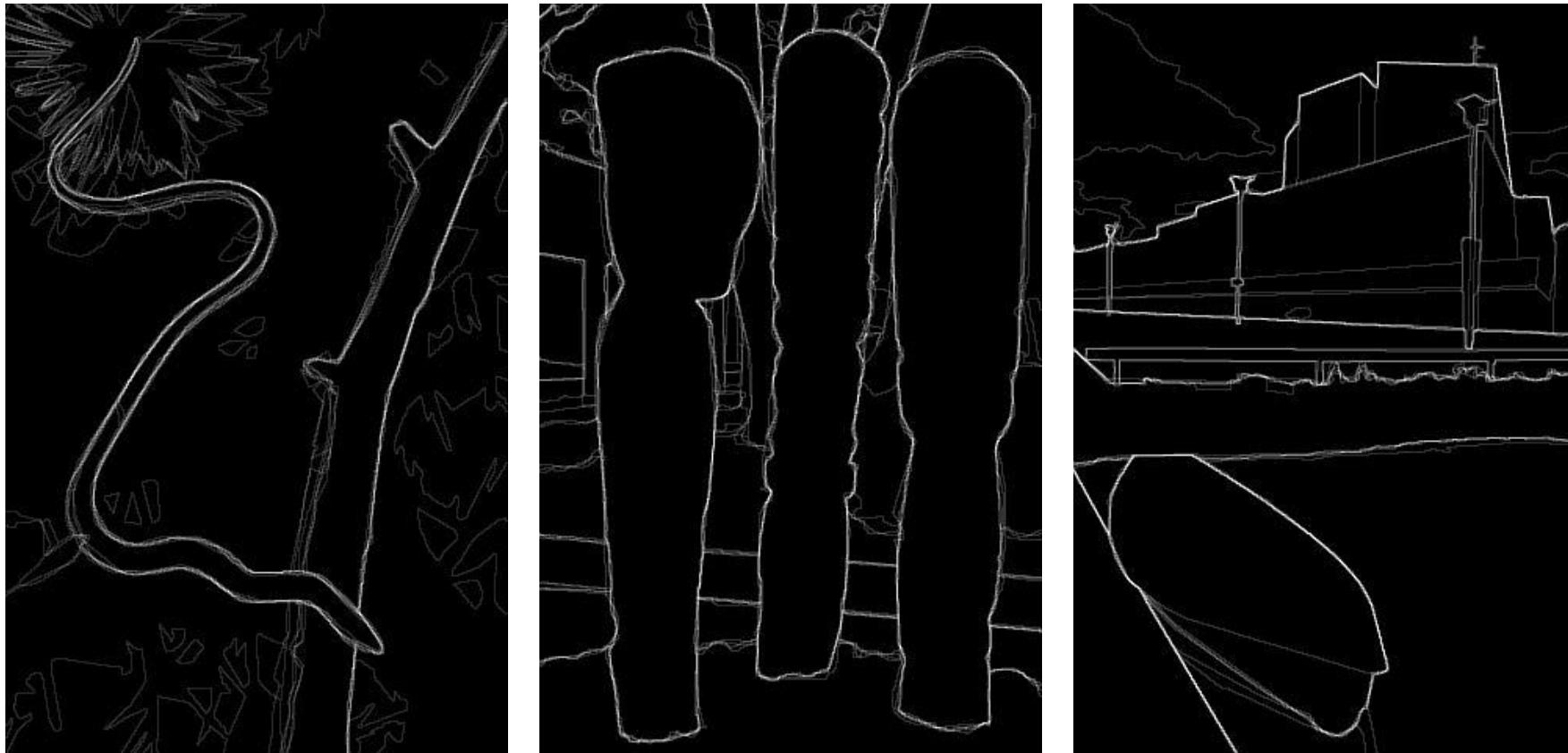


Gibt es die „ideale Segmentierung“?



Beispielbilder aus dem Berkeley Segmentation Dataset

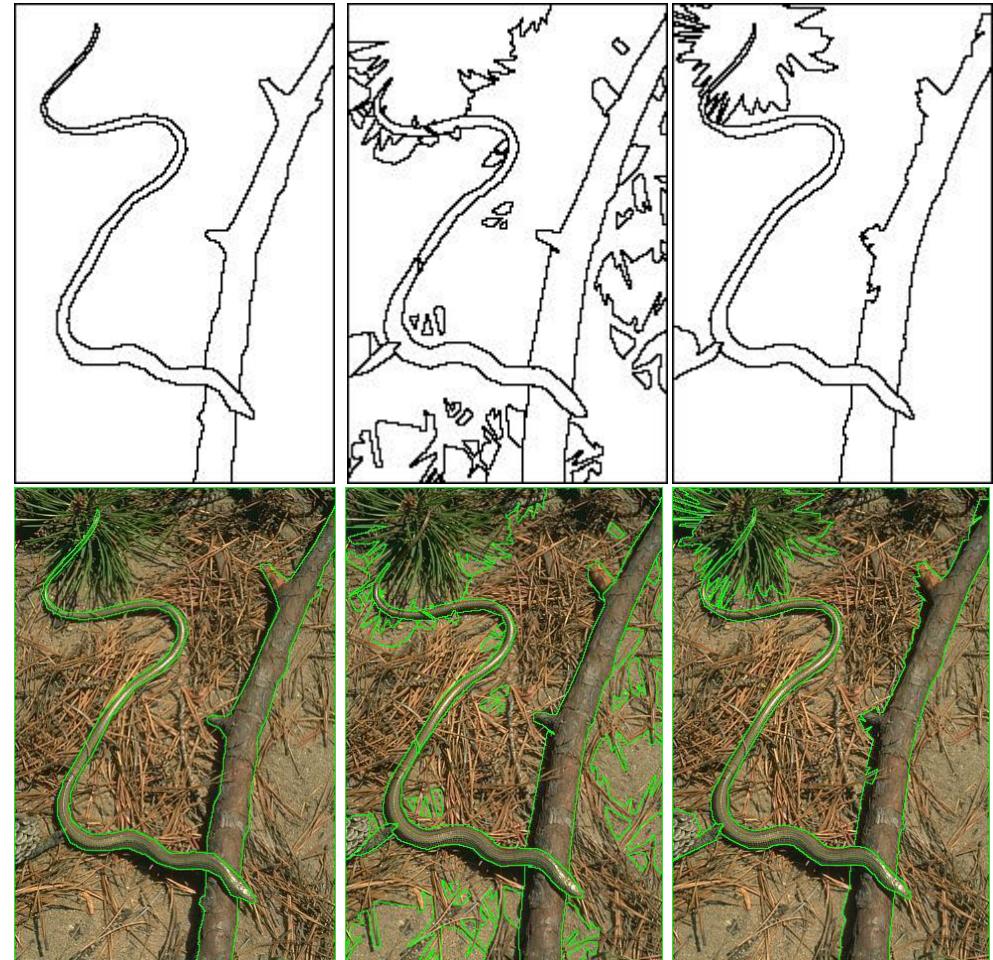
Referenzsegmentierungen



Helligkeit = Häufigkeit, mit der die betreffende Segmentgrenze gewählt wurde

Referenzsegmentierungen

- Es existieren zwar meist Segmente, die jeder als solche bezeichnet, aber
 - Was Hintergrund ist, wird oft unterschiedlich gesehen
 - Segmente selbst werden unterschiedlich detailliert gesehen
- Segmentierung ist nicht a-priori lösbar

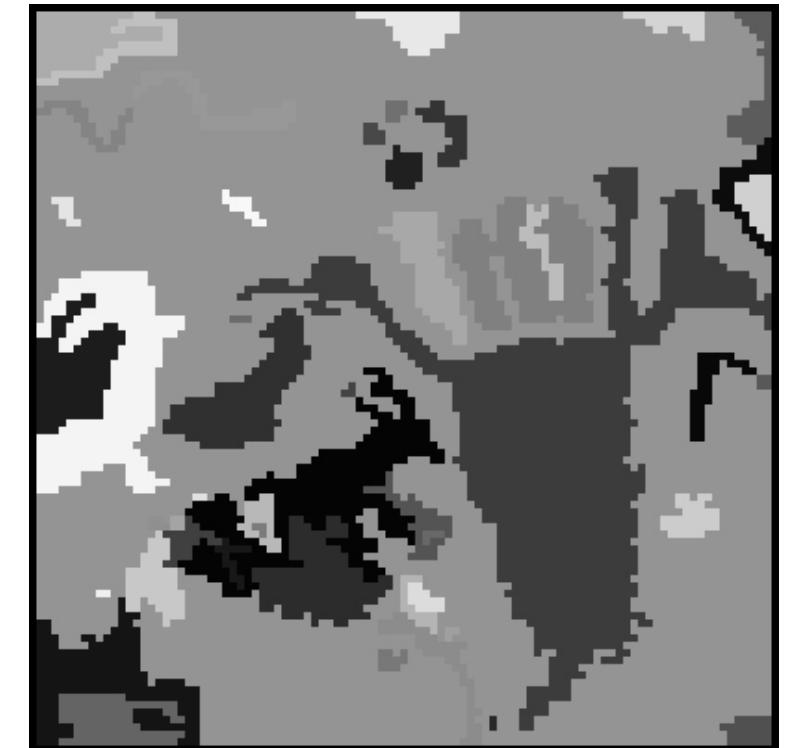


Segmente

- Segmente und Semantik
 - Unterschiedliche Segmente haben unterschiedliche Bedeutung
 - Pixel innerhalb eines Segments haben die gleiche Bedeutung
- Segmentierung ist ein *subjektiver* Prozess!
- Segmente können aus einem Bild generiert werden, falls
 - Pixel eines Segments andere Eigenschaften haben als die außerhalb eines Segments
 - jedes Pixel nur einem Segment angehört
- Es gibt viele Fälle, in denen diese Annahmen (näherungsweise) zutreffen

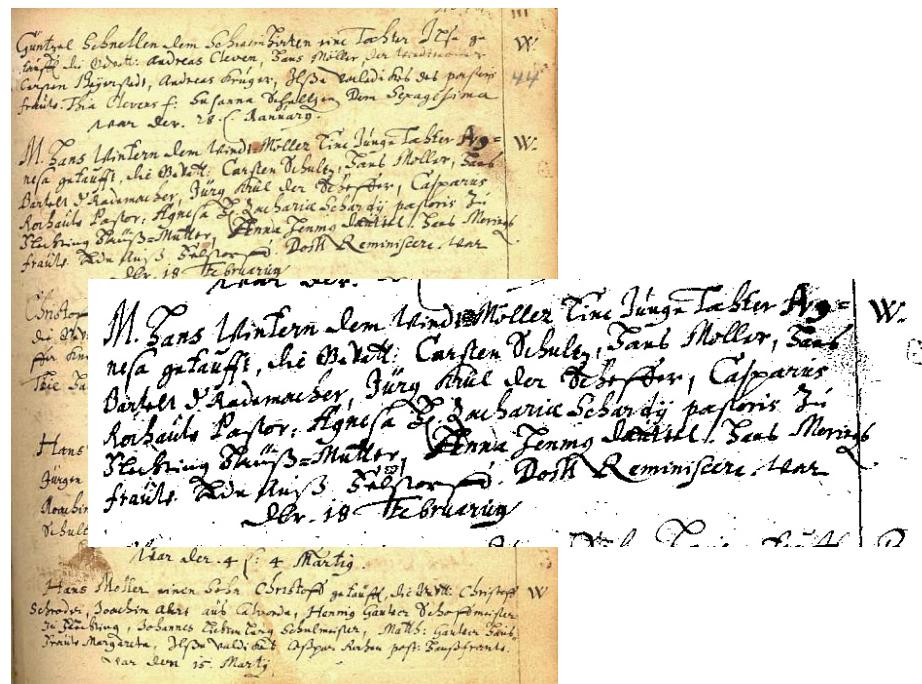
Segmentierung (abgeschwächt)

- Ziel: Pixelaggregate schaffen, denen Bedeutung zugeordnet werden *kann*
- Pixel gehören zusammen, wenn sie einem zusammenhängenden Gebiet angehören.
- Homogenitätsbedingung ist vom Bildinhalt unabhängig (Segmentierung ist datenbasiert).
- Modellbasierte Segmentierung: Zwitter zwischen Segmentierung und Analyse.



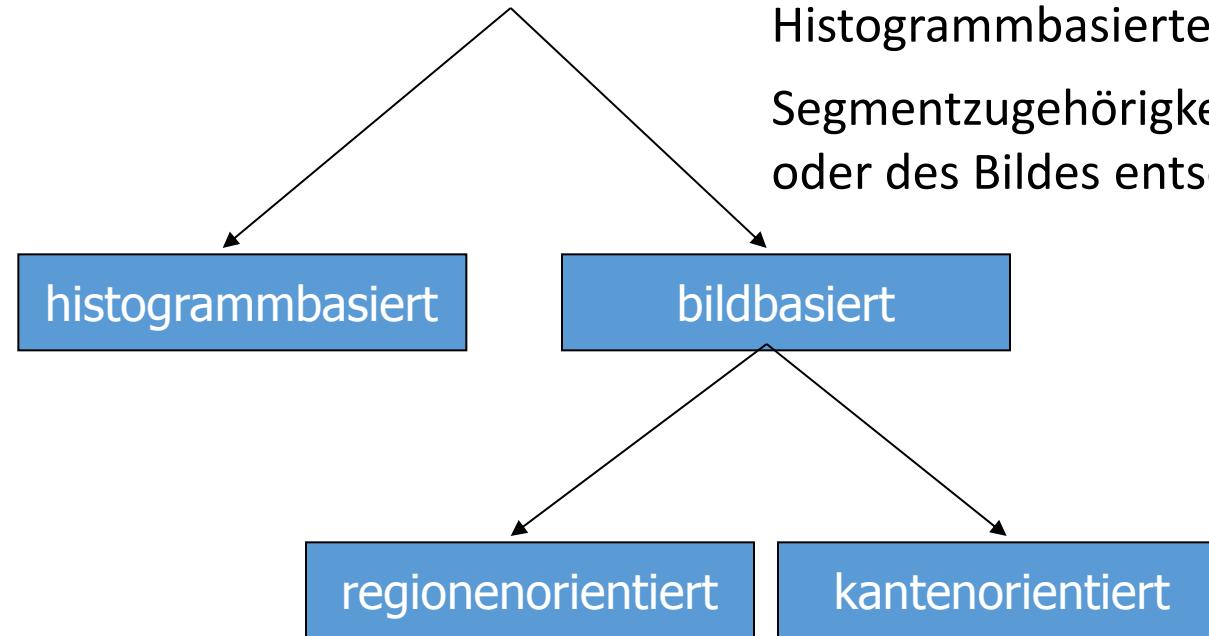
Segmentierung

- Zerlegung eines Bildes in semantische Einheiten.
 - **Segmente:** Träger der Bedeutung von Strukturen eines Bildes.
 - Eigenschaften einer Segmentierung:
 - **vollständig:** jedes Pixel ist einem Segment zugeordnet.
 - **überdeckungsfrei:** ein Pixel ist genau einem Segment zugeordnet.
 - **zusammenhängend:** jedes Segment bildet ein zusammenhängendes Gebiet.



Bsp. Figure-Ground-Segmentation

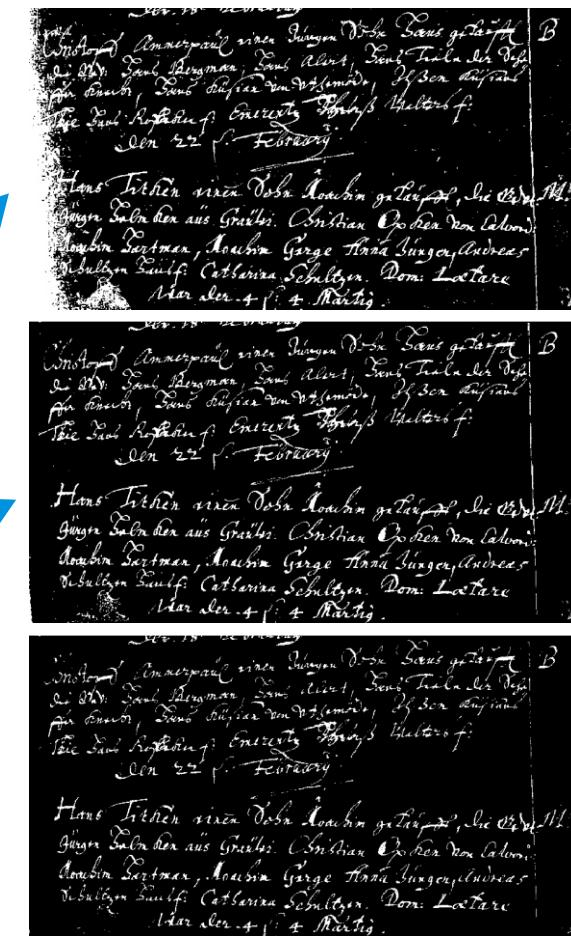
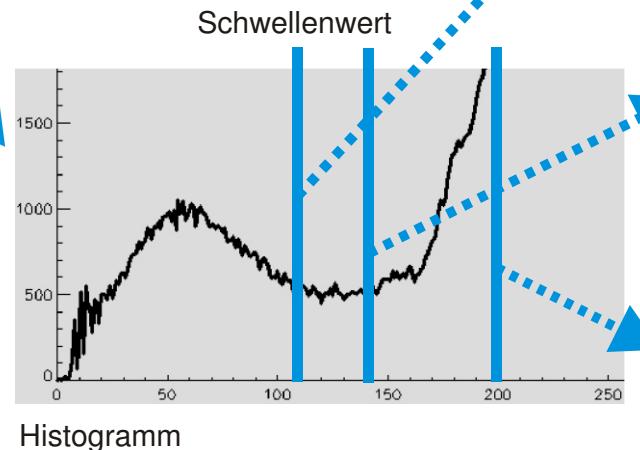
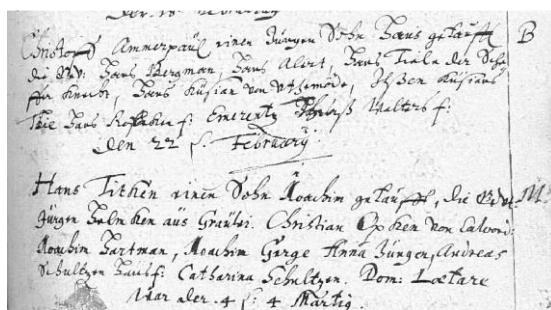
Datenbasierte Segmentierungsmethoden



Histogrammbasierte oder bildbasierte Segmentierung:
Segmentzugehörigkeit wird anhand des Histogramms
oder des Bildes entschieden

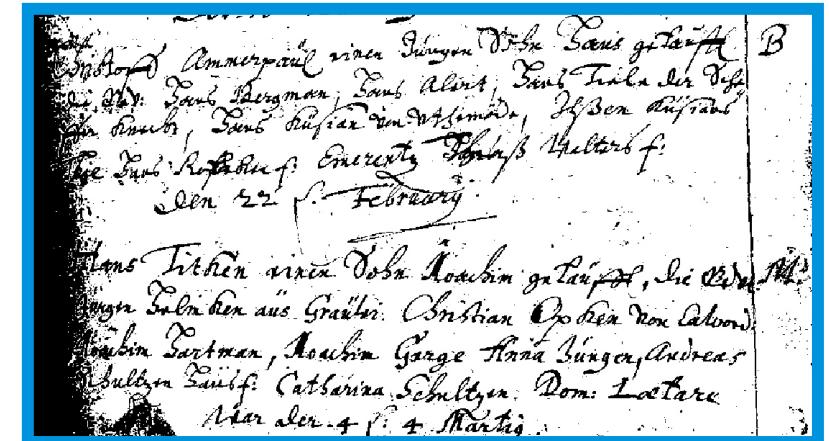
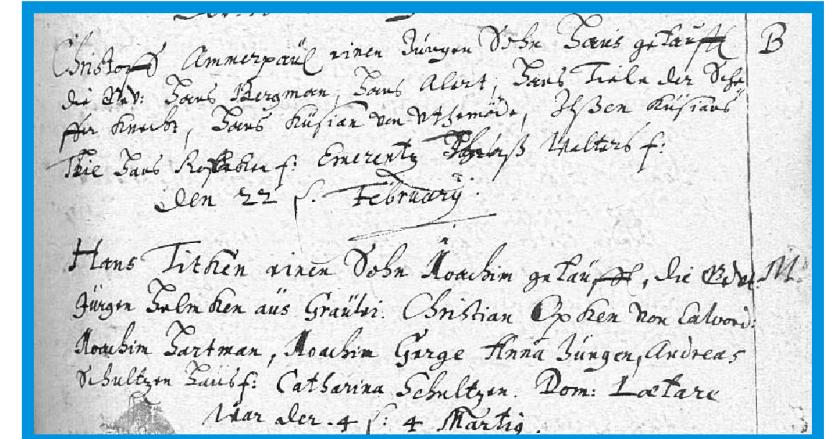
Histogrammbasierte Segmentierung

- Annahme:** Bild besteht aus zwei Anteilen die sich durch ihren Grauwert unterscheiden.
- Aufgabe:** Schwellenwert zwischen den beiden Grauwerten finden.
- $b(i,j) = \begin{cases} 1, & \text{falls } f(i,j) > T; \\ 0, & \text{sonst} \end{cases}$



Shading

- Helligkeitsvariationen zerstören die bimodale Verteilung der Häufigkeiten.
- ► Schwellenwert ist nicht mehr für das gesamte Bild definierbar.

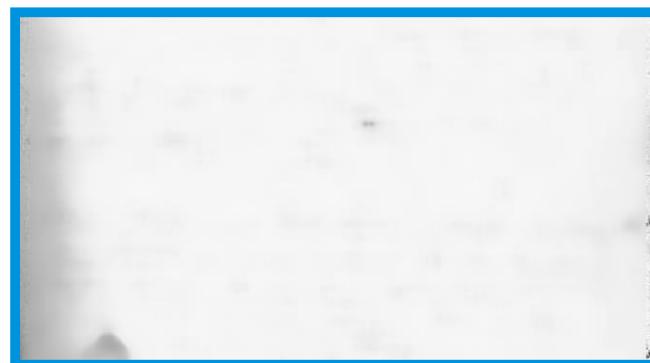
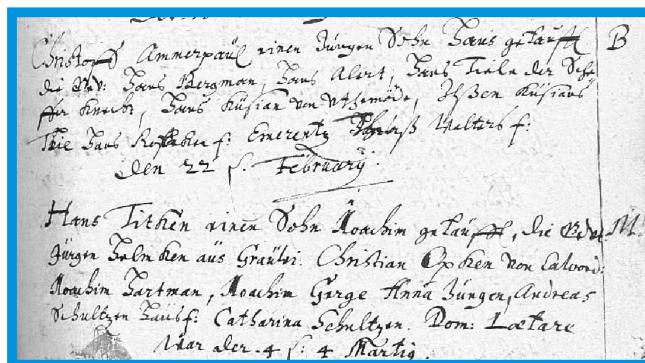


Berücksichtigung von Shading

- Homogenes Bild unter gleichen Bedingungen aufnehmen.
- Shading-Bild aus dem Bild selbst bestimmen.
- Variierende Schwellenwerte.

Bestimmung des Shadingbilds

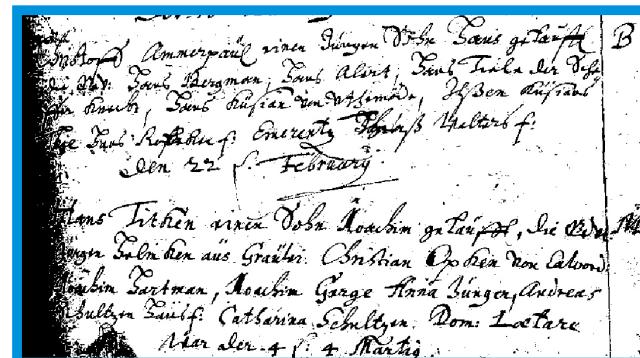
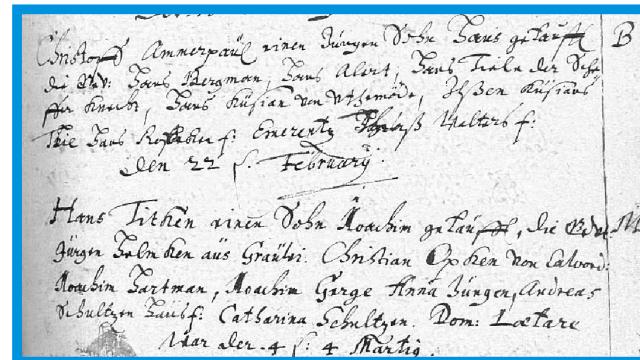
- Falls die überwiegende Mehrzahl des Bildes aus Vorder- oder Hintergrundpixeln besteht, kann das Shadingbild durch einen Rangordnungsfilter erzeugt werden.
- Das Rangordnungsfilter muss mindestens so groß sein, dass immer mindestens ein Vordergrund- und ein Hintergrundpixel enthalten ist.



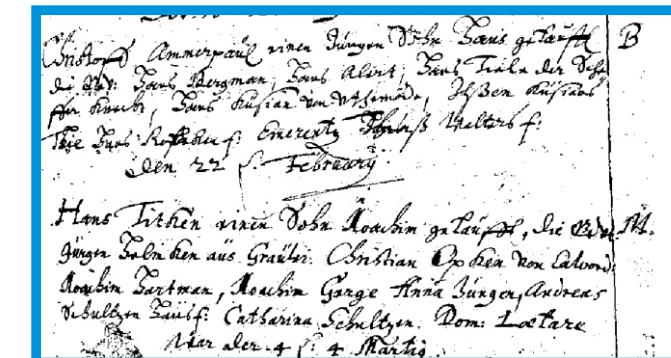
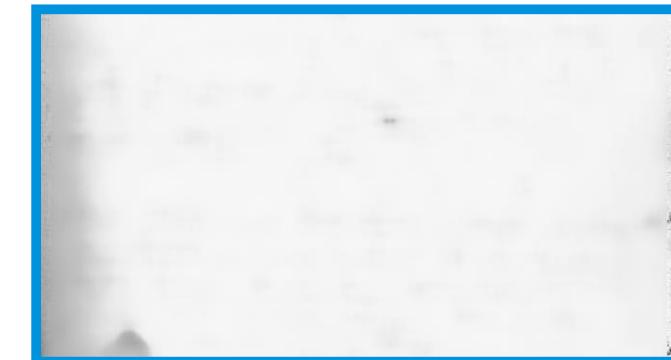
13×13 Maximumsfilter

Shading-Korrektur

- Berechne aus dem Hintergrundsbild Shadingfunktion $s(i, j)$
- Korrektur: $f'(i, j) = f(i, j) \cdot s(i, j)$
- Segmentierung auf dem korrigierten Bild.

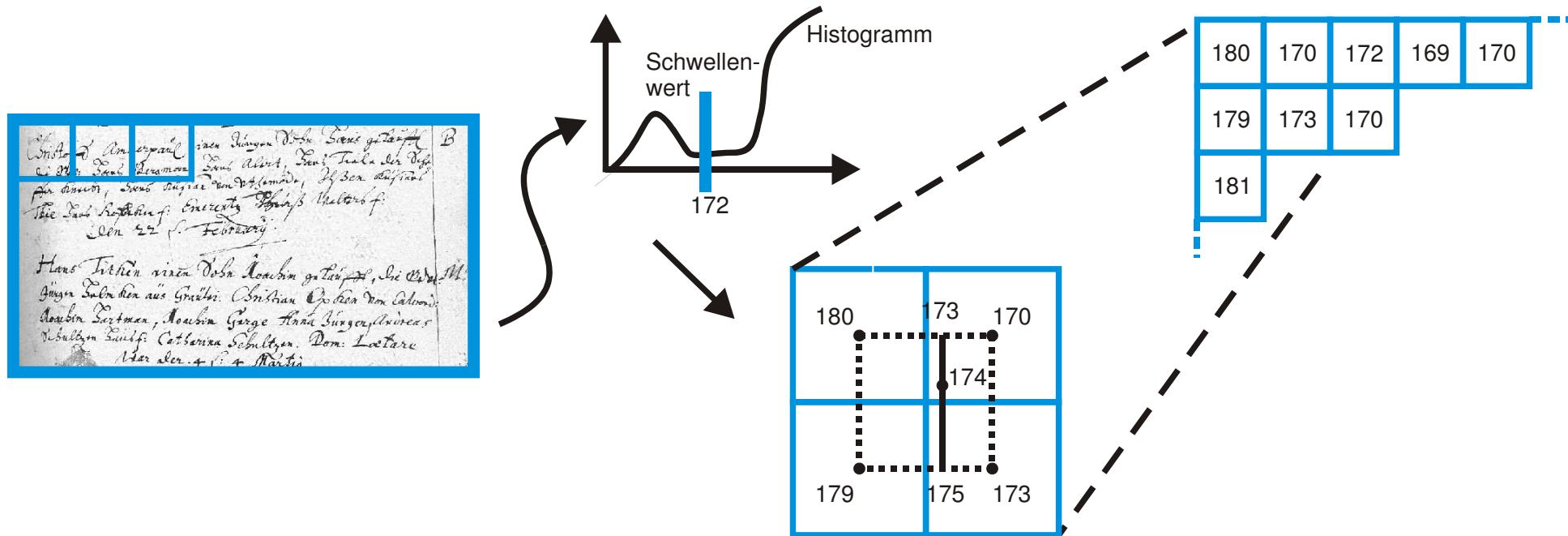


ohne Korrektur



mit Korrektur

Variierende Schwelle



Lokale Schwellenwerte aus Histogramme in Teilregionen.

(Lineare) Interpolation von Schwellenwerten $T(i, j)$ an allen anderen Punkten.

Segmentierung durch $f(i, j) > T(i, j)$.

Region Labeling

- Schwellenwert zerlegt das Bild in Vordergrund und Hintergrundsegmente.
- **Region Labeling** bestimmt Ort und Anzahl aller zusammenhängenden Gebiete im Binärbild b :

```
region.initialise() // Region der Größe M,N erzeugen und
label=1             // mit Null initialisieren, Startlabel=1
for (i,j) = 0, (M,N) do           // Doppelschleife über i und j
    if region.labels(i,j) = 0 then // dieser Ort ist noch nicht
                                    // Teil einer Region
        label = label+1          // neues Label vergeben
        region.flood_fill(i,j,label) // zusammenhängendes
                                    // Gebiet um (i,j) mit
                                    // Label füllen
```

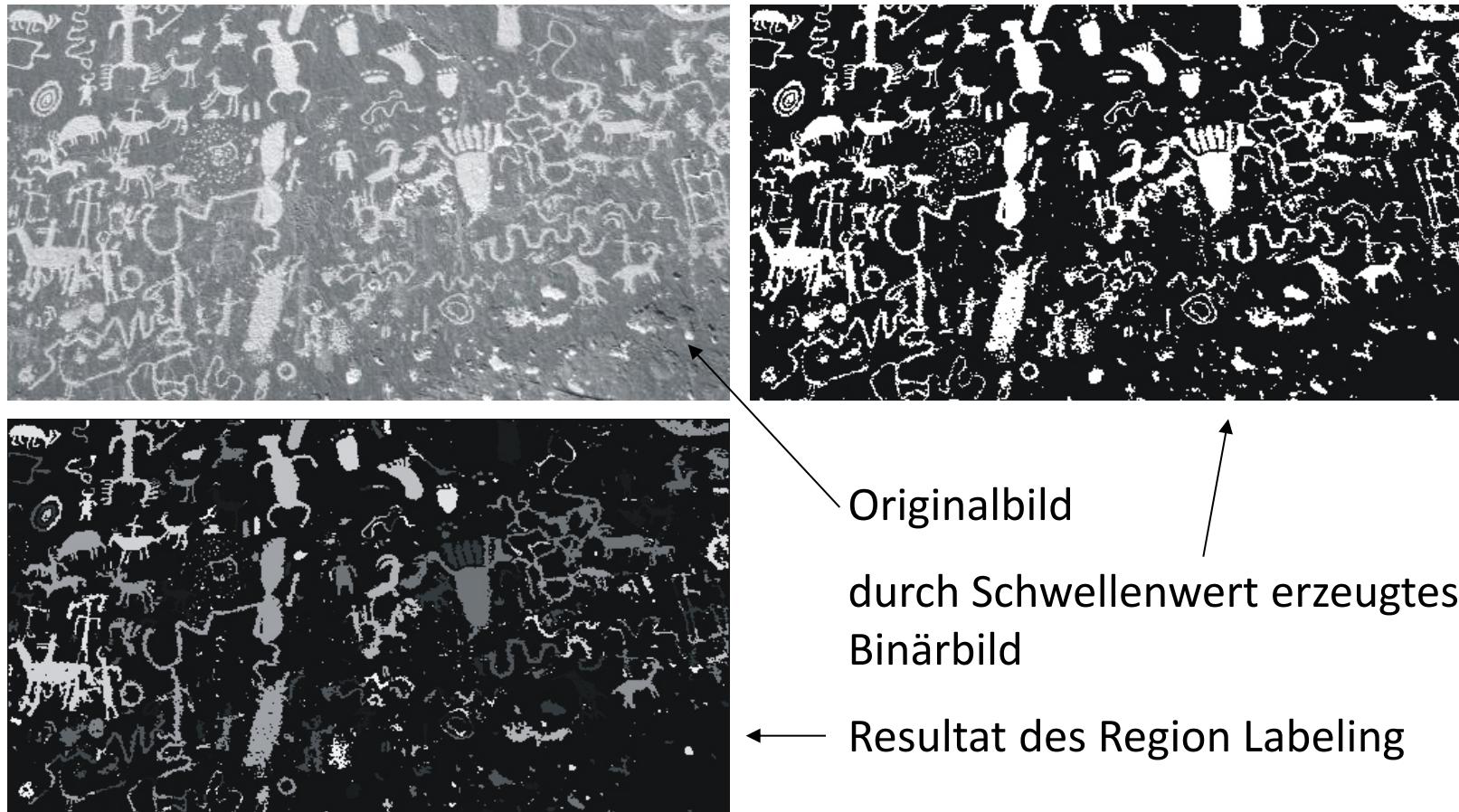
Flood Fill

```
flood_fill(i,j,label) // Variablen zur Auswertung der  
Zusammenhangsbedingung sind global  
verfügbar  
if f(i,j) erfüllt Zusammenhangsbedingung then  
    region(i,j) = label // Region an (i,j) mit Label  
        // versehen  
    flood_fill(i-1,j,label) // Nachbarpixel untersuchen  
    flood_fill(i,j-1,label)  
    flood_fill(i+1,j,label)  
    flood_fill(i,j+1,label)
```

Zusammenhangsbedingung:

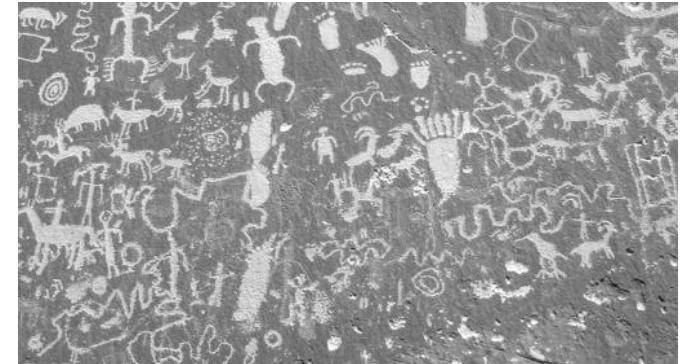
- hat den gleichen Grauwert wie Saatpunkt oder
- ist kein Randpunkt

Resultat

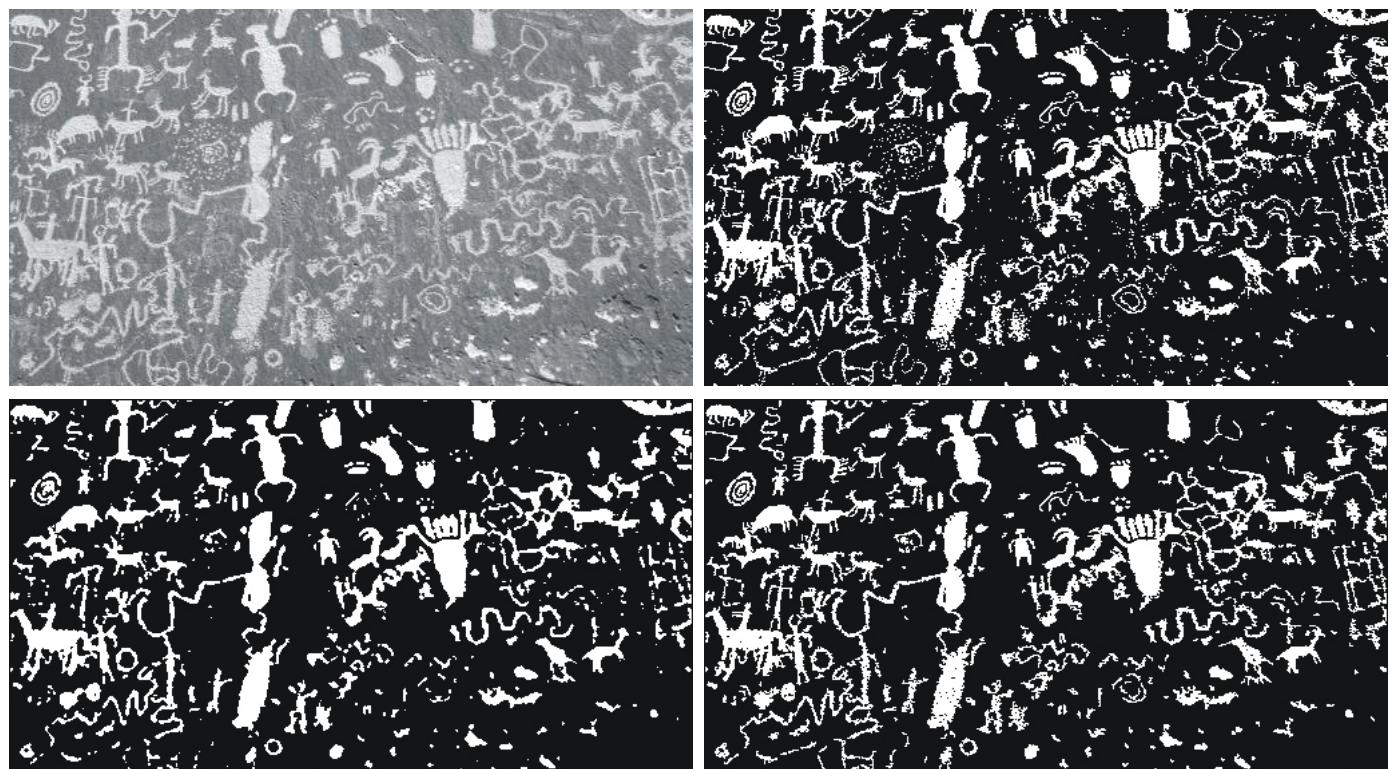


Nachverarbeitung

- Trennung nach Grauwerten wird nicht perfekt sein.
- Schwellenwertbild enthält falsche Regionen:
 - kleine fälschlicherweise als Segmente identifizierte Regionen.
 - Störungen am Rand von Regionen.
- Nachverarbeitung
 - Medianfilterung auf den Labeln
 - Entfernung von zu kleinen Regionen
 - Relaxation Labelling



Medianfilterung auf Labeln

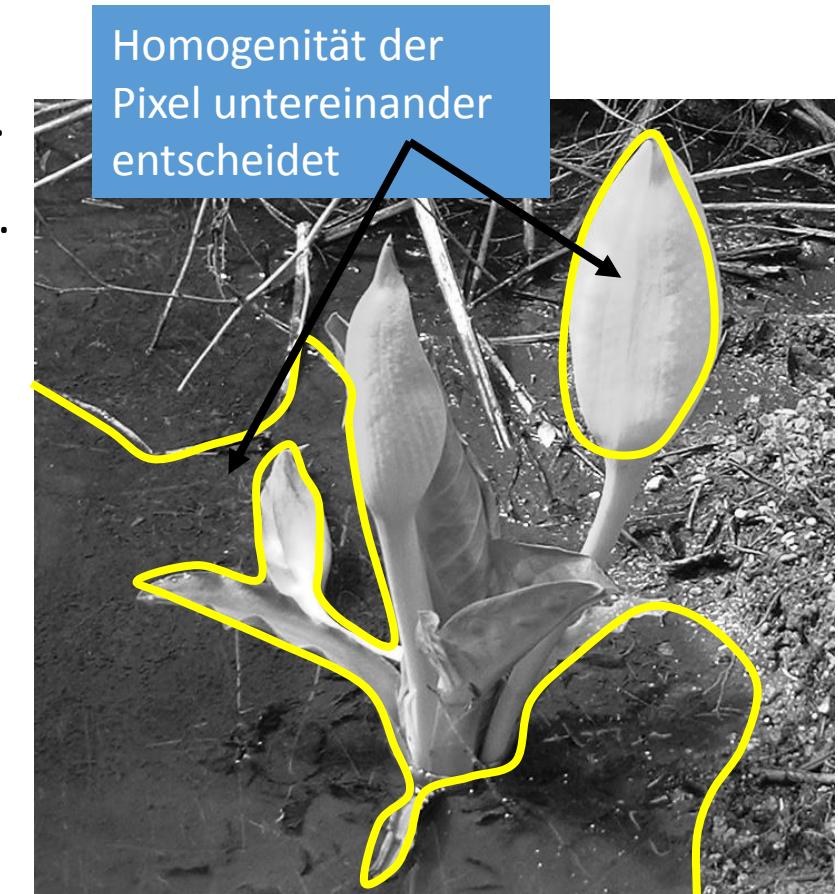


Entfernung kleiner Gebiete



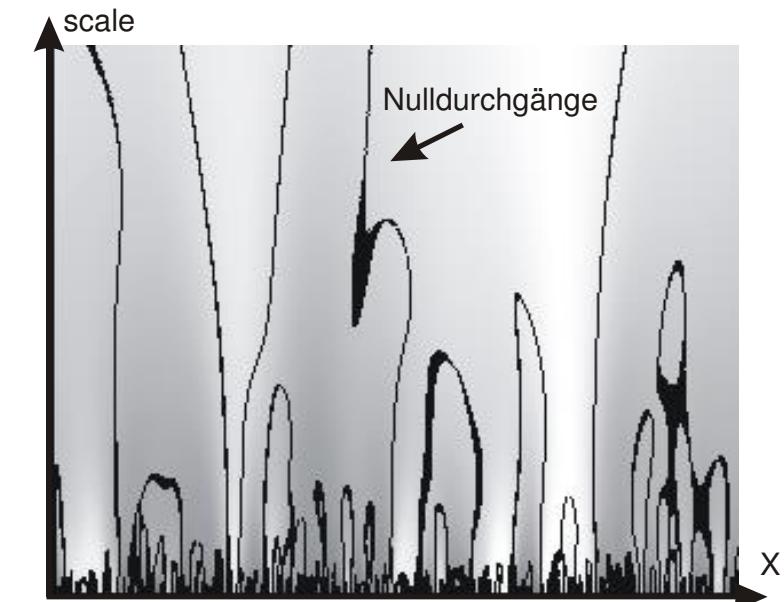
Regionenbasierte Segmentierung

- Homogenität im Inneren des Segments.
- Homogenitätsbedingung wird bei der Segmentierung ausgewertet.
- Homogenität ist relativ zu den Attributen eines Segments definiert.
- Globale Zusammenhänge über Multiskalenstrategie.



