

Bilder

Visual Computing Winter Semester 2018-2019



Prof. Dr. A. Kuijper

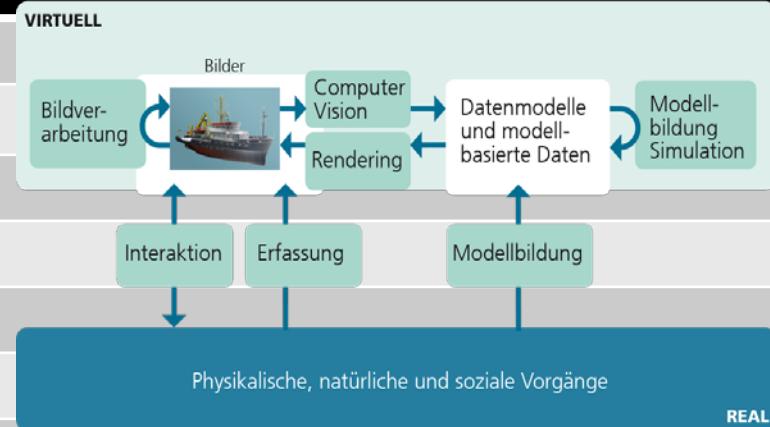
Mathematical and Applied Visual Computing (MAVC)
Graphisch-Interaktive Systeme (GRIS)
Fraunhofer IGD
Fraunhoferstrasse 5
D - 64283 Darmstadt

E-Mail: office@gris.tu-darmstadt.de
<http://www.gris.tu-darmstadt.de>
<https://www.mavc.tu-darmstadt.de>

Semesterplan



Datum	
20. Okt	Einführung + <u>Visual Computing</u>
27. Okt	<u>Wahrnehmung</u>
03. Nov	<u>Objekterkennung</u> und <u>Bayes</u>
10. Nov	<u>Fourier Theorie</u>
17. Nov	<u>Bilder</u>
24. Nov	Bildverarbeitung
01. Dez	Grafikpipeline & Eingabemodalitäten & VR+AR
08. Dez	Transformationen & 2D/3D Ausgabe
15. Dez	3D-Visualisierung
12. Jan	X3D – 3D in HTML
19. Jan	Informationsvisualisierung
26. Jan	Farbe
02. Feb	User Interfaces + Multimedia Retrieval
01. Mrz	Klausur



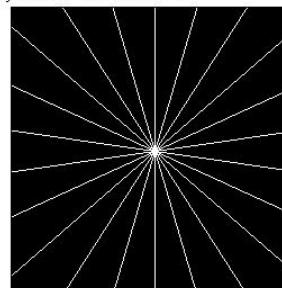


- Darstellung einer Funktion in einem anderen Raum
- Benutze Basisfunktionen -> vereinfachte Darstellung
 - Kartesisches Koordinatensystem: Vektoren für x - und y -Richtung
 - Polarkoordinaten: Bei konstanter Länge ist nur der Winkel wichtig
 - Fourierraum (Frequenz, Sinus und Kosinus):
 - 1D: Musik, z.B. $A = 440$ Hz
 - Ähnlich bei Bildern:
 - Grobe Struktur rund um Ursprung
 - Hoch frequentierte Strukturen (Rauschen!) „weiter“ weg, kleine Amplitude
- Vereinfachte Darstellung → wenige Koeffizienten
 - Guter Merkmalsvektor
 - Gute Kompression (Thema dieser Vorlesung)
 - Zu wenige Koeffizienten berücksichtigt: Aliasing ☹

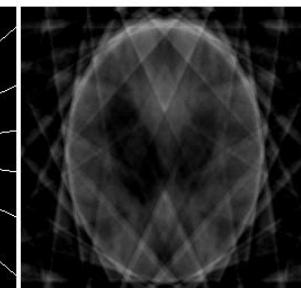
- Fast Fourier Transform (FFT)
 - Algorithmus zur effizienten Berechnung der Werte einer diskreten Fourier-Transformation (DFT)
- Compressive (sparsity) sampling / sensing
 - Benutzt Vorwissen (zB. wenig Struktur)
 - ***Suche eine geeignete Basis!***
- Beispiel: MRI Rekonstruktion Shepp-Logan Phantom
- <http://sunbeam.ece.wisc.edu/csaudio/>
http://archive.wired.com/magazine/2010/02/ff_algorithm/



• Sparse projections: 11 radial lines



available portion of the spectrum
(11 radial lines)



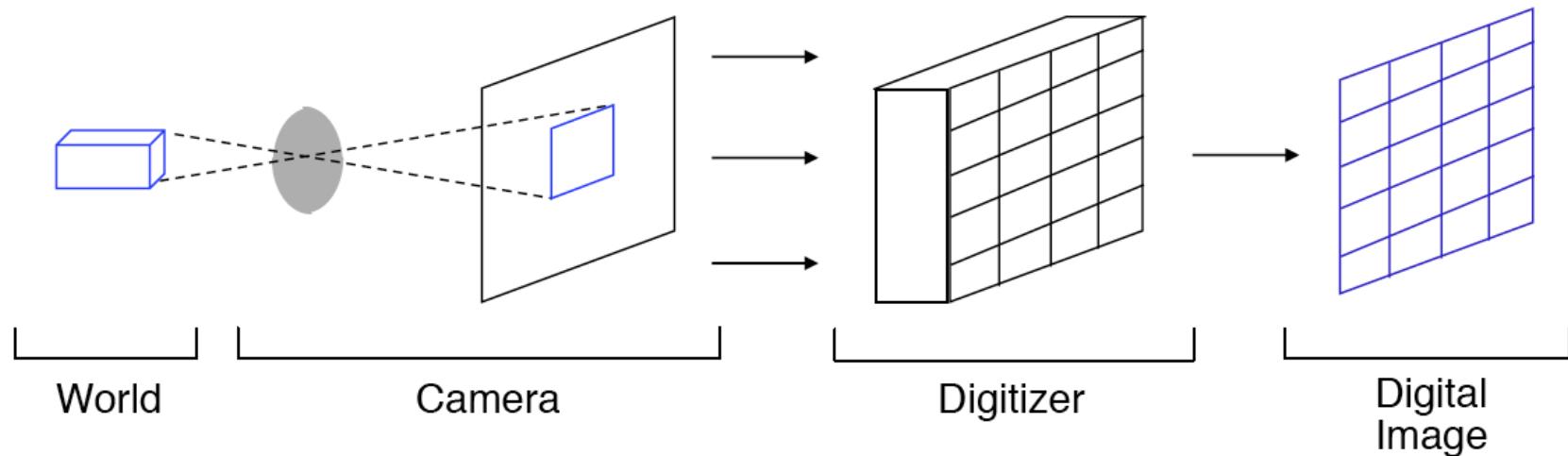
Back-projection estimate



Estimate after convergence
(exact reconstruction)

Digitale Kamera

- Bildverarbeitungsprozess:
 - (Loch-) Kameramodell
 - Rasterisierung zur Erzeugung eines digitalen Bildes

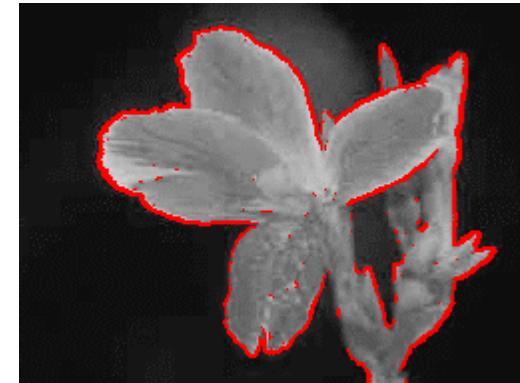


Digitale Bildverarbeitung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Kantenextraktion



Pixelcorrektur



Überblick



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- **Bildverbesserung**
 - Histogramm

- Bildfilterung
 - Im Ortsraum
 - Im Frequenzraum
- Bildkompression

Bildverbesserung



- Verbesserung und Aufbereitung der Bildinformation für den Betrachter
- Anwendungsspezifisch, keine „allgemeine Theorie“
- Angewendete Methode ist abhängig von Bild und Betrachter

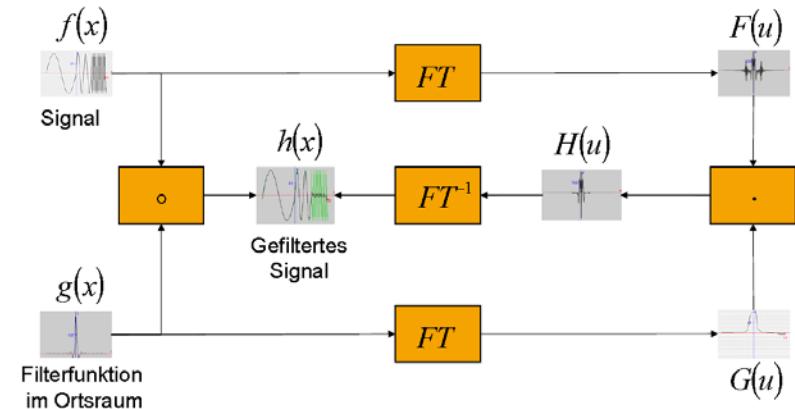


Ortsraum

- Direkte Manipulation der Pixelwerte im Bildbereich
 - Pixeloperationen
 - Filteroperationen

Frequenzraum

- Bildtransformation
 - DFT: Discrete Fourier Transform
 - DCT: Discrete Cosine Transform
 - ...
 - Manipulation der Transformierten
 - Rücktransformation in den Bildbereich
- Siehe Vorlesung „Fouriertheorie“



Typische Anwendungen



- Korrektur von Nicht-Linearitäten der Kamera
- Anpassung Helligkeit, Kontrast
- Bildbereiche hervorheben oder unterdrücken
- Bild ausgleichen

- Oft Lookup-Table-Implementierung

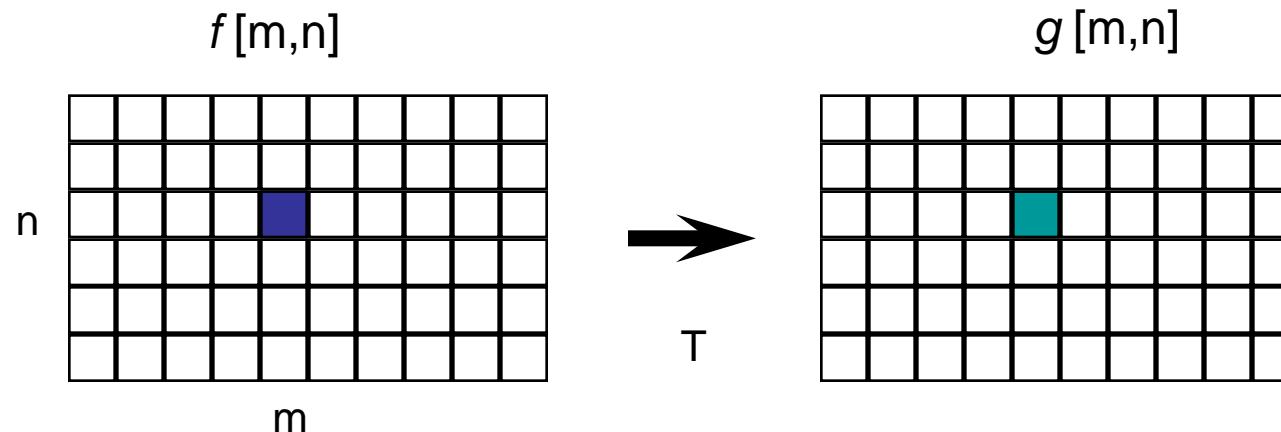
Pixeloperationen



- Manipulation eines Pixels **unabhängig** von seiner Nachbarschaft
- Globaler Kontext

→ **Grauwert Abbildung (Mapping)**

$$g[m, n] = T(f[m, n])$$



Pixeloperationen



- Manipulation eines Pixels **unabhängig** von seiner Nachbarschaft
- Globaler Kontext

→ **Grauwert Abbildung (Mapping)**

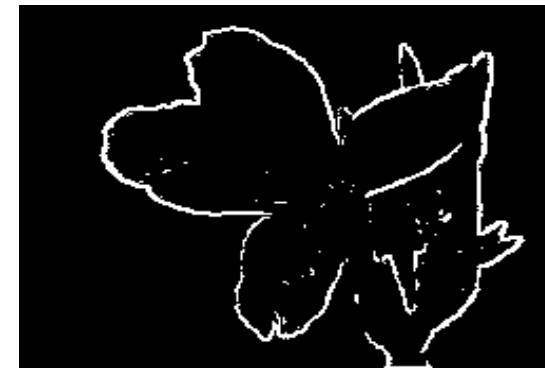
$$g[m, n] = T(f[m, n])$$

$f[m, n]$



\rightarrow
 T

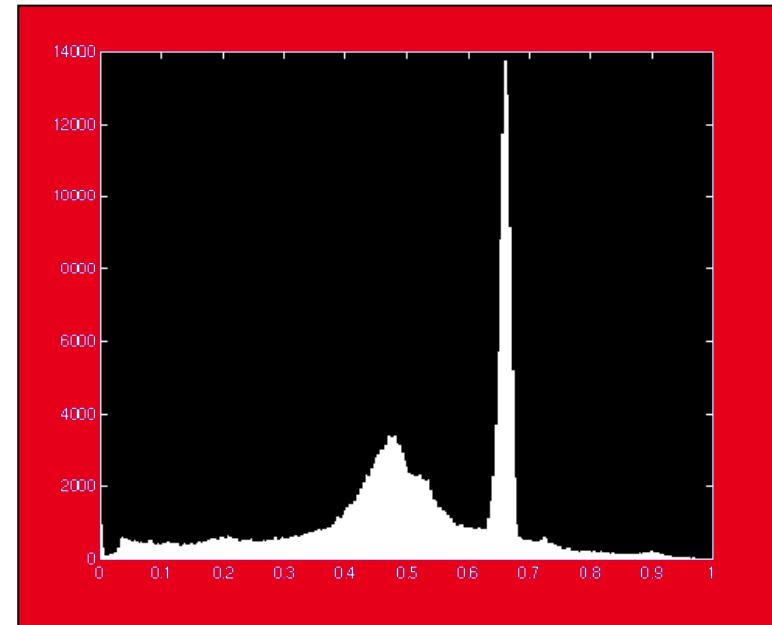
$g[m, n]$



Histogramm



- Graphische Darstellung der Häufigkeitsverteilung metrisch skalierten Merkmale.
- Statistische Verteilung der Grauwerte,
zB. Schwarz=0, Weiß=1 oder 2^8-1





Bilddynamik

- Bereich reeller Lichtintensitäten, der auf die Grauwertskala abgebildet wird

Bildkontrast

- Bereich der Grauwertskala, der zur Darstellung der Bildinformation ausgenutzt wird

Bildhelligkeit

- Beleuchtungsstärke (Grauwert)

Zusammenhang

- Bildhelligkeit eines Grauwertbildes = Mittelwert aller Grauwerte
- Bildkontrast = Varianz aller Grauwerte.

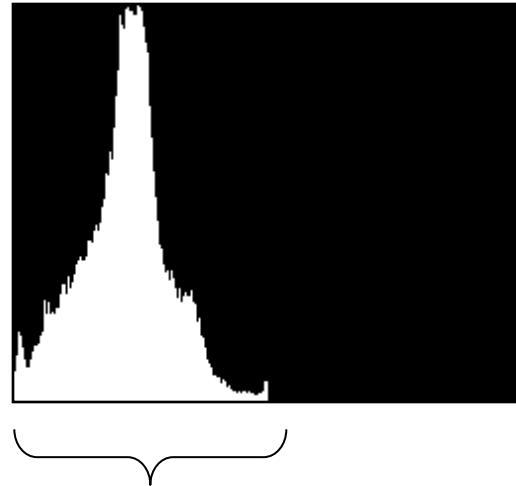
Bildhelligkeit (Mittelwert aller Grauwerte)



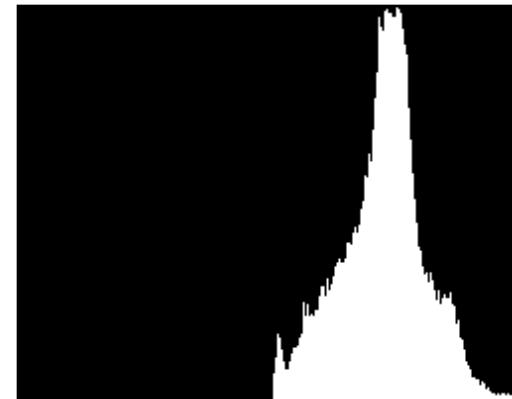
TECHNISCHE
UNIVERSITÄT
DARMSTADT



Dunkles Bild



gleicher Kontrast



Helles Bild



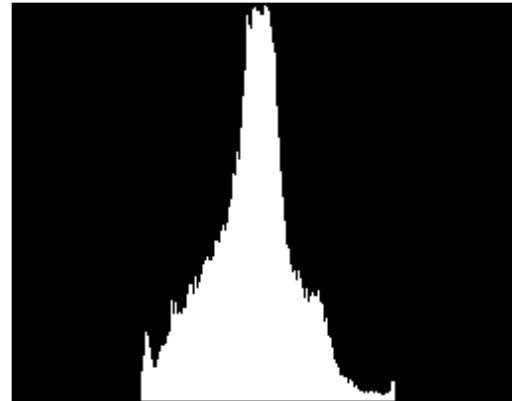
Bildkontrast (Varianz aller Grauwerte)



