

# Bildverarbeitung und Computer Vision

Prof. Dr. Wolfgang Konen

---

## Filterung von Bildern (2D-Filter)



Aktivierung

## Was, denken Sie, ist ein Filter in der BV? Welche Filter kennen Sie?

---

- neuer Pixelwert bilden aus Verknüpfung mit Nachbarwerten
- Tiefpass-, Hochpass-, Bandpass-Filter
- Glättungsfiler (Gauss, Box)

# Inhalt

---

- **Filter im Ortsraum: Die Faltung**
- Tiefpass-Filter
- Filter im Frequenzraum: Fouriertransformation
  - 1D-Fouriertransformation
  - 2D-Fouriertransformation
- Anwendungen der Fouriertransformation (kurz)
  - Bandstop
  - Unsharp Masking

# Filterung eines Bildes

## Implementierungsalternativen (lineare Filter)

---

Bei der Filterung eines Bildes werden die Bildpunkte (Pixel) in Abhängigkeit von ihrer Nachbarschaft manipuliert.

### □ Ortsraum

- Direkte Manipulation der Pixelwerte im Bildbereich

### □ Frequenzraum

- Bildtransformation (z.B. Diskrete Fourier Transformation / DFT)
- Manipulation der Transformierten
- Rücktransformation in den Bildbereich

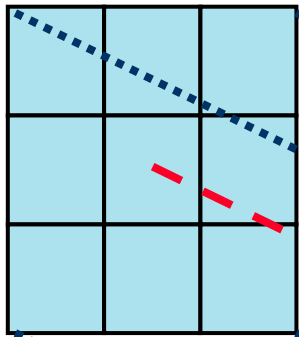
Viele Filteroperationen sind im Frequenzraum weniger aufwendig als im Ortsraum. Jedoch ist bei Filteroperationen, die nur eine kleine Bildnachbarschaft einbeziehen, der fixe Aufwand für die Bildtransformationen i.d.R. größer als der Aufwand für die Berechnung der Filterung selbst.

# Prinzip der diskreten Faltung (Konvolution)

## (Örtliche Faltung / Spatial Convolution)

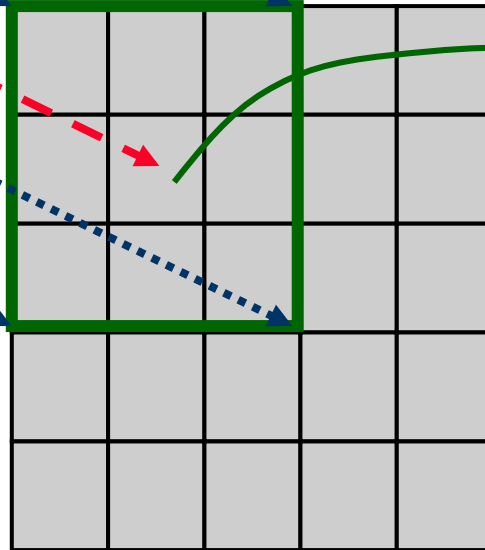
Filtermaske

(3 × 3)

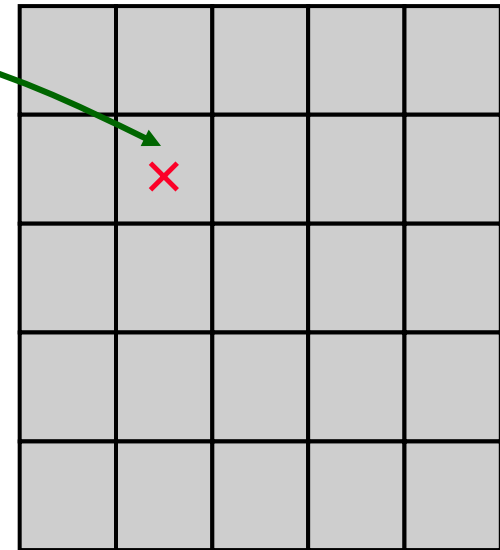


Bei der Faltung berechnet sich jeder Pixel des Ausgangsbildes als gewichtete Summe der Pixel einer Bildnachbarschaft.

Die Gewichte sind die Koeffizienten der Filtermaske (des Filterkerns).



Eingangsbild



Ausgangsbild

$W_{-1,-1}$	$W_{-1,0}$	$W_{-1,1}$
$W_{0,-1}$	$W_{0,0}$	$W_{0,1}$
$W_{1,-1}$	$W_{1,0}$	$W_{1,1}$

← Filterkoeffizienten (Gewichte)