Multiskalenstrategie

- relative Kriterien für Homogenität können über unterschiedliche Entferungen verschieden wirken.
- Segmentierung nach Multiskalenstrategie wertet Kriterien auf unterschiedlichen Skalierungen aus.
- Multiskalenstrategie
 - explizit auf einer Multiskalenrepräsentation
 - implizit in den Segmentierungsalgorithmus integriert.



Gaußpyramide

- Das Originalbild wird fortlaufend durch eine „reduce“-Operation skaliert.
- Jedes Pixel der nächsthöheren Skalierungsstufe repräsentiert 4 Pixel der aktuellen Stufe.
- Vor der Reduktion wird der Frequenzumfang durch Filterung vermindert:

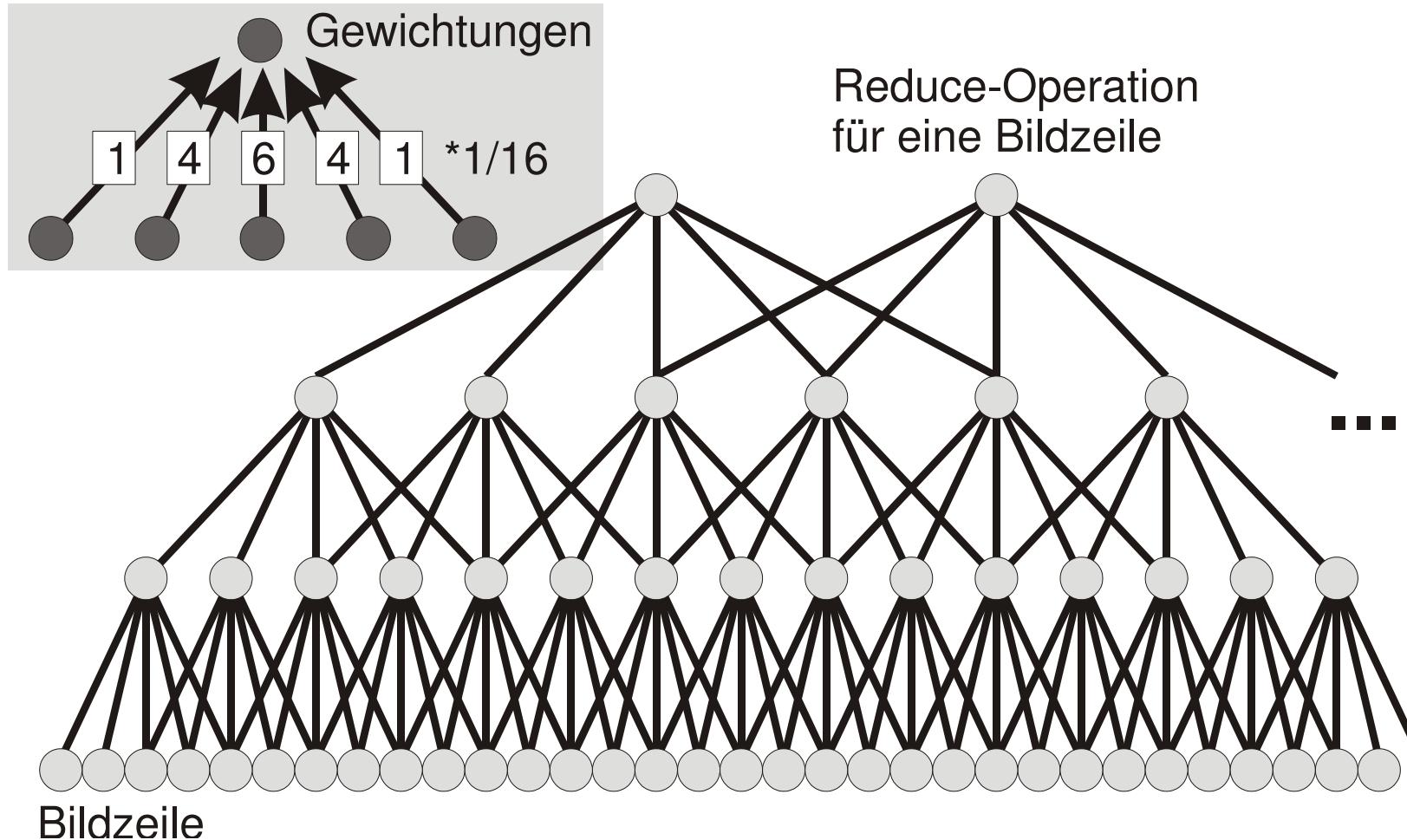
- Gaußfilter

$$\frac{1}{16} (0.87 \quad 3.91 \quad 6.44 \quad 3.91 \quad 0.87)$$

- Binomialfilter

$$\frac{1}{16} (1 \quad 4 \quad 6 \quad 4 \quad 1)$$

Gaußpyramide



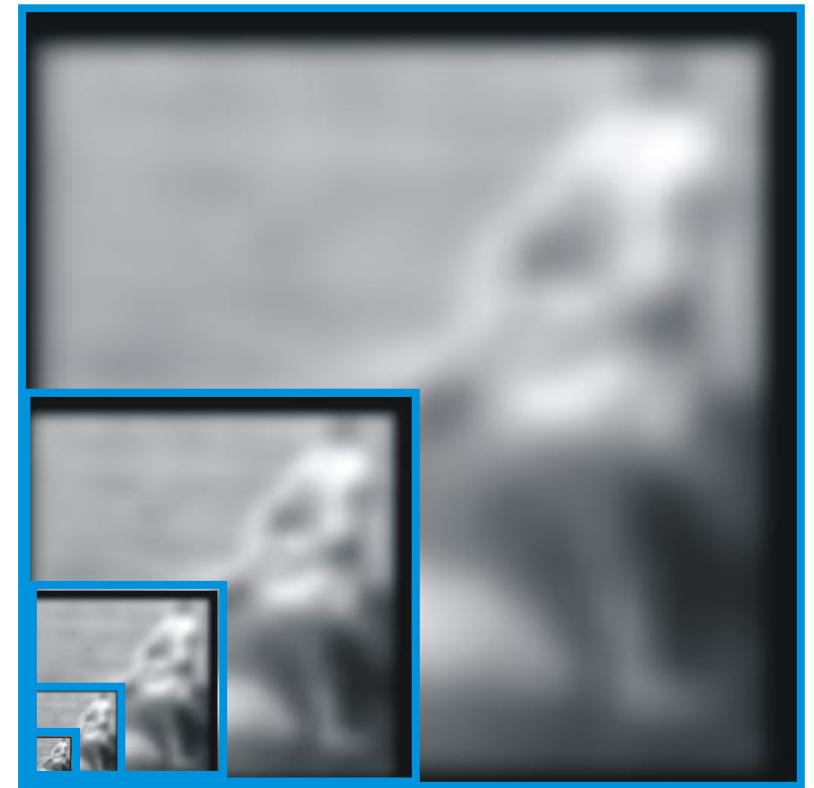
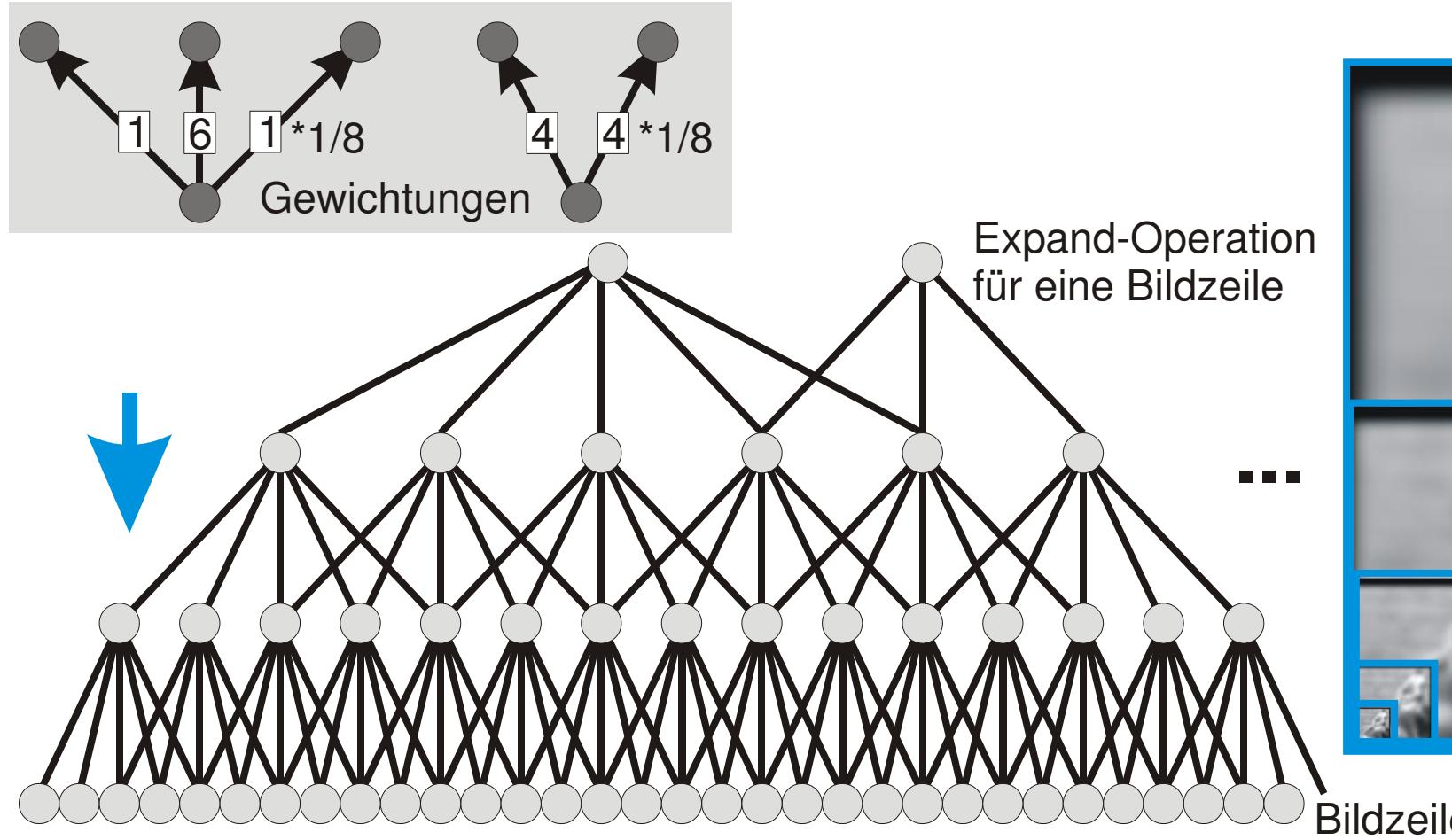
Expand-Operation

- Um die vorherige Skalierungsstufe zu erzeugen, wird eine „expand“-Operation definiert.
- Pixel der neuen Skalierungsstufe werden durch Interpolation erzeugt:
 - Pixelorte, die auf beiden Skalierungsstufen existieren:
 - Pixelorte, die nur auf der vorherigen Skalierungsstufe existieren:

$$\frac{1}{8.18} \begin{pmatrix} 0.87 & 6.44 & 0.87 \end{pmatrix} \text{ bzw. } \frac{1}{8} \begin{pmatrix} 1 & 6 & 1 \end{pmatrix}$$

$$\frac{1}{7.82} \begin{pmatrix} 3.91 & 3.91 \end{pmatrix} \text{ bzw. } \frac{1}{8} \begin{pmatrix} 4 & 4 \end{pmatrix}$$

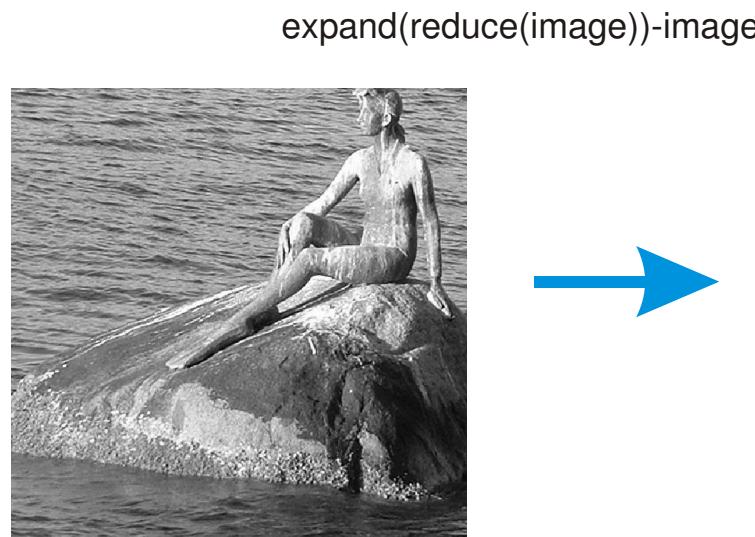
Expand-Operation



Die Expand-Operation ist nicht verlustfrei

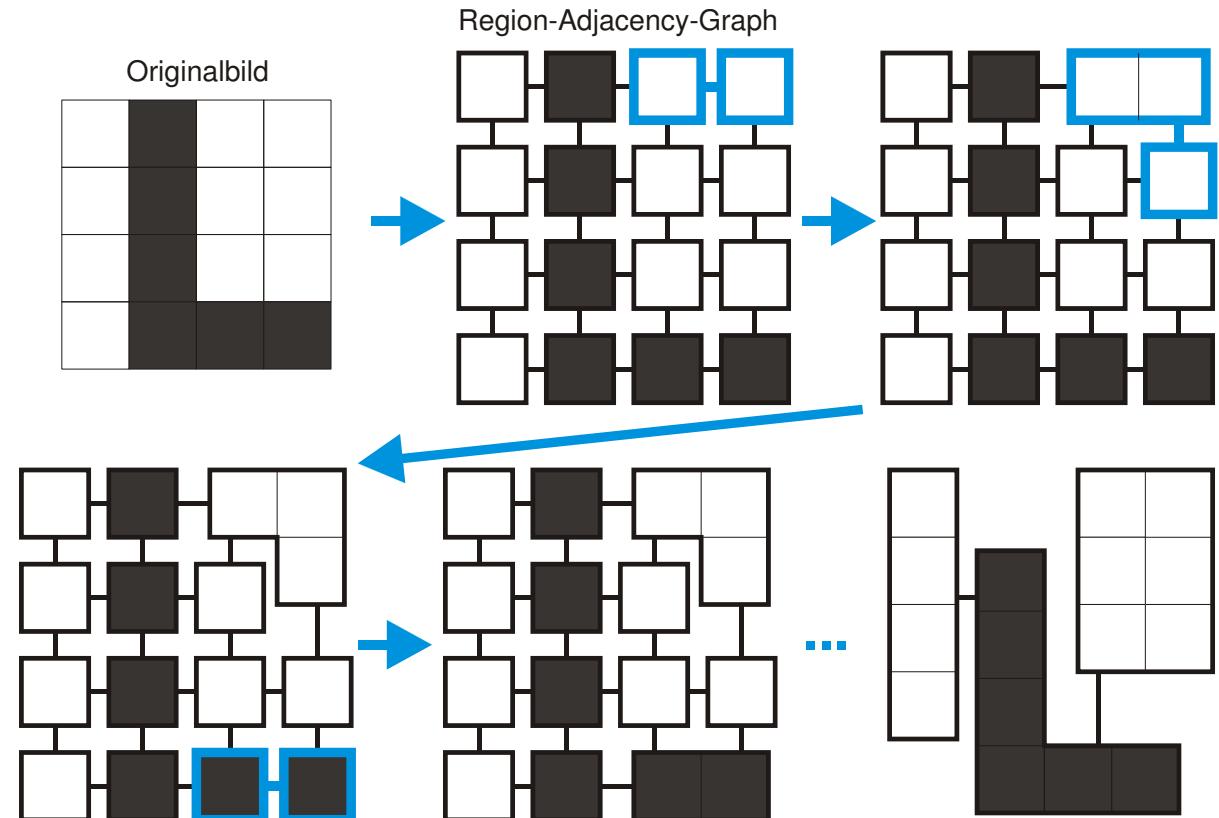
Laplace-Pyramide

- Jede Skalierungsstufe s enthält nur den Unterschied $f_s - \text{expand}(\text{reduce}(f_s))$
- Redundanzfreie Repräsentation



Region Merging

- Initial wird jedes Pixel zu einem Segment erklärt.
- Zwei benachbarte Regionen werden zusammengefasst, wenn sie auch gemeinsam das Homogenitätskriterium erfüllen.
- Die Segmentierung ist beendet, wenn keine zwei Regionen mehr existieren, die zusammengefasst werden können.
- Zwischenergebnisse werden in einem **Region Adjacency Graph (RAG)** gespeichert.



Region Merging

„von Pixeln zu Regionen“:

```

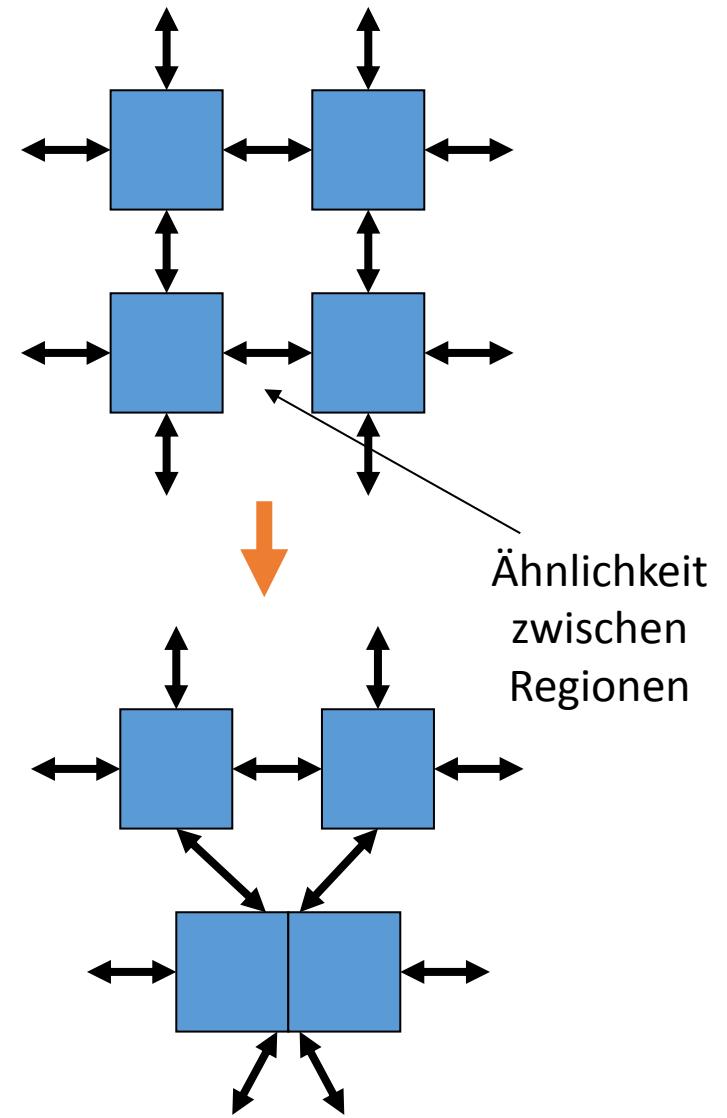
stopMerge = false
while not stopMerge do
    (r1, r2) = MaxSimilarity(region)
    if sim(r1, r2)>T then
        region.merge(r1, r2)
    else
        stopMerge=true

```

Ähnlichkeitskriterium:

z.B., maximaler Grauwertunterschied zwischen Pixeln von r1 und r2.

Region Labeling kann integriert werden.



Region Merging

- Homogenitätskriterium
 - Grauwertdifferenz innerhalb der Region.
 - Wahrscheinlichkeit, dass beide Regionen die gleichen Grauwertverteilungen haben.



Region Merging und Multiskalenstrategie

Modellannahme:

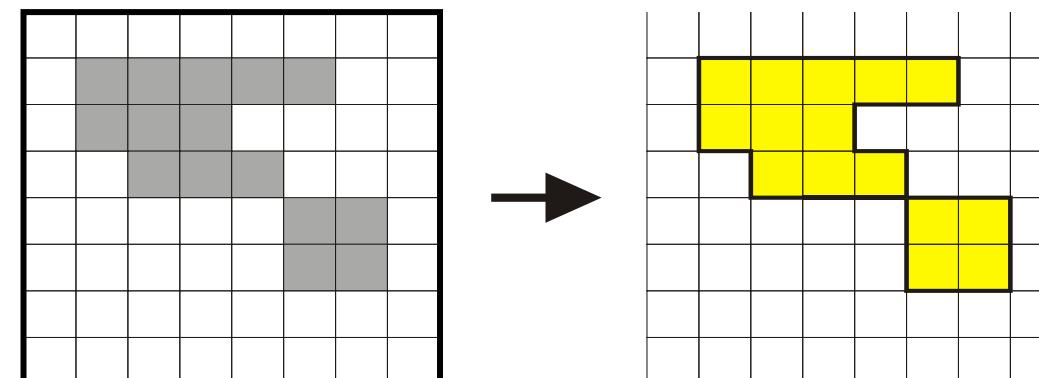
Die größte Skalierungsstufe, auf der sich segmentierungsrelevante Eigenschaften manifestieren, ist bekannt.

Prozess:

- Region Merging auf grober Skalierung
- Übertragung des Resultats auf die nächstfeinere Stufe.
- Alle Pixel, die zu Pixeln eines anderen Segments benachbart sind, werden nochmals geprüft.
- Verfahren endet, wenn die feinste Skalierungsstufe erreicht ist

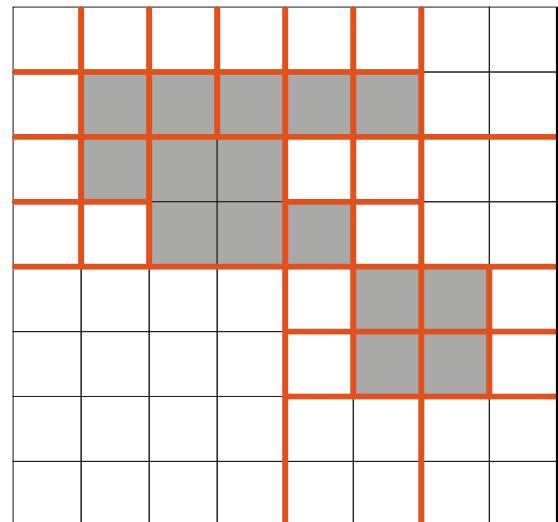
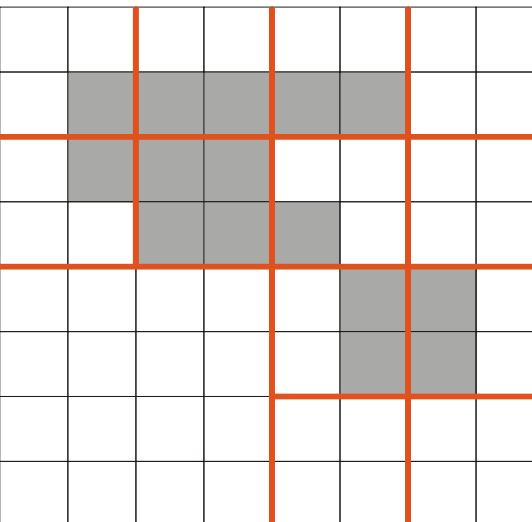
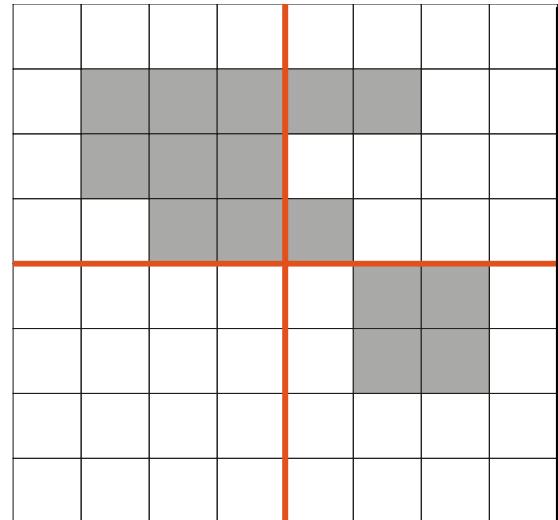
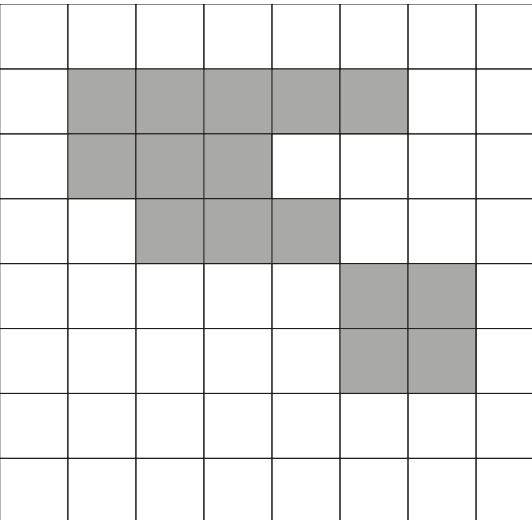
Split & Merge-Algorithmus (Regionenbasiertes Verfahren)

- **Startbedingung:** Das gesamte Bild ist ein Segment.
- Ein Segment wird solange in 4 Untersegmente zerlegt, wie es ein gegebenes Homogenitätskriterium nicht erfüllt
- Benachbarte Segmente werden zusammenfasst, wenn sie auch nach der Zusammenfassung das Homogenitätskriterium erfüllen.
- **Resultat** ist eine vollständige, überdeckungsfreie Zerlegung des Bildes (Segmentierung gemäß Definition)

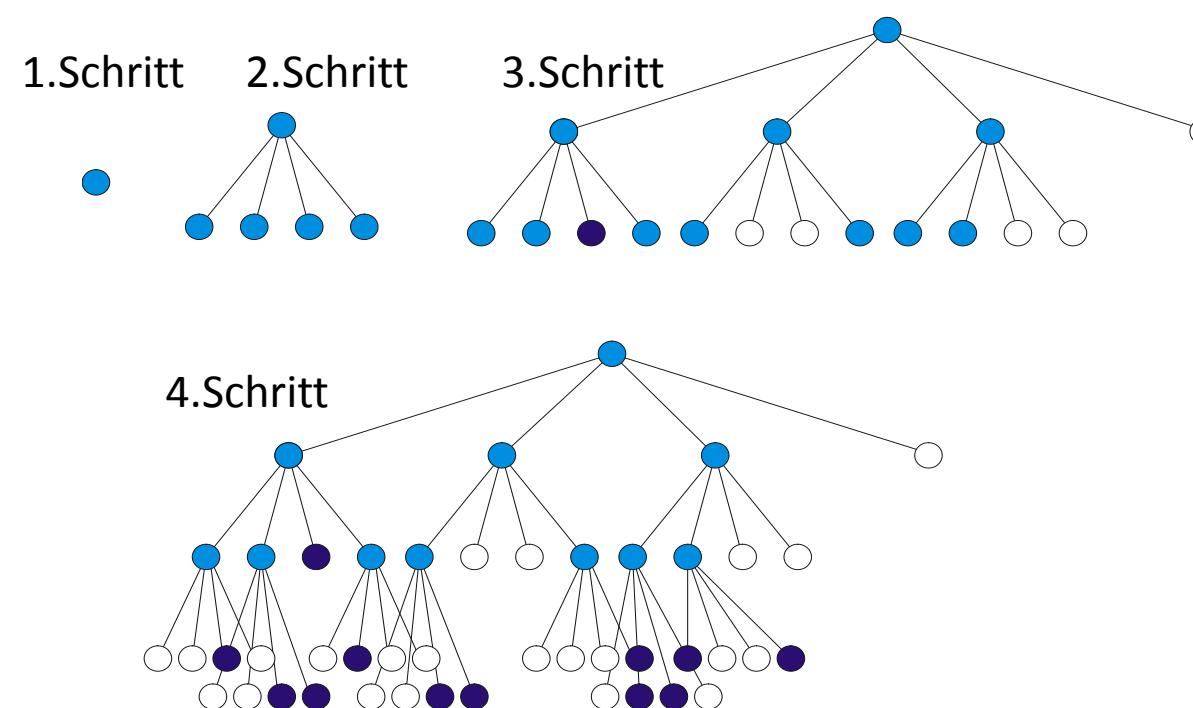


Zerlegungsschritt

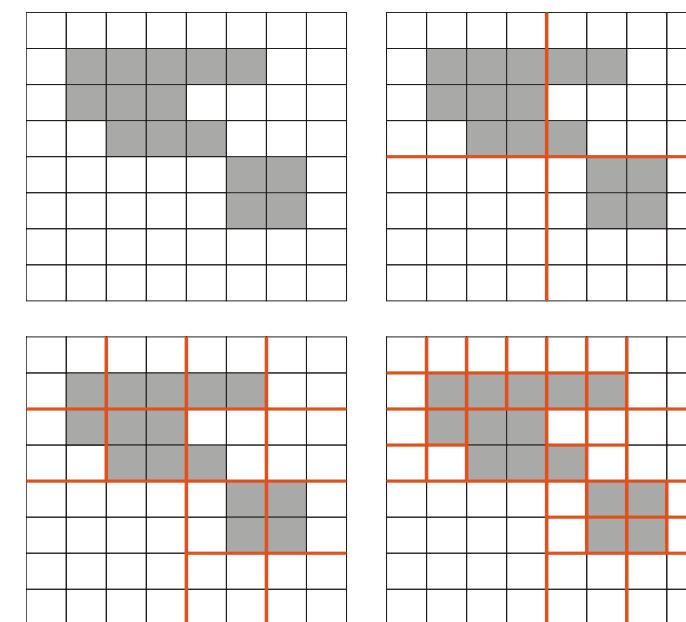
- Zerlegung terminiert spätestens auf Pixelebene.
- Problem: Datenstruktur zur Dokumentation der aktuellen Zerlegung



Zerlegungsschritt (Quad-Tree Repräsentation)



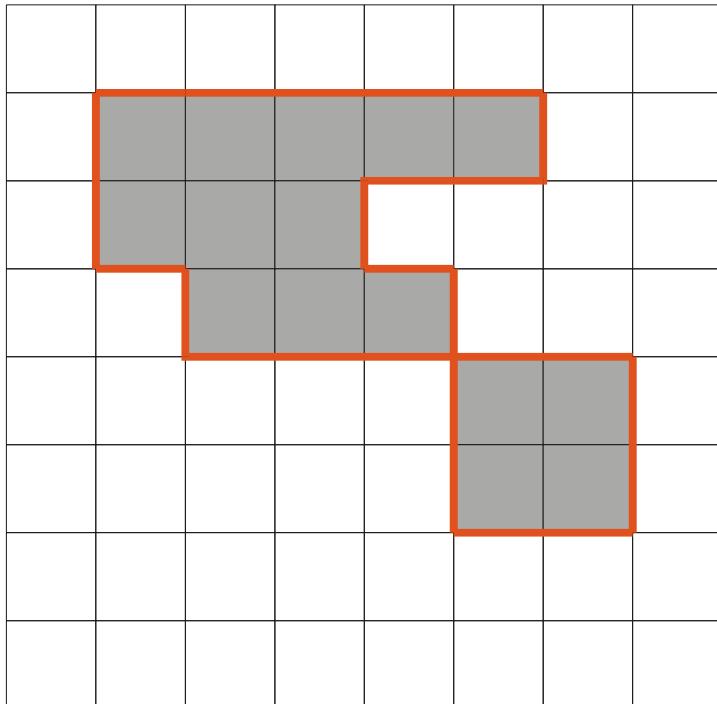
Wert des Homogenitätsmerkmals einer Region wird im entsprechenden Blatt des Quad-Tree abgelegt



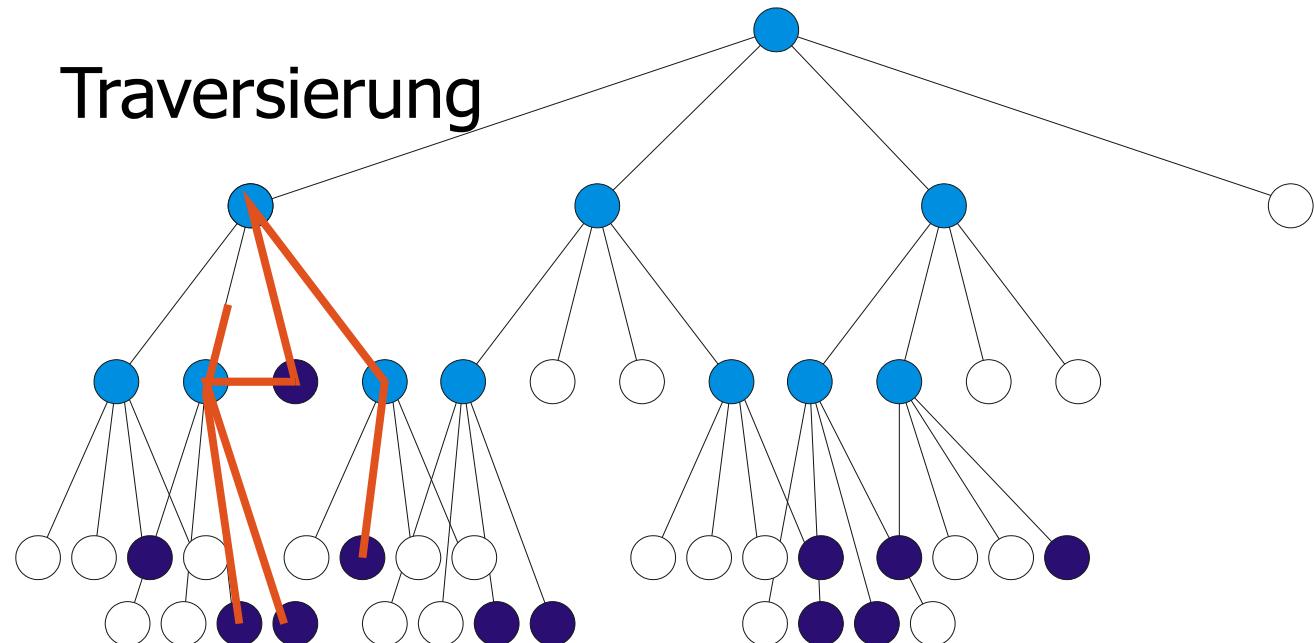
Merging

Quadtree wird traversiert und *in einen RAG überführt*.