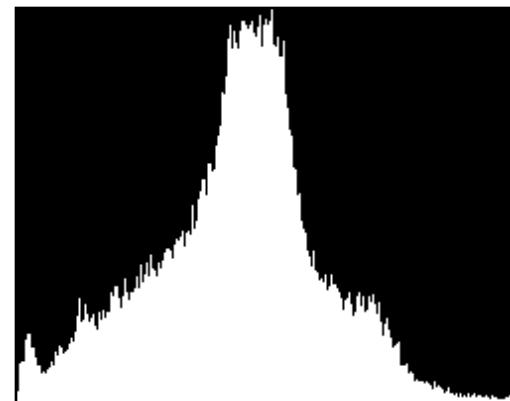
TECHNISCHE
UNIVERSITÄT
DARMSTADT



Kontrastarmes Bild



gleiche Helligkeit



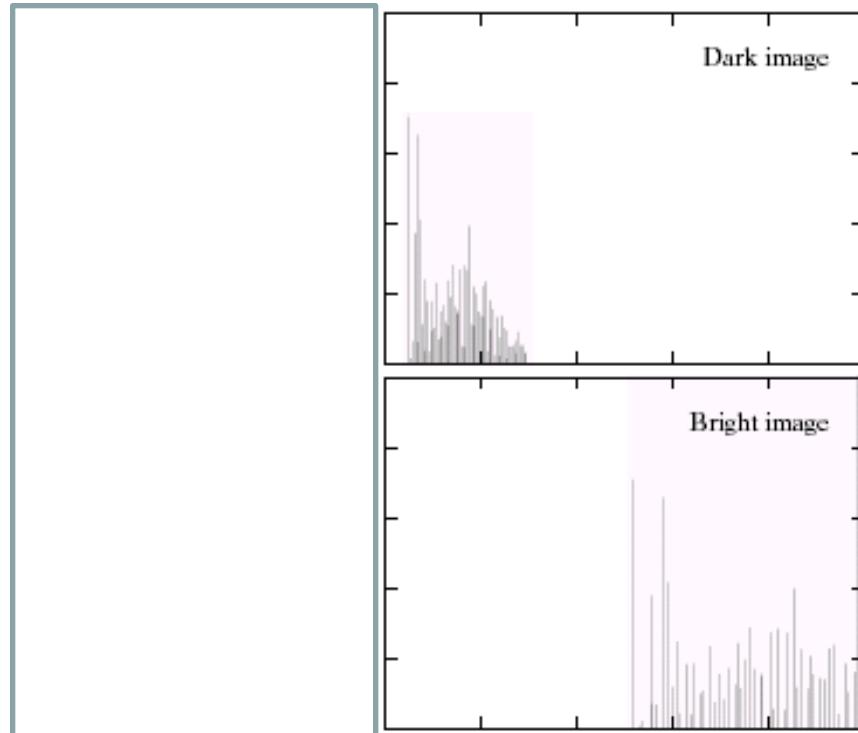
Kontrastreiches Bild



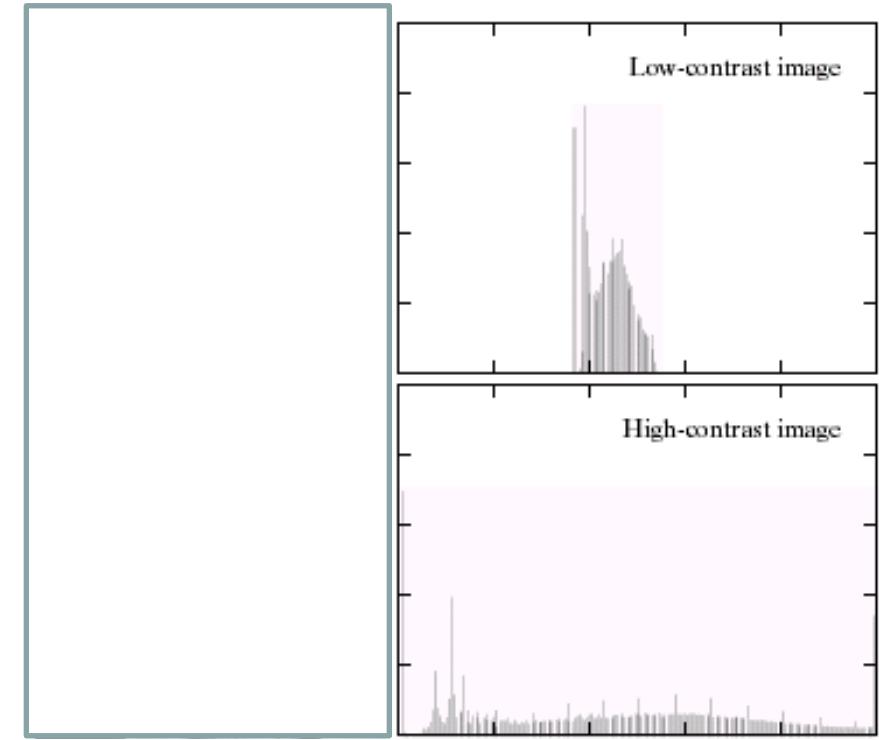
Wichtige Histogramm-Aussagen



Dunkles Bild



Kontrastarmes Bild



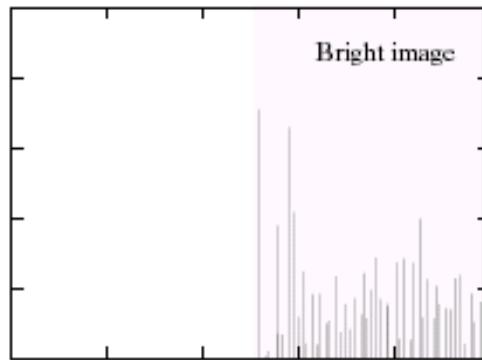
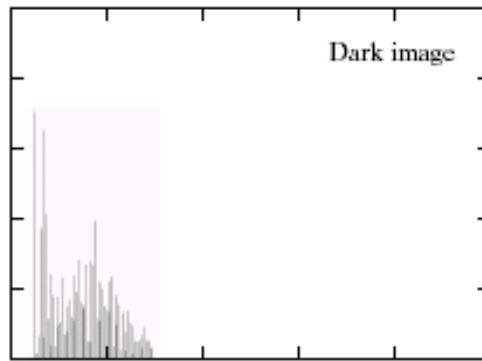
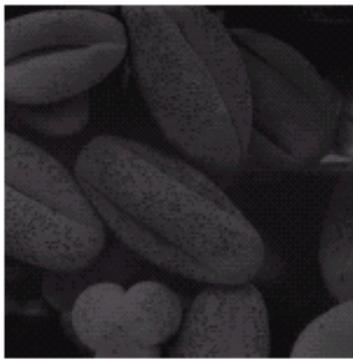
Helles Bild

Kontrastreiches Bild

Wichtige Histogramm-Aussagen

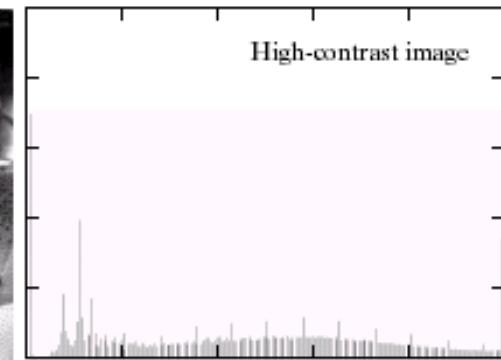
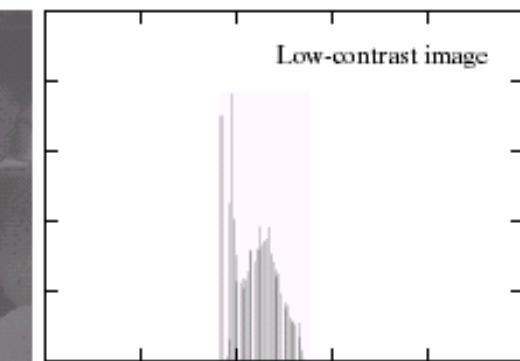
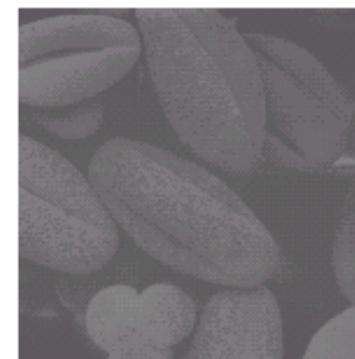


Dunkles Bild



Helles Bild

Kontrastarmes Bild



Kontrastreiches Bild

Pixeloperationen



- Negativ
- Binärisierung / Thresholding
- Fensterung
- Kontrastspreizung
- Dynamikkompression
- Gammakorrektur (Bildschirm!)
- Helligkeit
- Histogrammausgleich
- Differenz
- Mittelung

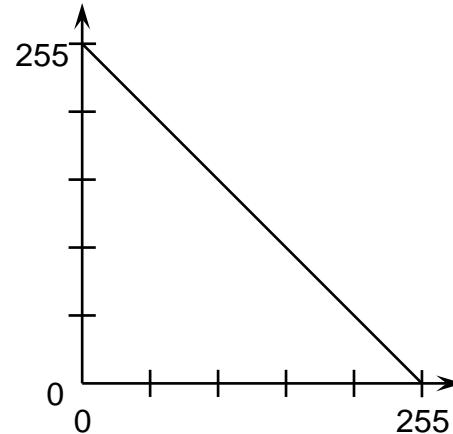
Sehe auch:

https://de.wikipedia.org/wiki/Punktoperator_%28Bildverarbeitung%29

Bildnegativ



$$g[m,n] = f_{\max} - f[m,n]$$



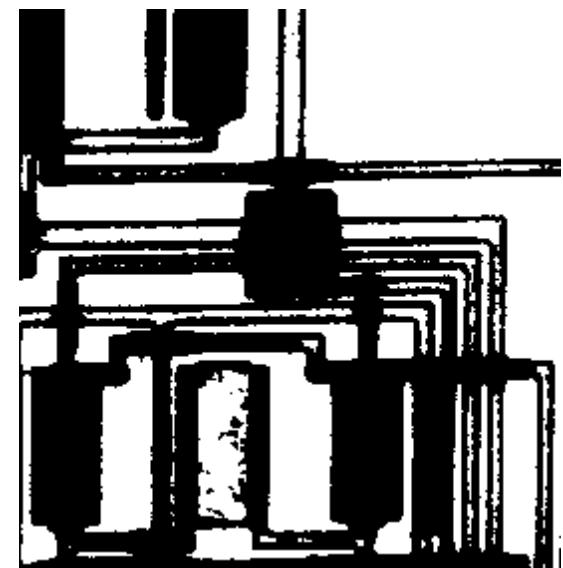
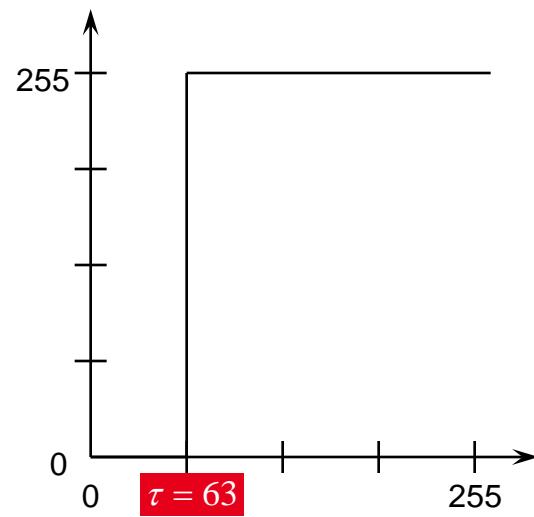
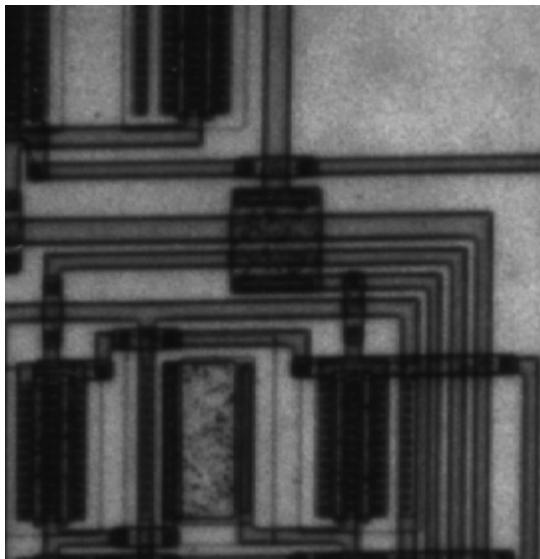
Binärisierung / Thresholding



Thresholding der Grauwerte eines Bildes

$$g[m,n] = \begin{cases} f_{\max} & f[m,n] > \tau \\ f_{\min} & f[m,n] \leq \tau \end{cases}$$

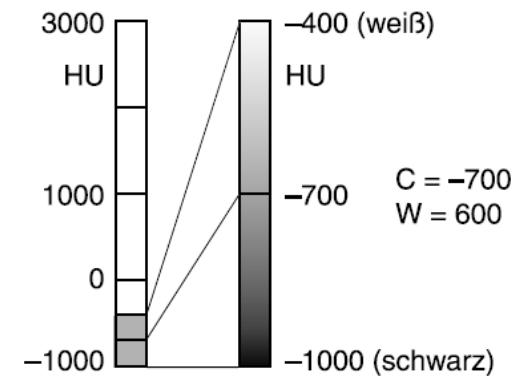
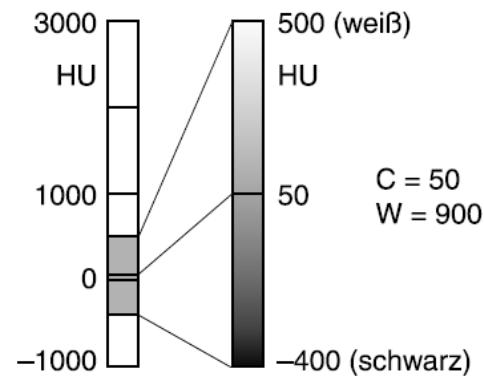
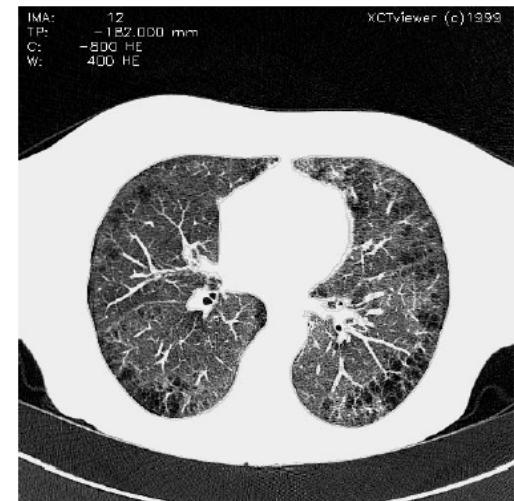
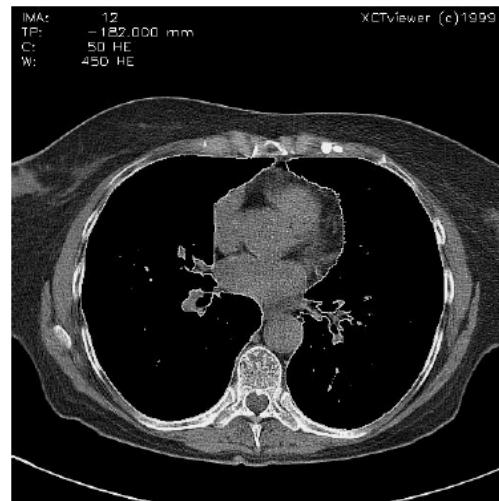
Beim Spezialfall Binärisierung gilt $f_{\max} := 1$ und $f_{\min} := 0$



Grauwertfensterung



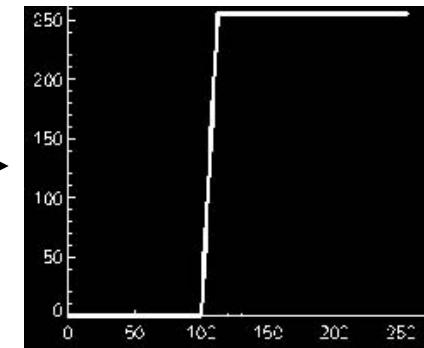
- Hervorheben eines bestimmten Intensitätsintervalls im Bild



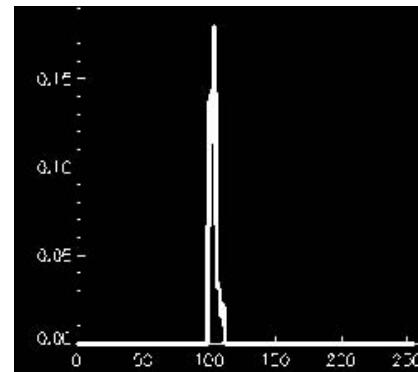
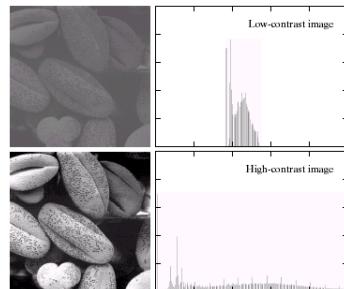
Kontrastspreizung



- Abbildung der Grauwerte auf eine neue Grauwertskala anhand einer
 - *einwertigen*
 - *monotonen*Funktion



Sehe Folien 19:



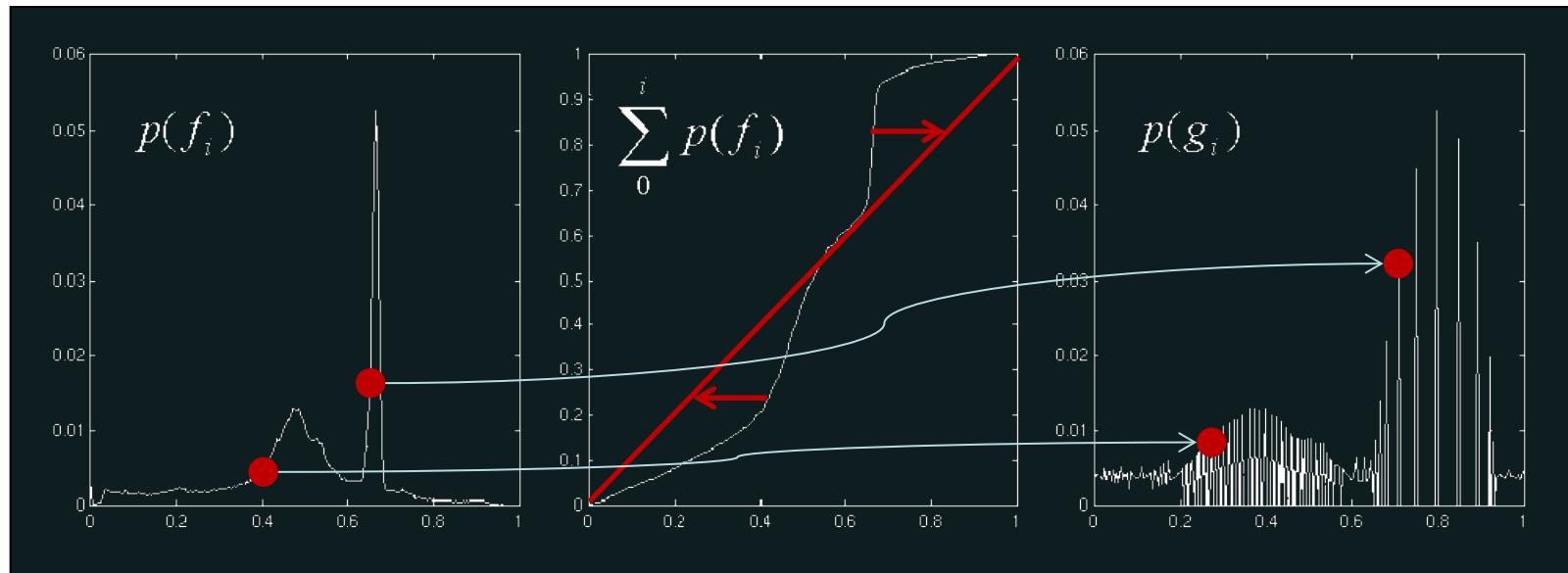
Histogrammausgleich



- Transformation der Grauwertskala anhand der Kurve der Summenwahrscheinlichkeit

$$p(g) = \max(\text{Intensität}) \cdot \sum_{i=0}^g p(i)$$

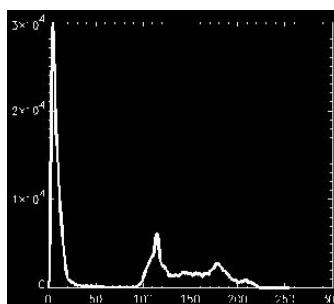
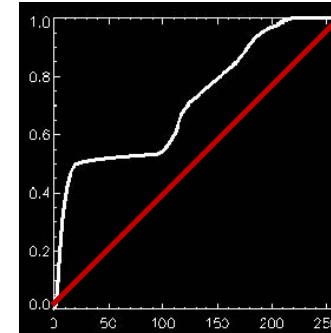
- Verlustbehaftet → nicht umkehrbar!