# Berechnung der diskreten Faltung

Eingangsbild  $I_{i,j}$

$I_{0,0}$	$I_{0,1}$	$I_{0,2}$	...	$I_{0,M}$
$I_{1,0}$	$I_{1,1}$	...		
$I_{2,0}$	...			
:				
$I_{N,0}$	...			$I_{N,M}$

Filtermaske  $M_{p,q}$

$M_{-1,-1}$	$M_{-1,0}$	$M_{-1,1}$
$M_{0,-1}$	$M_{0,0}$	$M_{0,1}$
$M_{1,-1}$	$M_{1,0}$	$M_{1,1}$

Maskengröße  $W = 3$

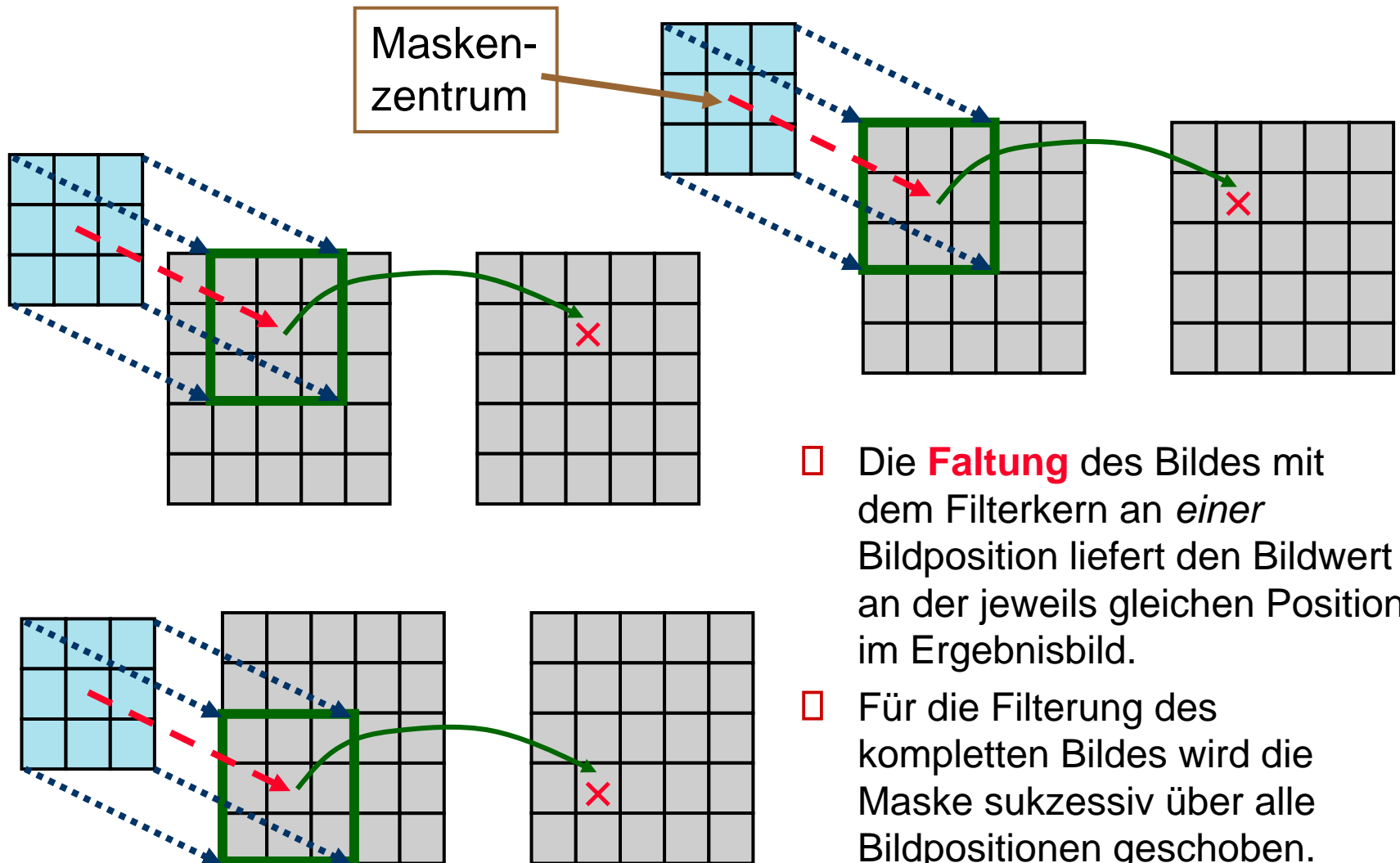
Ausgangsbild  $O_{i,j}$

$O_{0,0}$	$O_{0,1}$	$O_{0,2}$	...	$O_{0,M}$
$O_{1,0}$	$O_{1,1}$	...		
$O_{2,0}$	...			
:				
$O_{N,0}$	...			$O_{N,M}$

$$O[i, j] = \sum_{p=-W/2}^{W/2} \sum_{q=-W/2}^{W/2} I[i-p, j-q] \cdot M[p, q]$$

$$O = I * M \quad (\text{Faltung})$$

# Prinzip des Faltungsprozesses



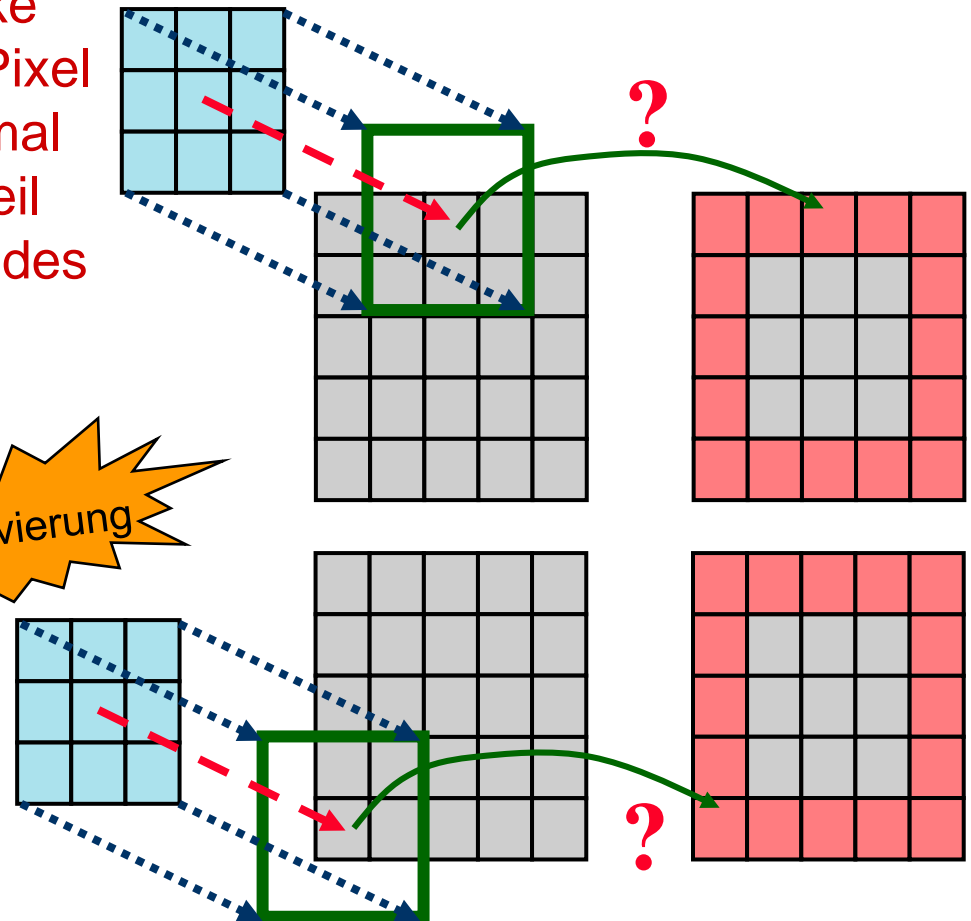
# Konvolution mit Filtermasken

## Randbetrachtungen

Je nach Größe der Filtermaske existiert ein Bildrand (min. 1 Pixel breit), dessen Pixel nicht normal berechnet werden können, weil ein Teil der Maske außerhalb des Eingangsbildes liegt.

Was sind gängige Methoden der Randbehandlung?

- am Rand kein Filter
- Zero- oder Grauwert-Padding
- Last-Value
- Periodische Fortsetzung
- Symmetrische Fortsetzung
- Interpolative Fortsetzung
- Konstruktion spezieller Filtermasken





# Randbehandlung

©Burger2005



(a)



(b)



(c)

## Zero- oder Grauwert-Padding

(Zero-Padding ist ImageJ's Default, wenn man getPixel an "out-of-bounds"-Positionen macht)