Auf dem RAG wird ein Region Merging durchgeführt.

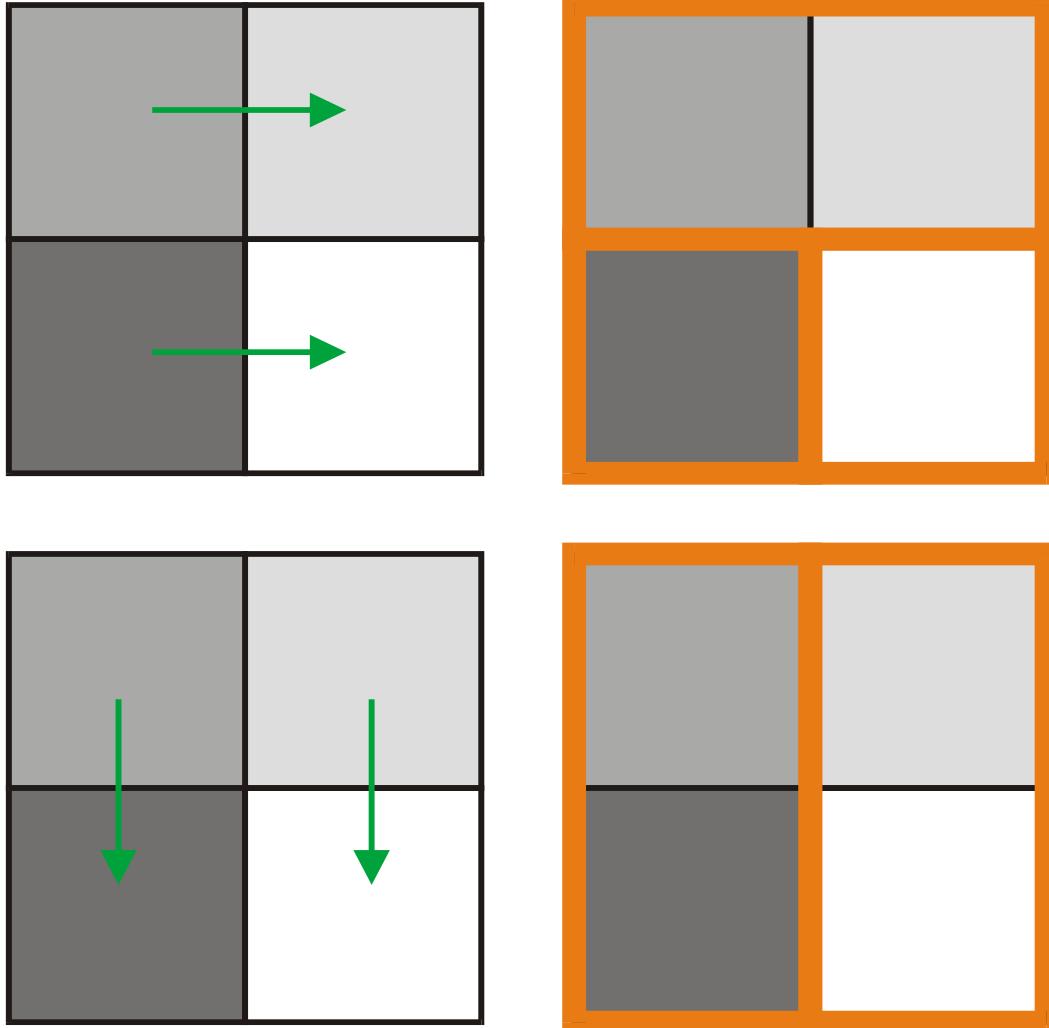


Traversierung



Split & Merge

- Resultat: Zerlegung des gesamten Bilds in Regionen.
- Multiskalenstrategie ist integriert
- Homogenitätsmerkmale wie bei Region Merging
- Probleme (Region Merging und Split & Merge):
 - Merge-Schritt ist bei relativen Homogenitätsmerkmal **nicht immer eindeutig**.
 - Keine absoluten Merkmale möglich.

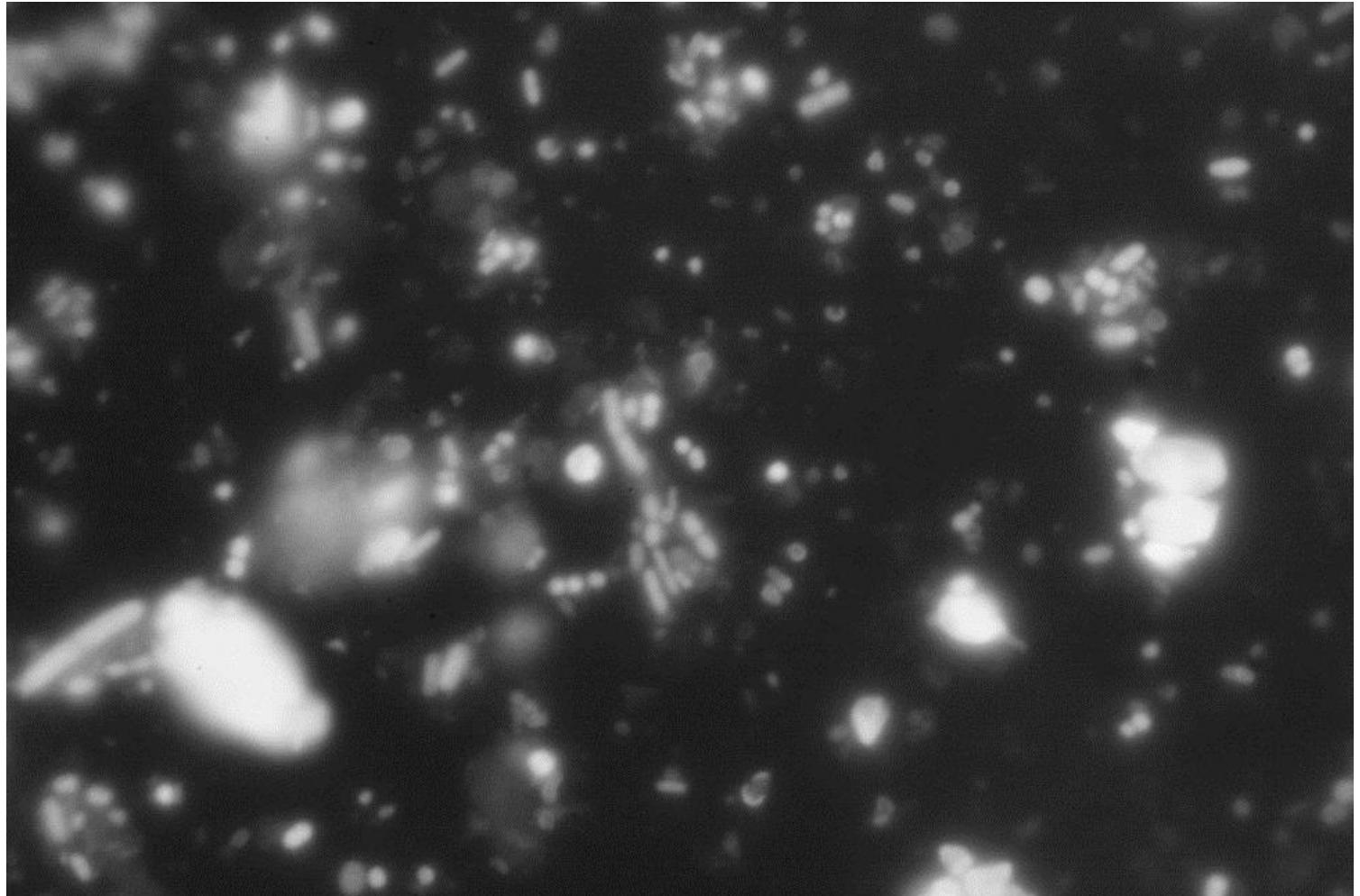


Was Sie heute gelernt haben sollten?

- Schwellenwertsegmentierung
- Region Labelling, Shadingkorrektur und Nachverarbeitung
- Gauß- und Laplacepyramide
- Region Merging
- Split and Merge

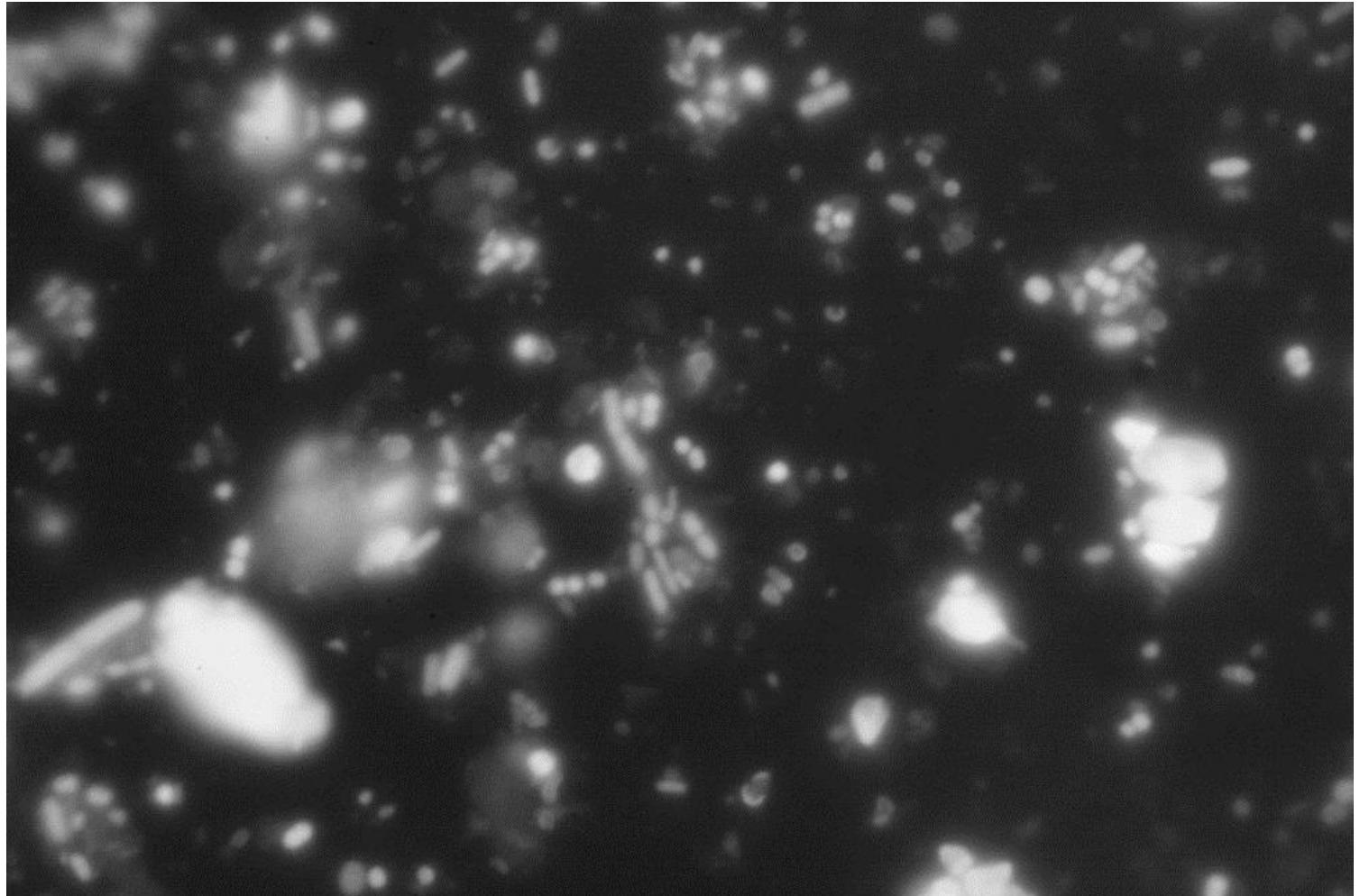
Famous Last Question

Wie könnte man diese
Mikroorganismen
segmentieren?



Famous Last Question

Wie könnte man diese
Mikroorganismen
segmentieren?



Kantenbasierte Segmentierung

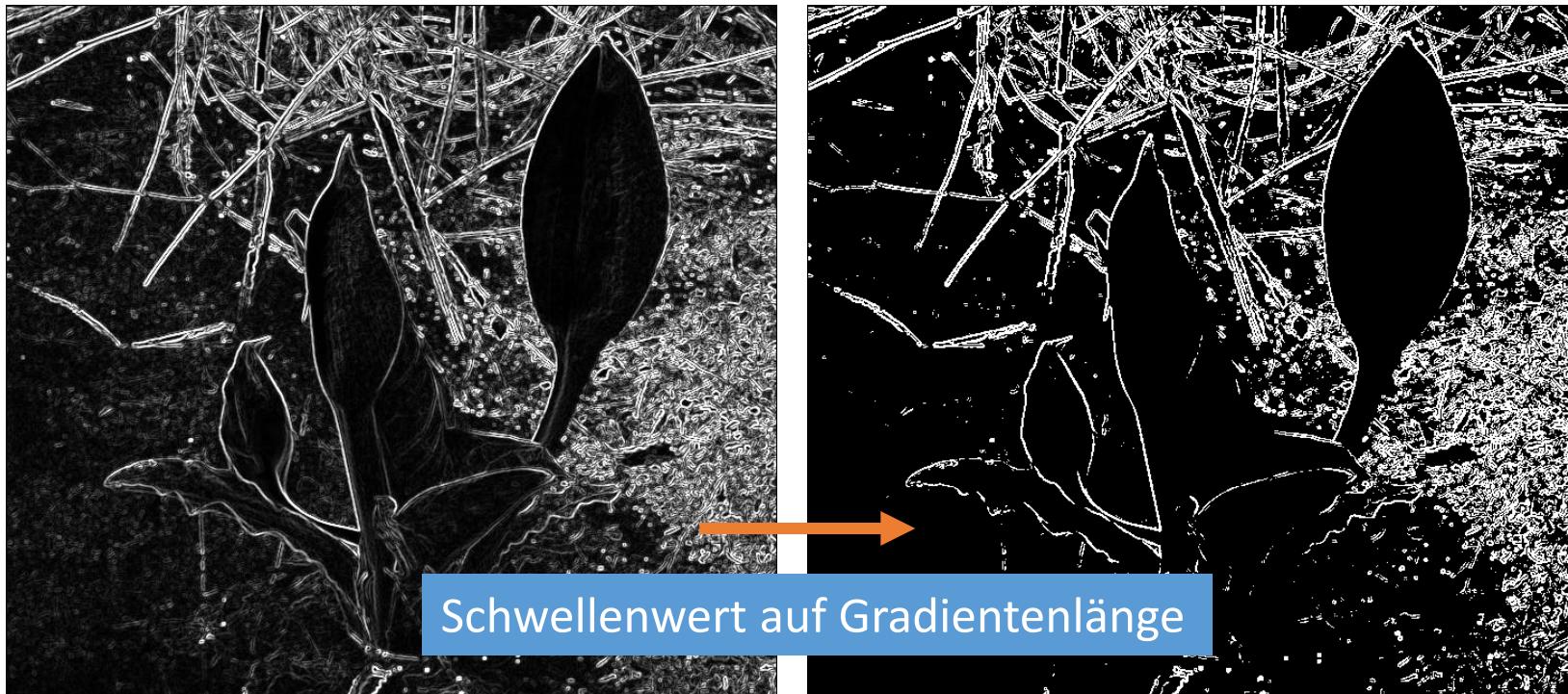
- Edge Linking und Canny Edge Operator
- Nulldurchgänge
- Wasserscheidentransformation
- Relaxation Labeling zur Nachverarbeitung

Segmentierung durch Kantenerkennung

- **Vorteil:** Kantenmerkmale sind robuster gegenüber Shading
- Einfache Methode
 - Gradientenberechnung
 - Kantenpunktdetektion (z.B. Schwelle auf Gradientenlänge)
 - Region Labeling basierend auf Kantenpunkten.

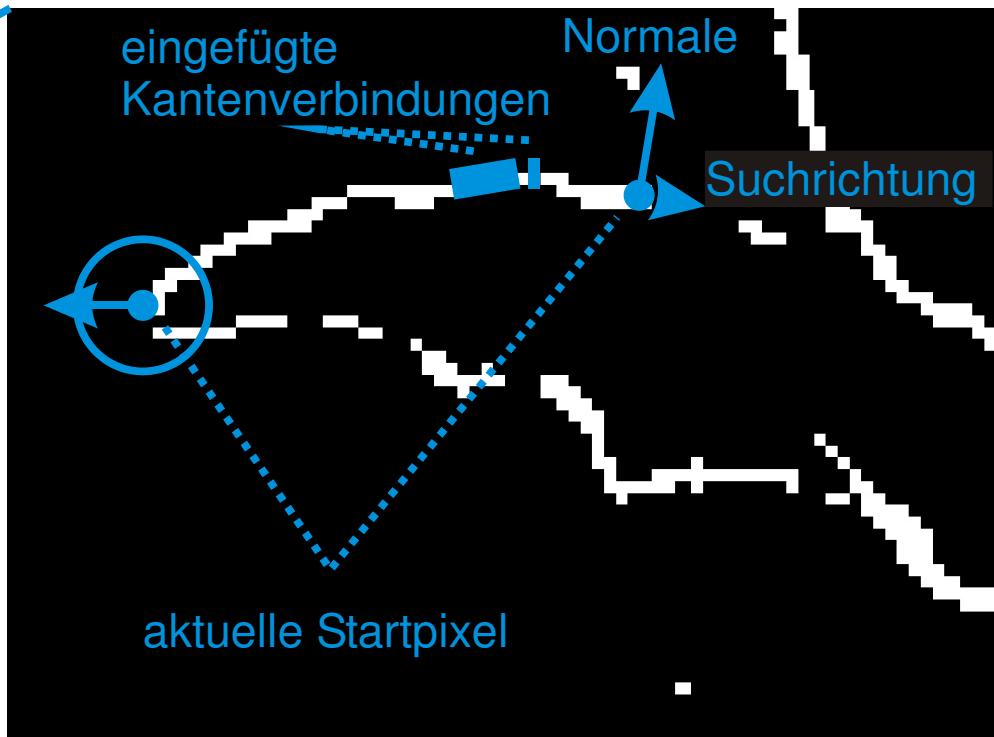
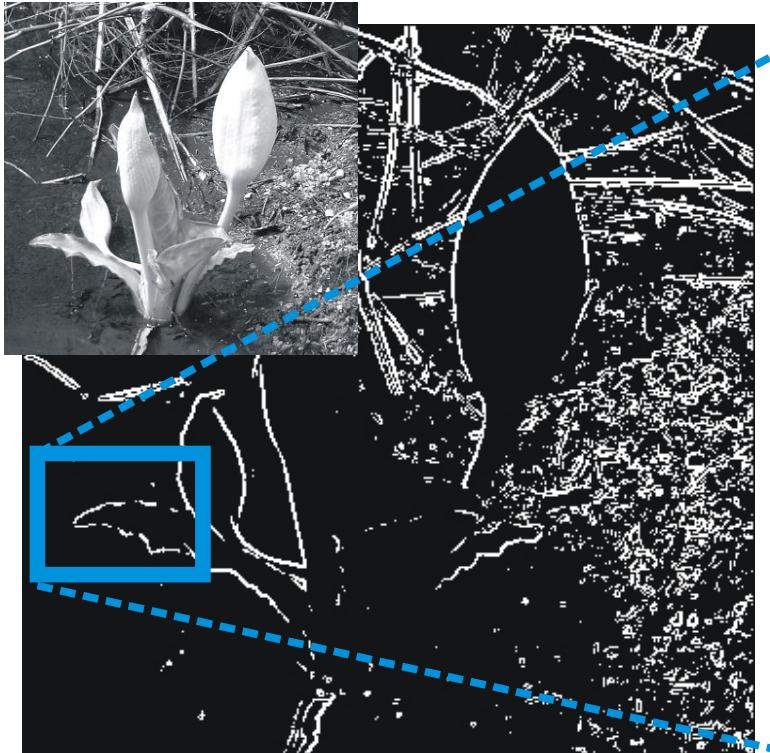


Segmentierung durch Kantenerkennung



Problem: Kantenpunkte sind nicht Ränder zusammenhängender Gebiete.

Edge Linking



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

1. Suche das nächste Kantenpixel, welches noch nicht bereits als „untersucht“ markiert wurde und erkläre es zum Startpixel eines Kantenzugs.
2. Falls sich in der Umgebung des Kantenpixels in einer der beiden Richtungen orthogonal zur Kantenrichtung unmarkierte Kantenpixel befinden, die eine ähnliche Gradientenrichtung und – stärke aufweisen:
 - a. Markiere die Pixel als zum Kantenzug zugehörig.
 - b. Erkläre diese Pixel zu neuen Startpixeln.
 - c. Gehe zu 2.
3. Falls sich in der Umgebung markierte Pixel befinden, die den obigen Bedingungen genügen, dann wurde eine Verzweigung von Kanten gefunden.
4. Falls kein Kantenpixel gefunden wurde, gehe zurück zu Schritt 1.

Canny Edge Operator

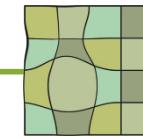
Ziele:

- möglichst viele Kanten fehlerfrei vom Hintergrund unterscheiden zu können (niedrige Rate von Fehldetektionen).
- (unverzweigte) Kanten genau zu lokalisieren.
- für jede Kante genau eine Detektorantwort zu liefern.

Canny Operator: *Kantenhervorhebung* und *Erzeugung von Kantenzügen*.

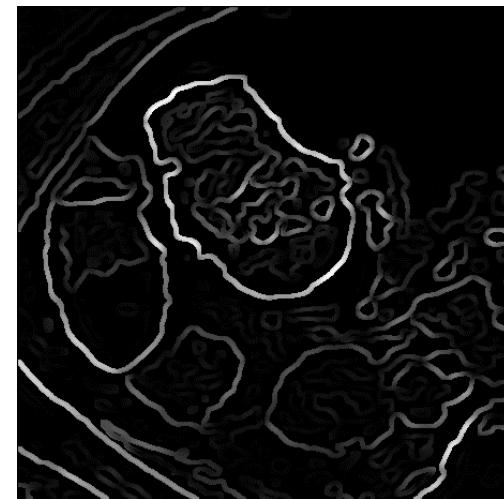
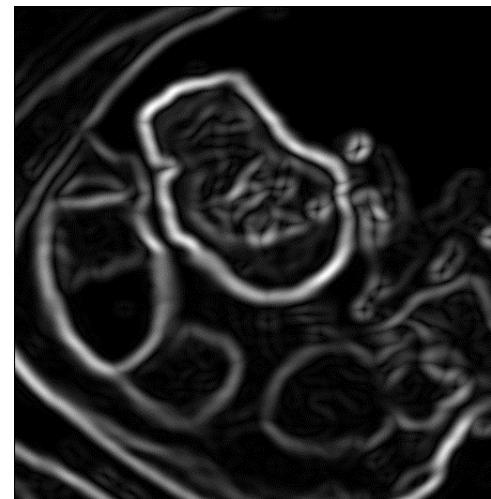
Optimale Kantenhervorhebung ist eine Filterung mit einer 1-D abgeleiteten Gaußfunktion orthogonal zur Kante.

Geringfügig schlechtere Ergebnisse erzielt man mit einem 2-D Gradientenoperator auf der Basis abgeleiteter Gaußfunktionen.



Canny Edge Detection (Beginn)

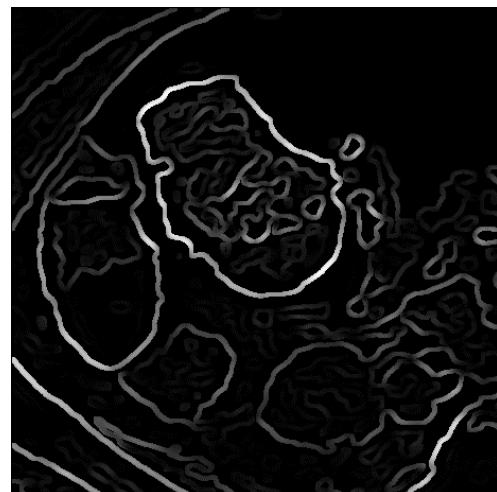
1. Anwendung eines Gradientenoperators (z.B. Sobel-Operator)
2. Non-Maximum Suppression (z.B. über Nulldurchgänge)



Canny Edge Detection (Fortsetzung)

3. Kantenverfolgung - Startpixelsuche

- Es wird immer dasjenige Pixel mit größter Gradientenlänge selektiert.
- Startpixel können nur Pixel sein, deren Gradientenlänge oberhalb einer Signifikanzschwelle T_1 liegt.

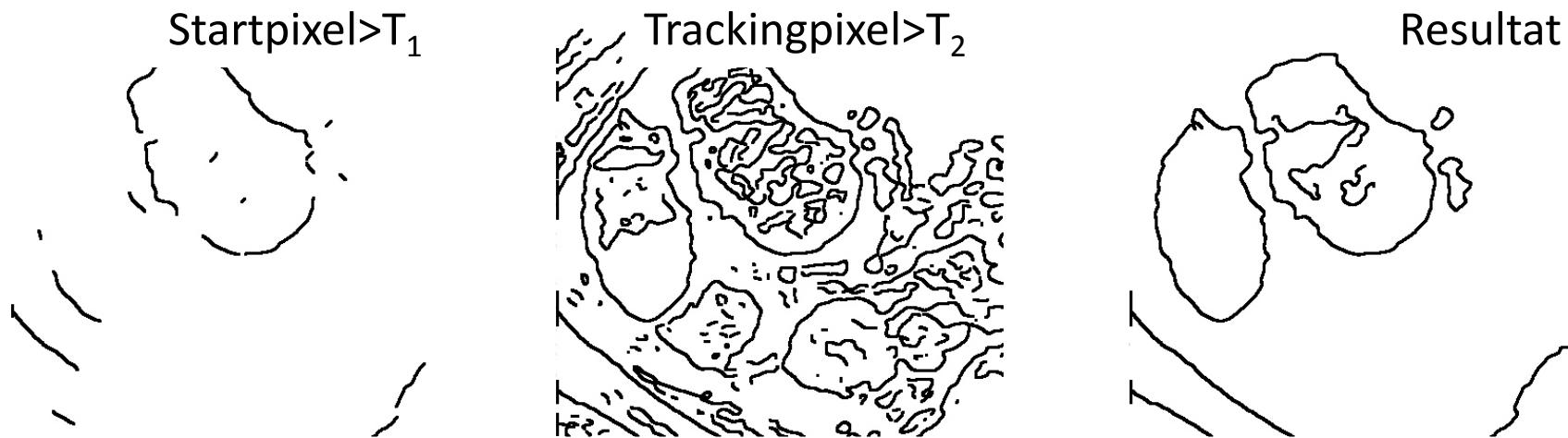


Canny Edge Detection (Fortsetzung)

4. Kantenverfolgung - Tracking

- Neue Kantenpixel haben eine *Gradientenlänge* $> T_2$ ($T_2 < T_1$) und sind zu einem bereits gefundenen Kantenpixel benachbart.

Verfahren endet, wenn keine neuen Startpixel gefunden werden.



Nulldurchgänge zur Segmentierung

- Die Orte der Nulldurchgänge der zweiten Ableitung sind Ränder von zusammenhängenden Gebieten.
- 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)



Nulldurchgänge

- Die Menge der Nulldurchgänge bildet immer geschlossene Kurven = Segmente
- Kombination des Laplace-Operators mit Glättungsoperator (z.B. als *LoG*-Operator) reduziert die Anzahl der Nulldurchgänge



Wasserscheidentransformation

- Wasserscheide: Menge aller Orte, die die Grenzen der Entwässerung in unterschiedliche Senken sind.
- **Beispiel:** Wasserscheide zwischen Nordsee und Mittelmeer verläuft entlang des Kamms der Berner Alpen.
- Wasserscheide in der Segmentierung: Generiere ein Höhenprofil so, dass die Wasserscheiden gerade die gesuchten Segmentgrenzen sind.

Wasserscheiden

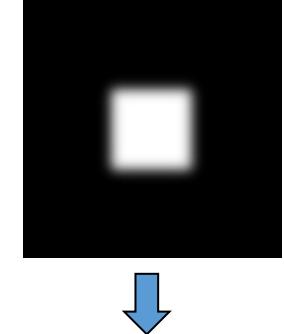
- Wasserscheiden sollen an Kanten verlaufen.
- Wasserscheiden sind „Gebirgskämme“
- „Geländehöhen“ sind die Längen der Grauwertgradienten.



Wasserscheidentransformation

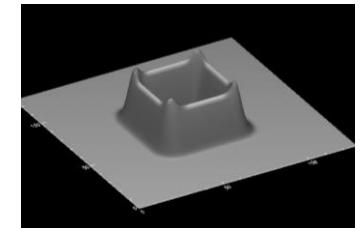
- Berechnung

Es fällt „Regen“ auf jedes Pixel. Anhand des Gradienten wird entschieden, wohin der Regen entwässert wird.

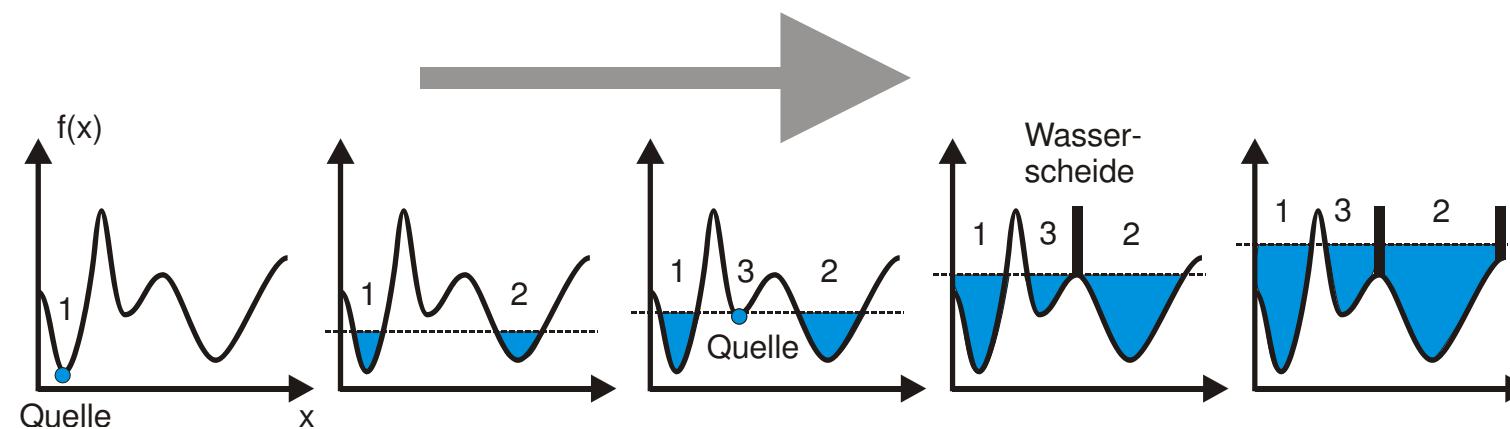


- Flutung

Die „Welt“ wird von den Senken her geflutet. Immer wenn Wasser aus zwei Senken zusammen fließt, entsteht eine Wasserscheide.



„Gradientengebirge“



Flutungsalgorithmus (Skizze)

Jedes neu überflutete Pixel (m_f, n_f) ist

- **in Isolation:**

Es nicht zu anderen überfluteten Pixeln der Höhen $h < h_{aktuell}$ benachbart.

Isolierte Pixel sind Kerne von neuen Segmenten.

- **Erweiterung:**

Es ist zu anderen überfluteten Pixeln der Höhen $h < h_{aktuell}$ mit gleichem Label benachbart.

Das Pixel wird dem Segment mit diesem Label zugeordnet.

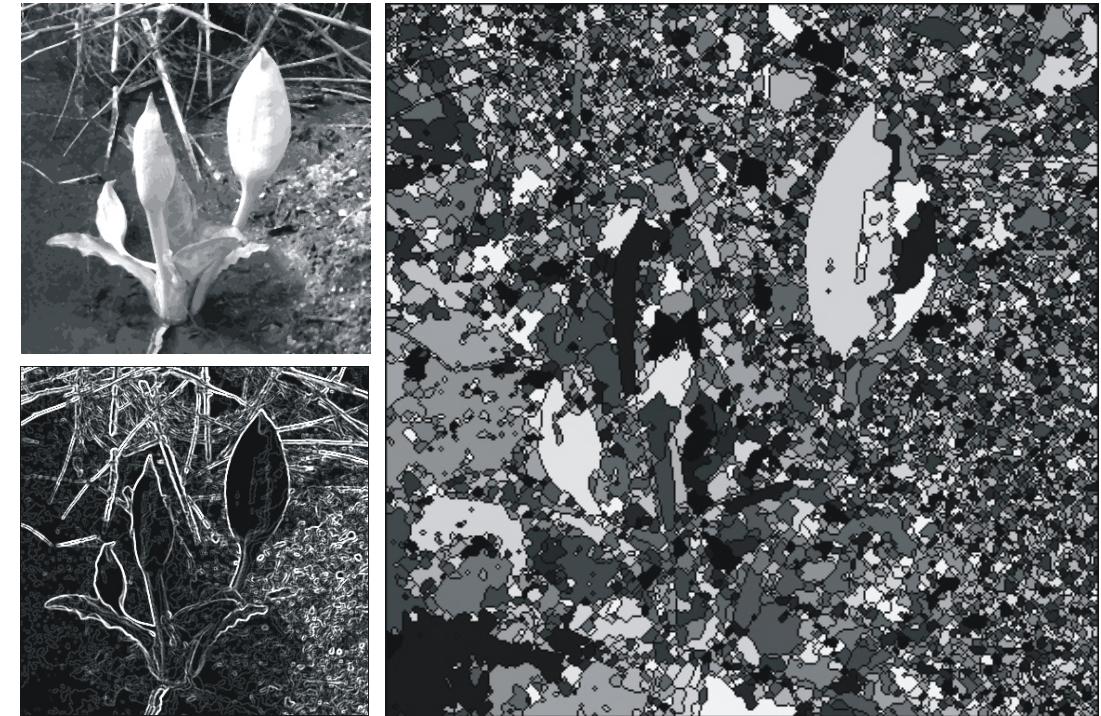
- **Wasserscheide:**

Es zu überfluteten Pixeln von mindestens zwei Regionen benachbart.

Dem Pixel wird das Label „Wasserscheide“ zugeordnet.

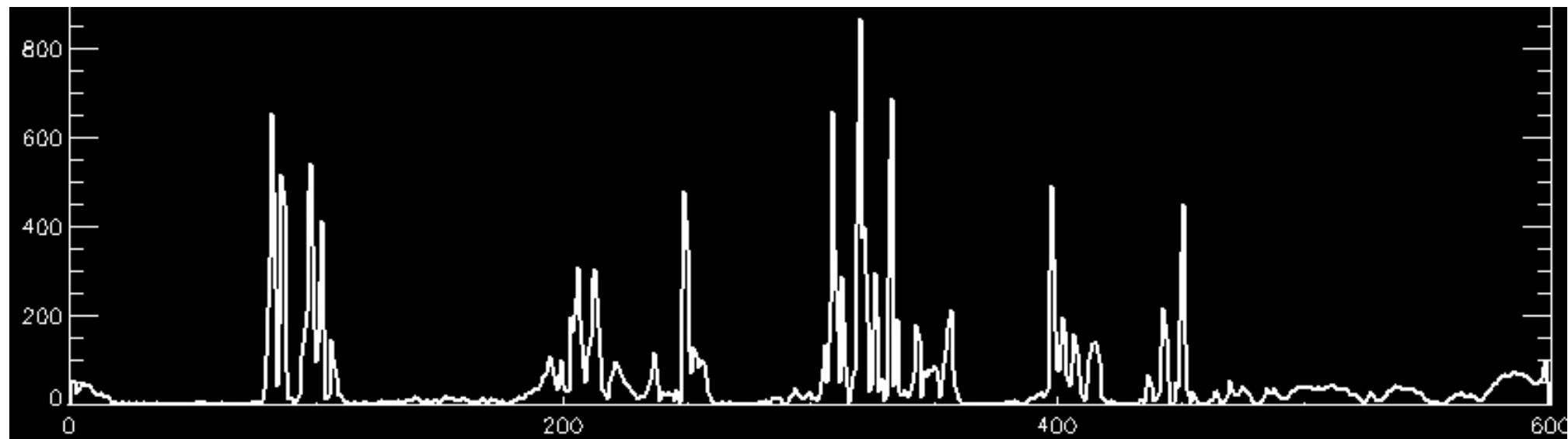
Resultat der WST

- WST auf Gradienten
 - Segmentgrenzen sind lokale Maxima in Gradientenrichtung
 - Maxima der ersten Ableitung = Nulldurchgänge der zweiten Ableitung
- ⇒ WST = Suche nach Nulldurchgängen



Problem Übersegmentierung

- Für die WST ist jedes lokale Minimum eine Senke.
- Die meisten Senken werden durch Rauschen verursacht.
- Senken durch Rauschen sind weniger tief als die von Kanten.