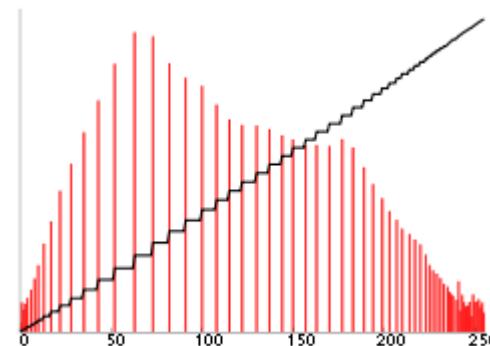
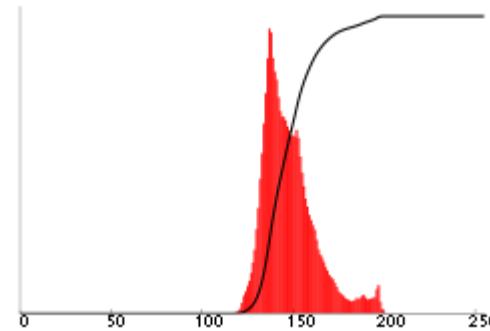
Histogrammausgleich



TECHNISCHE
UNIVERSITÄT
DARMSTADT



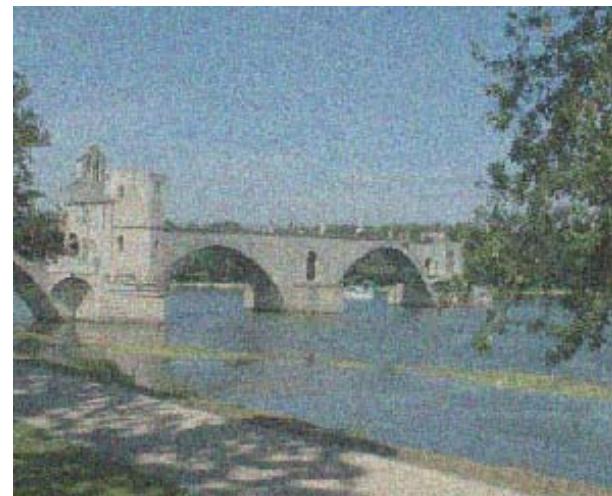
Histogrammausgleich



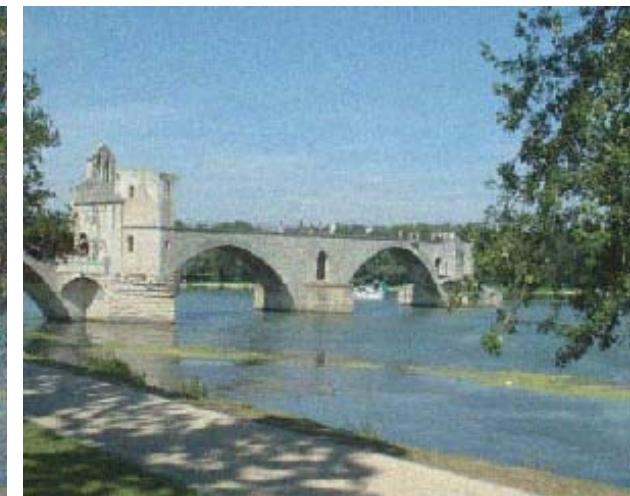
- Verteilung der Balken auf der x-Axe ändert sich.
- Sehe auch https://en.wikipedia.org/wiki/Histogram_equalization

Unterdrückung von unkorreliertem Rauschen durch Mittelung über k Aufnahmen

$$g(m, n) = \frac{1}{k} \sum_{i=0}^{k-1} f_i(m, n)$$



10 x



50 x

Überblick



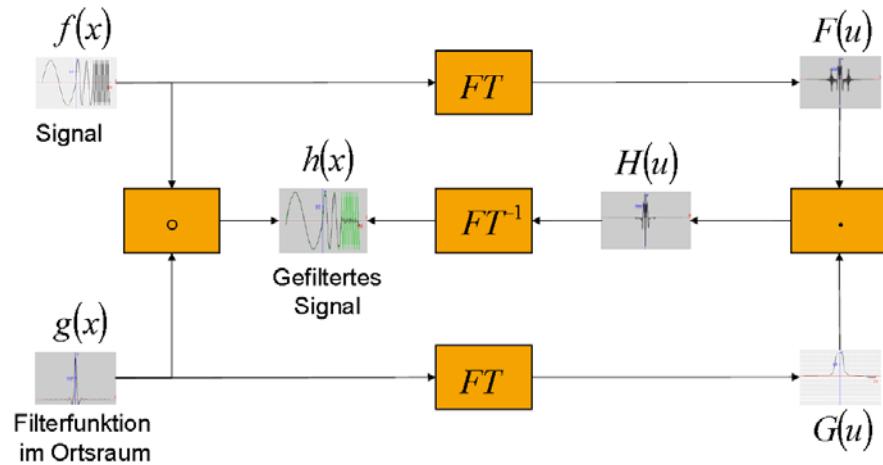
TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Bildverbesserung
 - Histogramm
- **Bildfilterung**
 - Im Ortsraum
 - Im Frequenzraum
- Bildkompression

Noch einmal:

- **Ortsraum**
 - Direkte Manipulation der Pixelwerte im Bild

- **Frequenzraum**
 - Bildtransformation
 - Manipulation der Koeffizienten im Frequenzraum
 - Rücktransformation in den Bildbereich



Filtermasken

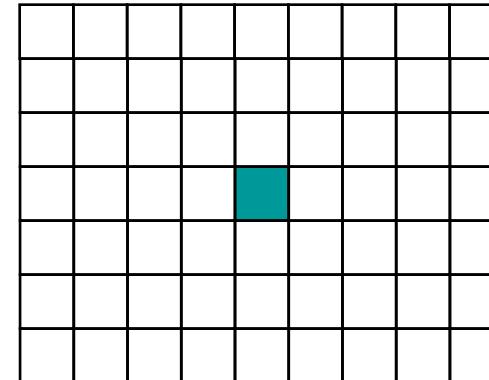
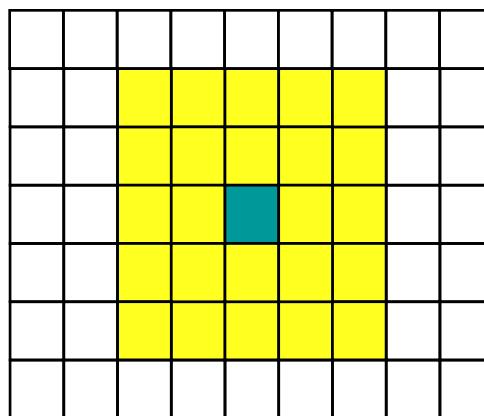


- Manipulation eines Pixels **abhängig** von seiner Nachbarschaft
- Lokaler Kontext



Faltung mit einer $k \times l$ Matrix

(Filter, Maske, Kernel, Template, Fenster, Konvolution...)



Lineare Filterung (Faltung)

$$(f * w)(m, n) = \sum_{i=-k/2}^{k/2} \sum_{j=-l/2}^{l/2} w(i, j) \cdot f(m + i, n + j)$$

	$w_{-2, -2}$	$w_{-2, -1}$	$w_{-2, 0}$	$w_{-2, 1}$	$w_{-2, 2}$	
	$w_{-1, -2}$	$w_{-1, -1}$	$w_{-1, 0}$	$w_{-1, 1}$	$w_{-1, 2}$	
	$w_{0, -2}$	$w_{0, -1}$	$w_{0, 0}$	$w_{0, 1}$	$w_{0, 2}$	
	$w_{1, -2}$	$w_{1, -1}$	$w_{1, 0}$	$w_{1, 1}$	$w_{1, 2}$	
	$w_{2, -2}$	$w_{2, -1}$	$w_{2, 0}$	$w_{2, 1}$	$w_{2, 2}$	

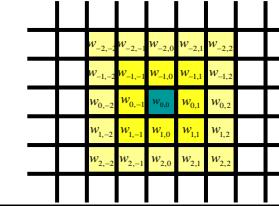
Tiefpass-Filter (Ortsraum)

$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$



- Koeffizienten ausnahmslos positiv
 - Koeffizienten normalisiert (Summe aller Koeffizienten ergibt 1)
 - Produzieren nur positive Werte
 - Randeffekte
-
- „averaging filters“ (*Weichzeichnungsfilter*)
 - Mittelwert Filter
 - Gaussian Filter

Mittelwert Filter



$$\frac{1}{9} \begin{matrix} & 1 & 1 & 1 \\ 1 & & & \\ & 1 & 1 & 1 \\ 1 & & & \\ & 1 & 1 & 1 \end{matrix}$$

3×3
„Boxfilter“

$$\frac{1}{25} \begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 1 & & & & \\ 1 & 1 & 1 & 1 & 1 \\ 1 & & & & \\ 1 & 1 & 1 & 1 & 1 \\ 1 & & & & \\ 1 & 1 & 1 & 1 & 1 \\ 1 & & & & \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$$

Mittelwert Filter

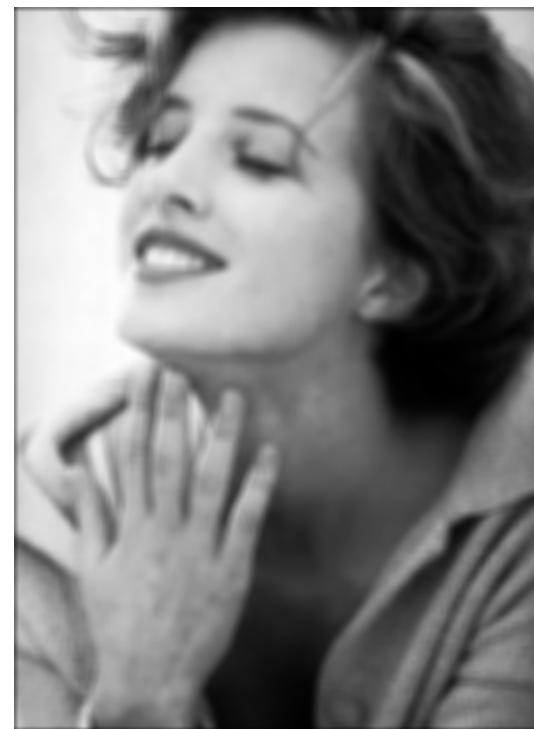
$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$



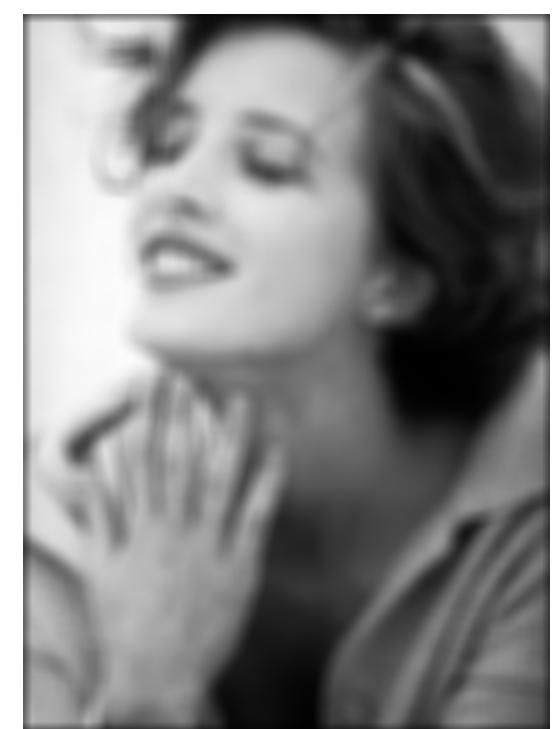
TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original



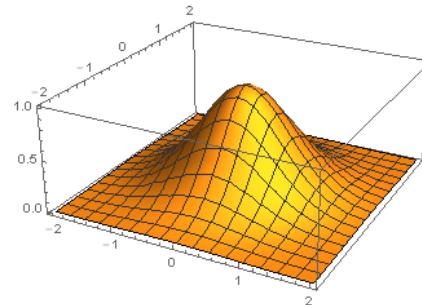
5 x 5



9 x 9

Gauss-Filter

Allgemein



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

Diskrete Approximation

$$\frac{1}{16} \begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix}$$

3×3

„Binomialfilter“

$$\frac{1}{273} \begin{matrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{matrix}$$

5×5

$$\sigma = 1$$

Gauss-Filter

$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$

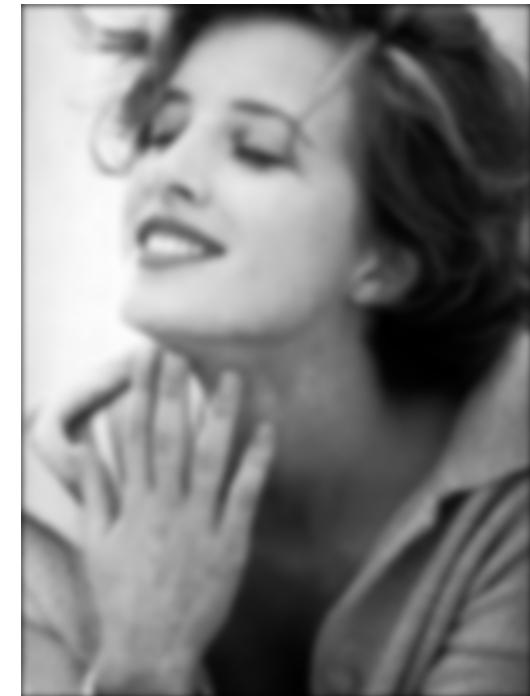


Original



5×5

$$\sigma = 1$$



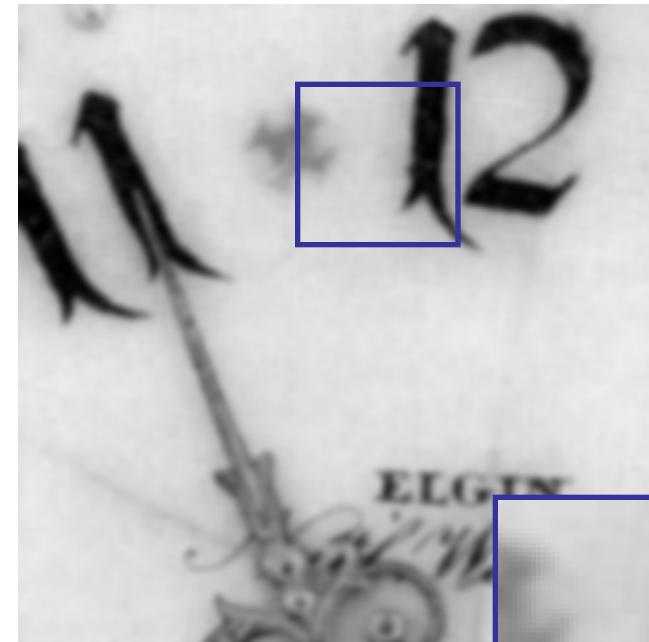
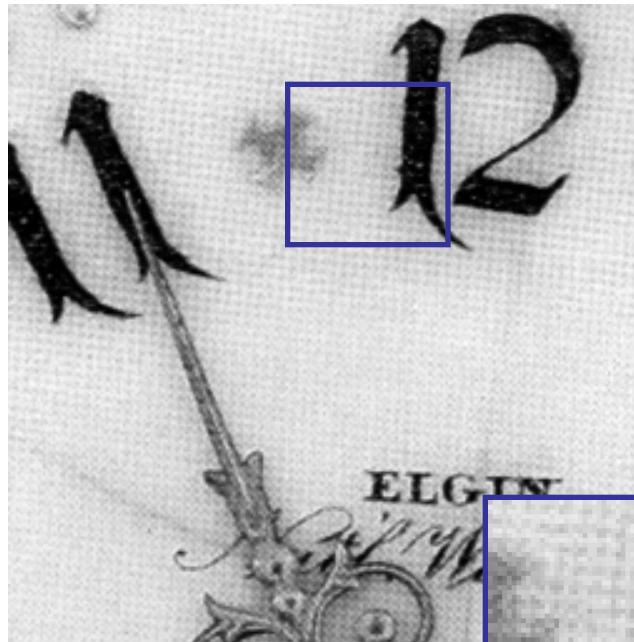
9×9

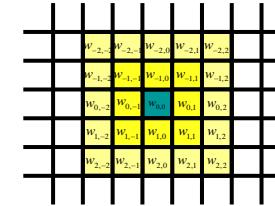
$$\sigma = 2$$

Anwendungsbereiche

$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$

Reduktion von periodischem Rauschen
Verringerung eines störenden Druckrasters





Nichtlinearer Filter!

Kann nicht als Faltung ausgedrückt werden!