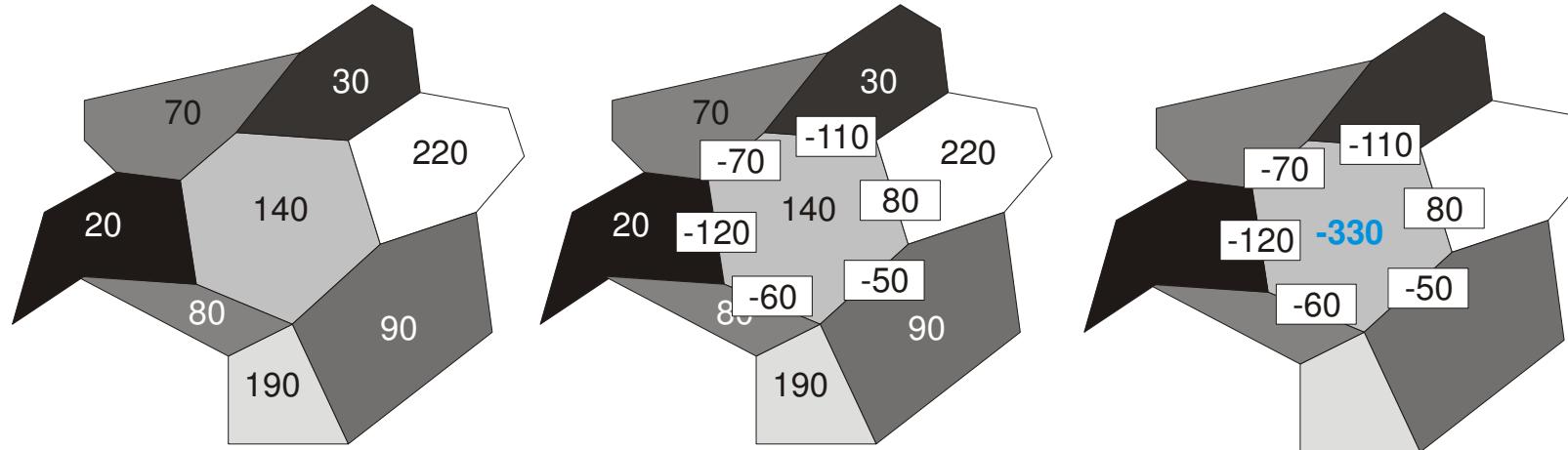
Hierarchische WST

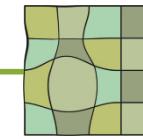
Multiskalenstrategie:

- Wasserscheidentransformation auf dem WST-Resultat.
- Jede Region erhält ihren durchschnittlichen Grauwert als Funktionswert.
- Die erste WST wird hauptsächlich durch Rauschen verursachte Senken finden.
- „Wahre“ Senken sollten über mehrere Stufen der Hierarchie erhalten bleiben.

Gradienten für die hWST

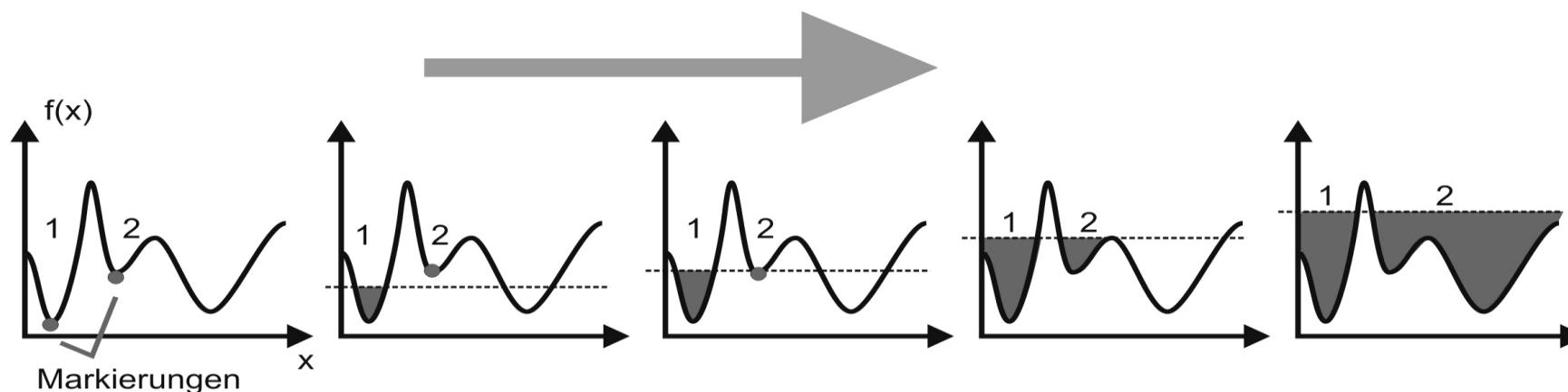
- Zu jeder der benachbarten Regionen wird die Differenz berechnet.
- Die Länge des Gradienten ist die durchschnittliche Differenz zu allen Regionen.
- Die Richtung ergibt sich aus der (mit der Regionengröße gewichteten) Vektoren zwischen dem Schwerpunkt der Region zu den Schwerpunkten aller benachbarten Regionen.





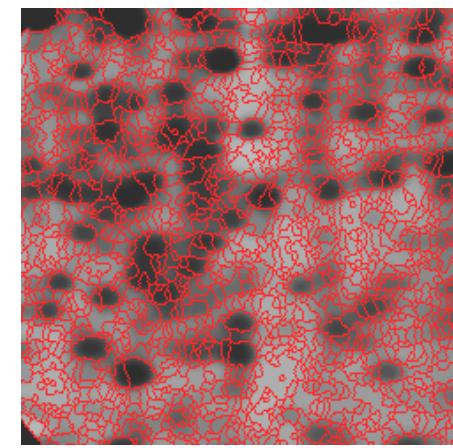
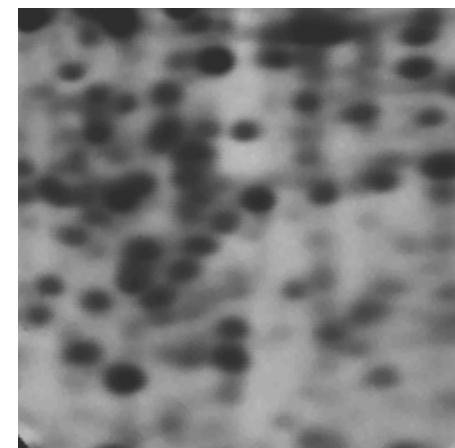
Markerbasierte WST

- Flutung erfolgt nur von vorher definierten Markierungen aus.
- Abwandlung des Flutungsalgorithmus:
 - Flutung erfolgt wie vorher von den Senken aus
 - Regionen erhalten das Label „undefined“, wenn sie nicht von einer Markierung aus geflutet werden
 - Wenn eine Region „undefined“ mit einer mit Label versehenen Region zusammenfließt, dann erhält sie dieses Label.



Markierungen suchen

- mWST ist im Gegensatz zur WST ein Segmentierungsverfahren, das Objektwissen (Objektorte) benötigt.
- Markierungen können z.B. erzeugt werden, falls die gesuchten Objekte mindestens in einem Punkt anhand der Helligkeit identifizierbar sind.
- Beispiel: Segmentierung in der Elektrophorese oder Zellsegmentierung.



Nachverarbeitung von Segmenten

- Übersegmentierung: Homogenität ist durch Rauschen gestört
- Charakterisierung der Störung: Rauschen verursacht sehr kleine Segmente
- Nachverarbeitung: kleinere Segmente werden benachbarten großen Segmenten zugeordnet
- Problem: Was ist das „Mastersegment“
- Lösung: relaxierendes Verfahren



Relaxation Labelling

- Jedes Pixel erhält für jedes Label eine vorgegebenen Wahrscheinlichkeit
(Bsp.: Schwellenwertresultat, Wahrscheinlichkeiten „70% weiß“ und „30% schwarz“ für weiße Pixel; umgekehrt für schwarze Pixel)
- Benachbarte kompatible Pixel unterstützen sich.
- Relaxationsprozess: Zuordnungswahrscheinlichkeiten ändern sich mit dem Maß der Unterstützung.
- Zu definieren:
 - Kompatibilität
 - Einfluss der Kompatibilität auf die Labelwahrscheinlichkeiten.

Labelwahrscheinlichkeit

- Umsortierung aller Pixel in Liste p_0, p_1, \dots, p_N .
- Initiale Labelwahrscheinlichkeit P^0 für jedes Pixel p_i und jedes Label l_k vergeben, z.B.

$$P^0(p_i, l_k) = \begin{cases} 0.8 & , \text{falls } l_k = l(p_i) \\ 0.2 & , \text{sonst} \end{cases}$$

- Labelwahrscheinlichkeit
 - gibt an, wie sicher man sich nach der Segmentierung über das zugeordnete Label ist
 - darf nicht 0 oder 1 sein (Gewissheiten werden nicht verändert)

Kompatibilität

- Ein Pixel p_i mit Label l_k hat eine Kompatibilität r mit einem Pixel p_j , dessen Label l_l sei:

$$r((p_i, l_k), (p_j, l_l))$$

- Kompatibilität für Binärbilder (2 Label) z.B.

$$r((p_i, l_k), (p_j, l_l)) = r(l_k, l_l) = \begin{cases} 1 & , \text{falls } l_k = l_l \\ 0 & , \text{sonst.} \end{cases}$$

(d.h., nur gleiche Label unterstützen sich)

- Kompatibilität bei mehr als zwei Labeln kann auch bedeuten, dass sich bestimmte Labelpaare unterstützen (*auch wenn es unterschiedliche Label sind*)

Unterstützung eines Pixels

Unterstützung $q^{(n)}$ von Pixel p_i von Pixel p_j zur Iteration n

$$q_j^{(n)}(p_i, l_k) = \sum_{l=0}^{K-1} P^{(n)}(p_j, l_l) \cdot r((p_i, l_k), (p_j, l_l))$$

- Erinnerung: Pixel p_j hat für *jedes* Label eine von Null verschiedene Labelwahrscheinlichkeit
- Labelwahrscheinlichkeit für ein Label l_l wird mit der Kompatibilität zwischen l und dem zu unterstützenden Label l_k gewichtet
- Summe ist dann die Unterstützung durch alle Label

Unterstützung eines Pixels

Unterstützung von p_i durch alle Pixel

$$Q^{(n)}(p_i, l_k) = \sum_{j=0}^{NM-1} c_{ij} q_j^{(n)}(p_i, l_k)$$

mit Einflussparameter c_{ij} , z.B.

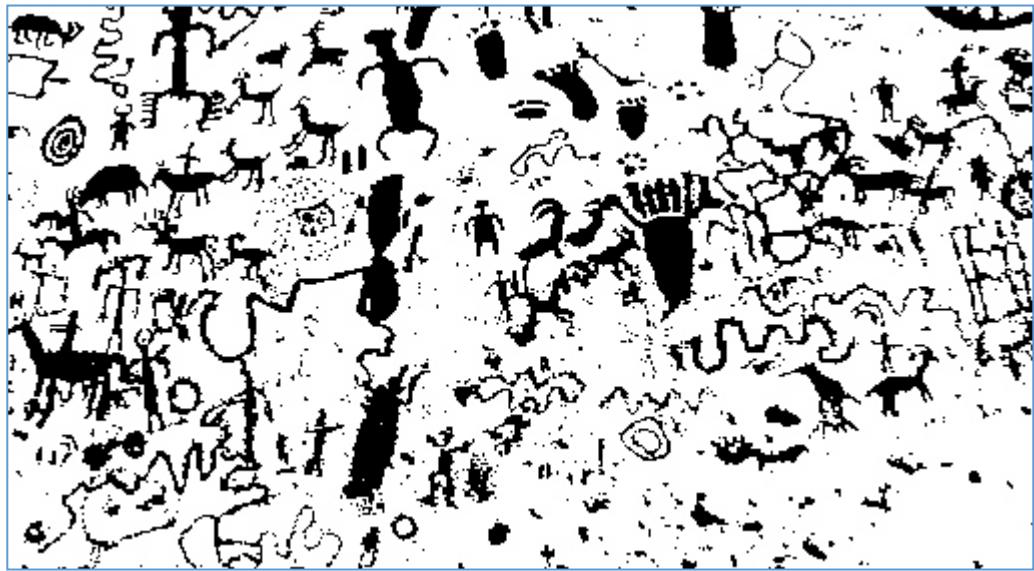
$$c_{ij} = \begin{cases} \frac{1}{8} & , \text{falls } p_j \in N_8(p_i) \\ 0 & , \text{sonst.} \end{cases}$$

Nachbarschaft kann sehr allgemein definiert sein und der „Grad der Nachbarschaft“ wird durch die Gewichtung c_{ij} berücksichtigt

Iterationsschritt

$$P^{(n+1)}(p_i, l_k) = \frac{P^{(n)}(p_i, l_k)[1 + Q^{(n)}(p_i, l_k)]}{\sum_{l=1}^{K-1} P^{(n)}(p_i, l_l)[1 + Q^{(n)}(p_i, l_l)]}.$$

- Unterstützung Q durch die umgebenden Pixel wird addiert.
- Division durch die Summe aller Labelwahrscheinlichkeiten dient der Normierung

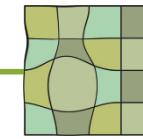


Confidence Map

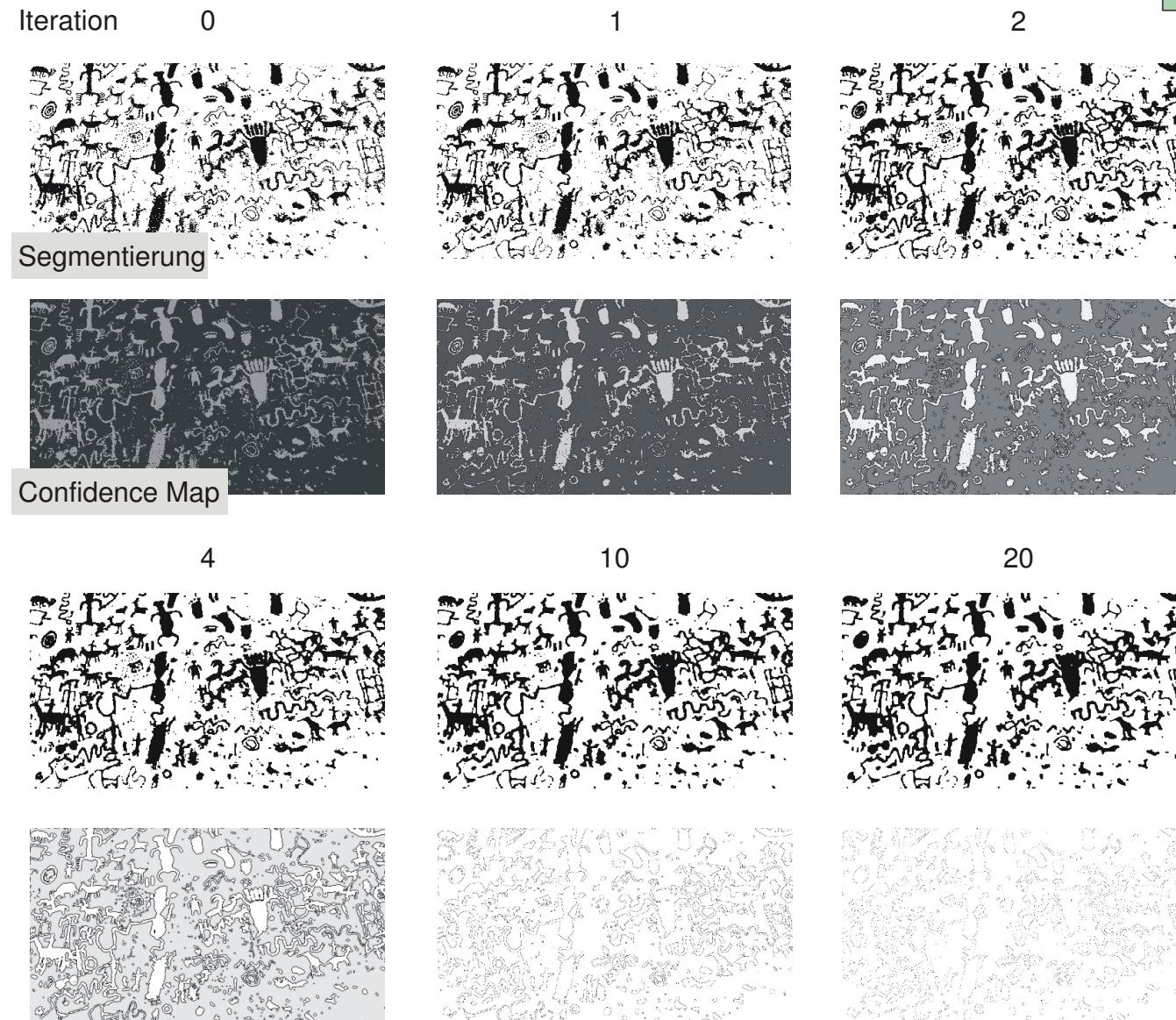
- Um das Konvergenzverhalten zu beobachten, kann eine **Confidence Map** erzeugt werden.
- Confidence Map: Gibt für jedes Pixel die Zuverlässigkeit der derzeitigen Entscheidung an
- Für 2-Label-Segmentierung: Differenz zwischen gewählten Label und nicht gewähltem Label (evtl. gewichtet mit der Anzahl der Pixel mit diesem Label).



Initiale Confidence Map

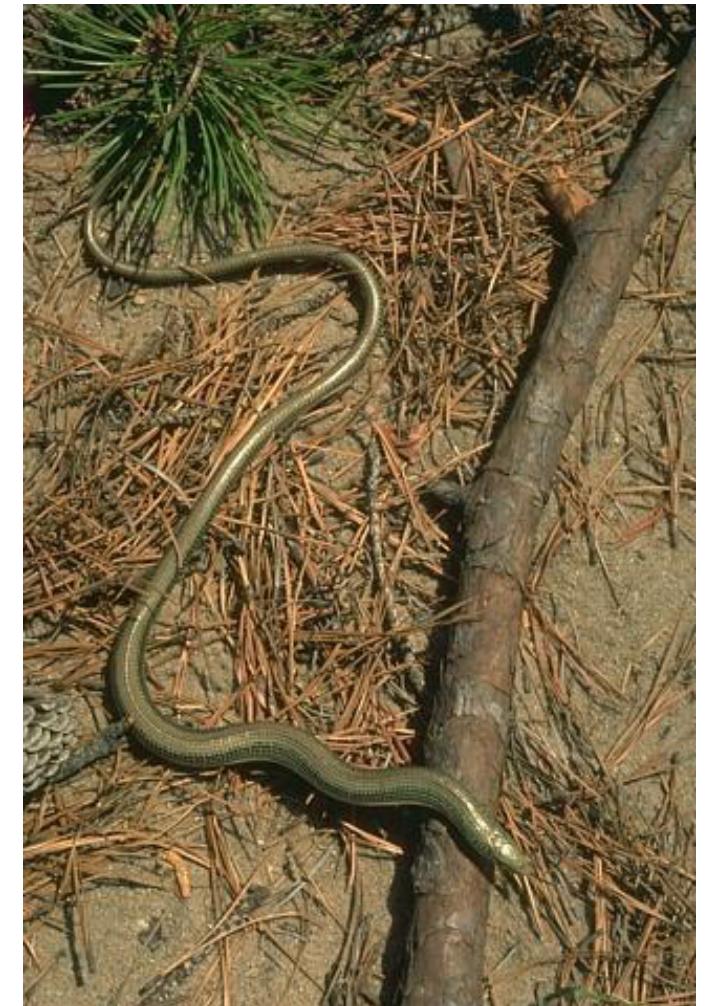


Konvergenz



Zwischenfazit Segmentierung

- Segmentierungskriterien
 - möglichst homogene Gebiete
 - möglichst kurze Segmentgrenzen
- Probleme
 - Homogenitätskriterium kann schwer zu definieren sein (vor allem bei Beleuchtungseinflüssen)
 - Homogenität kann in unterschiedlichen Bereichen etwas unterschiedliches bedeuten
 - Segmentgrenzen sind nicht immer Grauwertkanten



Berkeley Segmentation Dataset

Warum es sich dennoch lohnt

- Es gibt Anwendungen, bei denen die Bilder (die Bildklasse) nicht vorher bekannt ist, z.B.
 - Komprimierung (z.B. MPEG-7)
 - Suche in Bilddatenbanken
- Was hier benötigt wird
 - Segmente, die charakteristisch sind (d.h. in ähnlichen Bildern ähnlich gefunden werden)
 - Segmente, die mehr Bedeutung tragen als Pixel
- Man kann aber auch anders segmentieren, wenn ein bestimmtes Objekt gesucht ist \Rightarrow Segmentierung mit Objektwissen

Was Sie heute gelernt haben sollten

- Edge Linking und Canny Edge Operator
- Nulldurchgänge zur Regionensegmentierung
- Wasserscheidentransformation
 - Multiskalenstrategie
 - WST mit Markern
- Relaxation Labeling

Famous Last Question

Wie könnte man Segmentierung und Objektwissen verbinden, um die Figuren zu extrahieren?

Welche Art von Objektwissen?



Famous Last Question

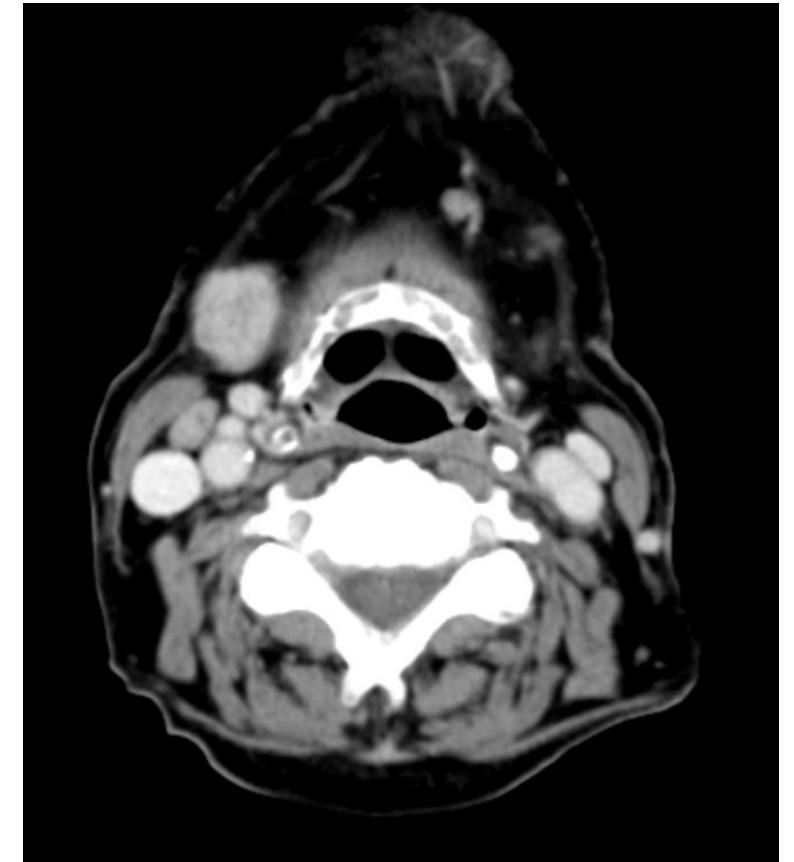
Wie könnte man Segmentierung und Objektwissen verbinden, um die Figuren zu extrahieren?

Welche Art von Objektwissen?



Modellbasierte Segmentierung

- Interaktive Suche
 - Markerbasierte WST
 - Region Growing
 - Kantenverfolgung
- Vollständige Suche
 - Template Matching
 - Hough Transformation



Modellbasierte Segmentierung

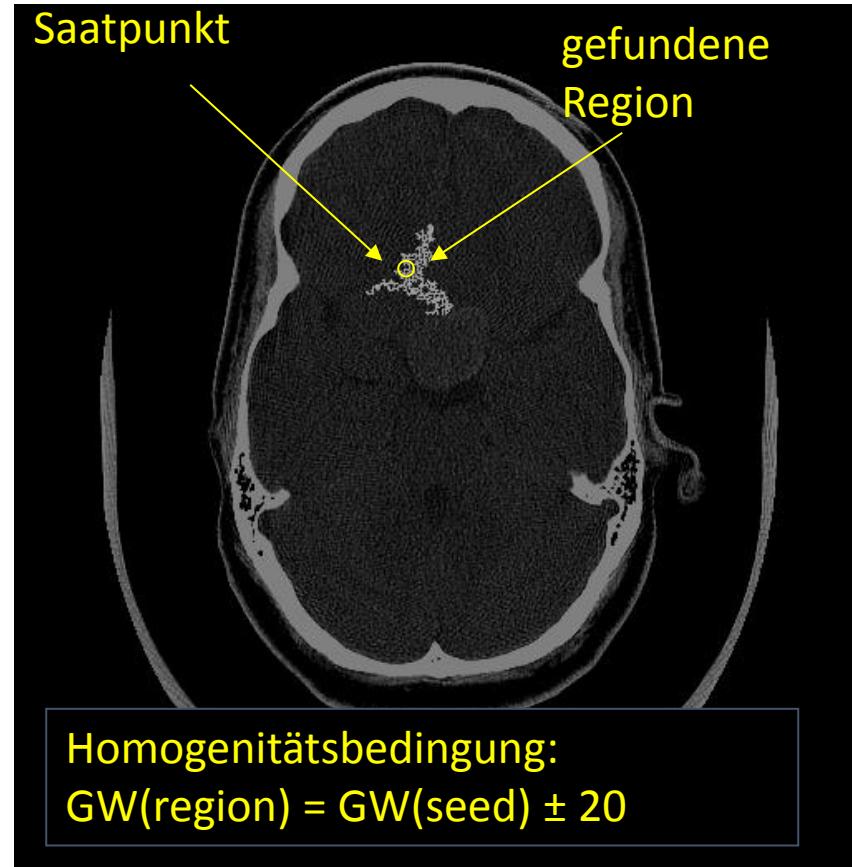
- Segmentierung: Generierung von Symbolen (Bedeutungsträgern) aus Pixeln.
 - Modell: Erwartete Bedeutung
- Widerspruch!

Modellbasierte Segmentierung: Mixtur zwischen Segmentierung und Analyse. Mit Vorwissen über ein bestimmtes Objekt wird nach Instanzen gesucht

- Interaktive Suche: Benutzer gibt Vorwissen interaktiv ein
- Vollständige Suche: Instanzen eines Modells mit wenigen Parametern werden gesucht
- Iterative Suche: Instanzen eines Modells mit vielen Parametern werden gesucht.

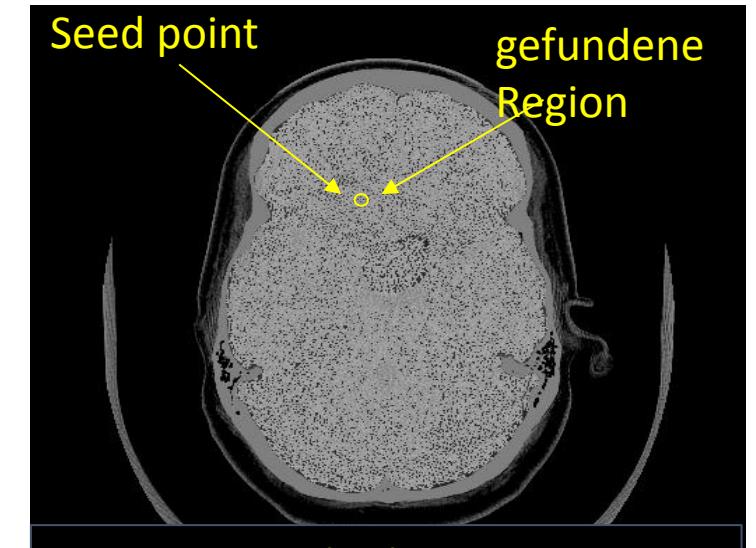
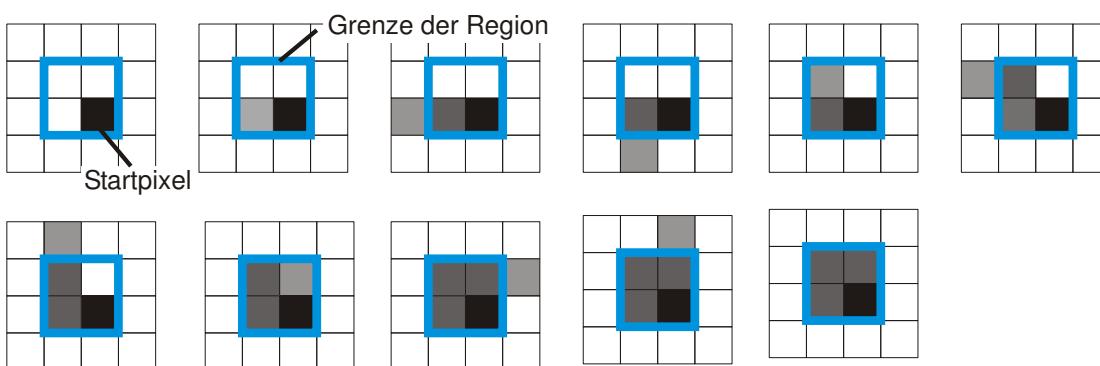
Region Growing

- Flood Fill auf Grauwertbild für einen einzigen (vorgegebenen) Saatpunkt.
- Homogenitätsbedingungen:
 - Grauwertbereich
 - Grauwertschwankung
- Annahmen und Ziel:
 - Gebiet ist intern von größerer Homogenität als an den Grenzen.
 - Selektion eines einzigen Gebiets.
 - Homogenitätsverhältnisse an anderen Orten interessieren nicht.

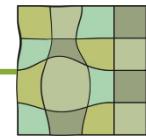


Region Growing

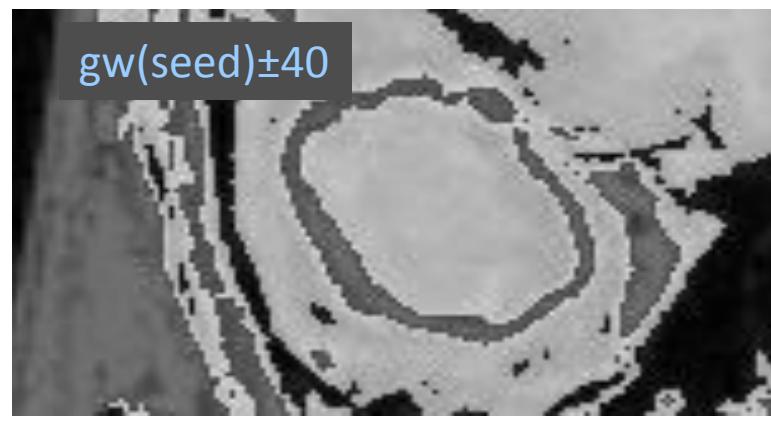
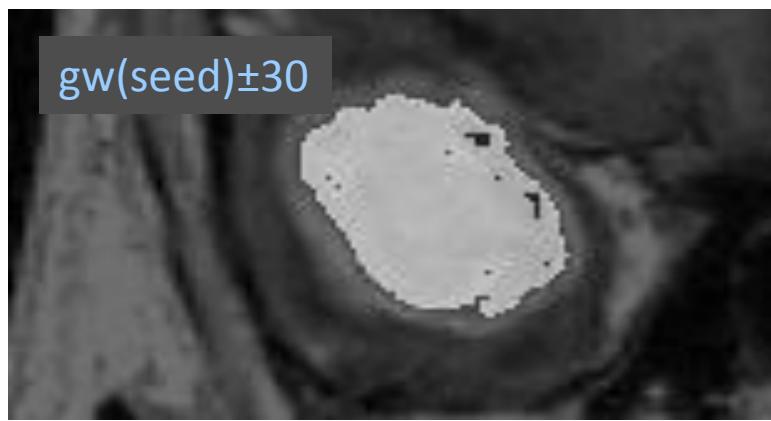
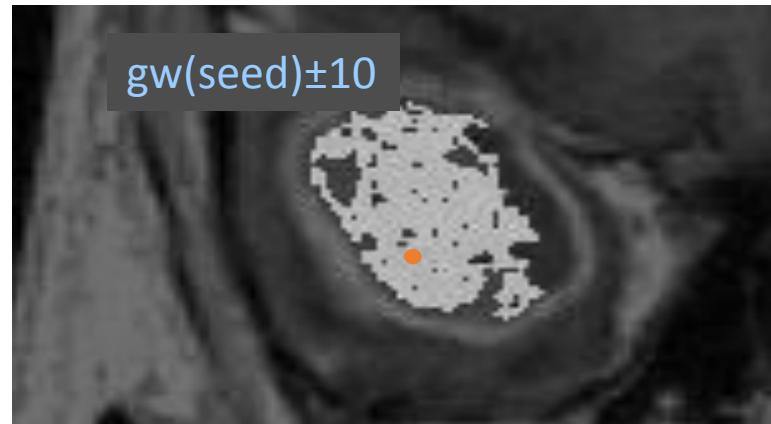
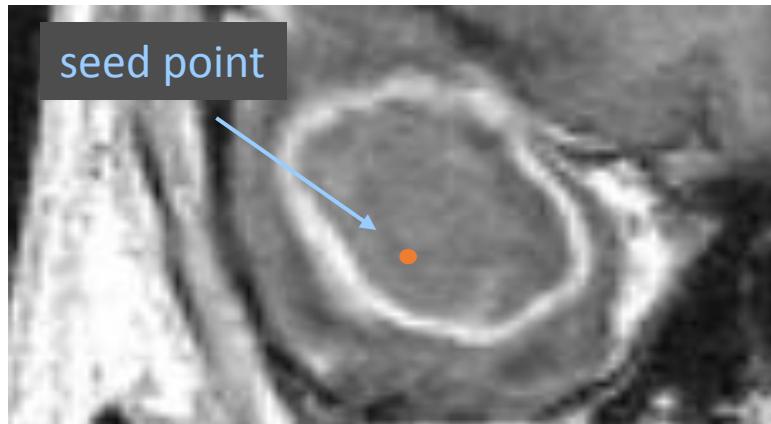
- Probleme:
 - „Auslaufen“ der Regionen
 - zu kleine Regionen
 - Rauschanfälligkeit
 - Shading im Bild