- Prinzip: Ersetzt ein Pixel mit dem Medianwert seiner Nachbarschaft
- Grautöne, die im Original erhalten waren (keine Interpolation neuer Werte!)
 - Unterdrückung von isolierten Punkten/Rauschen
 - Schärfe der Kanten bleibt erhalten
- Rechenintensiv (Sortierung)

Vergleich

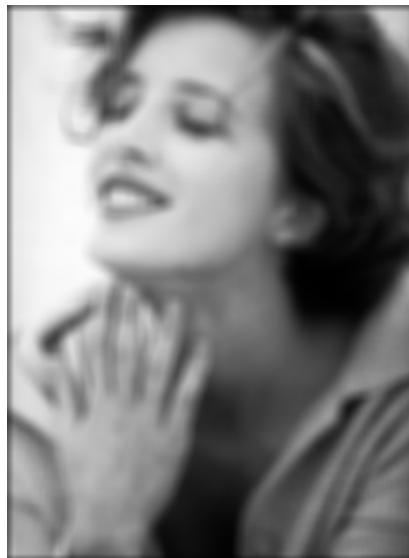
$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$



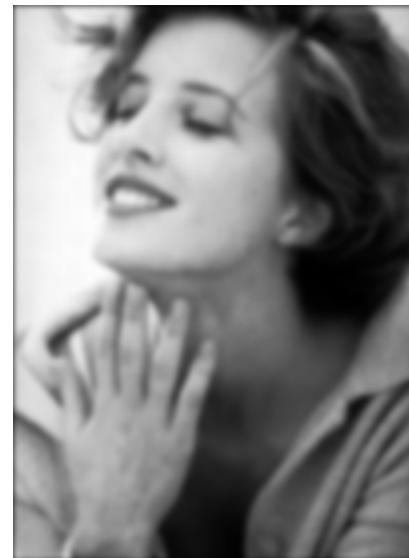
TECHNISCHE
UNIVERSITÄT
DARMSTADT



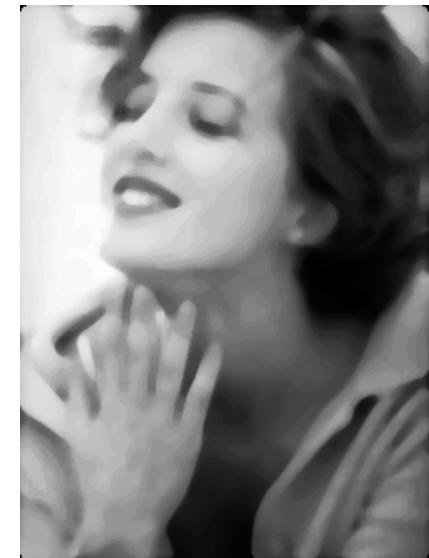
Original



Mittelwert



Gaussian



Median

Hochpass-Filter (Ortsraum)

$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$



- Koeffizienten sowohl negativ als auch positiv
- Koeffizienten normalisiert
(Summe der Koeffizienten ergibt 0)
- Produzieren positive und negative Werte!
- Erste Ableitung/Differenzfilter (partielle Gradienten)
- Zweite Ableitung

Ableitungen

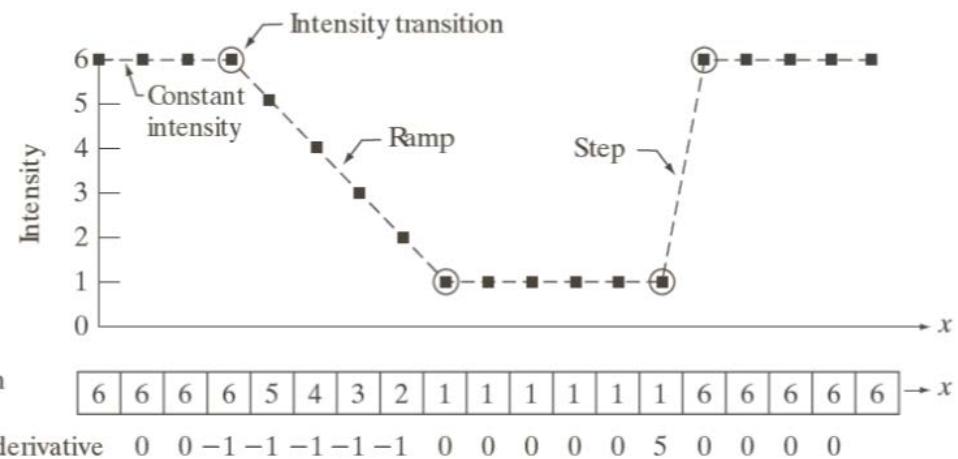
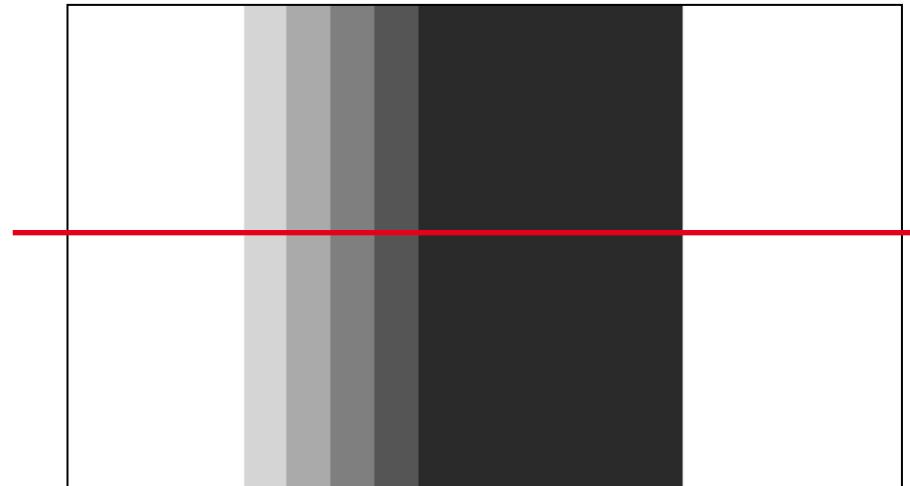


1 dimensional:

$$g(i) = f(i) - f(i-1)$$

Eigenschaften:

1. Null in homogenen Regionen
2. Nicht-Null an Übergängen
3. Grauton-Rampen:
Erste Ableitung Konstant



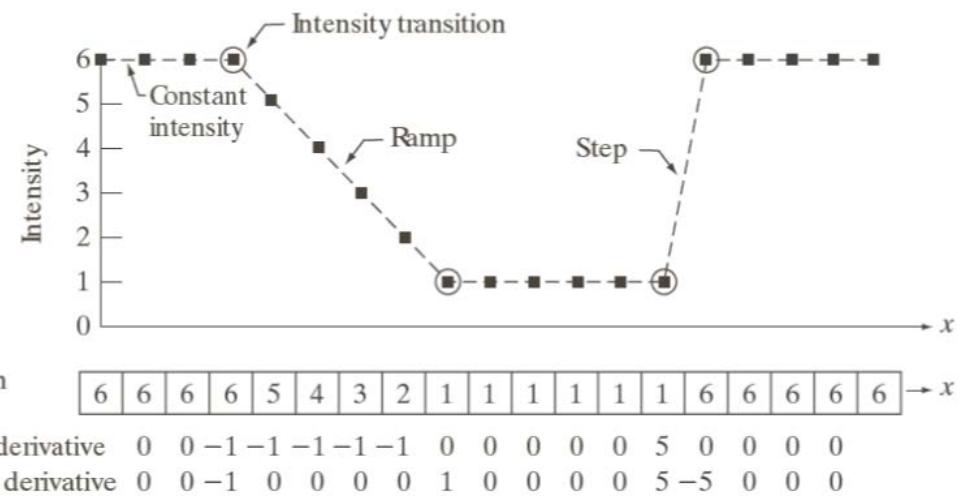
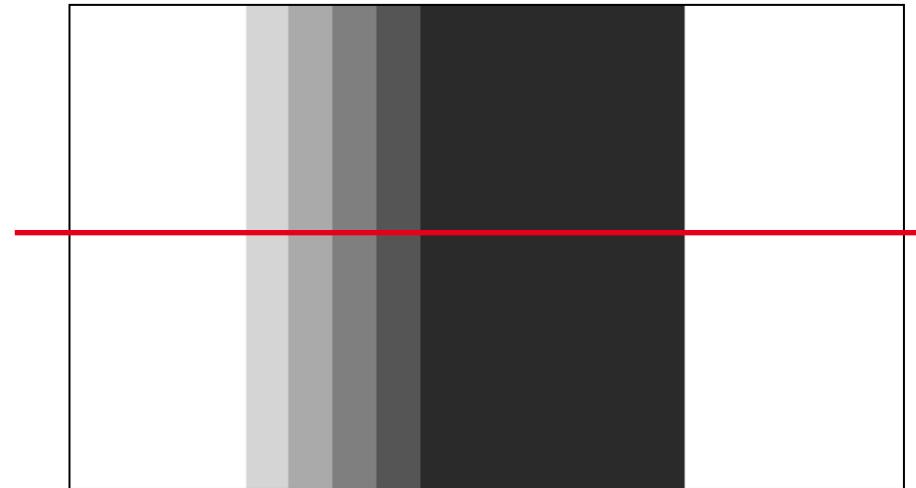
Zweite Ableitung



- „Doppelter“ Gradient:
„Ableitung der Ableitung“

$$g(i) = f(i+1) - 2f(i) + f(i-1)$$

- Großer Ausschlag bei scharfen Unstetigkeiten, einzelnen Punkten (in 2D: dünnen Linien!)
- Mittlerer Ausschlag bei Übergängen (Stufen)
- Kein Ausschlag bei Rampen und homogenen Flächen



In 2D: Laplacian-Filter

$w_{-2,-2}$	$w_{-2,-1}$	$w_{-2,0}$	$w_{-2,1}$	$w_{-2,2}$
$w_{-1,-2}$	$w_{-1,-1}$	$w_{-1,0}$	$w_{-1,1}$	$w_{-1,2}$
$w_{0,-2}$	$w_{0,-1}$	$w_{0,0}$	$w_{0,1}$	$w_{0,2}$
$w_{1,-2}$	$w_{1,-1}$	$w_{1,0}$	$w_{1,1}$	$w_{1,2}$
$w_{2,-2}$	$w_{2,-1}$	$w_{2,0}$	$w_{2,1}$	$w_{2,2}$



Allgemein

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

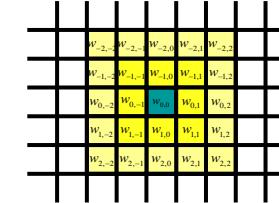
Diskrete Approximation

0	1	0
1	-4	1
0	1	0

3×3



Laplacian of Gaussian Filter

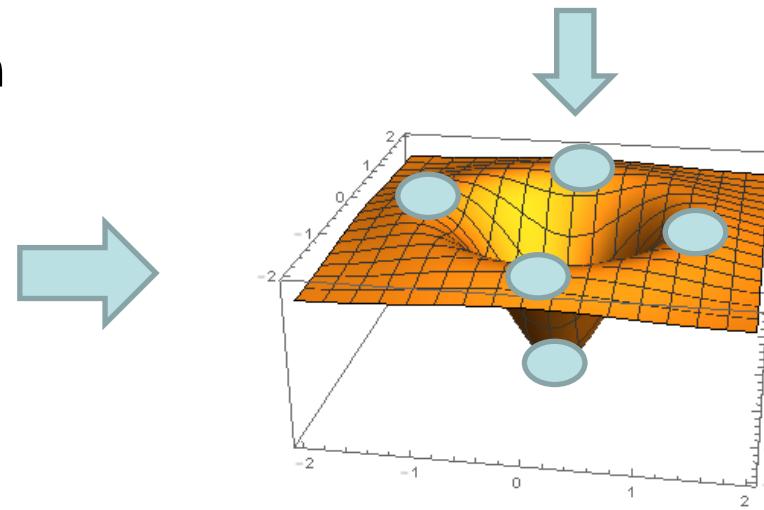


LoG: Anwendung des Laplace-Operators auf eine Gauß-Funktion

- = Marr-Hildreth-Operator
- = Laplacian of Gaussian (LoG)
- = Mexican Hat filter
- = Sombrero filter

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

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



0	1	0
1	-4	1
0	1	0

3 x 3

Laplacian of Gaussian Filter

$w_{-2,-2}$	$w_{-2,0}$	$w_{-2,2}$	$w_{-2,4}$	$w_{-2,6}$
$w_{-1,-2}$	$w_{-1,0}$	$w_{-1,2}$	$w_{-1,4}$	$w_{-1,6}$
$w_{0,-2}$	$w_{0,0}$	$w_{0,2}$	$w_{0,4}$	$w_{0,6}$
$w_{1,-2}$	$w_{1,0}$	$w_{1,2}$	$w_{1,4}$	$w_{1,6}$
$w_{2,-2}$	$w_{2,0}$	$w_{2,2}$	$w_{2,4}$	$w_{2,6}$



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original



5×5

$$\sigma = 1$$



9×9

$$\sigma = 2$$

Laplacian-Filter - Alternativen



Allgemein

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

Zweite Ableitung in x und y Richtung

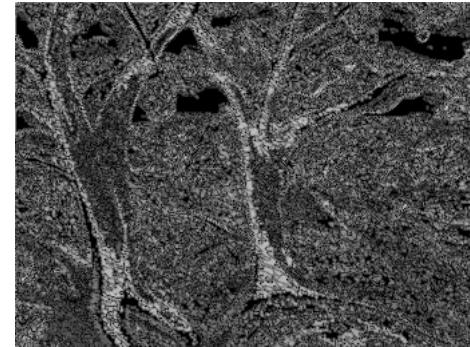
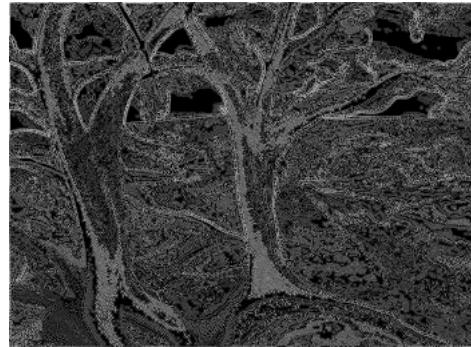
Diskrete Approximation:

$$\frac{1}{x} \begin{array}{|c|c|c|} \hline \alpha & 1-\alpha & \alpha \\ \hline 1-\alpha & -4 & 1-\alpha \\ \hline \alpha & 1-\alpha & \alpha \\ \hline \end{array}$$

$$1 \geq \alpha \geq -1$$

$$x = \begin{cases} 4 & 1 \geq \alpha \geq 0 \\ 4(1-\alpha) & 0 > \alpha \geq -1 \end{cases}$$

Laplacian-Filter



Original

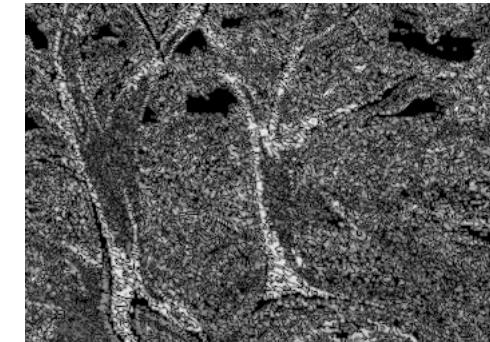
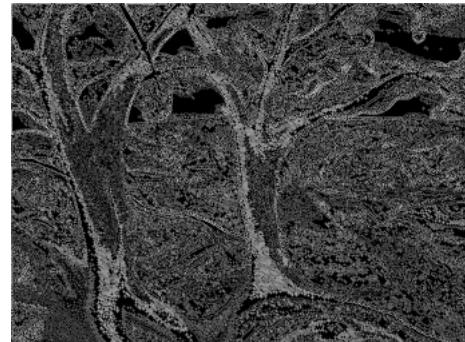
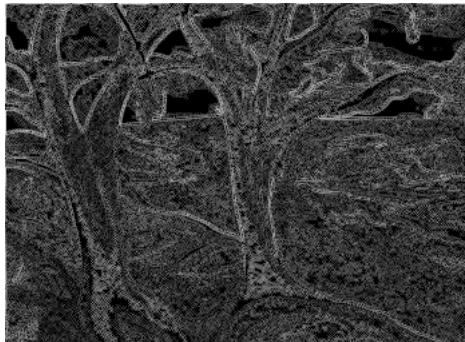
$$\frac{1}{4} \begin{array}{|c|c|c|} \hline 1 & 0 & 1 \\ \hline 0 & -4 & 0 \\ \hline 1 & 0 & 1 \\ \hline \end{array} \quad \alpha = 1.0$$

$$\frac{1}{8} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & -8 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \quad \alpha = 0.5$$

$$\frac{1}{4} \begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array} \quad \alpha = 0.0$$

$$\frac{1}{12} \begin{array}{|c|c|c|} \hline -1 & 3 & -1 \\ \hline 3 & -8 & 3 \\ \hline -1 & 3 & -1 \\ \hline \end{array} \quad \alpha = -0.5$$

$$\frac{1}{8} \begin{array}{|c|c|c|} \hline -1 & 2 & -1 \\ \hline 2 & -4 & 2 \\ \hline -1 & 2 & -1 \\ \hline \end{array} \quad \alpha = -1.0$$

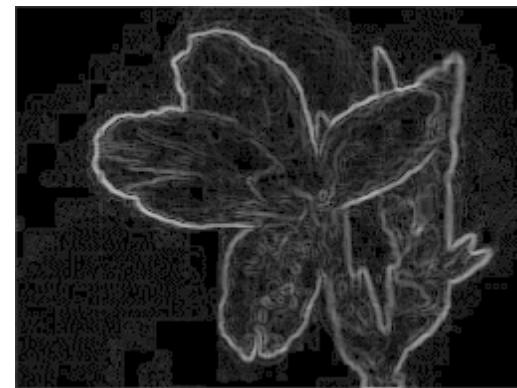


Anwendungsgebiete (2)



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Kantenextraktion



Hochpassfilterung



Binarisierung

Zum Weiterspielen



- <http://setosa.io/ev/image-kernels/>

Demonstriert sehr gut das Konzept von Faltungsfiltern allgemein.
Geniale Seite!

- <http://www.html5rocks.com/en/tutorials/canvas/imagefilters/>

Ein Beispiel mit Code in javascript

- <http://beej.us/blog/data/convolution-image-processing/>

Hier lassen sich vorgefertigte und eigene Filter auf ein Bild anwenden.