Homogenitätsbedingung:
 $GW(\text{region}) = GW(\text{seed}) \pm 70$



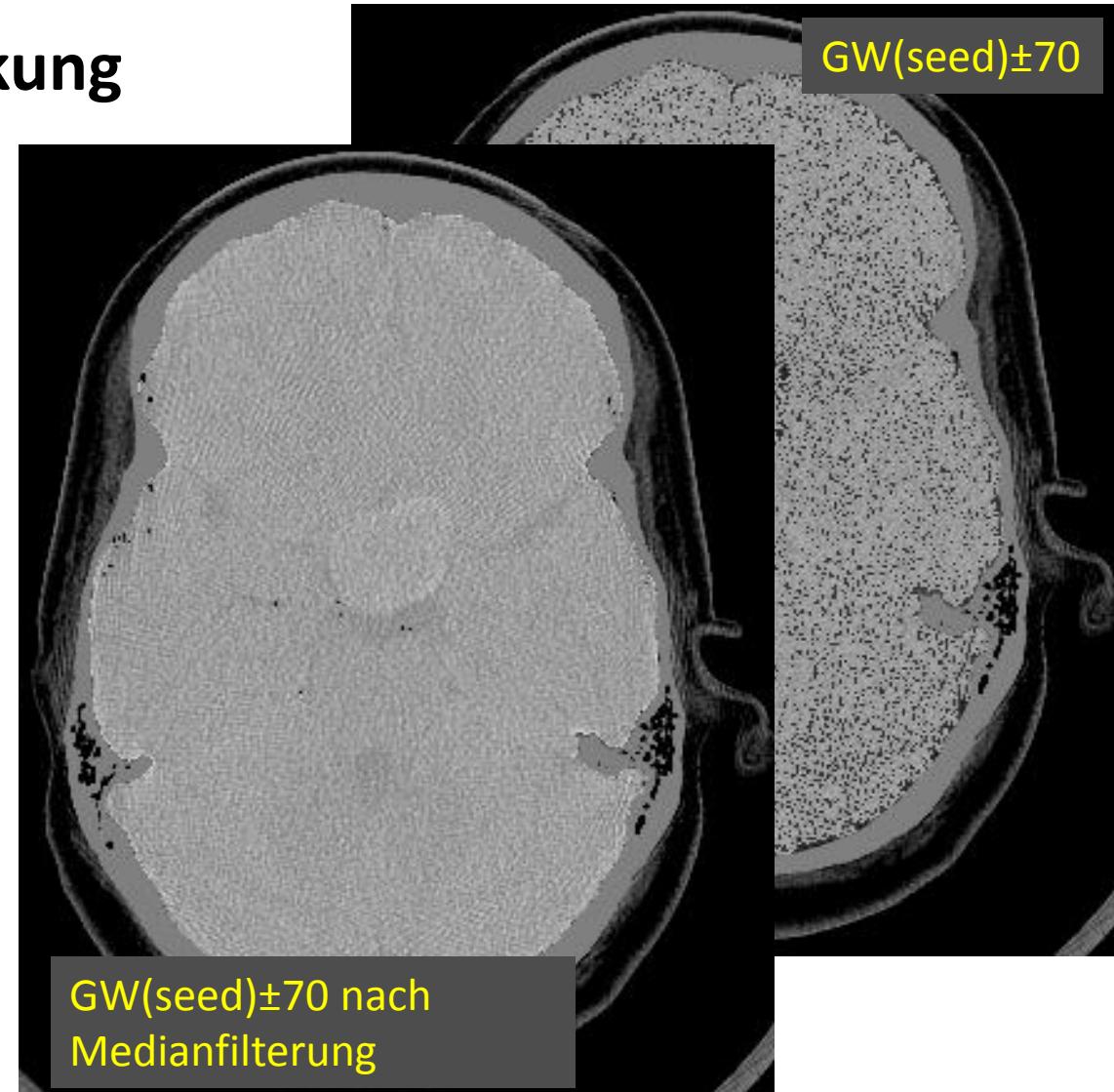
Auslaufen von Regionen



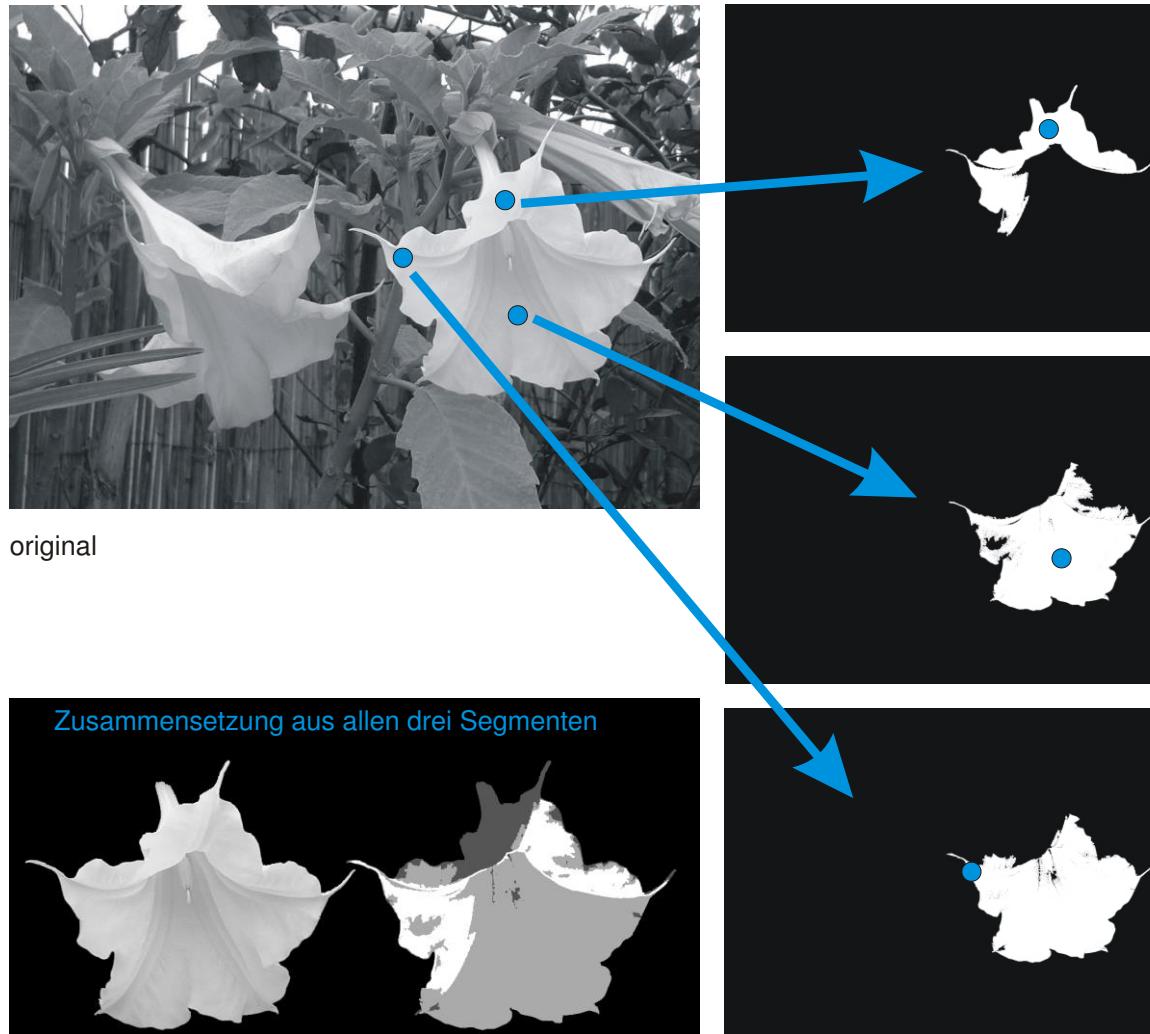
Region Growing und Rauschunterdrückung

Rauschen führt zu Anfälligkeit der Methode

→ vorherige [Rauschunterdrückung](#) oder
Nachverarbeitung



Mehrfaches Region Growing



Region Growing

- interaktives, recht intuitives Werkzeug
 - Startpunktselektion sollte für „Datenexperten“ trivial sein
 - Homogenitätskriterium kann schwieriger zu bestimmen sein.
- geeignet, falls das gesuchte Objekt homogen ist und sich überall vom Hintergrund abhebt.
- „Auslaufen“ kann durch unterschiedliche Maßnahmen begrenzt werden.
- Grundsätzlich auch in 3-d anwendbar, aber weniger intuitiv, da Resultate nicht einfach visualisiert werden können.

Gezielte Kantensuche

- Resultat der Kantenfilter:
- Liste von Kantenpunkt-kandidaten
- nicht alle Kandidaten gehören zur gesuchten Regionengrenze.
- die Regionengrenze kann Lücken aufweisen.
- Strategien mit Nutzerinteraktion
 - Optimale Kantenzüge
 - Hough Transformation



Optimale Kantenzüge (Pfadsuche)

Pixel sind Knoten in eines Graphen.

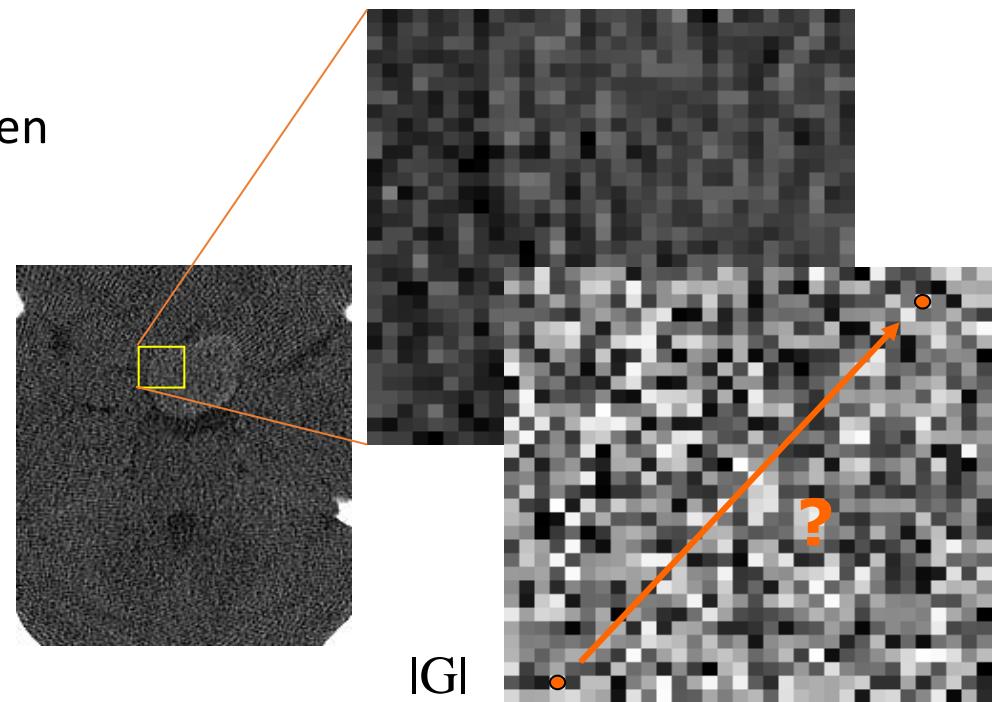
Benachbarte Pixel sind durch Kanten verbunden.

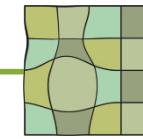
Knotenkosten spiegeln die Wahrscheinlichkeit wider, dass das betreffende Pixel zu einer Kante gehört

Gesuchter Kantenzug ist ein optimaler Pfad in diesem Graphen

Vorteil:

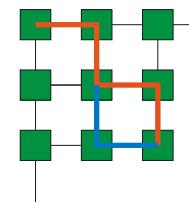
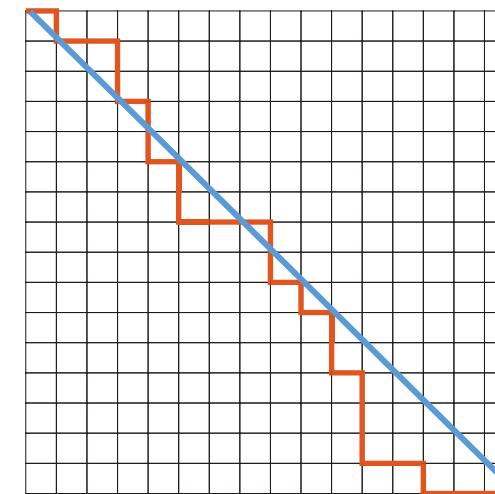
Globale Aspekte des Kantenzugs können
eingebbracht werden (z.B. Gesamtlänge des Kantenzugs)





Optimale Kantenzüge (Pfadsuche in Pixelgraph)

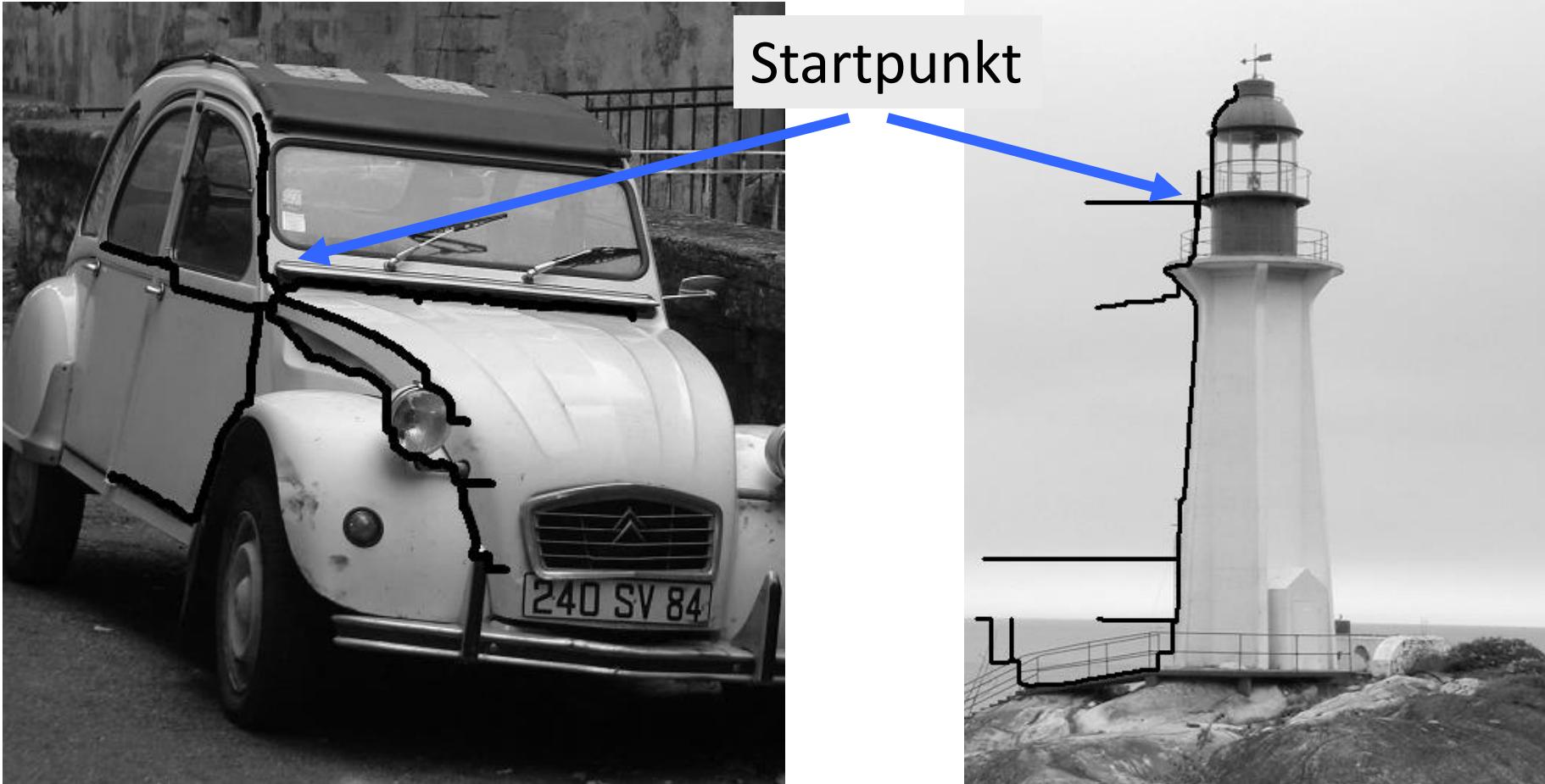
- Naheliegende Optimalitätskriterien:
 - Maximierung der Gradientenlänge (???)
 - Minimierung der Abweichungen von einer erwarteten Gradientenlänge
 - Minimierung der Pfadlänge
 - Minimierung der Richtungsänderungen
 - Minimierung der Grauwertänderungen



Live Wire

- Startpunkt liefert „Modellgradienten“
- Kriterien
 - wenig Abweichungen von der Länge des Modellgradienten
 - kurze Pfadlänge
- Methode
 - Dijkstra-Algorithmus zu allen möglichen Endpunkten
 - Pfad wird von gesetztem Endpunkt zurückverfolgt

Resultate



Live Wire - Kantensuche als Pfadsuche

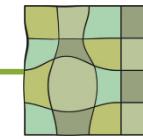
- Ist ein schnelles, intuitives Werkzeug zur interaktiven Segmentierung
- Eignet sich gut, wenn Kanten klar abgrenzbar sind, auch wenn die Gradientenstärke lokal variiert
- Kann sogar Lücken in Kantenzügen überbrücken (ist dann aber rauschanfällig)



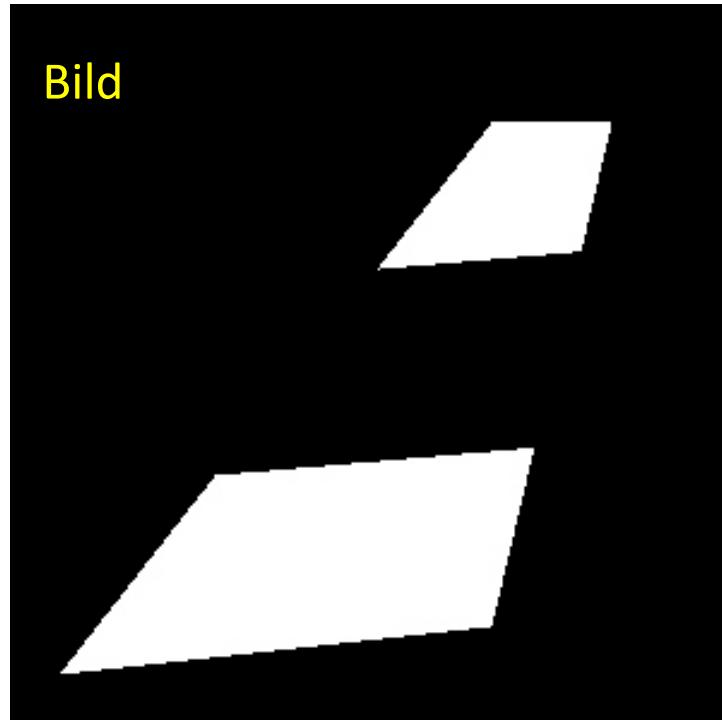
Template Matching

Segmentierung durch vollständige Suche:

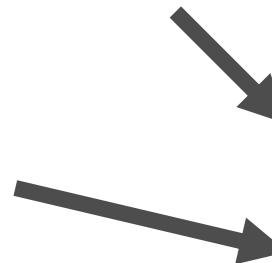
- Gezielte Suche nach einer vorgegebenen Form.
- Für jeden Ort im Bild wird ein Abstandsmaß zwischen Form und Bild berechnet.
- Der Ort, an dem das Maß am höchsten ist, ist die gesuchte Position des Objekts.
- Abstandsmaße zwischen Bild $f(x, y)$ und Template $t(x, y)$: z.B.
 - MAD: mean absolute difference: $MAD(x, y) = \frac{1}{MN} \cdot \sum_{xy} |f(x + m, y + n) - t(x, y)|$
 - Korrelationskoeffizient $cc(x, y)$



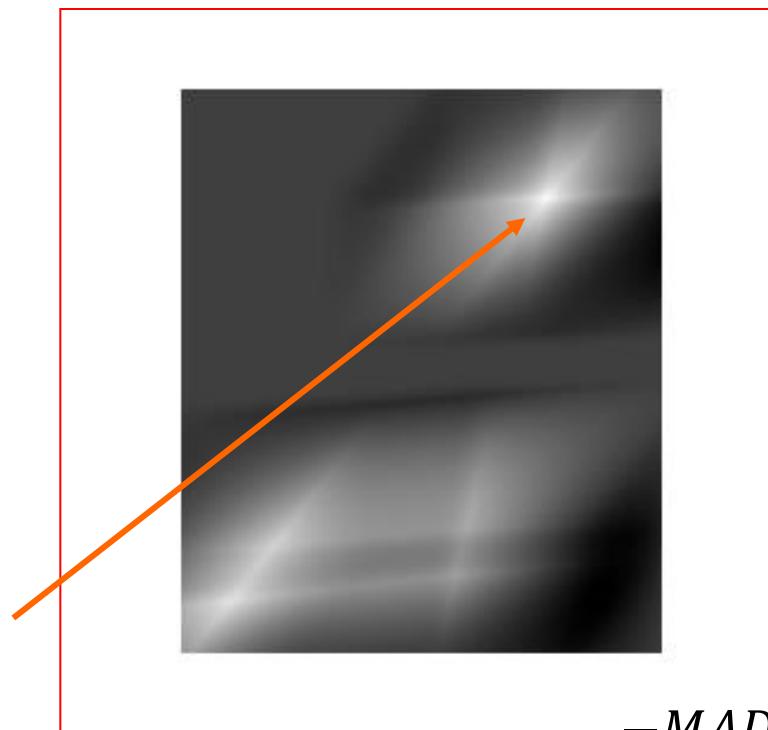
Template Matching



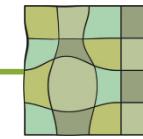
Template



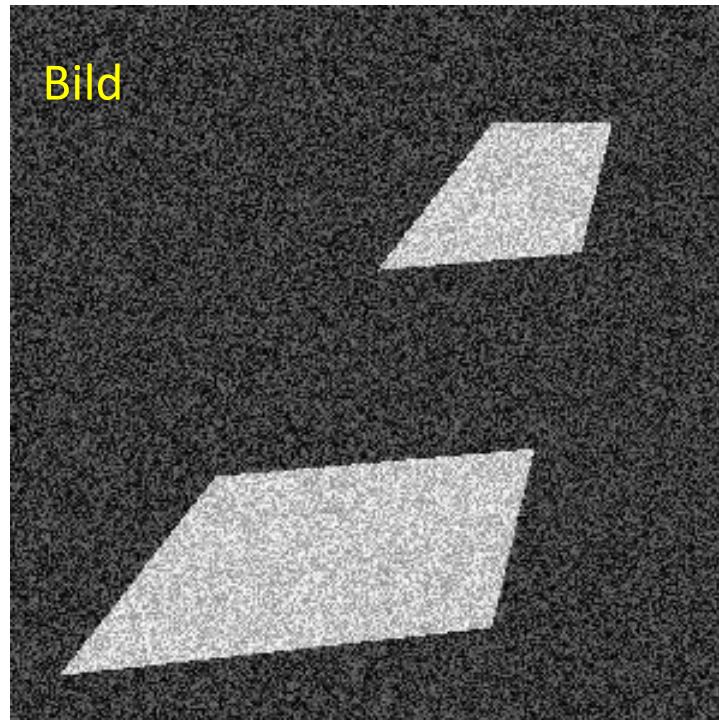
Maximum ist im Mittelpunkt
der gesuchten Struktur.



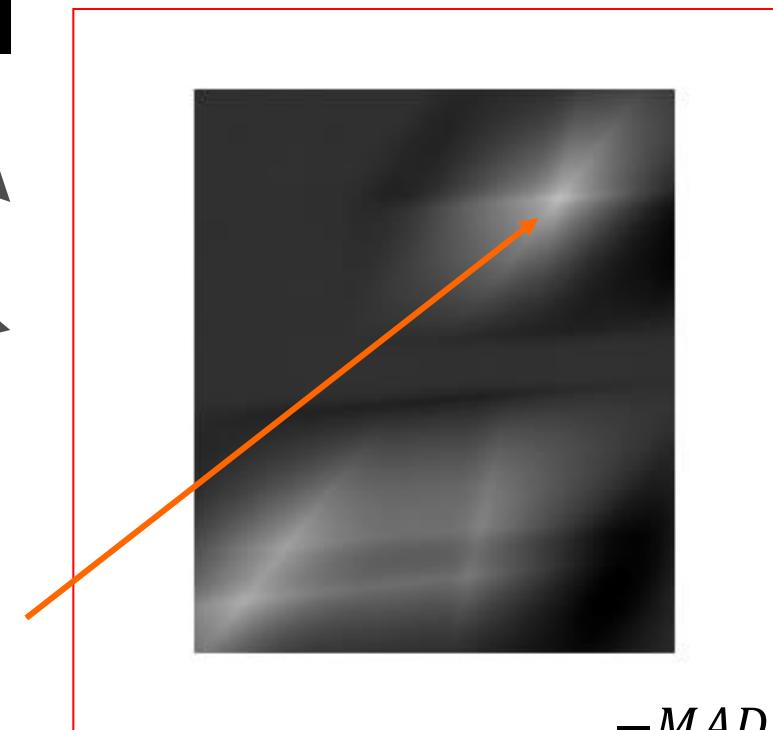
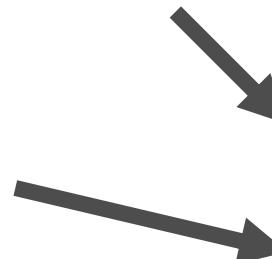
$$-MAD(x, y)$$



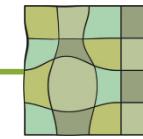
Template Matching - Rauschen



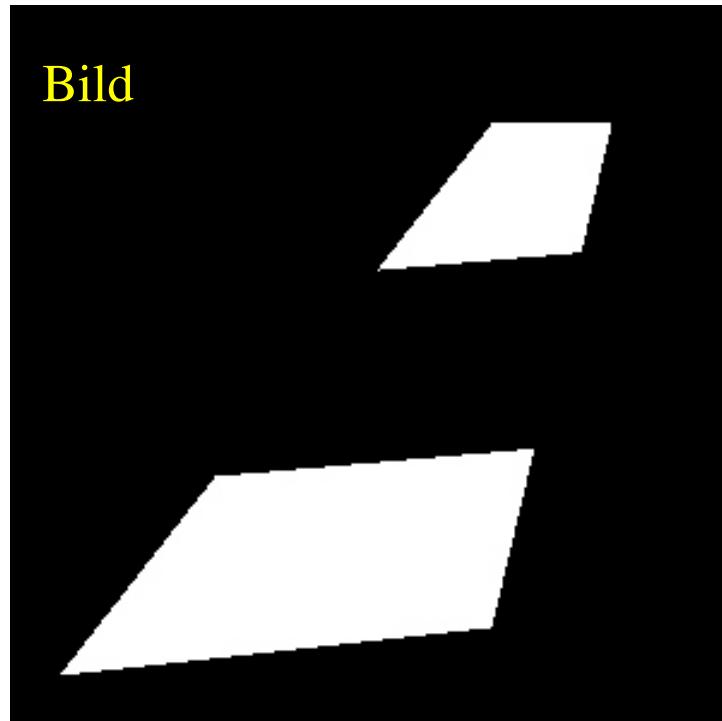
Template T_{org}



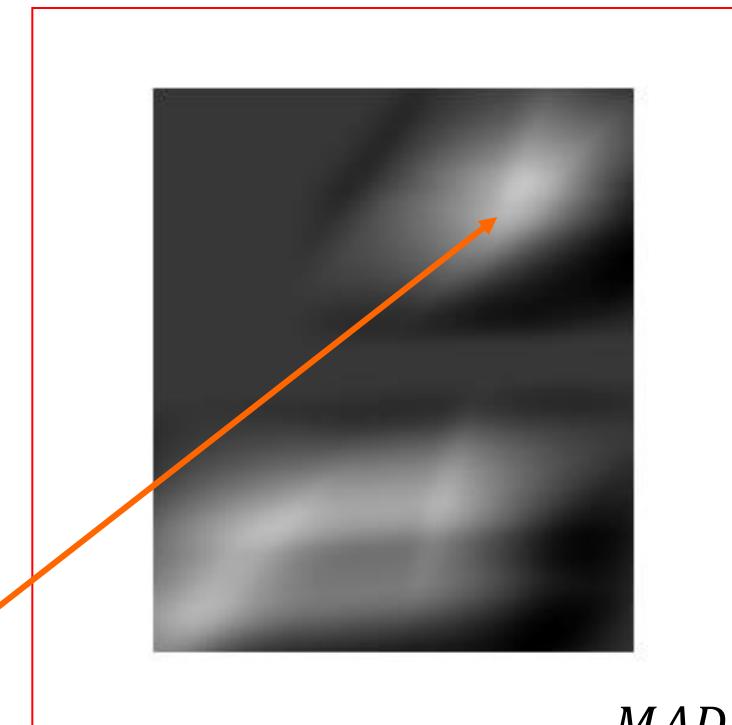
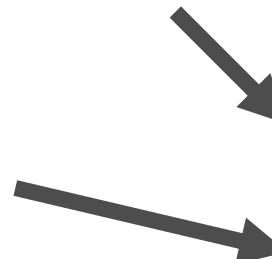
Wegen der integrierenden Komponente ist
Template Matching unempfindlich gegenüber
Rauschen



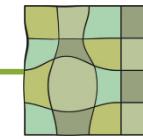
Template Matching - Probleme



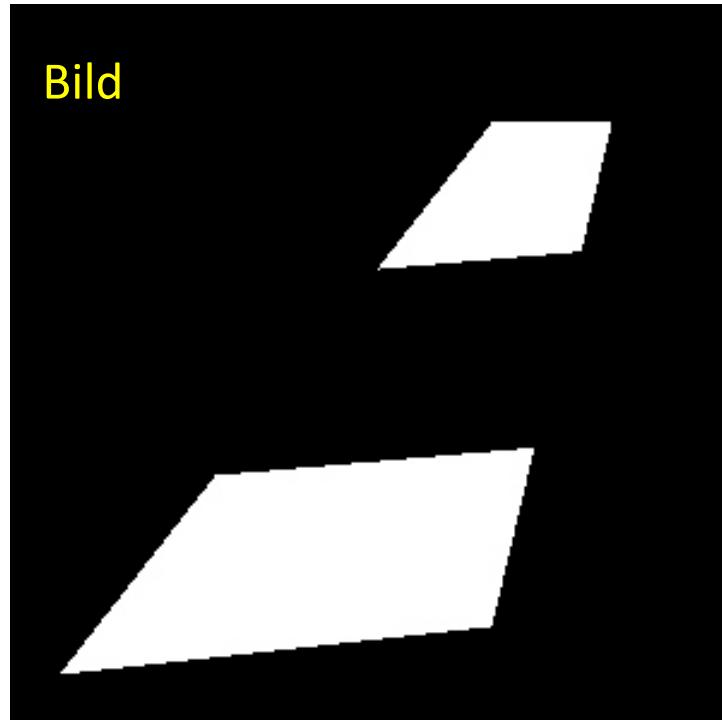
Template ($\text{Rot}[\text{Torg}, 15^\circ]$)



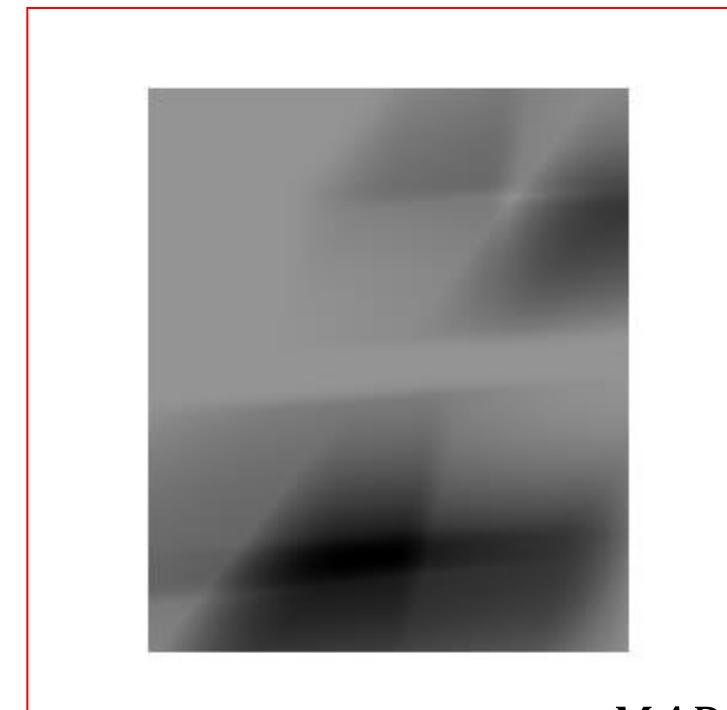
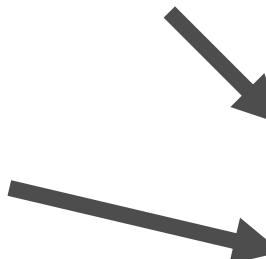
Match ist nicht perfekt: Werte der lokalen Maxima gleichen sich an.



Template Matching - Probleme

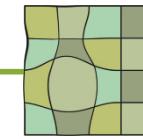


Template ($0.5 \cdot T_{org}$)

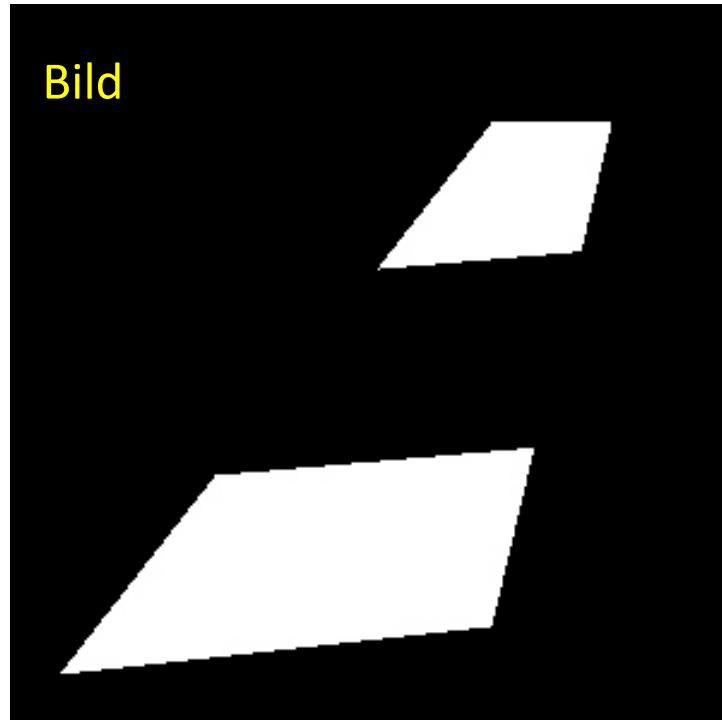


Falsch skalierte Grauwerte des Template:
Lokalen Maxima beginnen zu verschwinden.

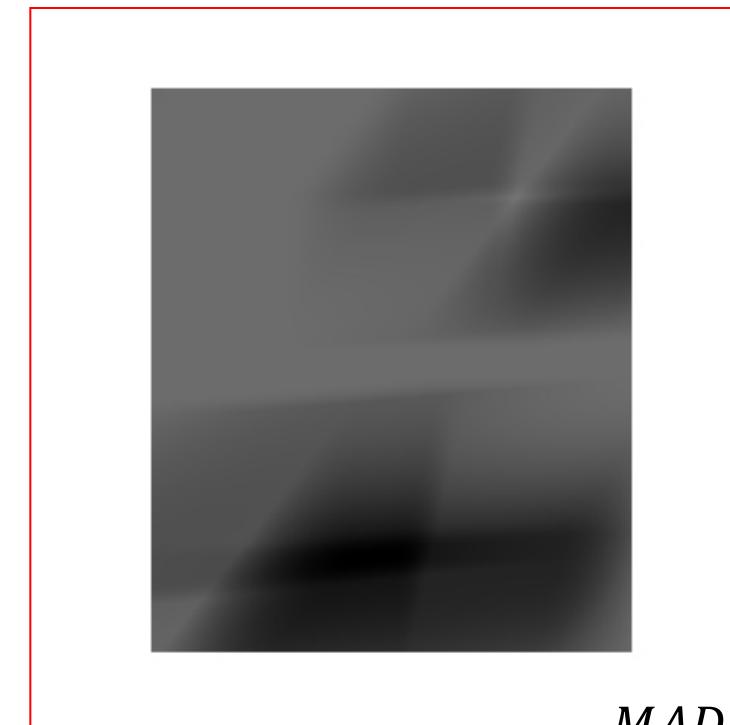
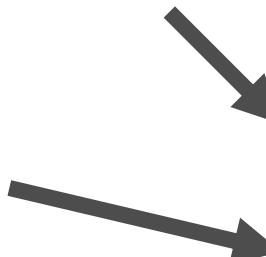
$-MAD(x, y)$



Template Matching - Probleme



Template ($T_{org} - 120$)



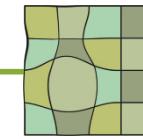
Falscher Durchschnittsgrauwert: Lokale Maxima beginnen zu verschwinden.

$-MAD(x, y)$

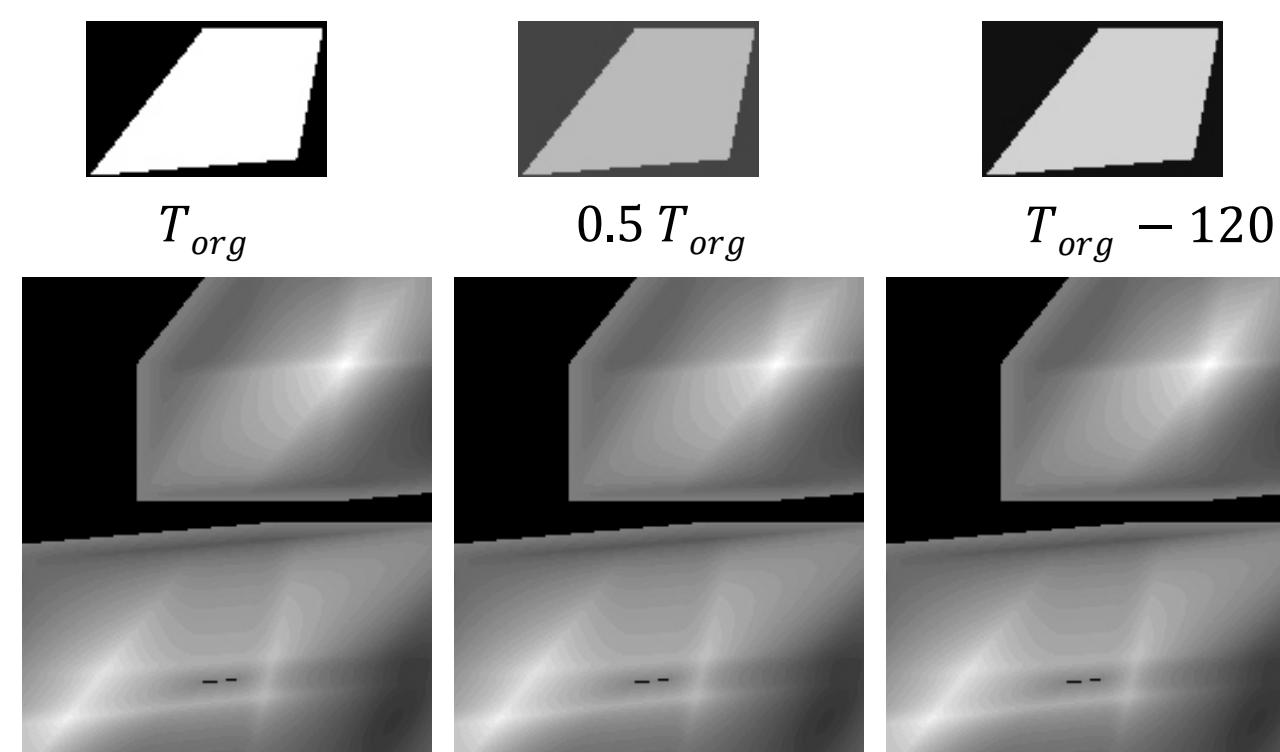
Korrelationkoeffizient

- Der Korrelationskoeffizient zwischen zwei Funktionen f und g ist ein Wert zwischen 1 und –1.
- Der Korrelationskoeffizient ist definiert als $c = \frac{\sigma(f,g)}{\sigma(f) \cdot \sigma(g)}$

wobei $\sigma(f)$, $\sigma(g)$ die Standardabweichungen von f und g , sowie $\sigma(f,g)$ die Kovarianz zwischen f und g ist
- Der Korrelationskoeffizient ist
 - 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.
- Der Korrelationskoeffizient ist damit als Ähnlichkeitsmaß unempfindlich gegenüber Skalierungen und Mittelwertveränderungen des Template.