Überblick



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Bildverbesserung
 - Histogramm
- **Bildfilterung**
 - Im Ortsraum
 - **Im Frequenzraum**
- Bildkompression

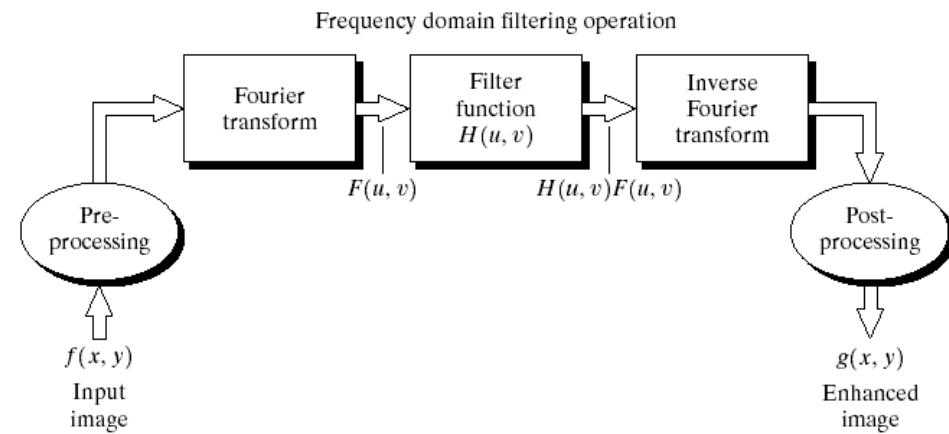
Filterung im Frequenzraum



Faltungssatz: Eine Faltung im Ortsraum entspricht einer Multiplikation im Frequenzraum (und umgekehrt)

$$f * h = F^{-1}(F(f) \cdot F(h))$$

- Transformation des Bildes f und des Filterkerns h
- Multiplikation im Frequenzraum
- Rücktransformation des Ergebnisses in den Ortsraum



[Komplexe Zahlen]



- Zwei Komponenten: Real- und Imaginärteil

$$z = a + ib, \quad i = \sqrt{-1}$$

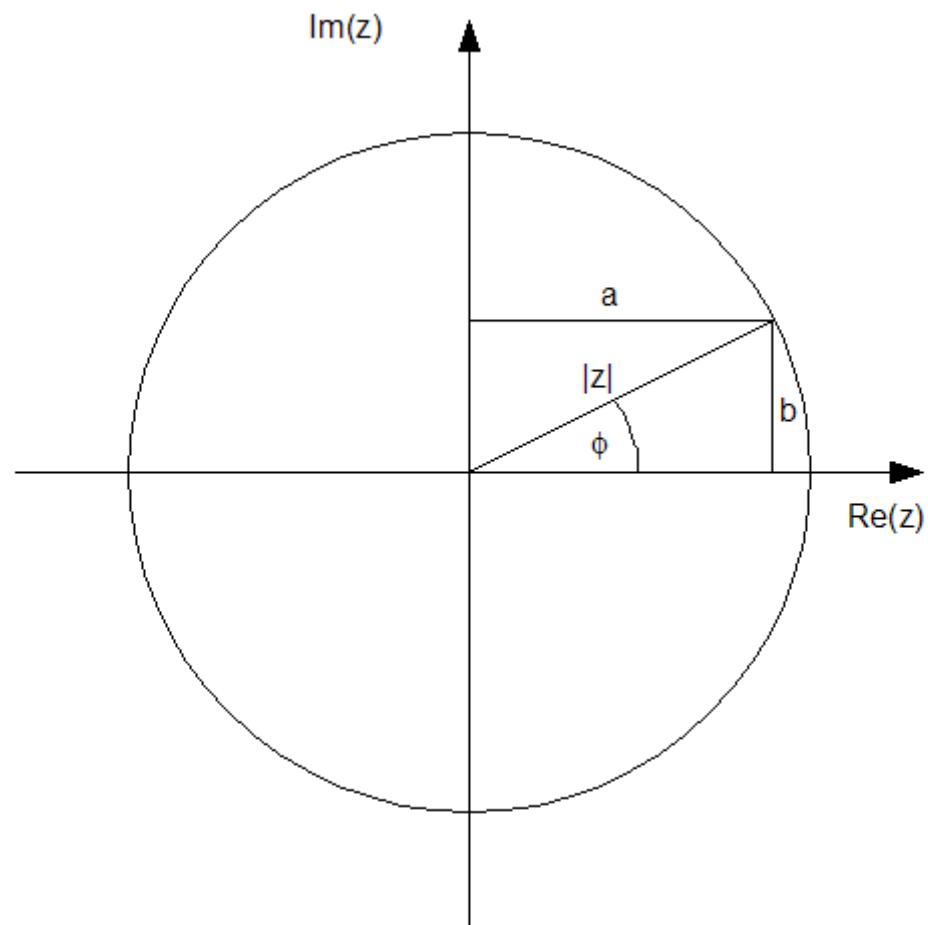
$$z = |z| \cdot e^{i\phi}, \quad \phi = \arctan\left(\frac{b}{a}\right)$$

- Euler-Identität ($|z|=1!$)

$$e^{i\phi} = \cos(\phi) + i \sin(\phi)$$

$$a = \cos(\phi) = \frac{e^{i\phi} + e^{-i\phi}}{2}$$

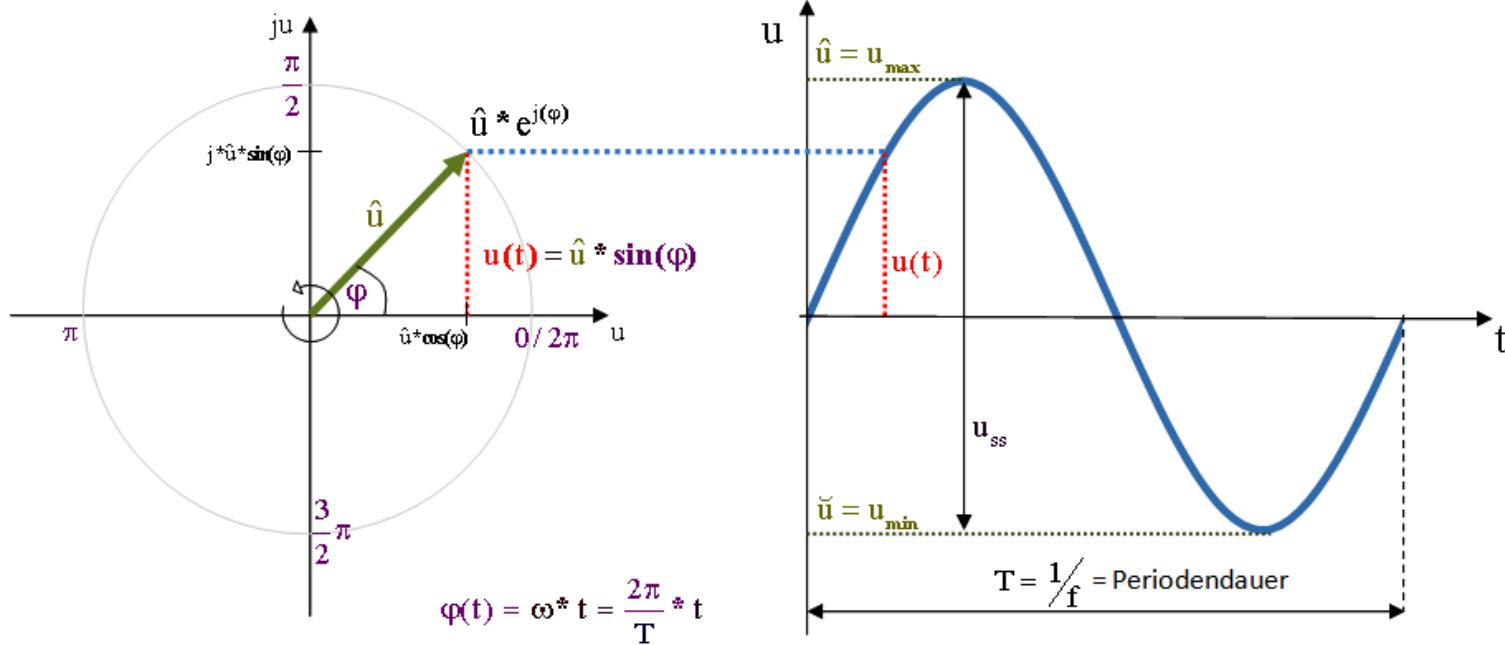
$$b = \sin(\phi) = \frac{e^{i\phi} - e^{-i\phi}}{2i}$$



Periodische Funktion



Elektrotechniker benutzen einen j statt einen i



- Sinusförmige Schwingung: $f(t) = u_{\max} \cdot e^{i\varphi(t)}$

u_{\max} : **Amplitude**, $\varphi(t)$: **Phase**

Die Fourier-Transformation 1D



- Fouriertransformation

$$f(x) \rightarrow F(u)$$

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux} dx$$

- inverse Fouriertransformation

$$F(u) \rightarrow f(x)$$

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{+i2\pi ux} du$$

- Oft ist $f(x)$ reell, $F(u)$ ist komplex:

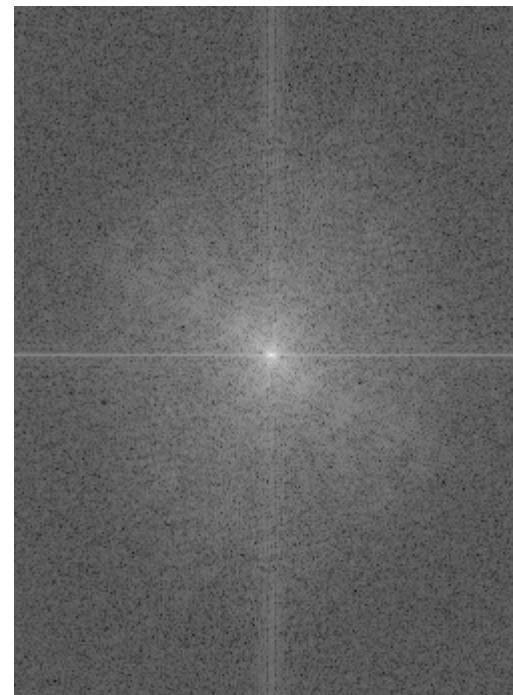
$$F(u) = \operatorname{Re}(F(u)) + i \operatorname{Im}(F(u))$$

- Fourerraum: Frequenzen mit Amplituden und Phasen

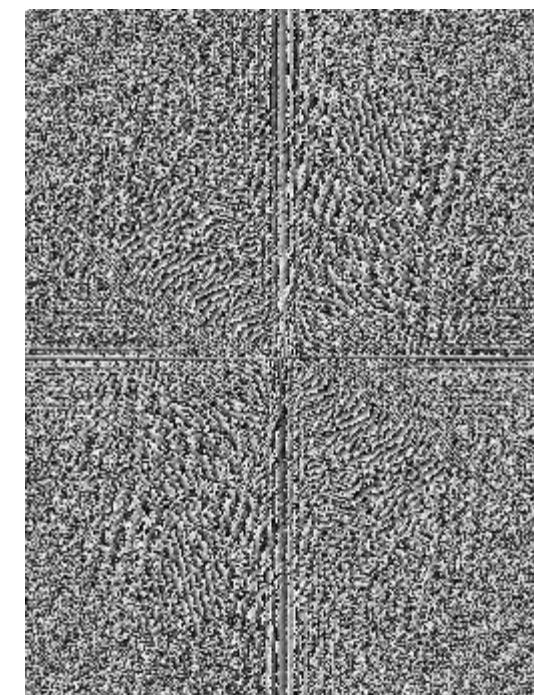
Fouriertransformation 2D



$$F(u, v) = \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \quad f(x, y) = \int \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du dv$$



Amplituden Spektrum



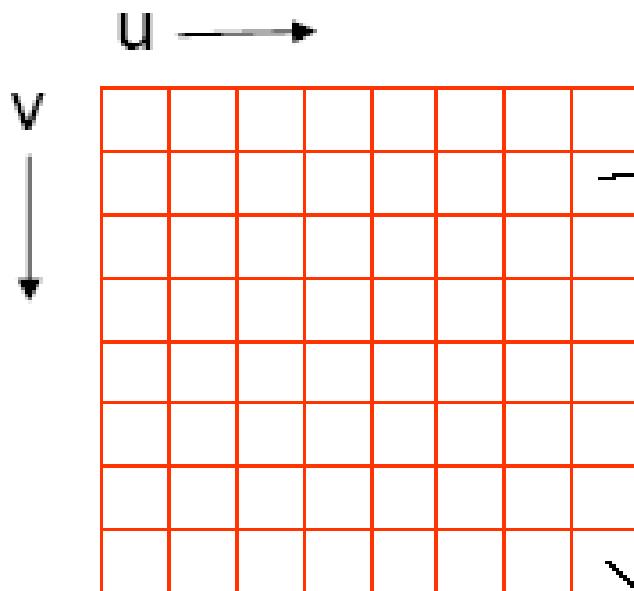
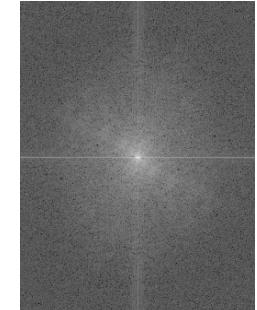
Phasen Spektrum

Amplitudenspektrum im Frequenzraum



Amplitudenspektrum:

Visualisierung im Frequenzraum der Amplituden (=Beiträge) der periodischen Funktionen verschiedener Frequenzen (u, v)



Frequenzraum

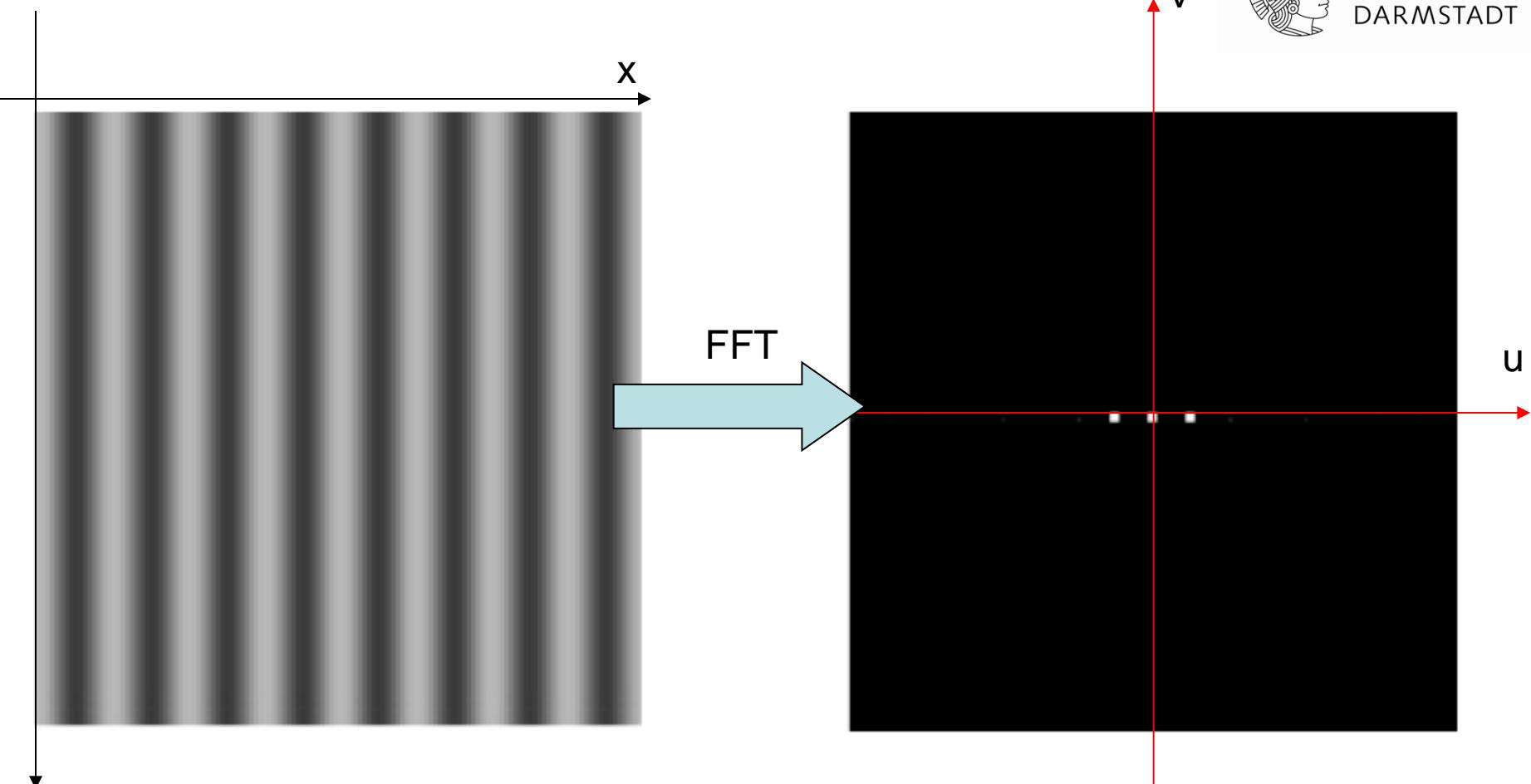
$F(u, v)$

$F(u, v) =$
Pixelwert im Frequenzraum =
Amplitude der periodischen
Funktion, die die Frequenzen
 u (in x-Richtung) und
 v (in y-Richtung) besitzt

Folien 57-65: S. Bonnet, TU Freiberg

23.11.2018 | VC - Bilder - A. Kuijper |

Weiterlesen:
<http://de.wikipedia.org/wiki/Frequenzspektrum>

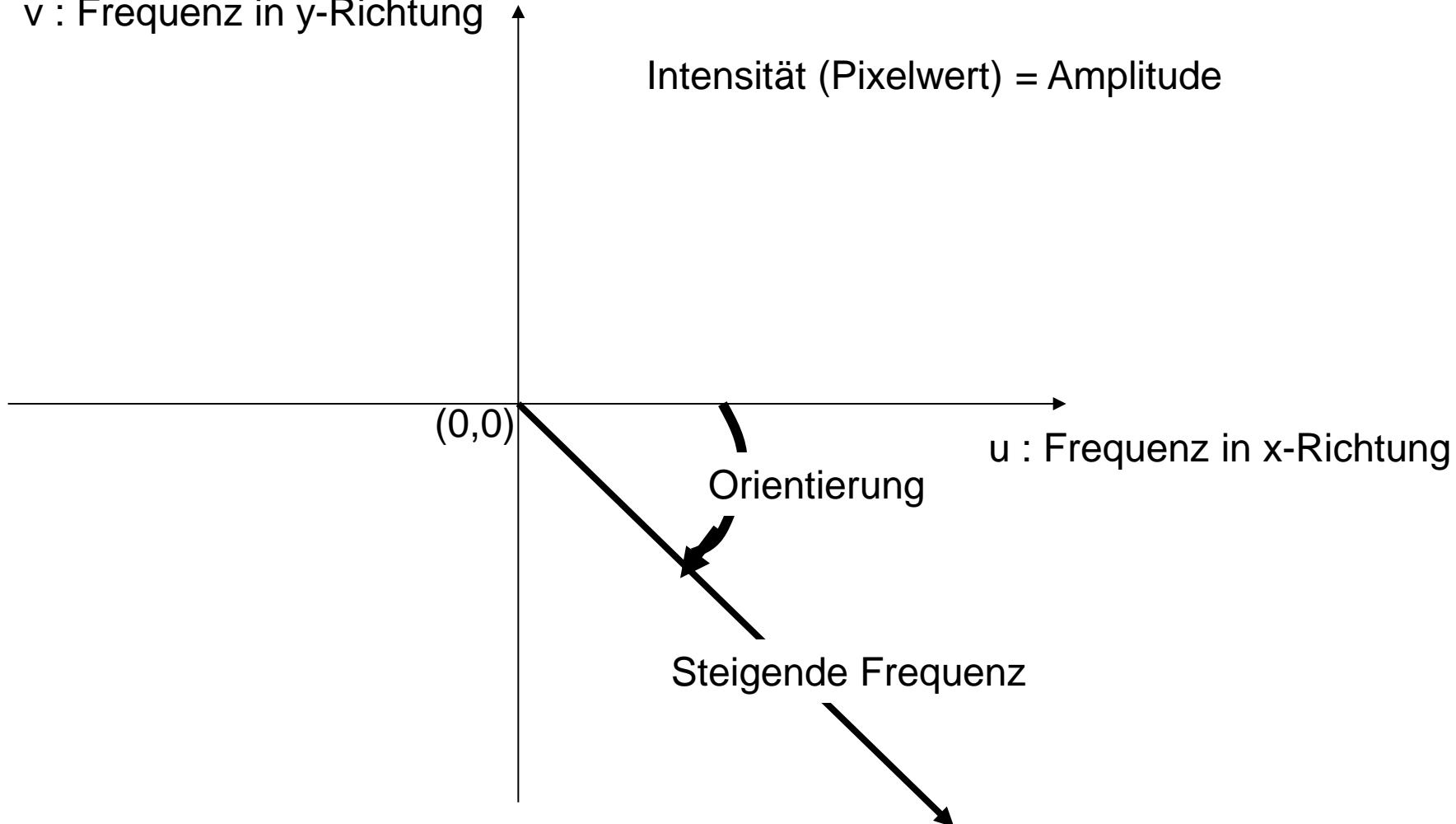


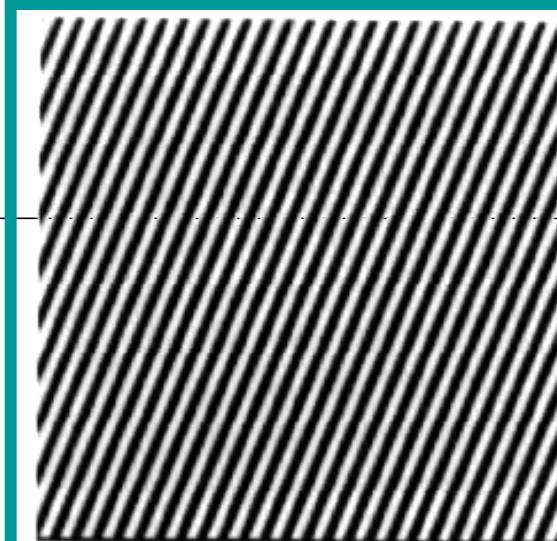
$|F(u,v)|$ bzw.
Amplitudenspektrum im
Frequenzraum

Amplitudespektrum im Frequenzraum



v : Frequenz in y-Richtung



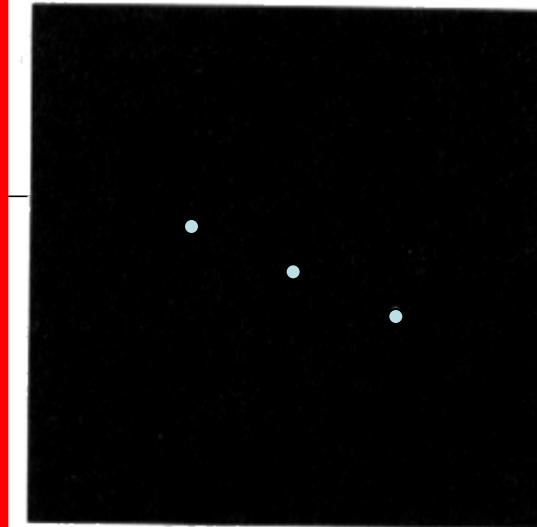


High Frequency Noise
Ortsraum (x,y)



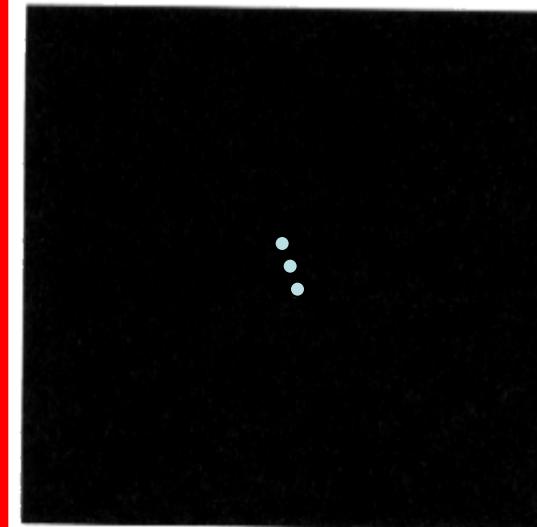
Low Frequency Noise

FFT



Fourier Transform
Frequenzraum (u,v)

FFT



Fourier Transform

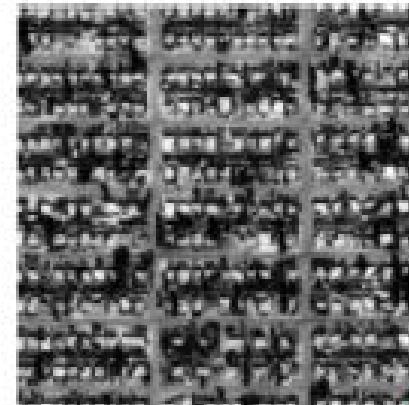
Bilder und dazugehörige Amplitudenspektren



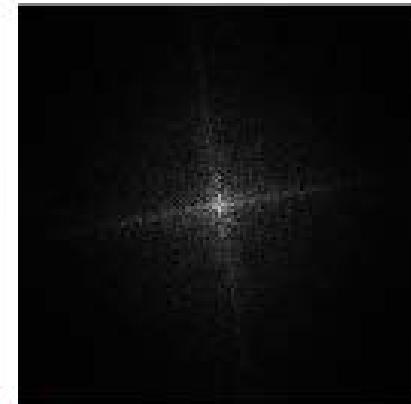
TECHNISCHE
UNIVERSITÄT
DARMSTADT



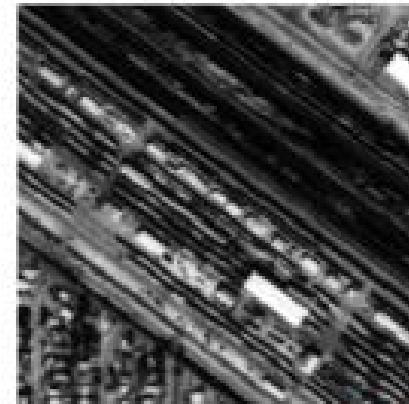
desert



streets



fields



railroad

Fourier-Filterung



- Durch eine Multiplikation jeder Frequenz-Komponenten $F(u,v)$ eines Bildes anhand einer bestimmten Gewichtungsfunktion (Filter) kann man bestimmte Frequenz-Komponenten erniedrigen und andere erhöhen (Erhöhung der Amplitude)
- Die zugehörigen Veränderungen sind im Ortsraum durch eine Rück-Transformation (FFT^{-1}) sichtbar
- Diese selektive Beseitigung von Frequenz-Komponenten heißt **Fourier-Filterung**

Filterungsart



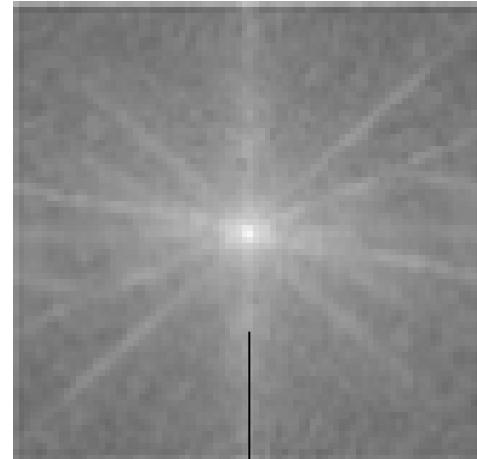
Filter werden eingesetzt, um z.B. den Einfluss von Datenfehlern oder Störsignalen zu verringern, hochfrequente von niederfrequenten Komponenten des Signals zu trennen, oder um bestimmte Frequenzbereiche in Signalen hervorzuheben.

Filter	Funktion $H(\omega)$	Eigenschaften
Hochpaßfilter	$H(\omega) = \begin{cases} 0 & \text{für } \omega < \omega_1 \\ 1 & \text{für } \omega \geq \omega_1 \end{cases}$	Abschneiden der tiefen Frequenzen $ \omega < \omega_1$, also können nur hohe Frequenzen passieren \Rightarrow scharfe Übergänge werden deutlicher
Tiefpaßfilter	$H(\omega) = \begin{cases} 1 & \text{für } \omega < \omega_1 \\ 0 & \text{für } \omega \geq \omega_1 \end{cases}$	Abschneiden der hohen Frequenzen $ \omega > \omega_1$, also können nur tiefe Frequenzen passieren \Rightarrow Rauschen wird eliminiert, Bild generell etwas unschärfer (blur)
Bandpaßfilter	$H(\omega) = \begin{cases} 0 & \text{für } \omega \leq \omega_1 \\ 1 & \text{für } \omega_1 < \omega < \omega_2 \\ 0 & \text{für } \omega \geq \omega_2 \end{cases}$	nur Frequenzen aus dem Band $\omega_1 < \omega < \omega_2$ können passieren

Filterung im Frequenzraum



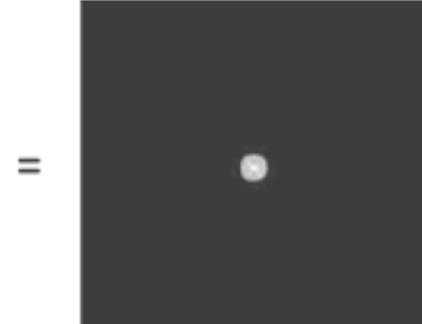
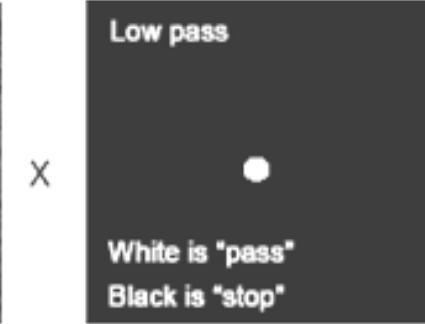
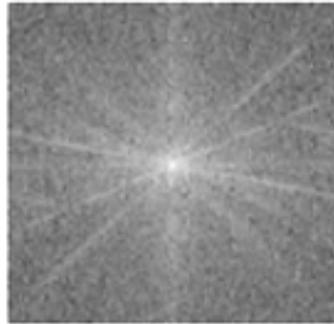
FFT
→



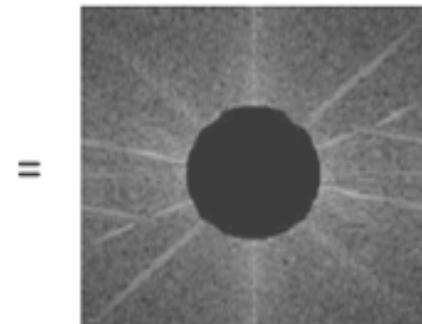
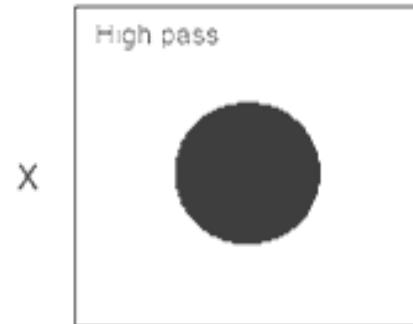
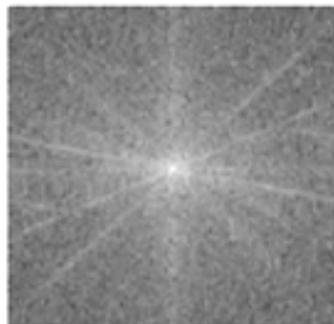
Gefiltertes Bildspektrum

Filter

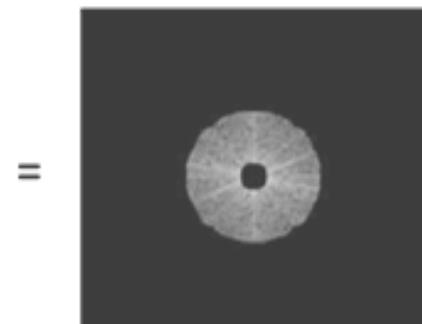
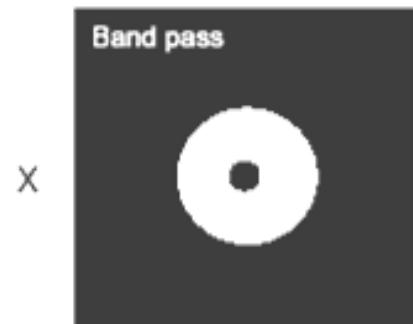
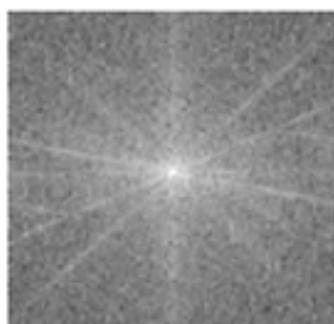
Beispiele



IFFT



IFFT



IFFT



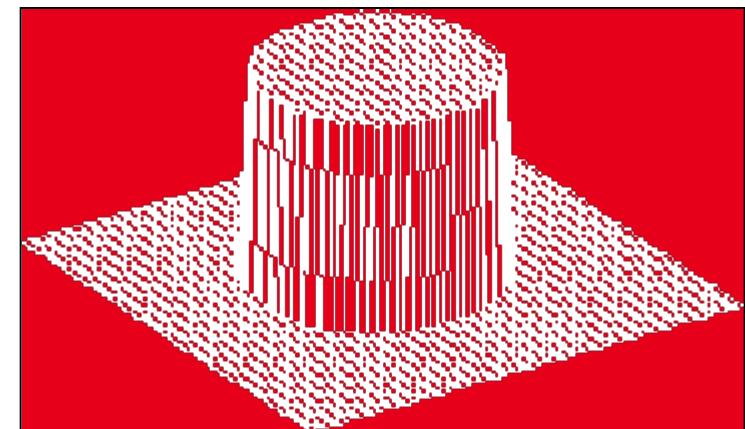
Idealer Tiefpass-Filter



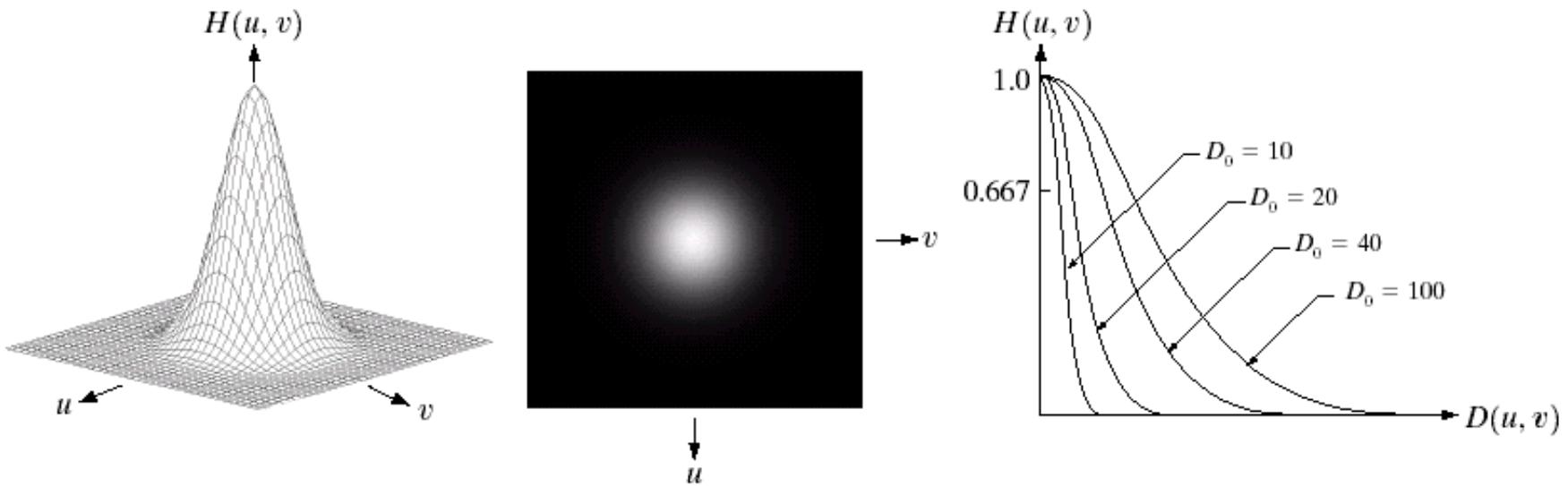
- Abschneiden von hohen Frequenzen jenseits einer Grenzfrequenz D_0
- Radial symmetrisch zum Ursprung
- Physikalisch nicht realisierbar (weil was war nochmal die Fouriertransformierte einer Rechteckfunktion?)

$$H(u, v) = \begin{cases} 1, & D \leq D_0 \\ 0, & D > D_0 \end{cases}$$

$$D(u, v) = \sqrt{u^2 + v^2}$$



Gauss'scher Tiefpass



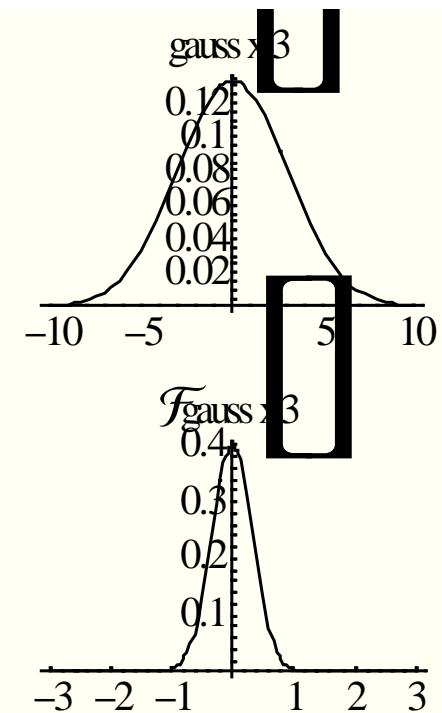
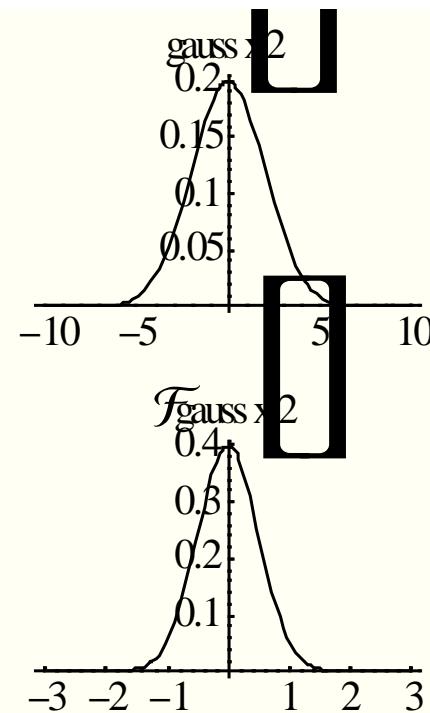
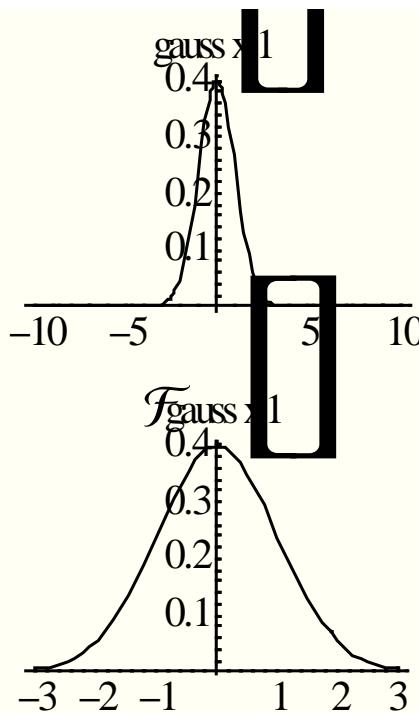
- Physikalisch realisierbar
(weil die Fouriertransformierte einer Gauß-Glocke ist?)

Gauss'scher Tiefpass



- ...eine Gauß-Glocke!