Template Matching mit Korrelationskoeffizient



Resultate der Korrelation mit optimalem, skalierten und geminderten Templatewerten

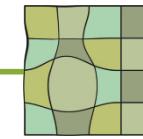
Korrelation durch FFT

Für $\mu_I = 0$ kann die Korrelation durch die Kovarianz abgeschätzt werden¹:

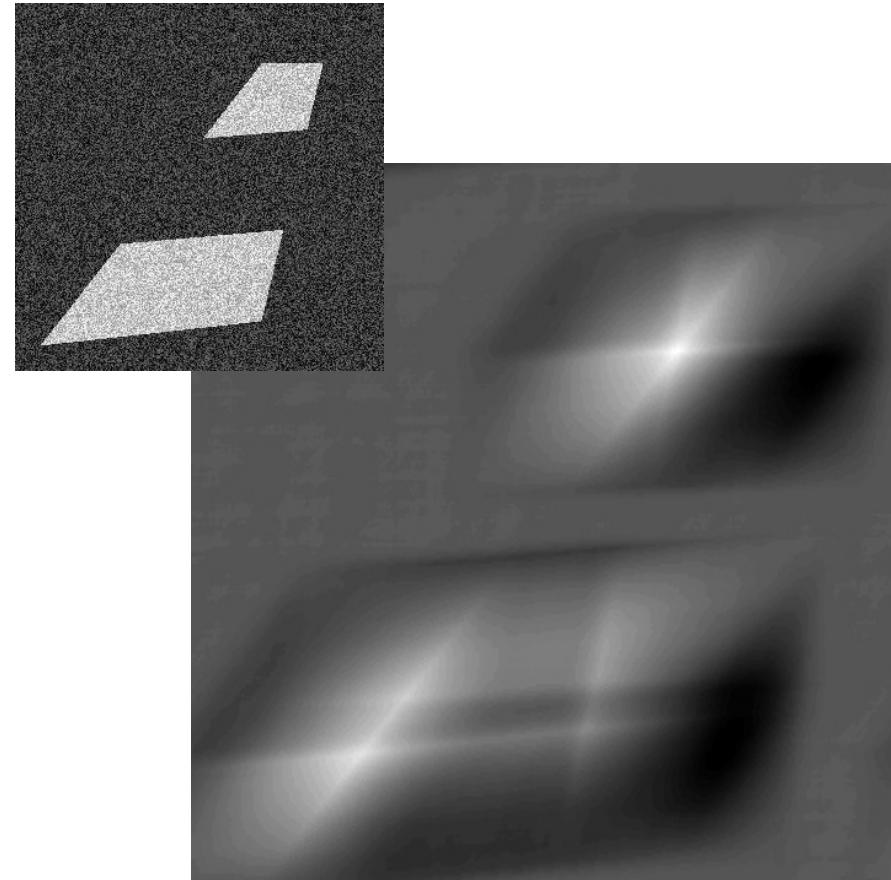
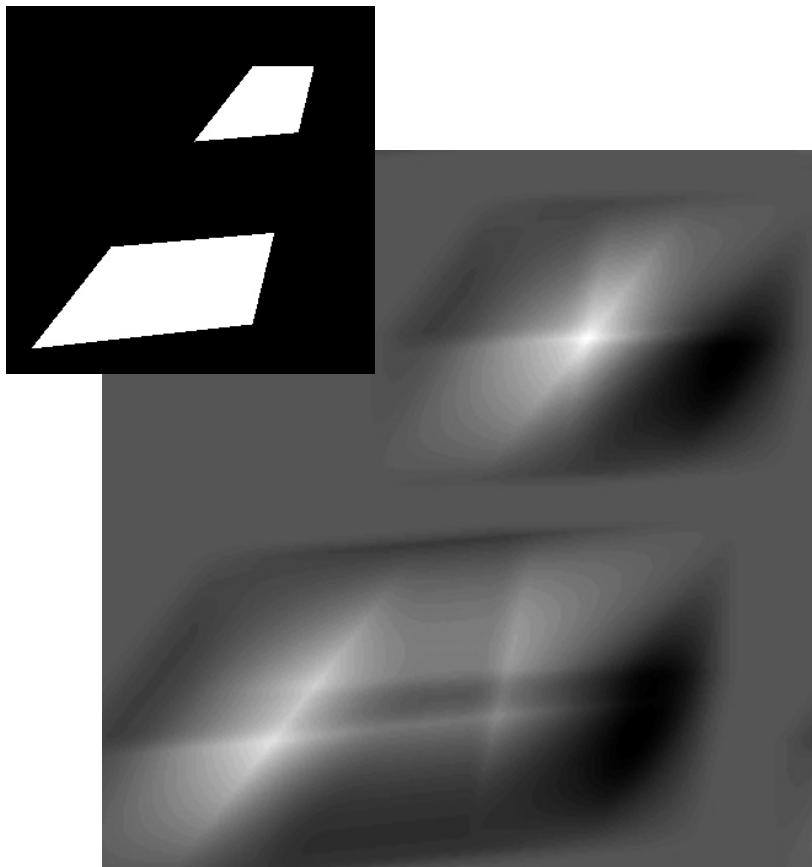
$$c(m, n) = \frac{\sigma(f, t)}{\sigma(f) \cdot \sigma(t)} = \frac{\sum_{i,j} (f(i+m, j+n) - \mu_f)(t(i, j) - \mu_t)}{\sigma(f) \cdot \sigma(t)}$$

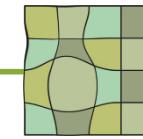
¹bis auf einen von Standardabweichungen von Template und überdeckter Region abhängigen Skalierungsfaktor.

$$\begin{aligned} c_F(m, n) &= \sum_{i,j} (f(i+m, j+n) - \mu_f) \cdot (t(i, j) - \mu_t) = \sum_{i,j} f(i+m, j+n) \cdot t(i, j) \\ FT(c_F(m, n)) &= F(u, v) \cdot T^*(u, v) \end{aligned}$$

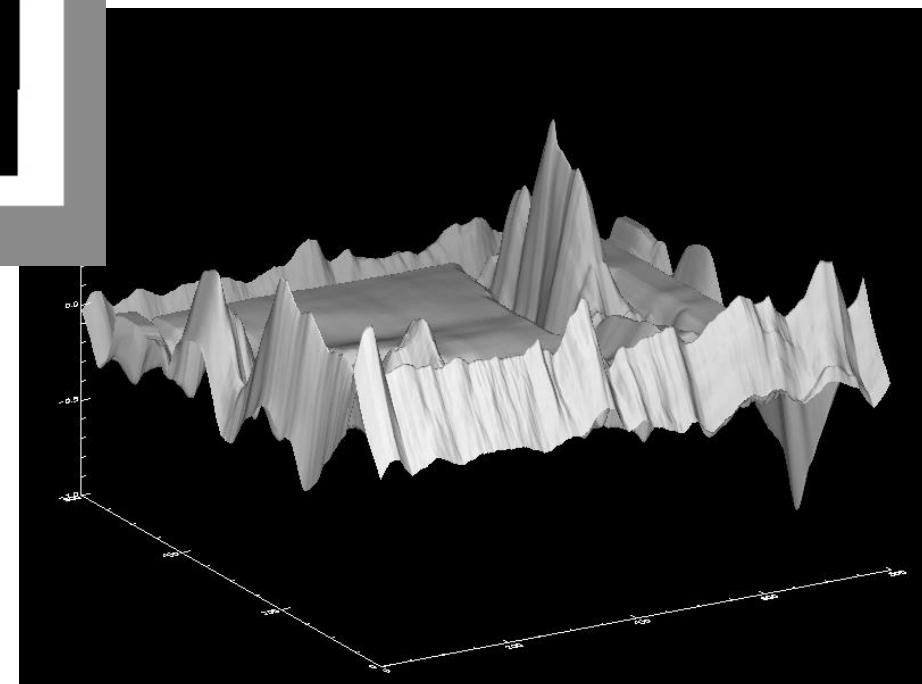


Korrelation durch FFT

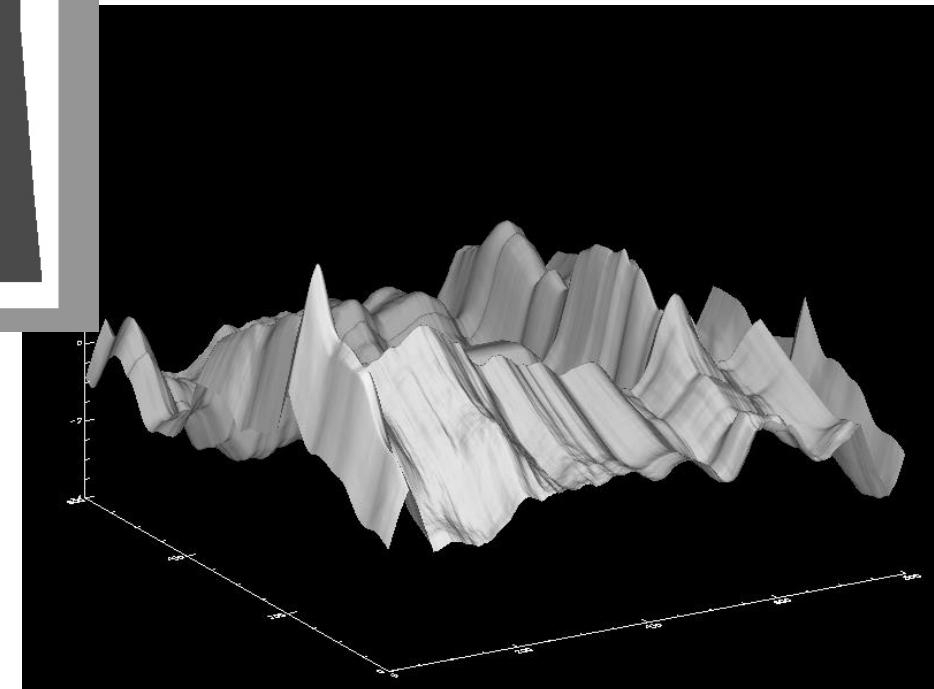
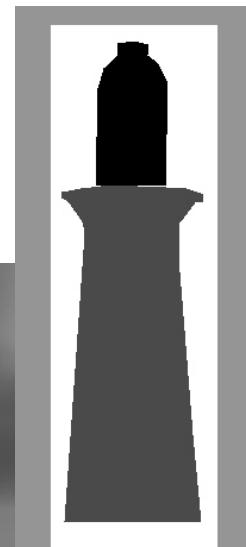




Resultate

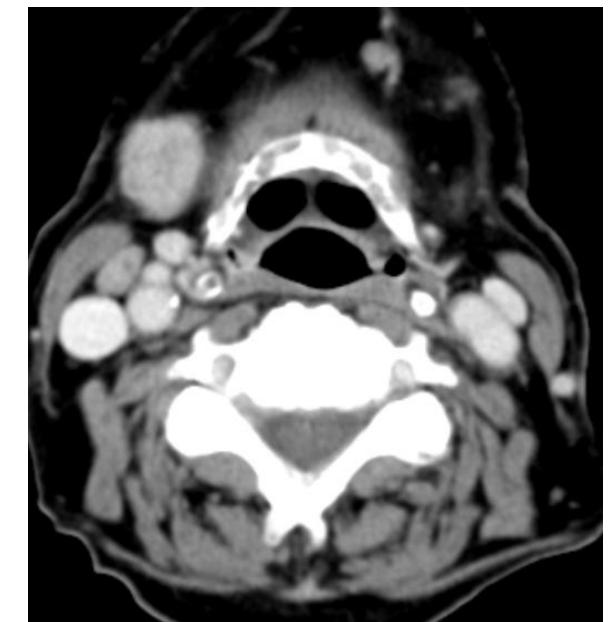


Resultate



Template Matching

- ist ein schnelles, automatisches Segmentierungsverfahren
- eignet sich gut, wenn die Form des gesuchten Objekts wenig variabel ist
 - Verkehrszeichen, Warenzeichen, Piktogramme
- Skalierungs- und Rotationsabhängigkeit kann zusätzliche Strategien für schnelle Berechnung erfordern
- kann auch zur Suche nach Regions of Interest benutzt werden, wenn die Form (in Maßen) variiert



Anwendung: ROI-Suche bei der Lymphknotendetektion

Hough Transformation

Modellbasierte Suche:

- 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.

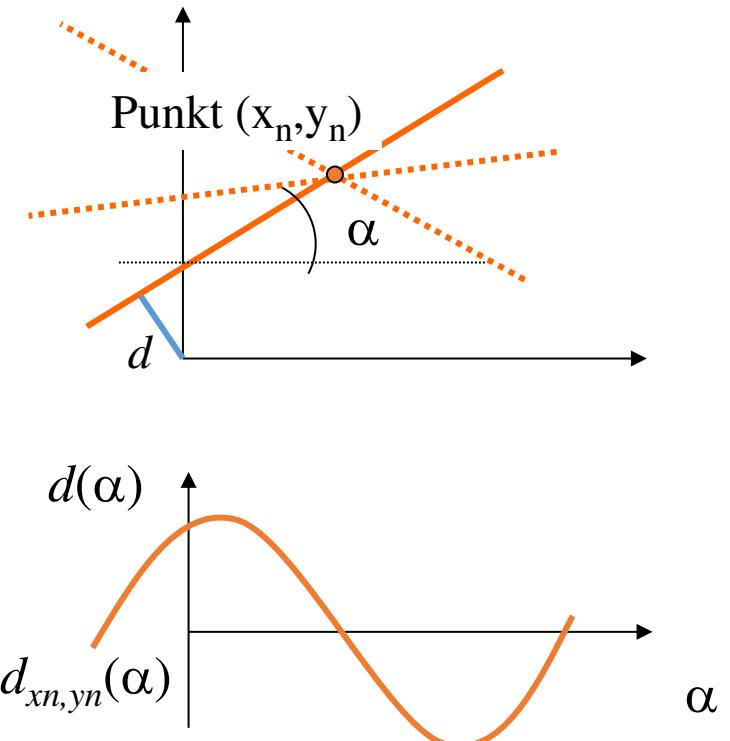
Geradenrepräsentation: $x \cos(\alpha) + y \sin(\alpha) - d = 0$

Hough-Transformation:

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

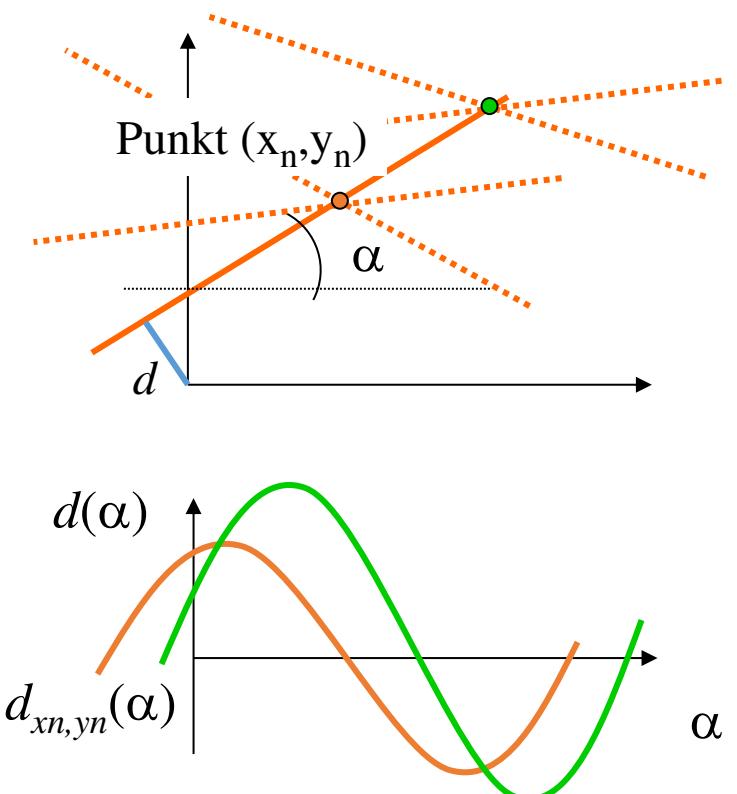
$$d(\alpha) = x_n \cos(\alpha) + y_n \sin(\alpha)$$

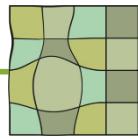
Der Raum, der durch (α, d) aufgespannt wird, heißt *Hough-Raum*.



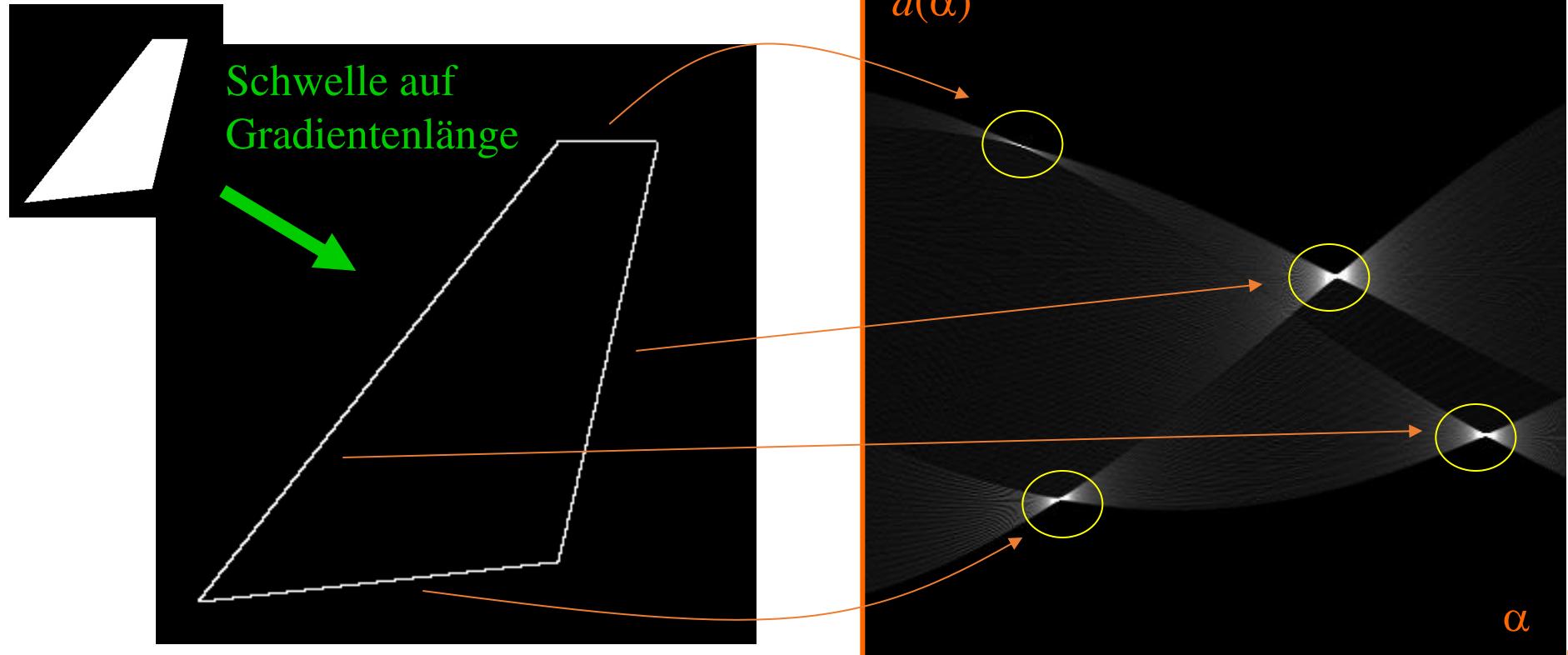
Berechnung der HT

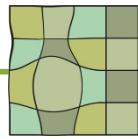
- Erzeugung eines Kantenbilds durch Schwellenwertsetzung auf Gradientenlängen.
- Diskretisierung des (α, d) -Raums (Zerlegung in Akkumulatoren)
- Für jeden Punkt x_n, y_n wird eine Kurve im (α, d) -Raum diskretisiert.
- Jeder Akkumulator wird inkrementiert, sobald eine Kurve durch in verläuft.
- Parameter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





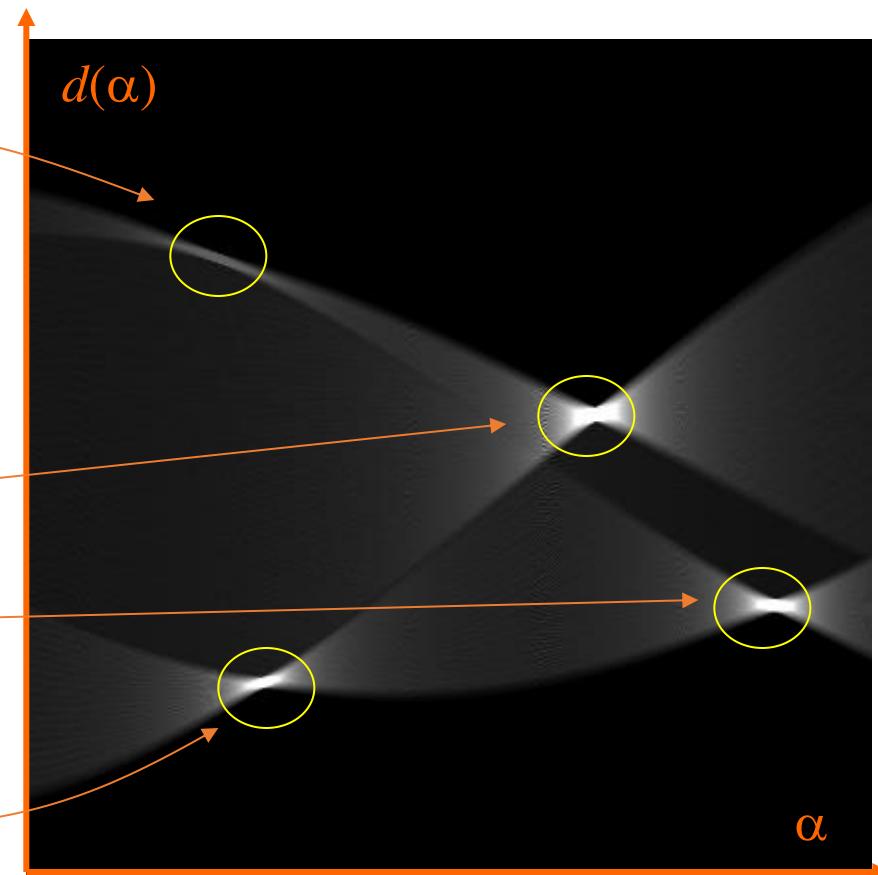
Hough Transformation

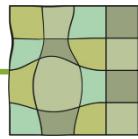




Hough Transformation

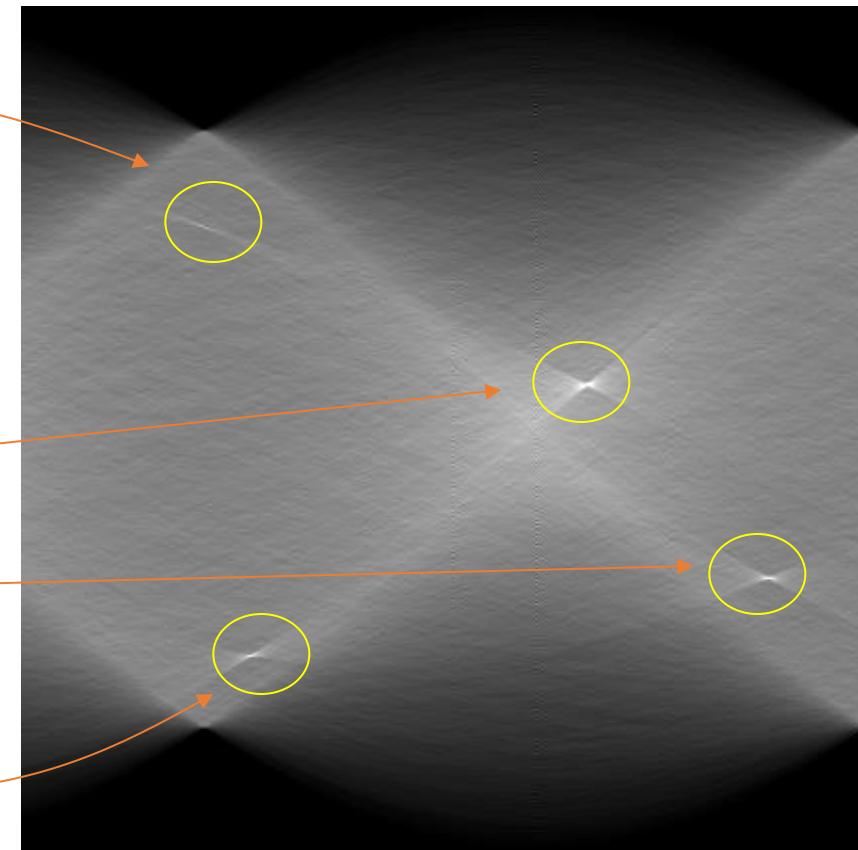
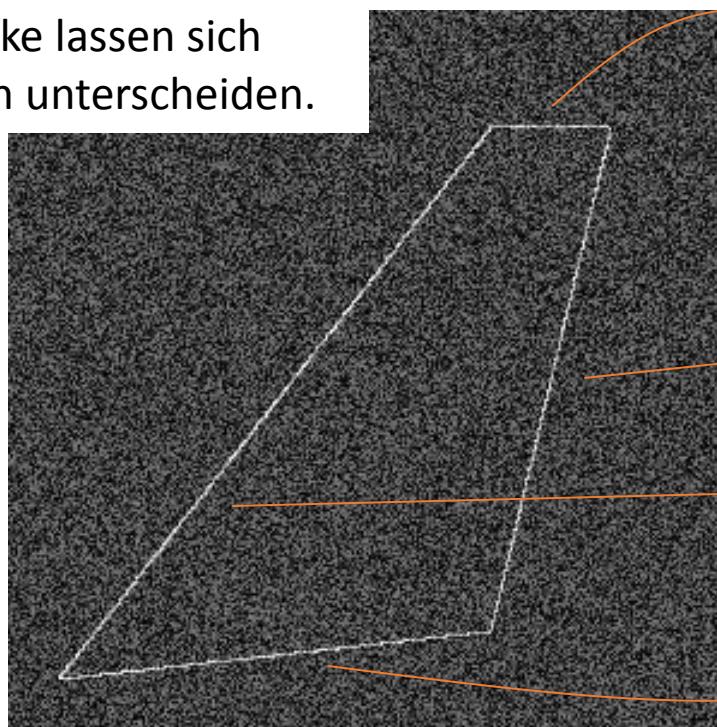
Breite Kanten führen zu flacheren Maxima im Hough-Raum

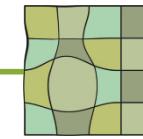




Hough Transformation in nicht-binären Bildern

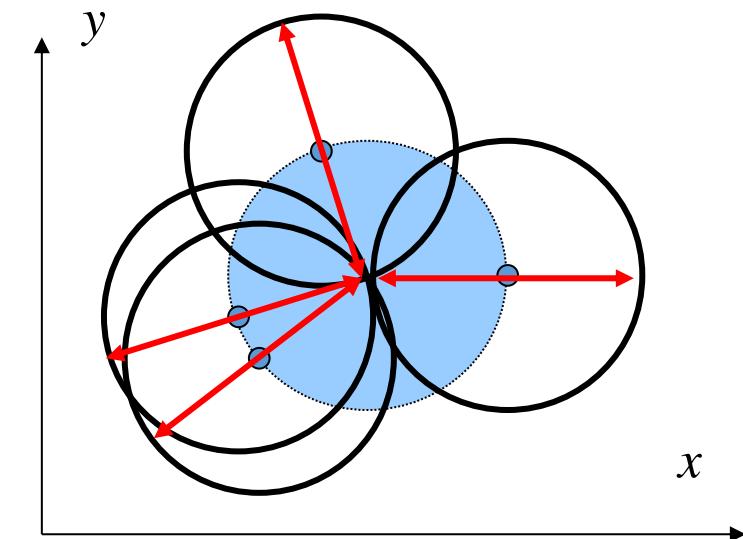
Bei Gewichtung der Stimmanzahl durch die Kantenstärke lassen sich Kanten von Rauschen unterscheiden.

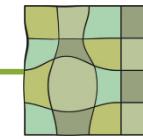




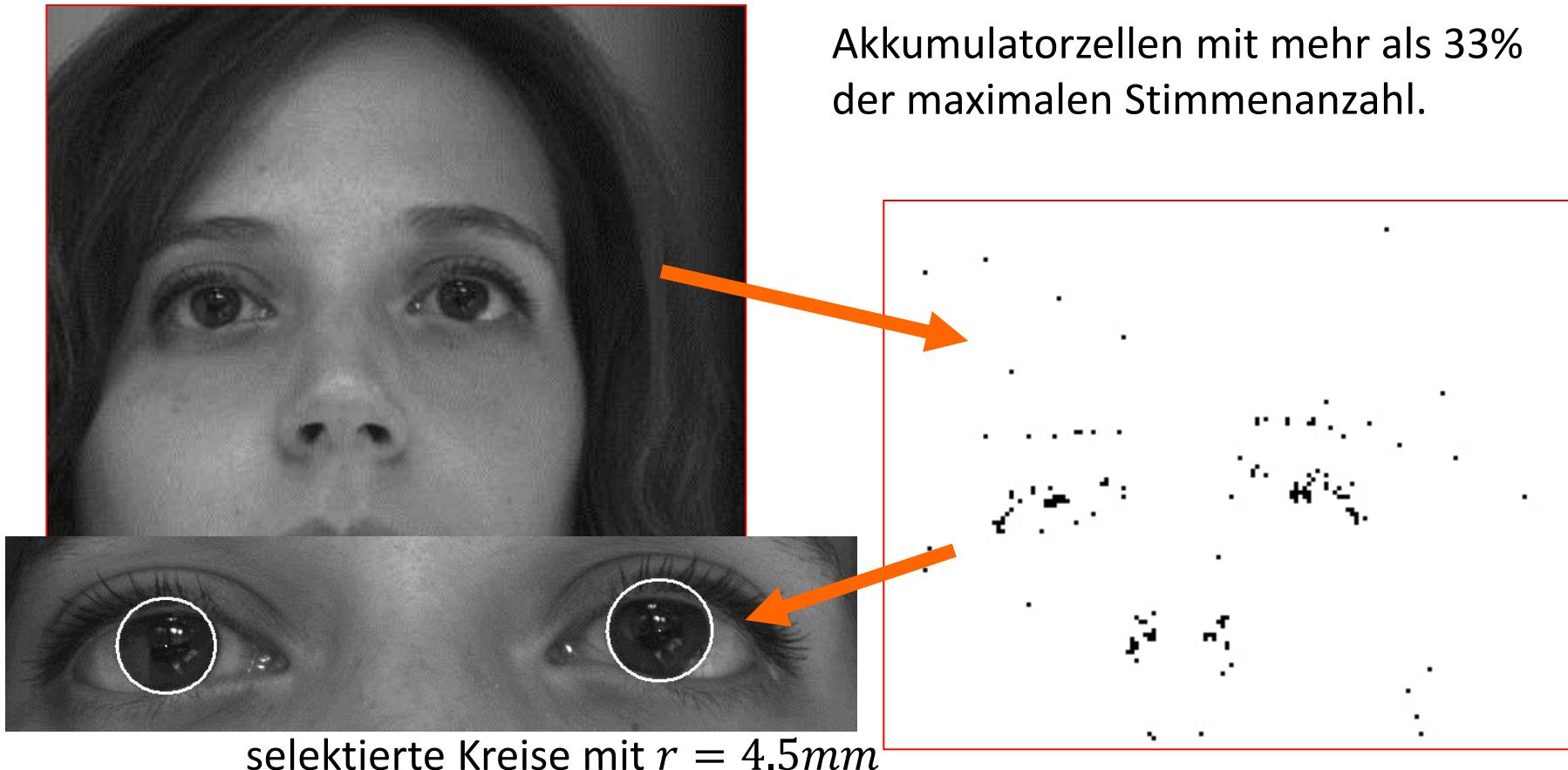
Hough Transformation für Kreise

- 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 Kantepunkt wird ein Kreis mit Radius r diskretisiert.
- Beschleunigung: Akkumulator wird nur in Distanz r in und entgegen der Gradientenrichtung inkrementiert.





Hough Transformation für Kreise

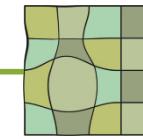


Hough Transformation

- Automatisches Verfahren, das Segmentränder und Teile von Segmenträndern findet
 - flexibler als Template Matching
 - kann aber Nachverarbeitung erfordern (Zusammensetzung von Randteilen)
- Das Voting-System ist interessant für Echtzeitanwendungen
 - Bereits eine Teilmenge der Stimmen erlaubt Vorhersagen auf das Endergebnis
 - Stimmabgabe kann gesteuert werden, so dass „wichtige“ Stimmen zuerst abgegeben werden.

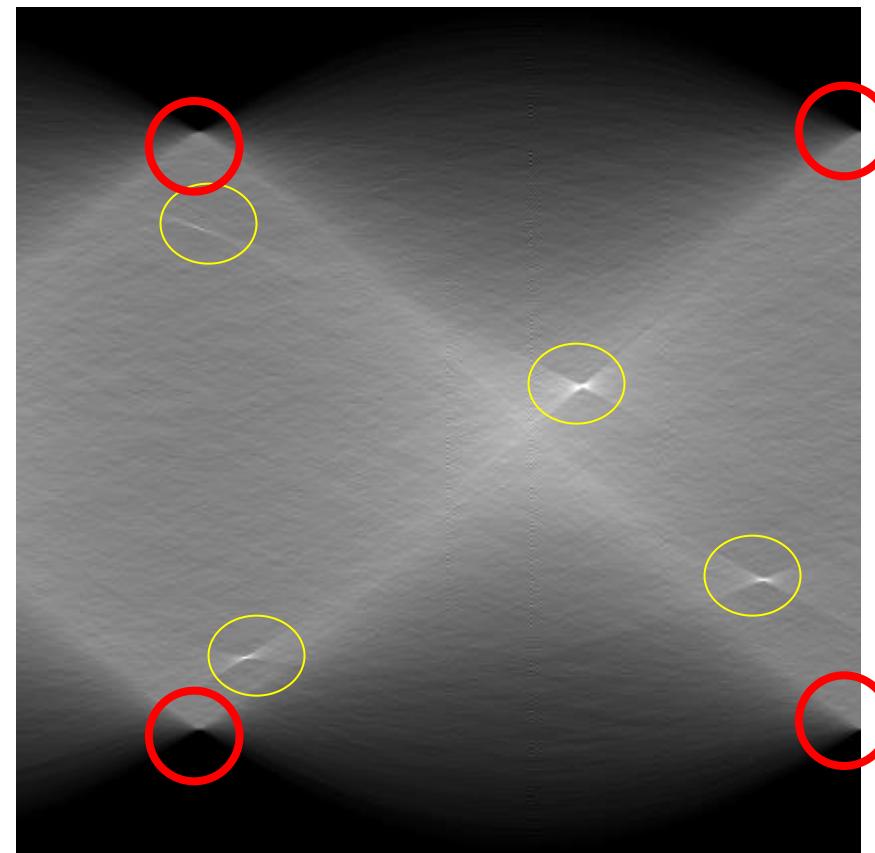
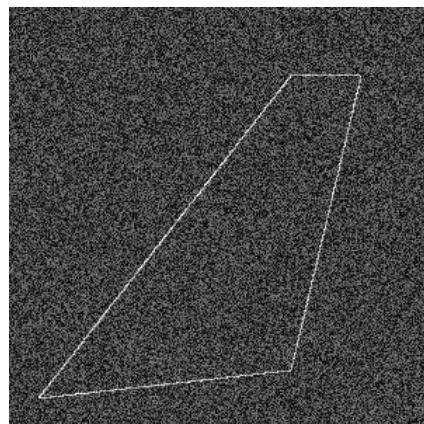
Was sollten Sie gelernt haben?

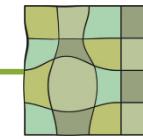
- Rolle des Modellwissens bei der Segmentierung
- Methoden mit interaktivem Einbringen von Modellwissen
 - Welche Art von Modellwissen
 - Wo und warum erfolgreich
- Methoden, die eine vollständige Suche verwenden
 - Welche Art von Modellen
 - Voraussetzungen



Famous Last Question

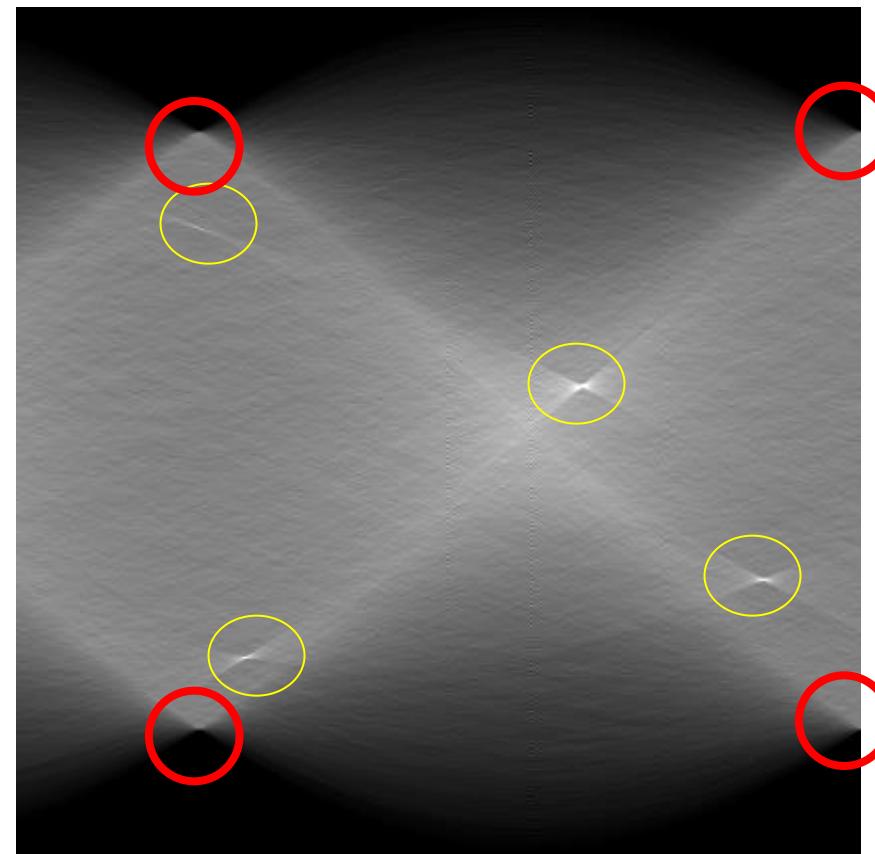
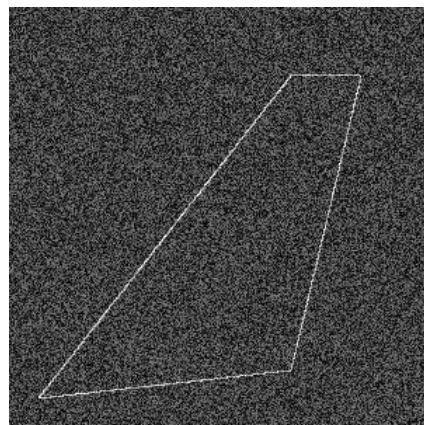
Wo kommen die zusätzlichen lokalen Maxima her?





Famous Last Question

Wo kommen die zusätzlichen lokalen Maxima her?

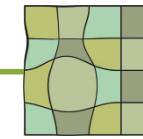


Morphologische Operationen

- Erosion und Dilatation
- Opening und Closing
- Ränder und Distanzen, Morphing
- Hit-or-Miss-Operator
- Skelettierung

Morphologische Operationen

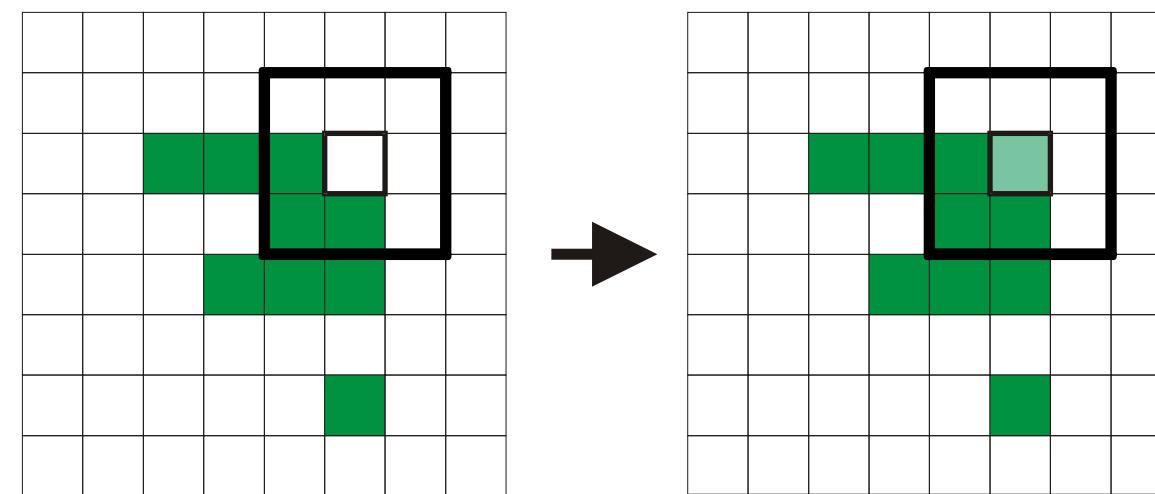
- Morphologisch: die äußere Gestalt betreffend
- morphologische Operationen:
 - Operationen auf der Gestalt von Objekten
 - setzt die Extraktion einer Gestalt voraus
 - also: in erster Linie Operation auf Segmenten (d.h., auf Binärbildern)
- Wozu ist es gut?
 - Veränderung der Gestalt, um Störungen nach einer Segmentierung zu beseitigen
 - Berechnung von Formmerkmalen
 - Suche nach bestimmten Formen (also: Analyse)

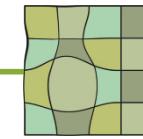


Dilatation

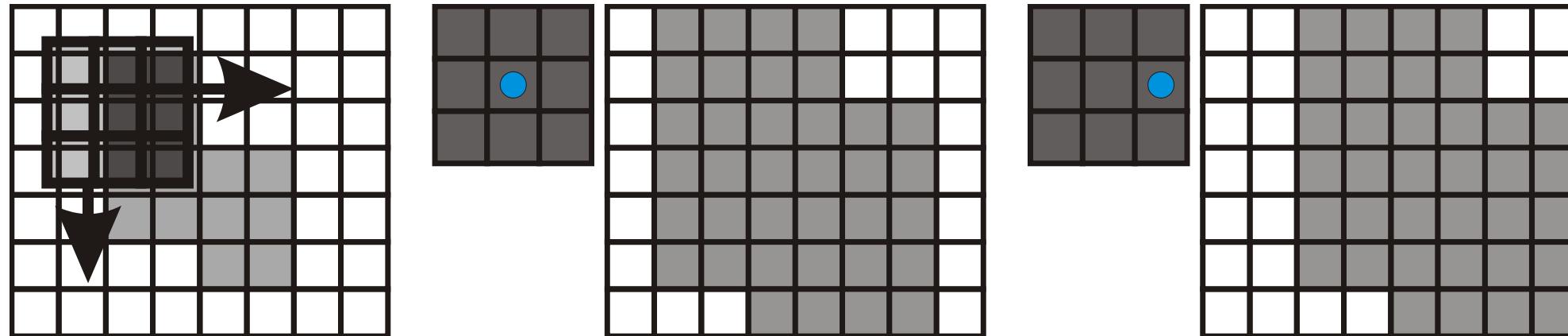
Dilatation (Ausdehnung): $G \oplus S$ mit Strukturelement S

$$g(m, n) = \vee_{(m_k, n_k) \in s} b(m + m_k, n + n_k)$$





Dilatation

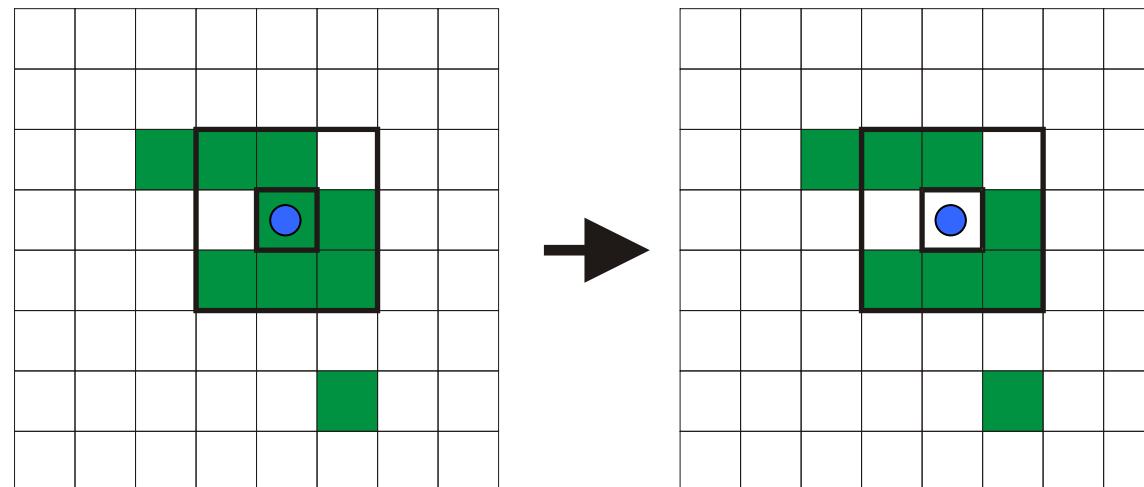


Dilatation wird (wie jede morphologische Operation) für einen **Ankerpunkt** ausgeführt.

- Dilatation:
- verbindet Strukturen
 - füllt Löcher
 - vergrößert

Erosion

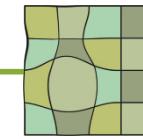
$$g(m,n) = \bigwedge_{(m_k, n_k) \in S} b(m + m_k, n + n_k).$$



Erosion: $G \ominus S$ mit
Strukturelement S

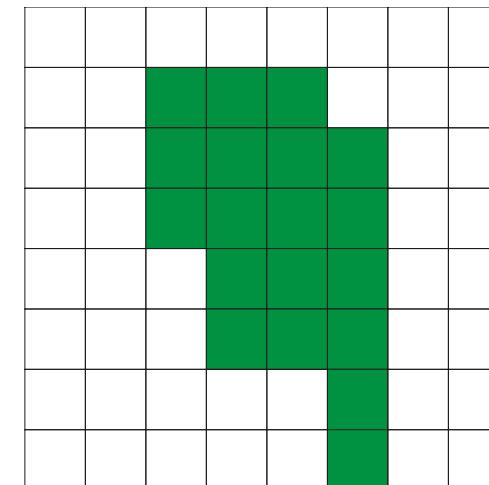
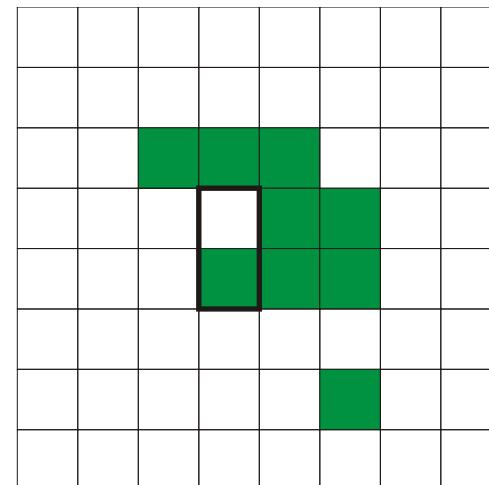
Erosion:

- löst Strukturen auf
- entfernt Details
- verkleinert



Strukturelemente

- Ein Strukturelement einer morphologischen Operation entspricht dem Faltungskern bei einer Konvolution.
- Mit einem gezielt geformten Strukturelement können genau definierte Formveränderungen erzeugt werden.

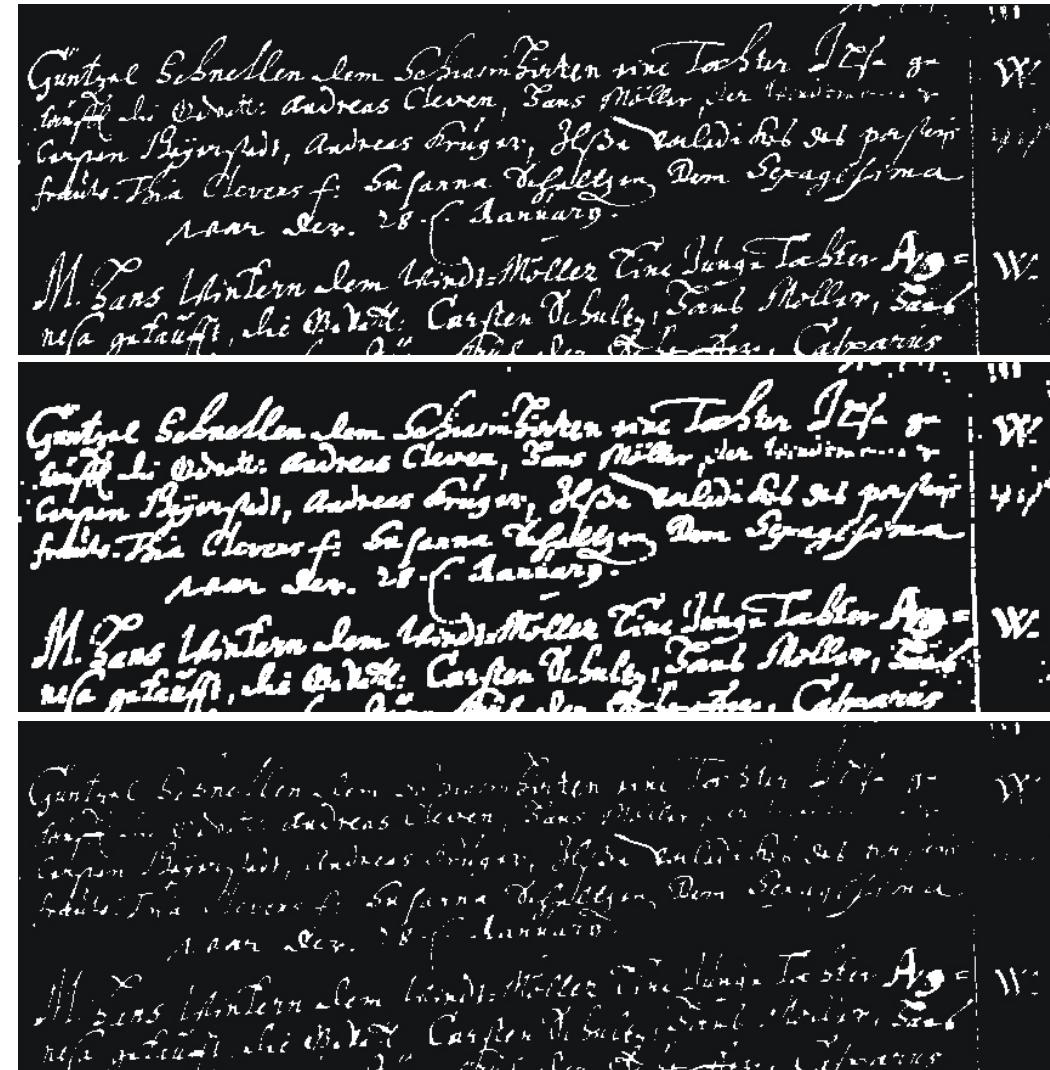


Dilatation

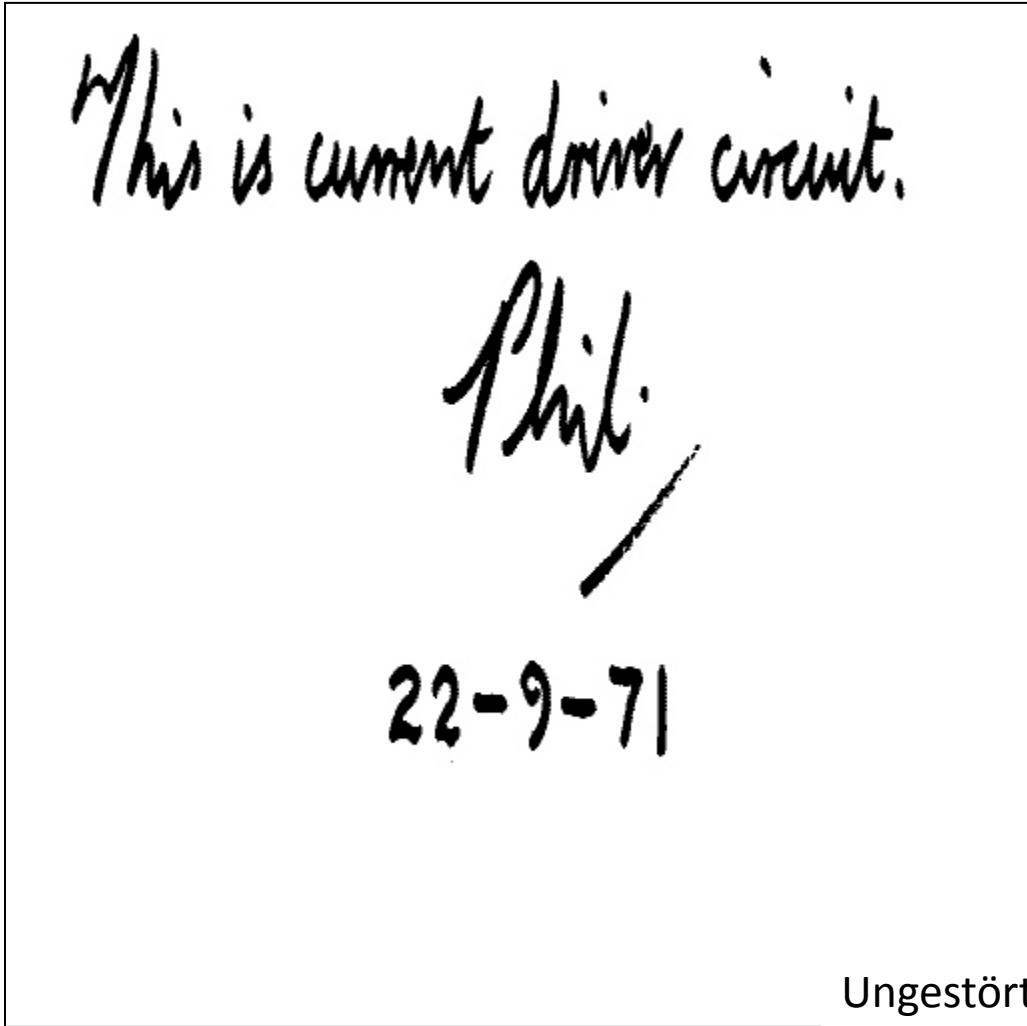


Strukturelement

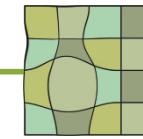
Beispiel



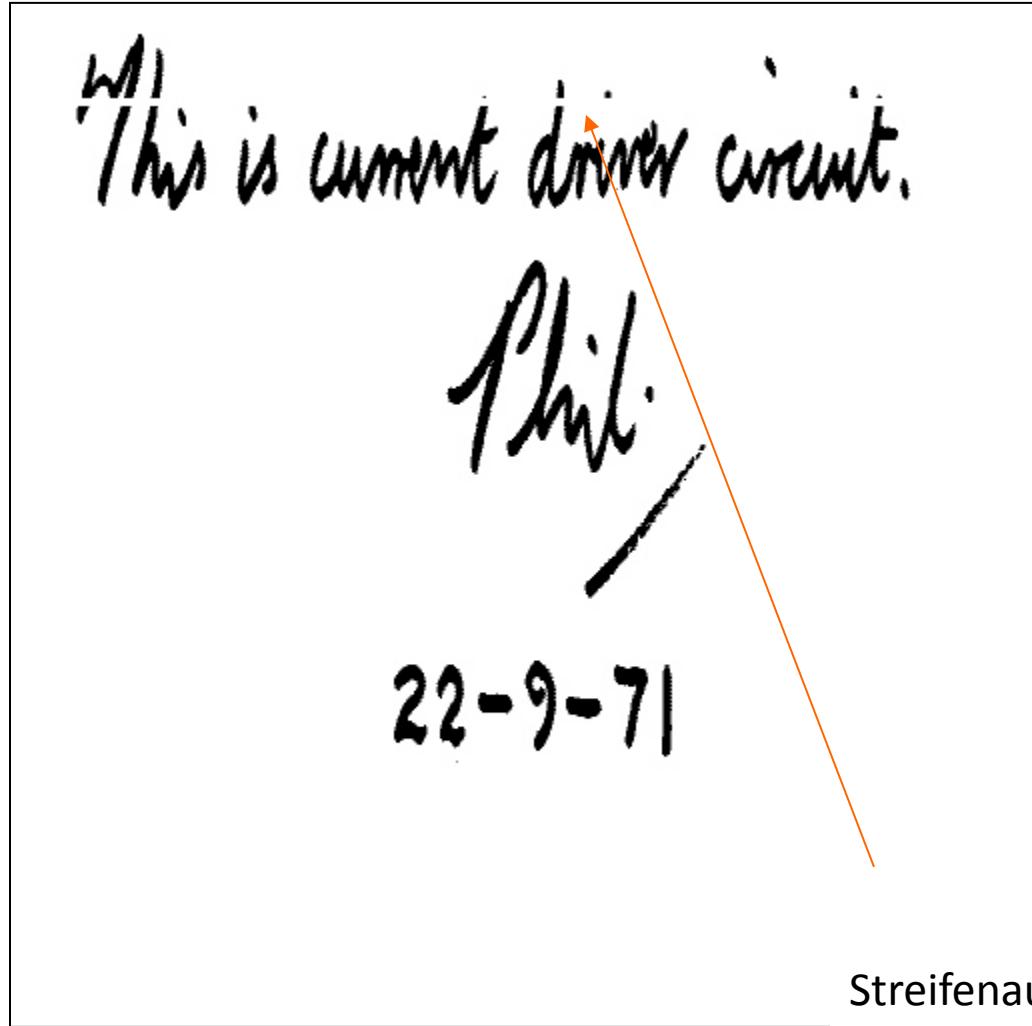
Gezielter Einsatz



Ungestörtes Binärbild

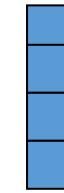
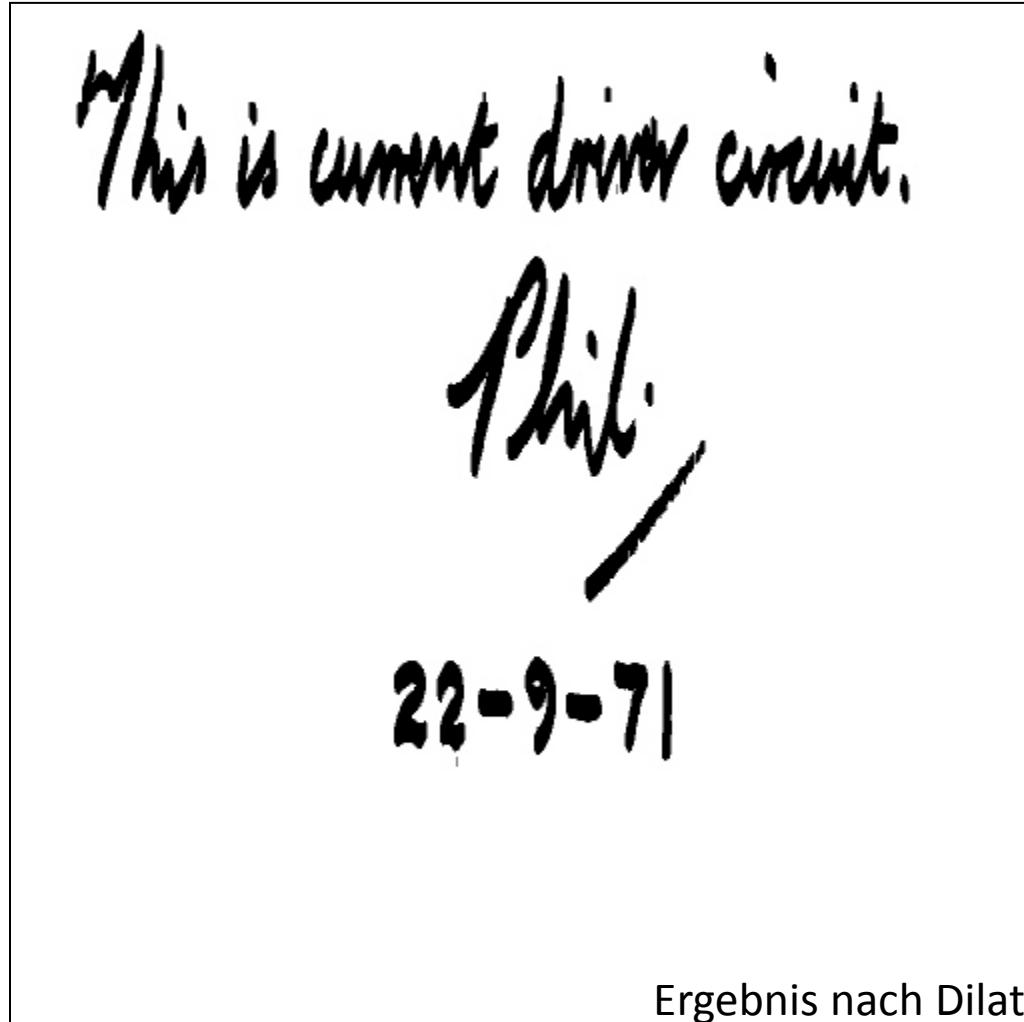


Gezielter Einsatz



Strukturelement zum
Schließen des Streifens

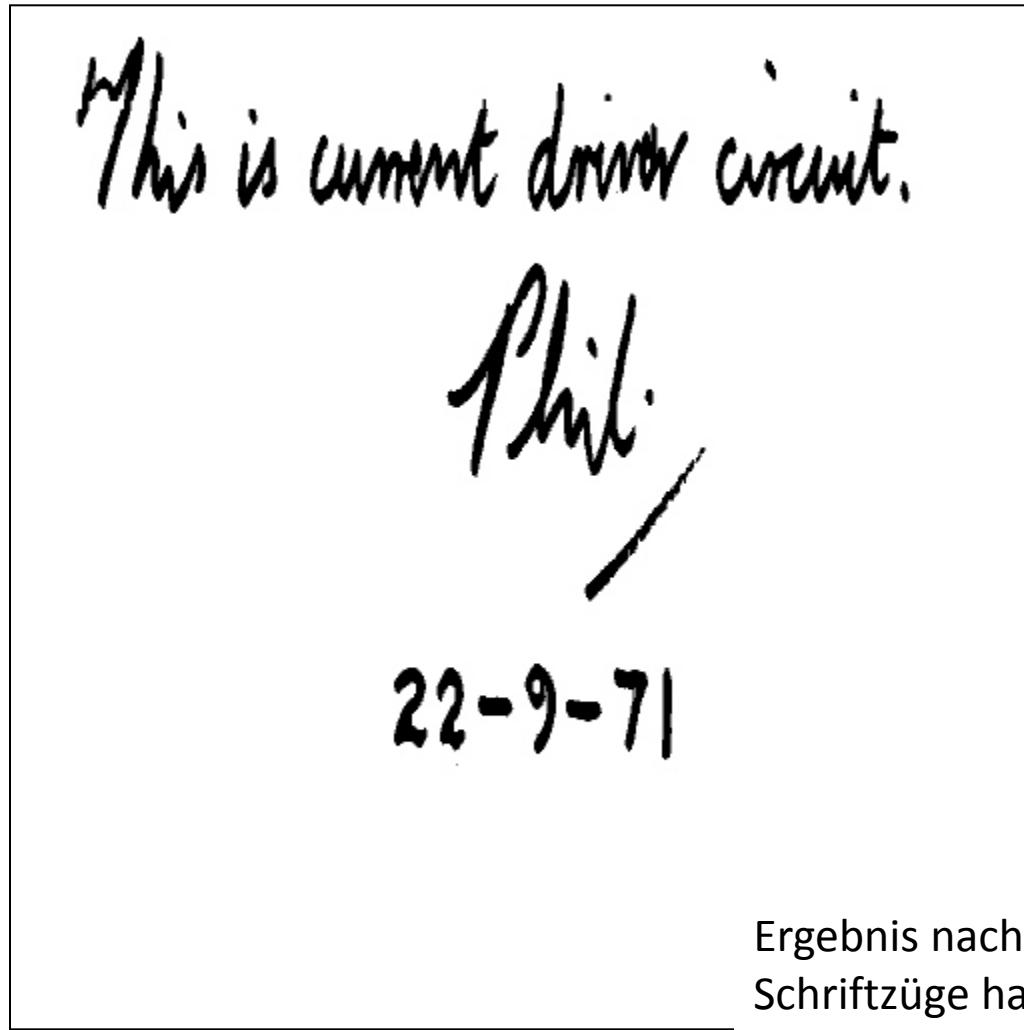
Gezielter Einsatz



Strukturelement zum
Schließen des Streifens

Ergebnis nach Dilatation: Streifen ist geschlossen

Gezielter Einsatz



Strukturelement zur Erosion
des zu breiten Schriftzugs

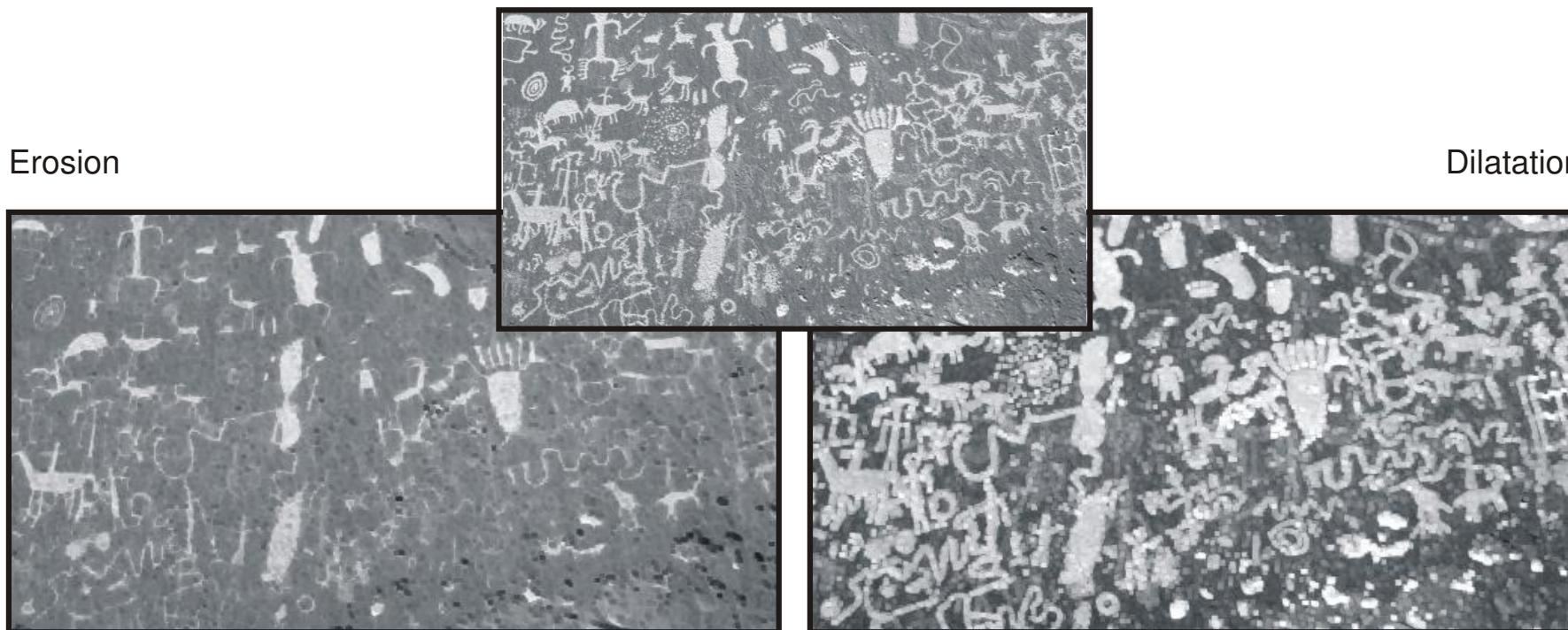
Ergebnis nach nachfolgender Erosion:
Schriftzüge haben ihre Ursprungsstärke

Einige Eigenschaften von morphologischen Operatoren

- **Verschiebungsinvarianz:** Wegen der Beschreibung von Erosion/Dilatation als Faltung sind beide Operationen genau wie eine Faltung verschiebungsinvariant.
- **Kommutativität und Assoziativität:** $M_1 \oplus M_2 = M_2 \oplus M_1$ aber $M \ominus M_2 \neq M_2 \ominus M_1$
es gilt jedoch $(G \ominus M_1) \ominus M_2 = G \ominus (M_1 \ominus M_2) = (G \ominus M_2) \ominus M_1$
- **Dualität:** $\overline{G} \ominus M = \overline{G \oplus M}$ und $\overline{G} \oplus M = \overline{G \ominus M}$

Morphologische Operationen auf Grauwertbildern

- Dilatation:
$$g(m,n) = \max_{(m_k, n_k) \in s} (b(m + m_k, n + n_k))$$
- Erosion:
$$g(m,n) = \min_{(m_k, n_k) \in s} (b(m + m_k, n + n_k))$$



Opening

Opening (Öffnen):

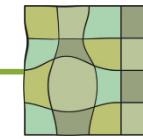
Kombination von Erosion gefolgt von einer Dilation am Ankerpunkt gespiegelten Strukturelement S'

$$G \circ S = (G \ominus S) \oplus S'$$

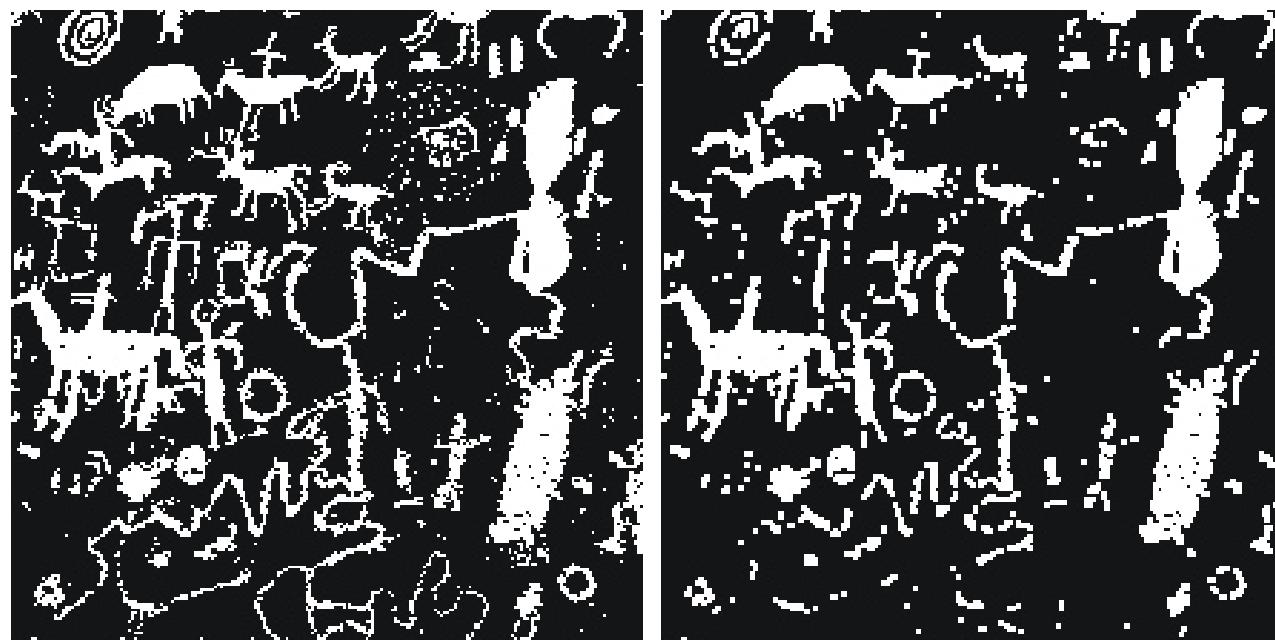
Ziel:

Erosion - Entfernung aller (Teil-)strukturen, die kleiner als das Strukturelement sind