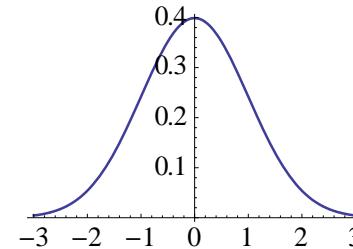
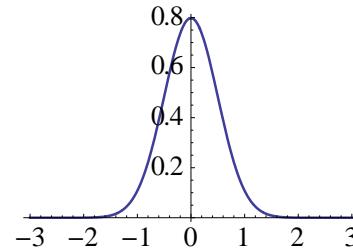
$$\frac{e^{-\frac{x^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \xrightarrow{F} \frac{e^{-\frac{1}{2}\sigma^2\omega^2}}{\sqrt{2\pi}}$$



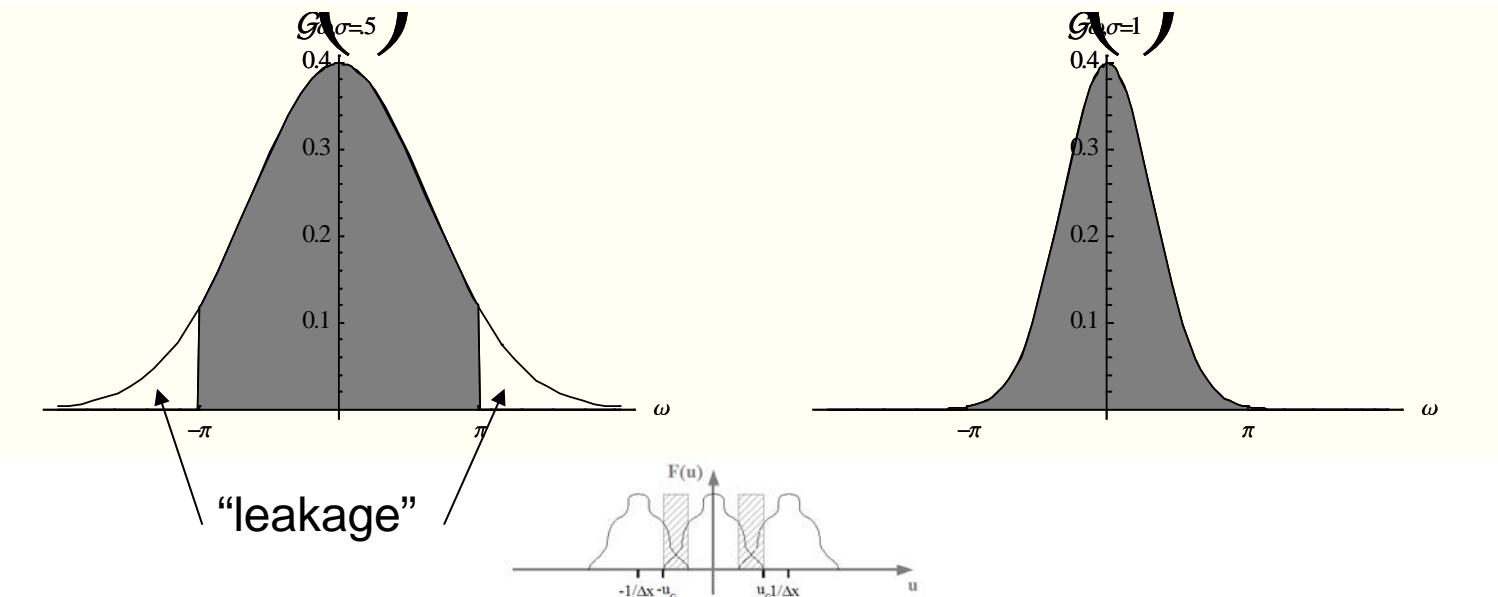
Aliasing!



- Sollte nicht zu klein sein: Sub-Pixel-Sampling im Ortsraum



- In Fourieraum:

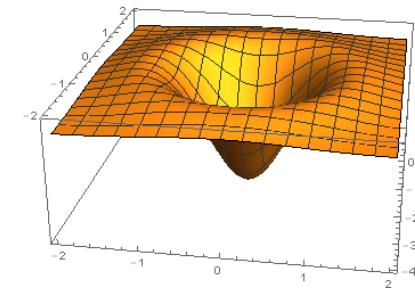


Idealer Hochpass-Filter

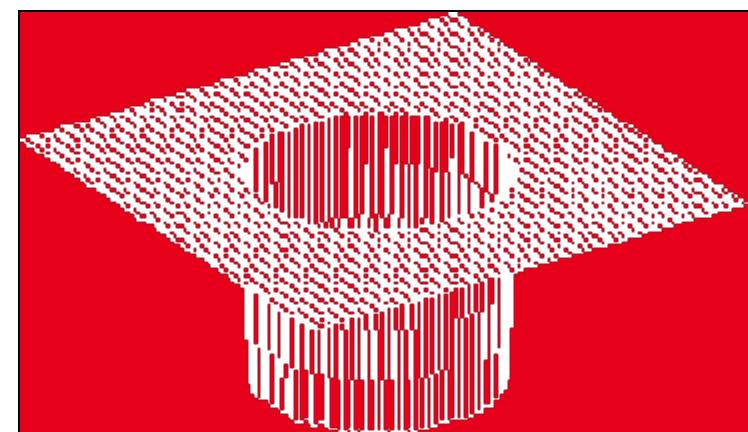


- Abschneiden von tiefen Frequenzen jenseits einer Grenzfrequenz D_0
- Radial symmetrisch zum Ursprung
- Physikalisch nicht realisierbar

Approximation: Mexican Hat!



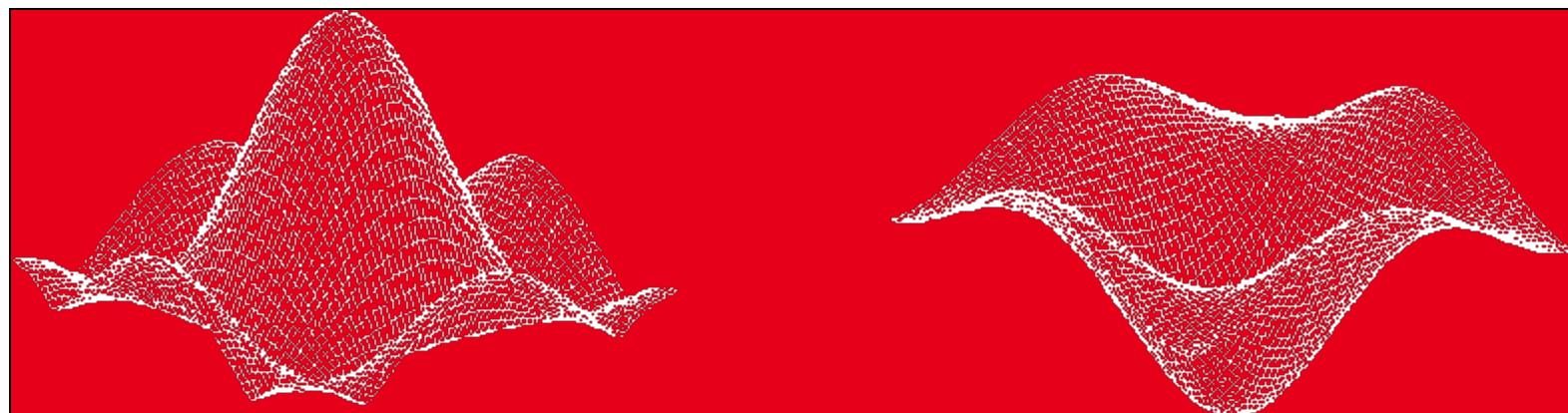
$$H(u, v) = \begin{cases} 0, & D \leq D_0 \\ 1, & D > D_0 \end{cases}$$



$$D(u, v) = \sqrt{u^2 + v^2}$$

Frequenzraum- gegenüber Ortsraum-Filtern

- + schnelle Berechnung (FFT)
- + einfache Handhabung (Filterdesign im Frequenzraum intuitiv)
- Approximation der Spezifikation aus dem Frequenzraum (keine unendlich breiten Filter im Ortraum möglich, Abschneiden führt zu Artefakten)



Ortsraum Tiefpass Filter

Ortsraum Hochpass Filter

Überblick



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Bildverbesserung
 - Histogramm
- Bildfilterung
 - Im Ortsraum
 - Im Frequenzraum
- **Bildkompression**

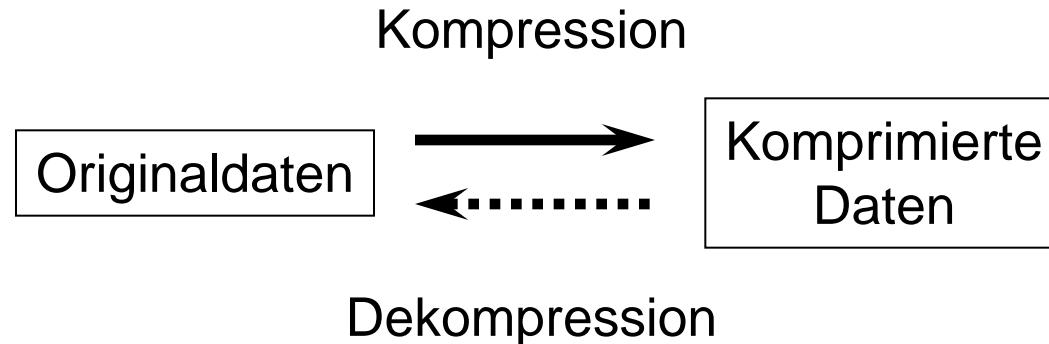


- Rasterung und Abtastung einer Intensitätsfunktion von Licht erzeugt eine „Unmenge“ von Daten (a.k.a digitales Photographieren)
 - ➔ Unpraktisch für
 - Datenspeicherung
 - Datenübertragung
- Wunsch nach kompakteren Darstellungen (ohne oder mit vertretbarem Qualitätsverlust) jenseits der herkömmlichen Repräsentation digitaler Bilder und Videos



- Bildkompression behandelt das Problem, die Menge der Daten zur Repräsentation einer gegebenen Menge an visueller Information zu reduzieren
 - ➔ Eliminierung von redundanten Daten
 - Kodierungen
 - Nachbarschaftsbeziehungen
 - räumlich
 - zeitlich
 - Psychovisuelle Eindrücke
 - Wahrnehmungsmodell des Menschen
 - Farbauflösung des menschlichen Auges

Modell



Klassifikation

- Verlustlose Kompression
- Verlustbehaftete Kompression



Variable-Length-Coding

- Huffman Code
- Arithmetischer Code

Bit-Plane Coding

- Bit-Plane Slicing / Run-Length Coding

Predictive Coding

Lempel-Ziv-Welch-Algorithmus (LZW)

- GIF, TIFF, ...
 - „Online“-Codebucherstellung
 - Kombination von Variable-Length- und Run-Length-Coding
-
- <http://www.data-compression.com/lossless.shtml>

Verlustbehaftete Kompression



- Komprimierte Kodierung der Bildinformationen, so dass
 - nicht alle Eigenschaften des Bildes berücksichtigt werden und
 - ggf. die exakte Rekonstruktion des Bildes nicht mehr möglich ist
- Erlaubt dem Anwender die Steuerung des Verhältnisses von Qualität zu Kompressionsgrad



- Verwendet häufig Modelle der menschlichen Wahrnehmung
 - zur Identifizierung von für den Betrachter irrelevanten Bildeigenschaften, die nicht kodiert werden müssen

Bildkompression

Harmonische Transformationen

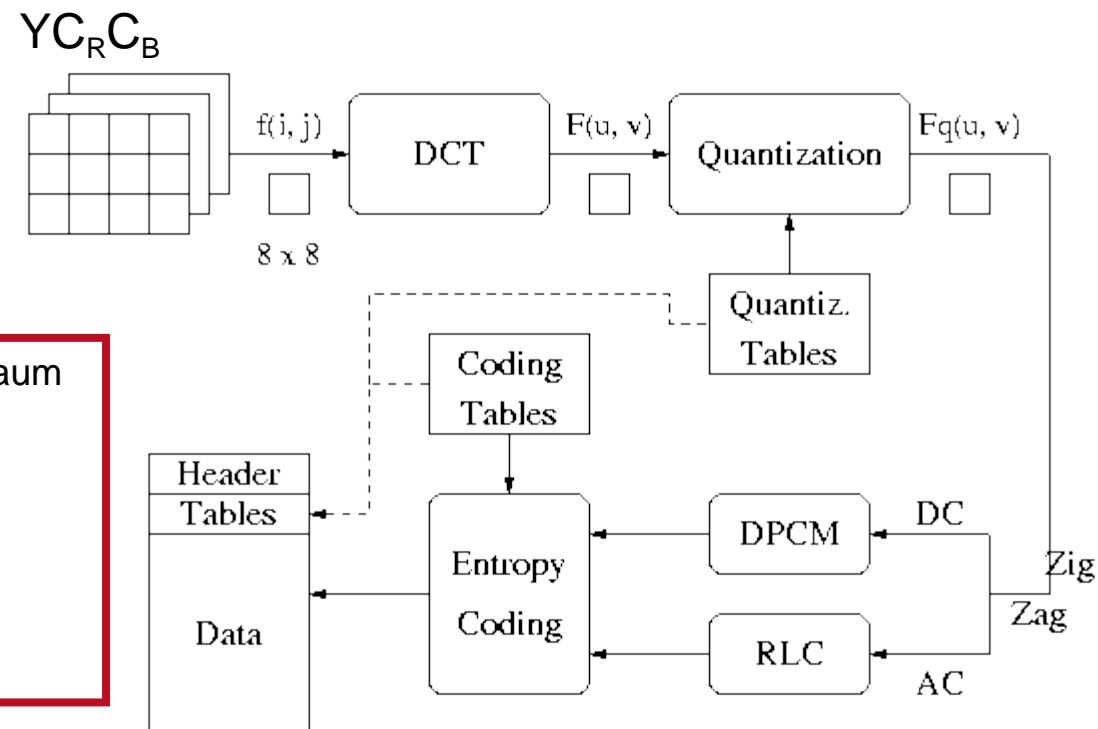


- Zerlegung der Daten in verschiedene „Frequenzanteile“
 - Fourier-Transformation
 - Wavelet-Transformation

- Typischer Vertreter:
JPEG



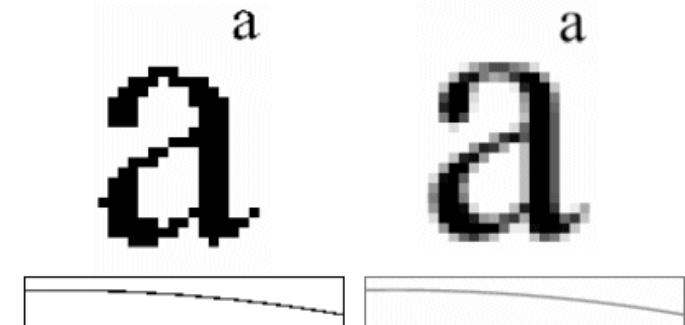
1. Umwandlung in den $YC_R C_B$ -Farbraum
2. Farb-Subsampling
3. Diskrete Kosinustransformation
4. Quantisierung
5. Kodierung der Koeffizienten



Bildkompression

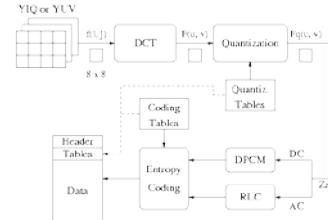
JPEG (Joint Photographic Experts Group)

- Familie von Algorithmen zur Kompression digitalisierter Standbilder in Echtfarbqualität
- Sammlung unterschiedlichster Verfahren 1993 unter der Bezeichnung ISO 10918 standardisiert.
- Verlustbehaftete und verlustfreie Kompression
- Verlustbehaftete JPEG-Prozesse
 - Für fotografische Aufnahmen mit fließenden Farbübergängen optimiert
 - Für andere Arten von Bildern weniger geeignet (z.B. Bilddaten mit harten Kontrasten, Liniengrafiken oder Texte)
- Für 24-Bit-RGB-Farbbilder
 - Kompressionsraten von 12 bis 15 für visuell verlustfreie Bilder und bis zu 35 für noch gute Bilder

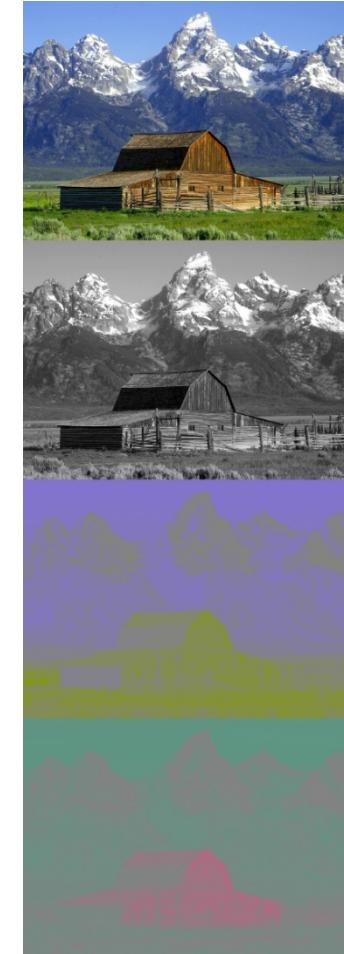


JPEG – „Baseline Codec“

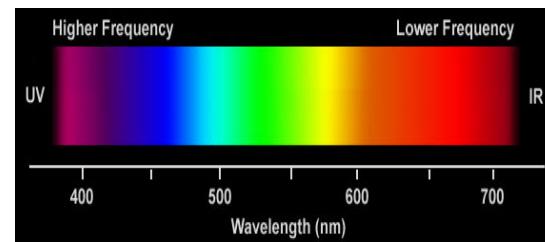
1. Umwandlung in den $YC_R C_B$ -Farbraum



- Kodierung der Farben als
 - Y Helligkeitswert
 - C_R Abweichung vom Grau in Richtung rot
 - C_B Abweichung vom Grau in Richtung blau

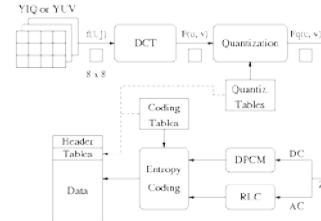


$$\begin{bmatrix} Y' \\ Cb \\ Cr \end{bmatrix} \approx \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,168736 & -0,331264 & 0,5 \\ 0,5 & -0,418688 & -0,081312 \end{bmatrix} \cdot \begin{bmatrix} R'_d \\ G'_d \\ B'_d \end{bmatrix}$$



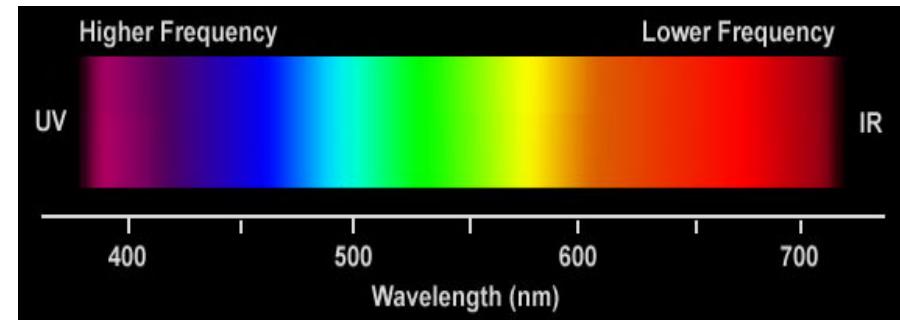
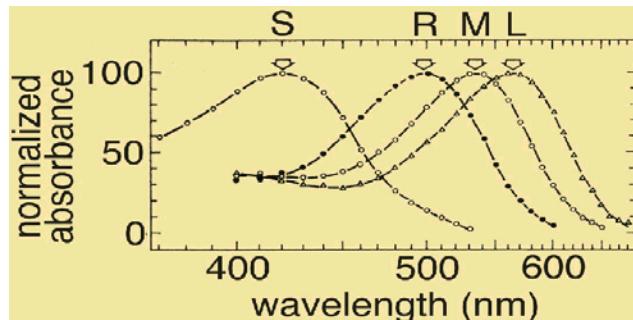
JPEG – „Baseline Codec“

2. Farb-Subsampling



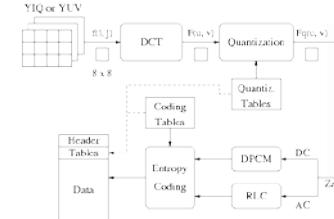
TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Verlustbehaftete Komprimierung der Farbrepräsentation
- Grundlage: Höhere Genauigkeit der menschlichen Ortsauflösung im Helligkeitsbereich (Grünbereich) als im Farbbereich
- für ein kleines Gebiet werden die Farbdifferenzwerte C_R und C_B gemittelt und für das gesamte Gebiet zusammengefasst angegeben
 - Übliche Größe: 2×2 Pixel
 - $4Y, 1 C_B, 1 C_R$ Wert pro 2×2 Pixel ($Y:Cb:Cr = 4:1:1$)



JPEG – „Baseline Codec“

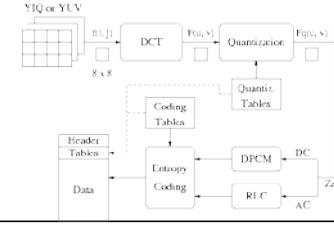
3. Diskrete Kosinustransformation



- Umwandlung der Bildinformationen in den Frequenzbereich
 - Rasterung jeder Komponente (Y , C_B , C_R) in 8x8 Bildblöcke
 - Bildblöcke werden DCT unterzogen und als Vektoren interpretiert
- Ziel:
 - Überführung der Bildinformation in eine Darstellung, die besser für die folgenden Schritte geeignet ist
 - Filterung (durch den folgenden Quantisierungsschritt) von hohen Frequenzen in den Farbanteilen, die vom menschlichen Auge nicht/kaum wahrgenommen werden können
- Fourierreihe mit nur Realteil: Kosinus
 - Einfacher zur Berechnung
 - Effektiv für Multi-Media-Kompression
 - Häufig benutzt

JPEG – „Baseline Codec“

3. Diskrete Kosinustransformation

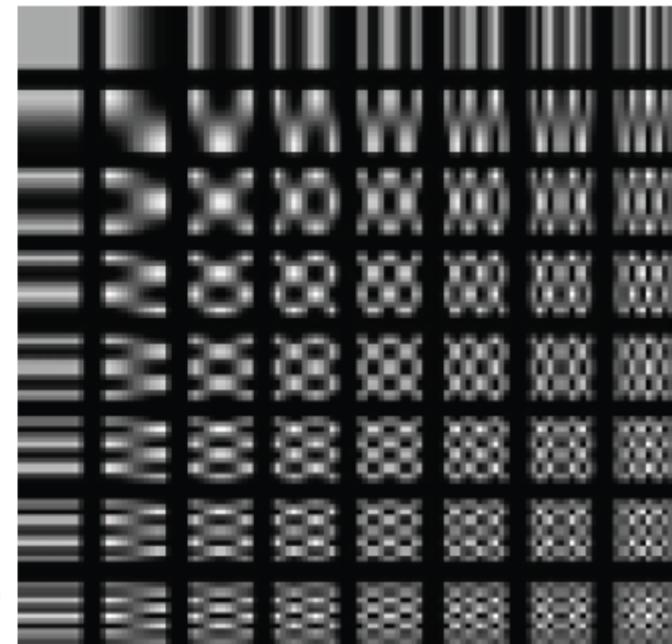


- Basisvektoren

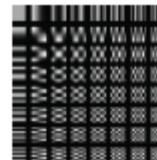
$$C_{u,v} = \frac{1}{4} a_u a_v \sum_{i=0}^7 \sum_{j=0}^7 \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16}$$

$$a_u, a_v = \begin{cases} 1/\sqrt{2} & u, v = 0 \\ 1 & sonst \end{cases}$$

$$\begin{matrix} u = 0 & & u = 7 \\ v = 0 & & \\ & & \end{matrix}$$



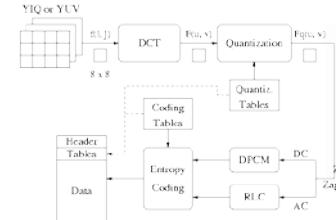
- Basisvektoren als 8x8 Grauwertbilder



$$v = 7$$

JPEG – „Baseline Codec“

3. Diskrete Kosinustransformation

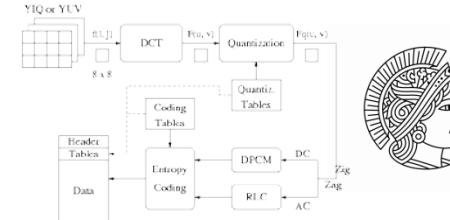


- Der Vorteil der DCT:
 - Wenn sich benachbarte Bildpunkte kaum unterscheiden...
 - d.h. das Bild keine scharfen Kanten hat (Kanten = hohe Frequenzen)
 - ...dann sind in der Koeffizientendarstellung...
 - ...der DC-Koeffizient $F(0,0)$ und einige niederfrequente AC-Koeffizienten $F(u,v)$ ungleich 0
 - ...alle anderen $F(u,v)$ häufig fast oder gleich 0
 - Geeignete Repräsentation für weitere Komprimierungsschritte

JPEG – „Baseline Codec“

4. Quantisierung

- Division der DCT-Koeffizienten durch die Quantisierungsmatrix
 - Betonung homogener Regionen
 - Isotrope Abstand von DC
- Beseitigung von Informationsanteilen, die das menschliche Auge nicht oder nur schlecht wahrnimmt
- $F^Q = \text{Floor}(\text{Round}(F/Q))$



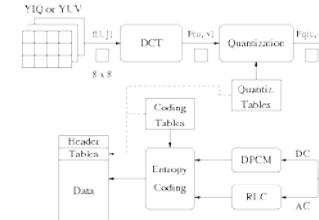
$$Q = \begin{bmatrix} 10 & 15 & 25 & 37 & 51 & 66 & 82 & 100 \\ 15 & 19 & 28 & 39 & 52 & 67 & 83 & 101 \\ 25 & 28 & 35 & 45 & 58 & 72 & 88 & 105 \\ 37 & 39 & 45 & 54 & 66 & 79 & 94 & 111 \\ 51 & 52 & 58 & 66 & 76 & 89 & 103 & 119 \\ 66 & 67 & 72 & 79 & 89 & 101 & 114 & 130 \\ 82 & 83 & 88 & 94 & 103 & 114 & 127 & 142 \\ 100 & 101 & 105 & 111 & 119 & 130 & 142 & 156 \end{bmatrix}$$

$$F = \begin{bmatrix} 782,91 & 44,93 & 172,52 & -35,28 & -20,58 & 35,93 & 2,88 & -3,85 \\ -122,35 & -75,46 & -7,52 & 55,00 & 30,72 & -17,73 & 8,29 & 1,97 \\ -2,99 & -32,77 & -57,18 & -30,07 & 1,76 & 17,63 & 12,23 & -13,57 \\ -7,98 & 0,66 & 2,41 & -21,28 & -31,07 & -17,20 & -9,68 & 16,94 \\ 3,87 & 7,07 & 0,56 & 5,13 & -2,47 & -15,09 & -17,70 & -3,76 \\ -3,77 & 0,80 & -1,46 & -3,50 & 1,48 & 4,13 & -6,32 & -18,47 \\ 1,78 & 3,28 & 4,63 & 3,27 & 2,39 & -2,31 & 5,21 & 11,77 \\ -1,75 & 0,43 & -2,72 & -3,05 & 3,95 & -1,83 & 1,98 & 3,87 \end{bmatrix}$$

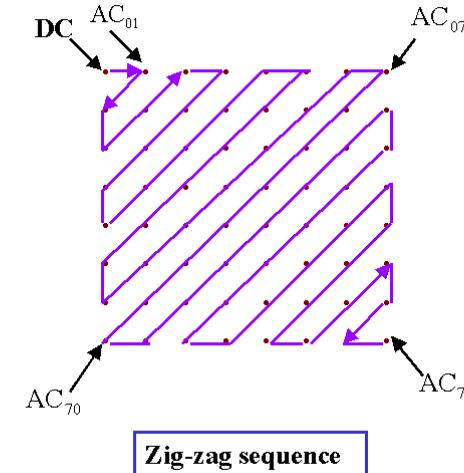
$$F^Q = \begin{bmatrix} 78 & 3 & 7 & -1 & 0 & 1 & 0 & 0 \\ -8 & -4 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

JPEG – „Baseline Codec“

5. Kodierung der Koeffizienten

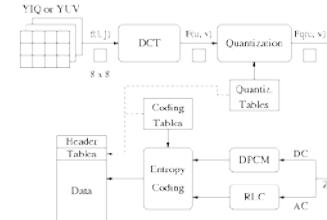


- Aus den 8x8 Blöcken wird ein sequentieller (eindimensionaler) Bitstrom erzeugt
- DC Koeffizienten werden als Differenzen zum vorhergehenden DC Koeffizienten kodiert
 - Durch die Kohärenz der DC-Koeffizienten ergeben sich auch hier wieder kleinere Werte
- Die 63 AC-Koeffizienten werden anhand einer Zick-Zack-Kurve kodiert
 - Da die hohen Frequenzanteile oft sehr klein bzw. Null sind, entsteht eine für die weitere Kompression der Bilddaten günstige Reihenfolge.



JPEG – „Baseline Codec“

5. Kodierung der Koeffizienten



- Die bisher beschriebenen Verfahren beinhalten noch keine explizite Kompression, sondern stellen nur eine, bei starker Quantisierung der DCT-Koeffizienten, recht grobe Transformation der Bilddaten dar
- Typische Kompressionstechniken
 - Komprimierung durch Huffman-Algorithmus
 - Arithmetisches Codieren
- Arithmetisches Codieren komprimiert zwar besser als das Huffman-Verfahren, ist jedoch mit verschiedenen Patenten belegt

JPEG Beispiele



TECHNISCHE
UNIVERSITÄT
DARMSTADT

■ Originalbild

2048x1536 Pixel
(3MPixel)
3x8 Bit
9.437.184 Bytes



JPEG

Qualitätsfaktor 75%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



25.398 Bytes
Kompressionsrate 1:6

JPEG

Qualitätsfaktor 50%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



15.980 Bytes
Kompressionsrate 1:10

JPEG

Qualitätsfaktor 25%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



11.334 Bytes
Kompressionsrate 1:15

JPEG

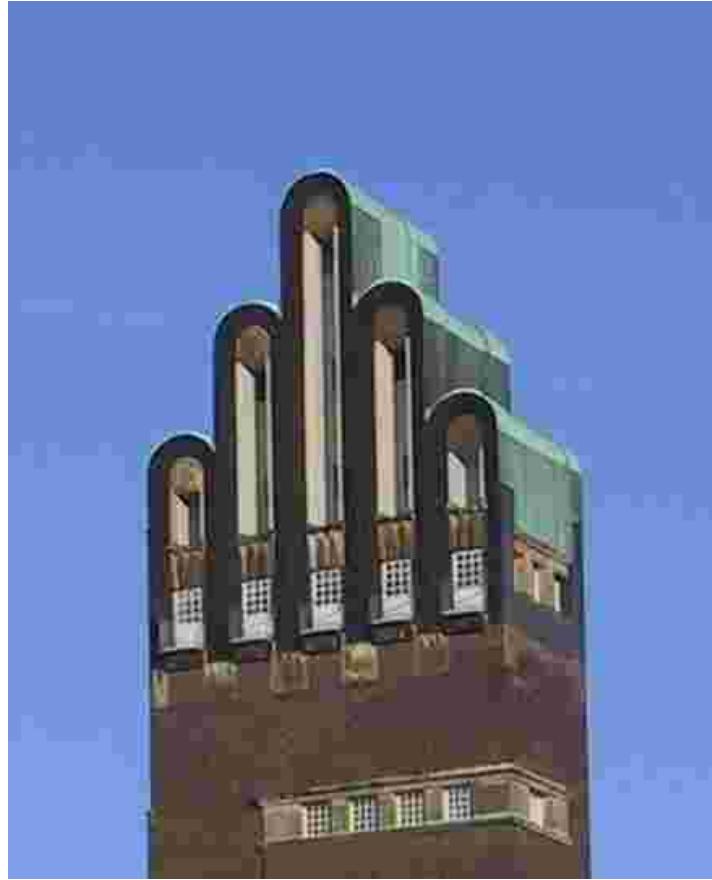
Qualitätsfaktor 10%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



7.742 Bytes
Kompressionsrate 1:22

JPEG

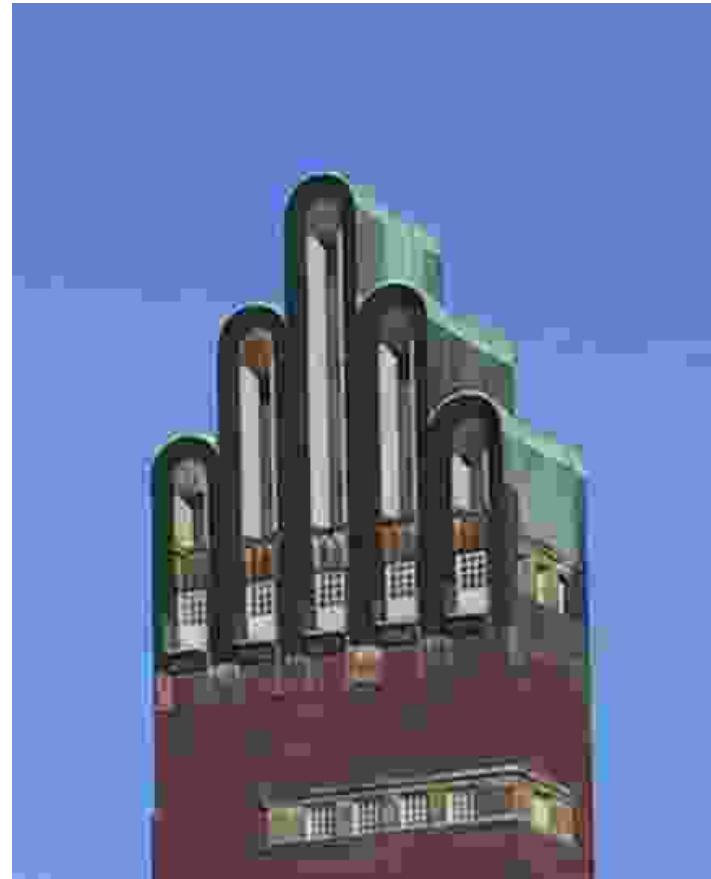
Qualitätsfaktor 5%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



6.194 Bytes
Kompressionsrate 1:27

JPEG

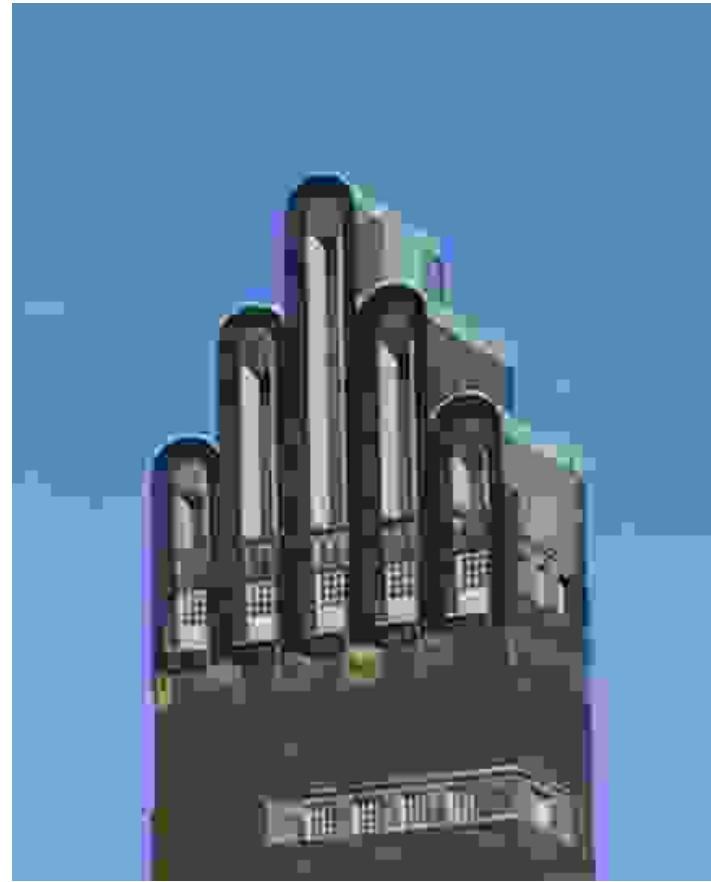
Qualitätsfaktor 3%



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Original - Ausschnitt (441x541 Pixel)
167.015 Bytes (715.743 Bytes)



5.547 Bytes
Kompressionsrate 1:30

JPEG

Zusammenfassung



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Komplexes Verfahren zur Bildkompression, mit dem Kompressionsraten bis zu 1:20 bis 1:35 zu erreichen sind
- In den Hauptmodi und Hauptanwendungsbereichen verlustbehaftet
- Basiert auf einer diskreten Kosinustransformation mit intelligenter Quantisierung und Komprimierung der Koeffizienten

Komprimieren ist überall



- Audio
 - Nicht komprimiert: AIFF, WAV, ...
 - Verlustlos: MPEG-4-ALC, Apple Lossless (ALAC), WMA Lossless, ...
 - Mit Verlust: MP3, Ogg Vorbis, MPEG-Audio, AAC (iTunes), WMA, ...
- Bilder
 - Nicht komprimiert: BMP, RAW,
 - Verlustlos: TIFF, GIF, PNG, (VI JPEG/-2000)...
 - Mit Verlust: JPEG (Disc. Cosine Transf), JPEG2000 (Wavelets), ...
- Video
 - Nicht komprimiert: nicht wirklich optimal!
 - Verlustlos: gibt es, aber...
 - Mit Verlust: H.264 (DivX,QuickTime) MPEG-4 part 2 (Xvid,DivX), WMV, ...



Vielen Dank für die Aufmerksamkeit