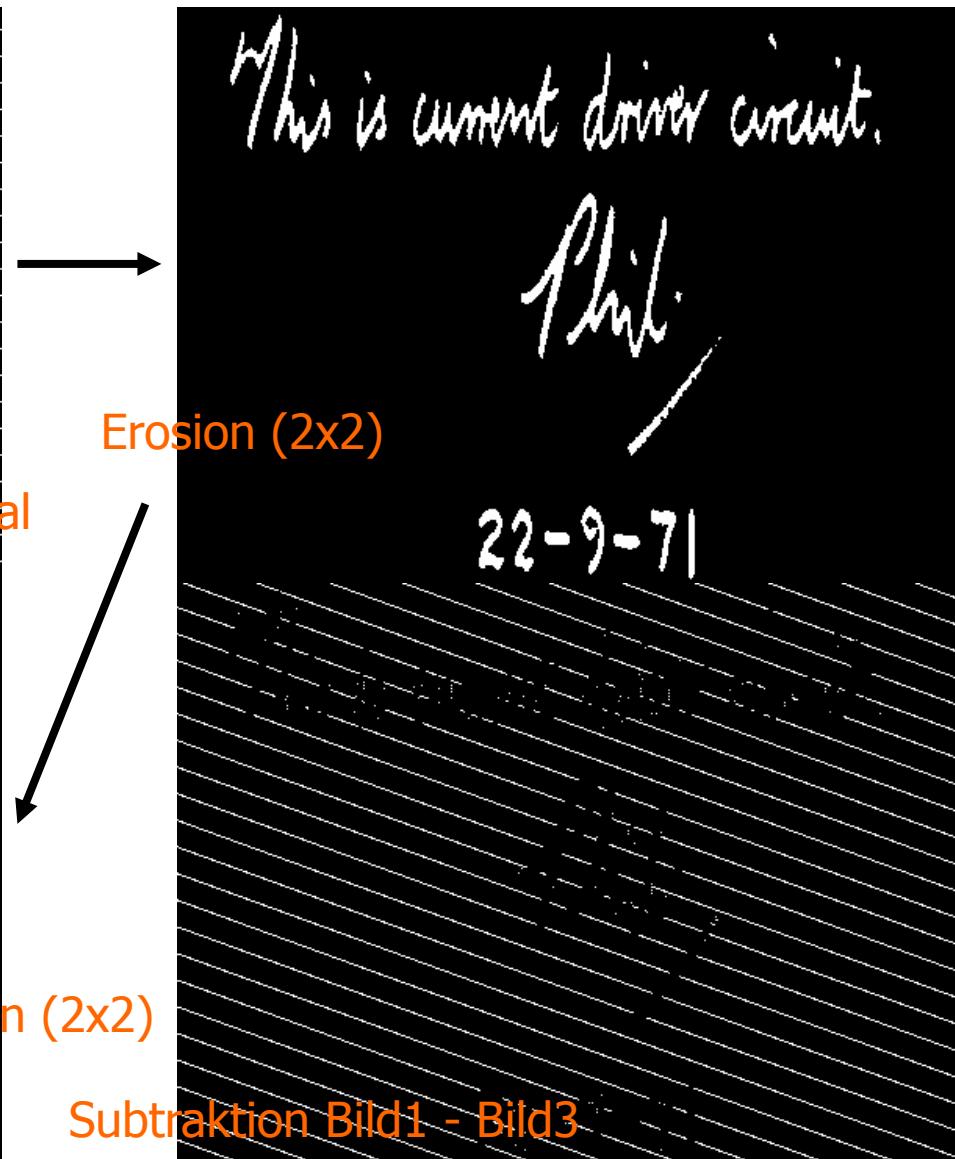
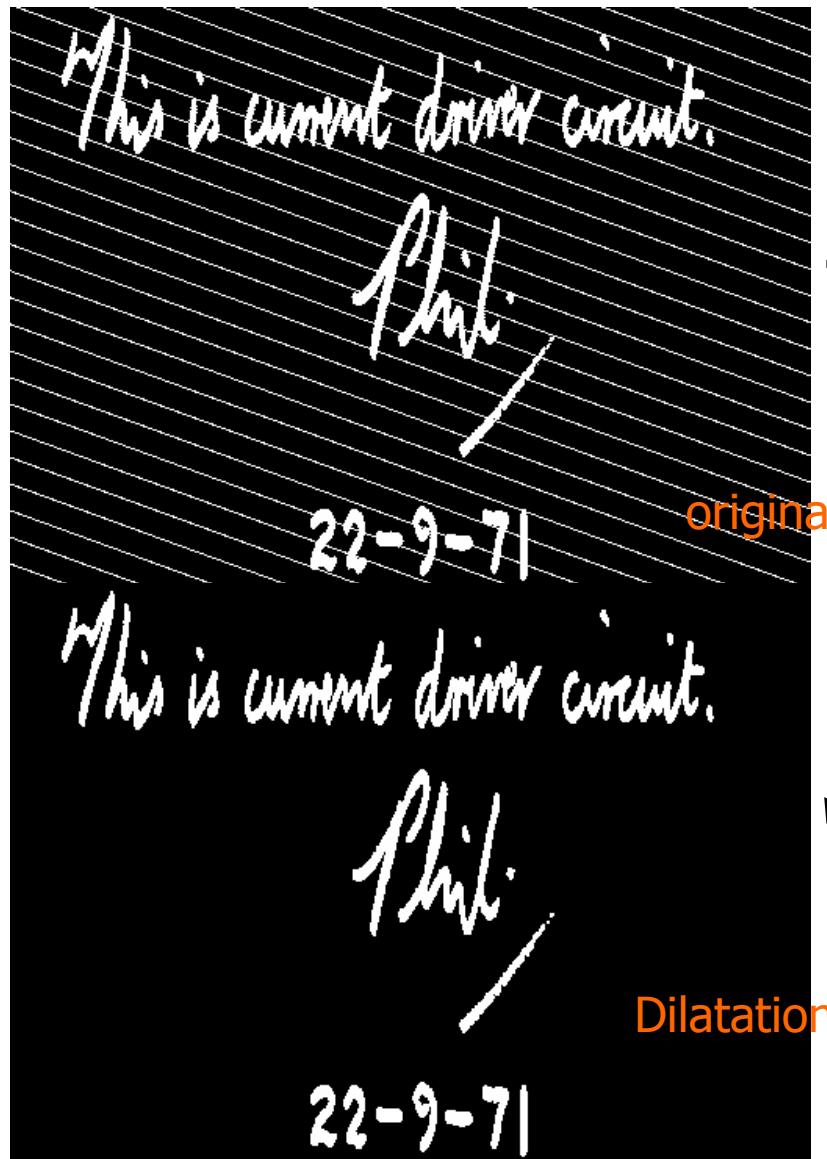
Dilatation - Wiederherstellung der ursprünglichen Größe des Objekts mit Ausnahme der vollständig entfernten Teilstrukturen



Beispiel Opening



Entfernung von Linien



Closing

Closing (Schließen):

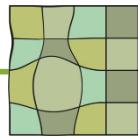
Kombination von Dilatation gefolgt von einer Erosion mit einem am Ankerpunkt gespiegelten Strukturelement S'

$$G \bullet S = (G \oplus S) \ominus S'$$

Ziel:

Dilatation - Schließen von kleinen Löchern (kleiner als das Strukturelement)

Erosion - Wiederherstellung der ursprünglichen Größe des Objekts



Opening vs. Closing



original



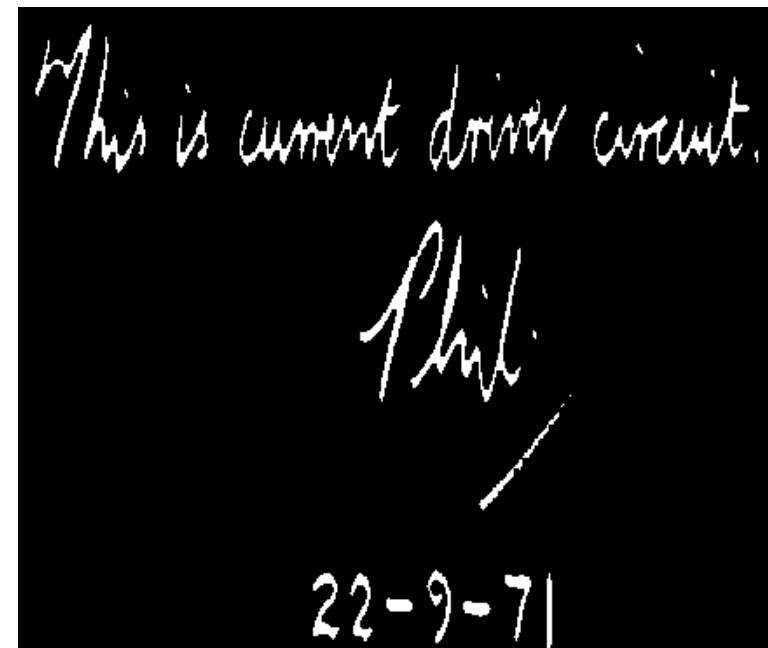
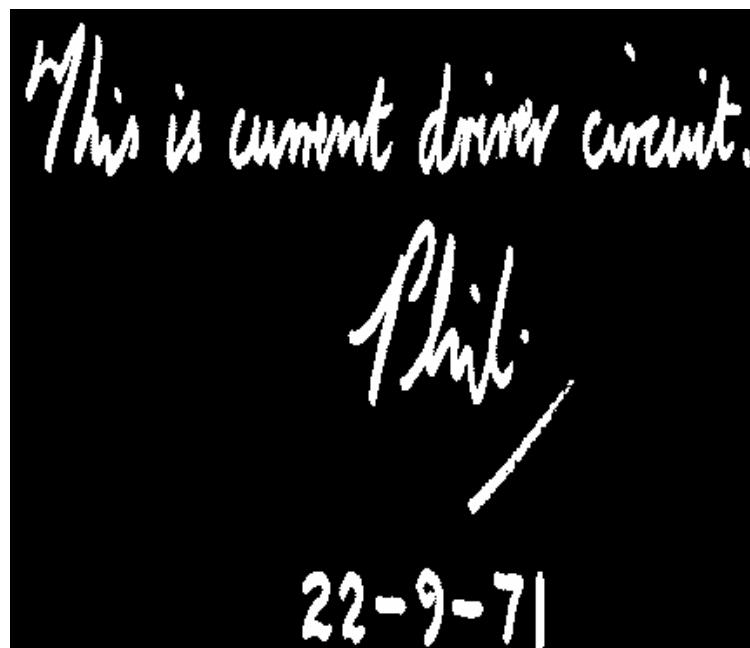
opening



closing

Extraktion von Rändern

- Erosion mit S_{b4} bzw. S_{b8} entfernt alle Objektpixel, in deren 4- bzw. 8-Nachbarschaft sich Hintergrundpixel befinden.
- Der Rand kann nun durch Differenzbildung zwischen Ursprungsbild und erodiertem Bild erzeugt werden: $\partial G = G \setminus (G \ominus S_b)$



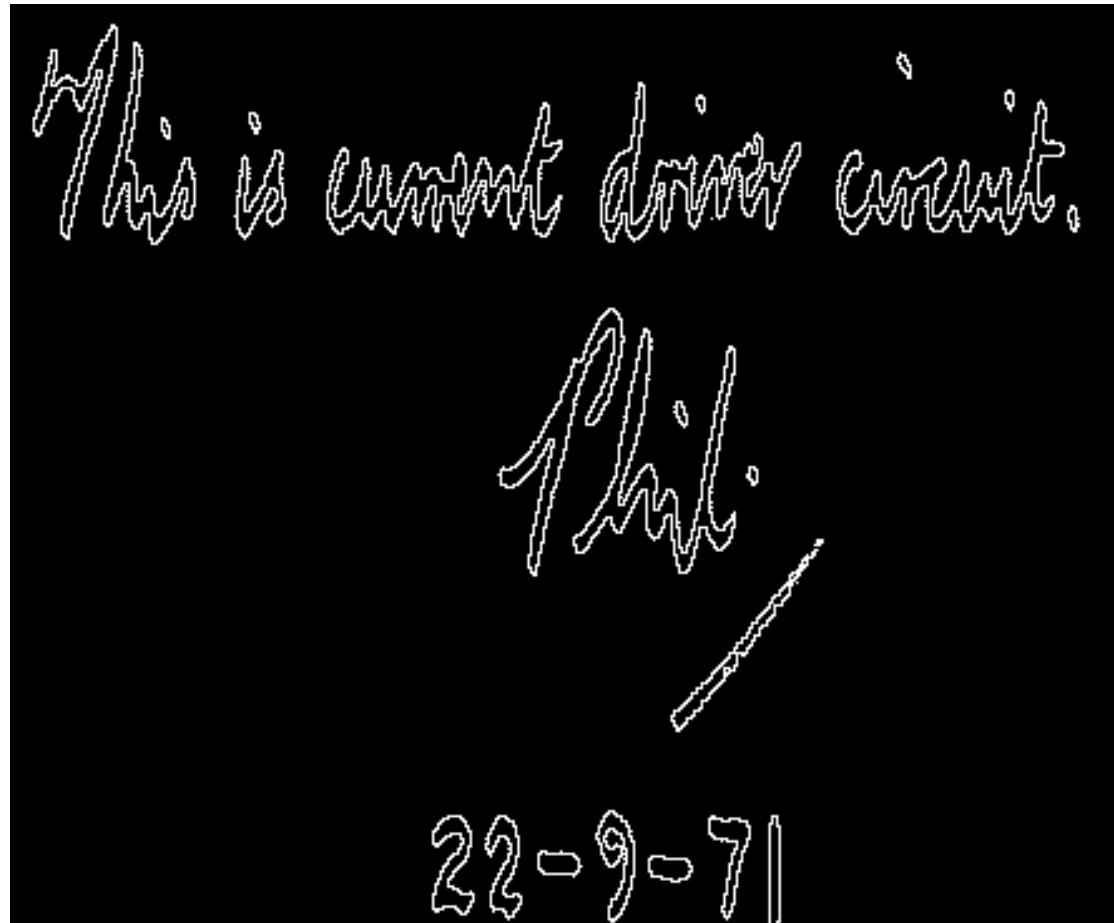
$$S_{b4} = \begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{matrix} \quad S_{b8} = \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

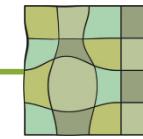
Extraktion von Rändern

$$\begin{aligned}\partial G &= G \setminus (G \ominus M_b) \\ &= G \cap \overline{(G \ominus M_b)} \\ &= \underline{G \cap (G \oplus M_b)}\end{aligned}$$

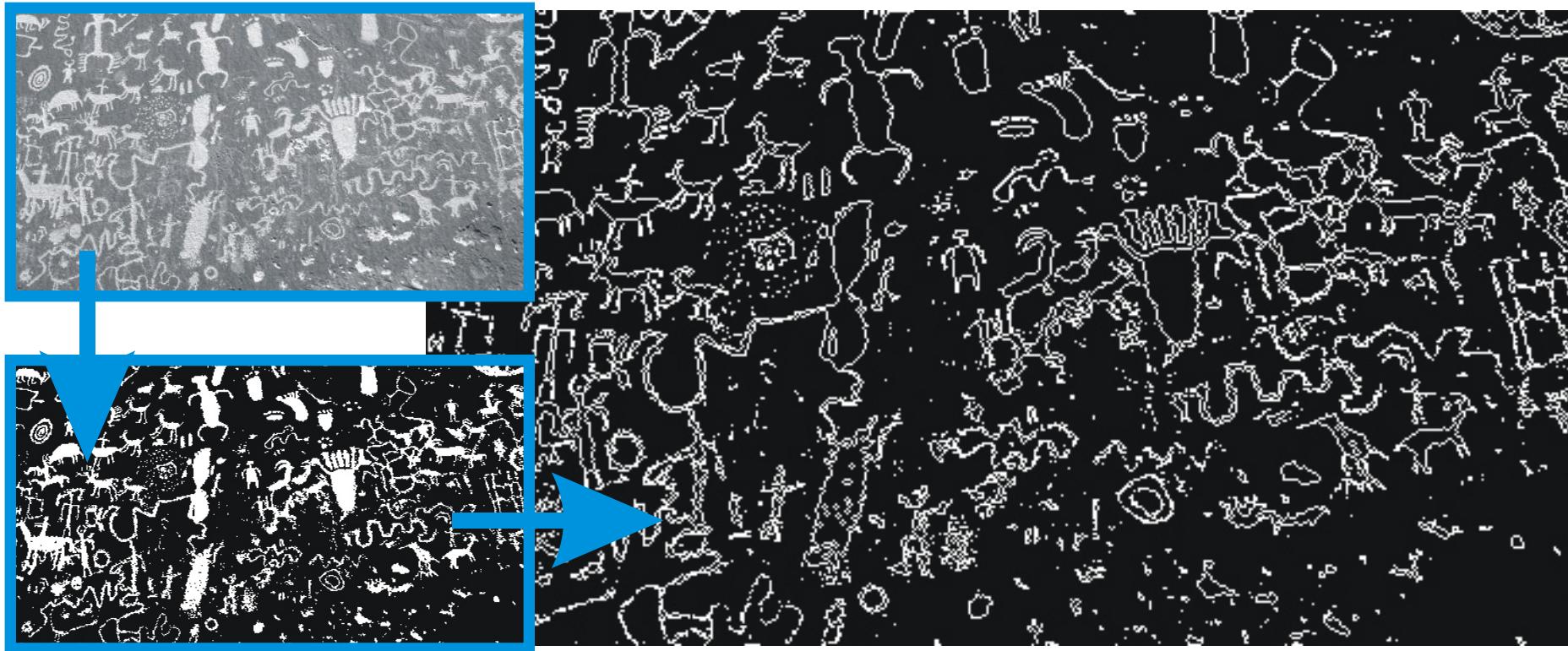
Hintergrundrand:

$$\partial G_B = (G \oplus M_b) \setminus G$$





Beispiel



Distanztransformation

Resultat der Randoperation $\partial G_0 = G \setminus (G \ominus S_b)$:

Menge aller Pixel, die den **Abstand 1** zum Rand haben.

Falls die gleiche Operation auf dem um den Rand verminderten Bild nochmals angewendet wird:

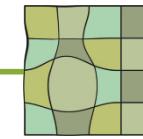
$$\partial G_1 = (G \ominus S_b) \setminus (G \ominus S_b \ominus S_b)$$

Menge aller Pixel, die den **Abstand 1** zum Rand haben.

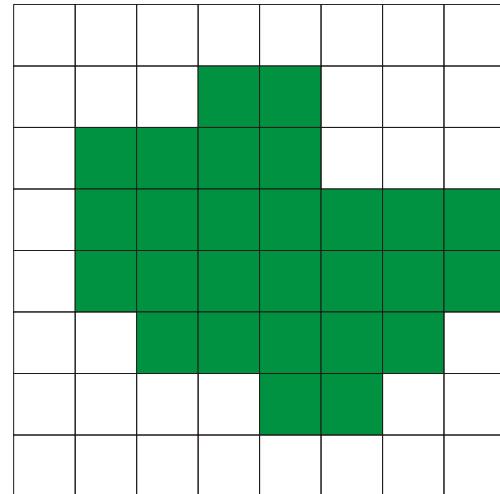
Fortgesetzte Extraktion von immer weiter vom Rand entfernten Linien und Multiplikation der jeweiligen Resultate mit der aktuellen Entfernung überführt das Binärbild in ein **Distanzbild D**:

$$D = \bigcup_{n=1,\infty} [(G \ominus S_b^{n-1}) \setminus (G \ominus S_b^n) \cdot n],$$

wobei die Operation \cdot die punktweise Multiplikation der n -ten Randkurve mit der Zahl n (dem aktuellen Abstand) darstellt.

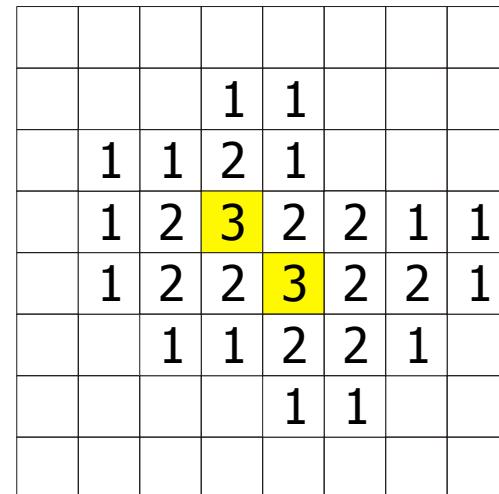
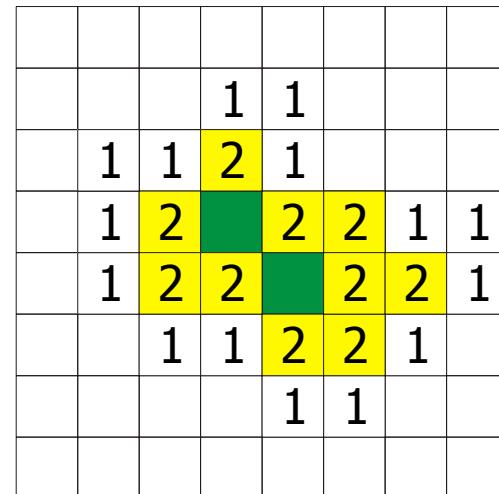
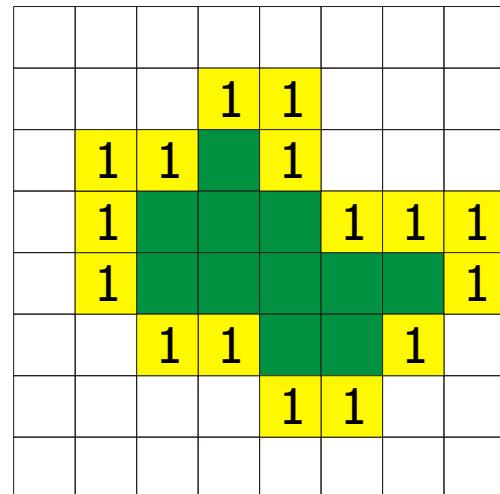


Beispiel



Originalbild

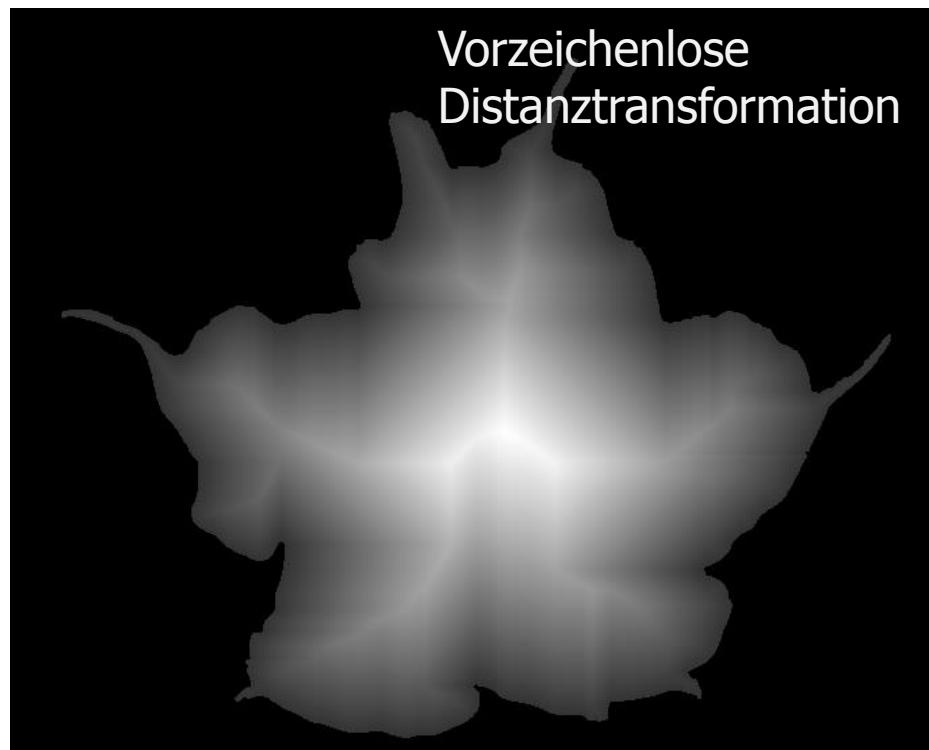
- Objektinneres (nach fortgesetzter Erosion)
- Randpixel nach der n-ten Erosion einschließlich Distanz



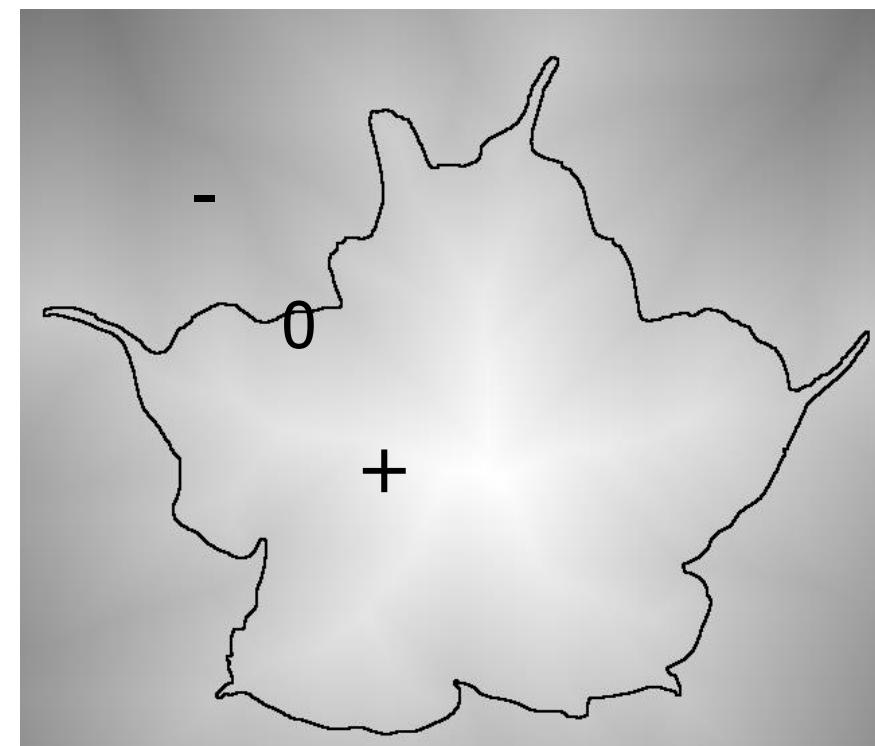
Vorzeichenbehaftete DT

- Distanztransformation auf dem Vordergrund (1-Pixel)
- Distanztransformation auf dem Hintergrund (Operation auf dem negierten Bild)
- Kombination beider DTs
 - Übernahme der Ergebnisse auf dem Vordergrund
 - Hintergrunddistanzen werden negativ eingetragen

Beispiel



vorzeichenbehaftete
Distanztransformation

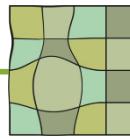


Morphing

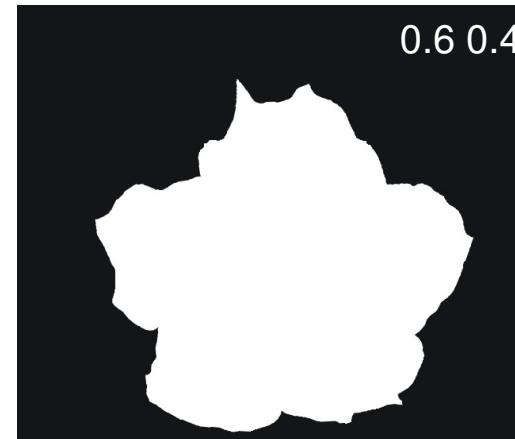
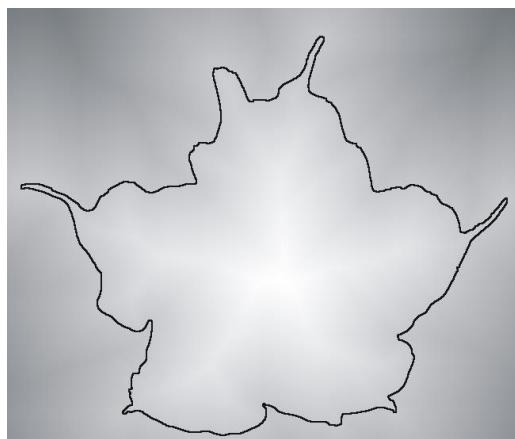
- Vorzeichenbehaftete Distanztransformation auf Binärbildern b_A und b_B durchführen.
- Für $i=0,N-1$ Distanzbilder linear aus den Distanzbildern A_A und A_B interpolieren

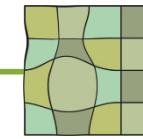
$$A_i = \frac{i \cdot A_B + (L - i) \cdot A_A}{L}$$

- Objekt einer Zwischenstufe i sind diejenigen Pixel, für die im i -ten Distanzbild A_i die Distanzen positiv sind.

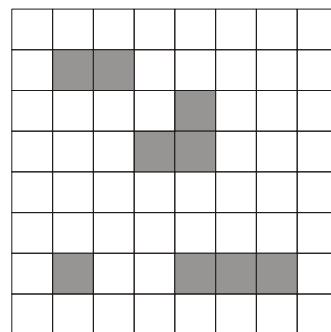


Beispiel

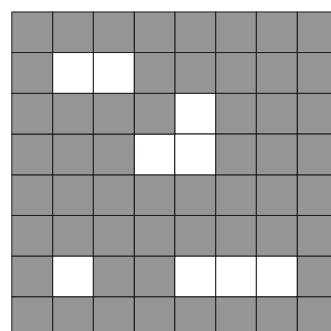
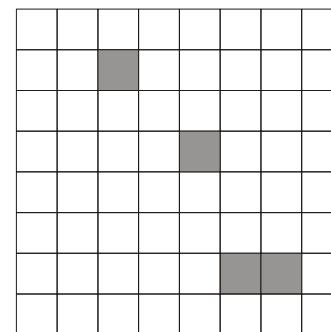




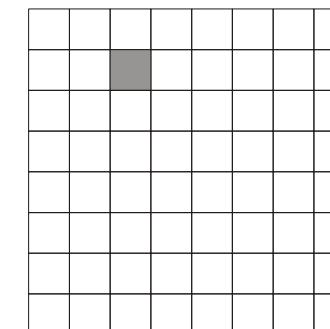
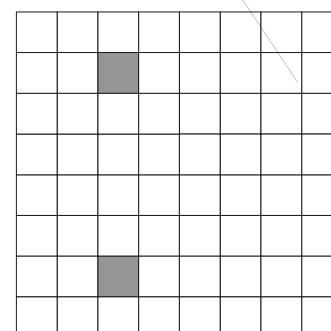
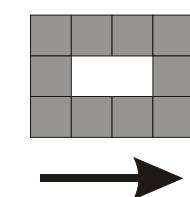
Hit-or-Miss Operator

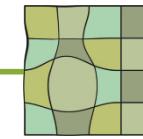


M_1
Erodieren
mit



M_2
Erodieren
mit





Hit-or-Miss Operator

Hit-or-Miss Operator:

$$\begin{aligned} G \otimes (M_1, M_2) &= (G \ominus M_1) \cap (\overline{G} \oplus M_2) \\ &= (G \ominus M_1) \cap (\overline{G \oplus M_2}) \end{aligned}$$

mit $M_1 \cap M_2 = \emptyset$ (sonst wäre das Resultat der Operation die leere Menge)

Hit-or-Miss-Operator für variable Strukturgrößen, z.B.:



Hit

Miss

führt zur Akzeptanz von horizontalen Linien von 3, 4, und 5 Pixeln Länge.

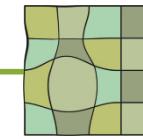
Notation für Hit-or-Miss-Operator:

0 - Miss

1 - Hit

x - weder Miss noch Hit

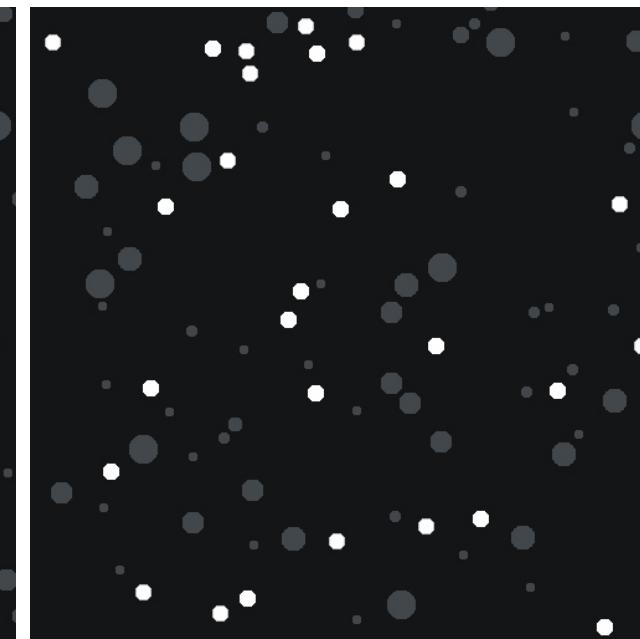
$$M = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x & 1 & 1 & 1 & x & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$



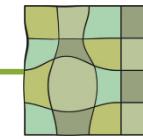
Beispiel



Kreise mit Radius von 6 Pixel



Kreise mit Radius 6-7 Pixel



Hit-or-Miss-Operatoren

$$M_I = \begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$$

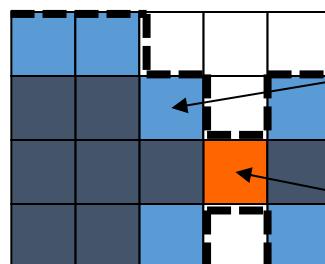
Entfernung einzelner Pixel

$$M_C = \begin{matrix} x & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$$

detektiert untere, rechte Ecken eines Objekts

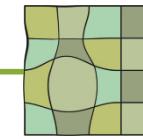
$$M_{T1} = \begin{matrix} 0 & 0 & 0 \\ x & 1 & x \\ 1 & 1 & 1 \end{matrix}$$

findet alle Randpunkte von oben, die ein Objekt nicht teilen würden, wenn sie entfernt würden.



Diese Punkte würden gefunden werden.

Dieser Punkt würde nicht gefunden werden.



Thinning mit Hit-or-Miss-Operatoren

$$S_{T1} = \begin{matrix} 0 & 0 & 0 \\ x & 1 & x \\ 1 & 1 & 1 \end{matrix}$$

$$S_{T2} = \begin{matrix} 0 & x & 1 \\ 0 & 1 & 1 \\ 0 & x & 1 \end{matrix}$$

$$S_{T3} = \begin{matrix} 1 & 1 & 1 \\ x & 1 & x \\ 0 & 0 & 0 \end{matrix}$$

$$S_{T4} = \begin{matrix} 1 & x & 0 \\ 1 & 1 & 0 \\ 1 & x & 0 \end{matrix}$$

$$S_{T5} = \begin{matrix} 0 & 0 & x \\ 0 & 1 & 1 \\ x & 1 & 1 \end{matrix}$$

$$S_{T6} = \begin{matrix} x & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & x \end{matrix}$$

$$S_{T7} = \begin{matrix} 1 & 1 & x \\ 1 & 1 & 0 \\ x & 0 & 0 \end{matrix}$$

$$S_{T8} = \begin{matrix} x & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & x \end{matrix}$$

Ziel: Skelettierung

Methode: Randpixel solange entfernen, bis der zusammenhängende Schriftzug aufgelöst werden würde.

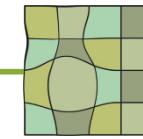
Thinning-Operator von oben:

$$G \oslash S_T = G \setminus (G \otimes S_{T1})$$

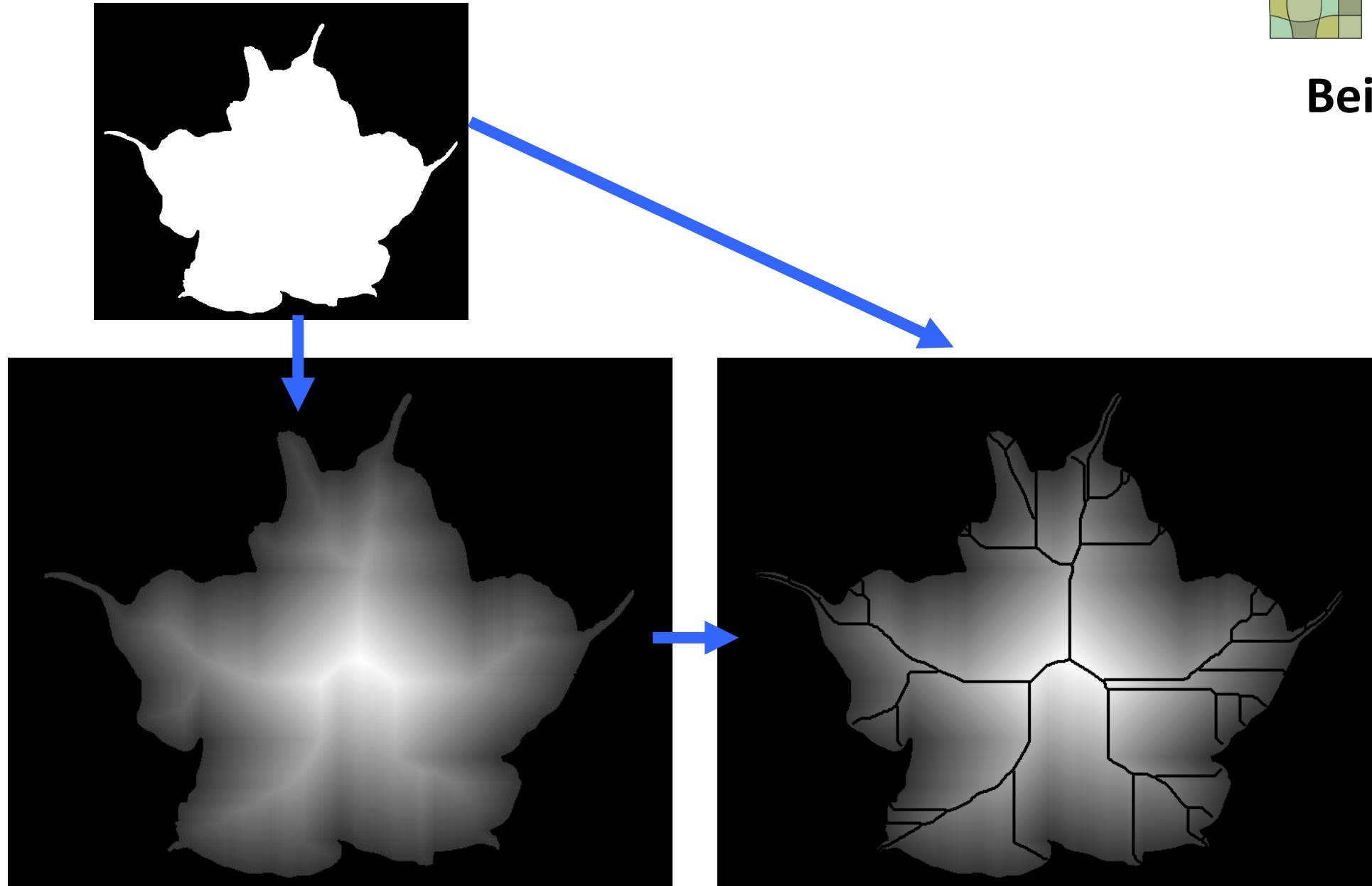
Symmetrisches Thining:

$$G \oslash S_T = G \setminus \bigcup_{n=1,8} (G \otimes S_{Ti})$$

Thinning wird wiederholt, bis $G \oslash S_T = G$ ist.



Beispiel



Was sollten Sie gelernt haben?

Morphologische Operationen: Formverändernde oder formauswertende Operationen auf Segmenten.

Morphologische Filter zur:

- Unterdrückung von Artefakten nach einer Segmentierung
- Suche nach vorgegebenen Formen
- Randbestimmung, Distanztransformation und Morphing
- Skelettierung von Segmenten

Famous Last Question

Wie lässt sich das folgende
Bild nach Textur
segmentieren?