## Last Value

(meist einfach und sinnvoll für Filter im Ortsraum)

## Periodische Fortsetzung

(sieht unsinnig aus, aber z.B. die Fouriertransformation arbeitet so)

# Inhalt

---

- Filter im Ortsraum: Die Faltung
- **Tiefpass-Filter**
- Filter im Frequenzraum: Fouriertransformation
  - 1D-Fouriertransformation
  - 2D-Fouriertransformation
- Anwendungen der Fouriertransformation (kurz)
  - Bandstop
  - Unsharp Masking

# Diskrete Tiefpass-Filter (Filtermasken)

- Glättungseffekt (Smoothing, Blur)
- Koeffizienten sind ausnahmslos positiv
- Koeffizienten sind normalisiert  
(Summe aller Koeffizienten ergibt 1)
- Beispiel: **Mittelwertfilter**

$$\frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$3 \times 3$

$$\frac{1}{25} \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

$5 \times 5$

# Lineare und Separierbare Filter

- Ein Filter ist **linear**, wenn er Pixelwerte linear verknüpft.  
Die **Faltung** ist ein linearer Filter
- Ein Filter heißt **separierbar**, wenn er in mehrere Faltungen aufgespalten werden kann:  $O = I * M \Leftrightarrow O = I * M_1 * M_2$
- Der Mittelwertfilter ist x/y-separierbar, denn

$$M_1 = \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{und} \quad M_2 = \frac{1}{3} [1 \quad 1 \quad 1]$$

liefert

$$M = M_1 * M_2 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

# Mittelwertfilter (Beispiel)

---



Original

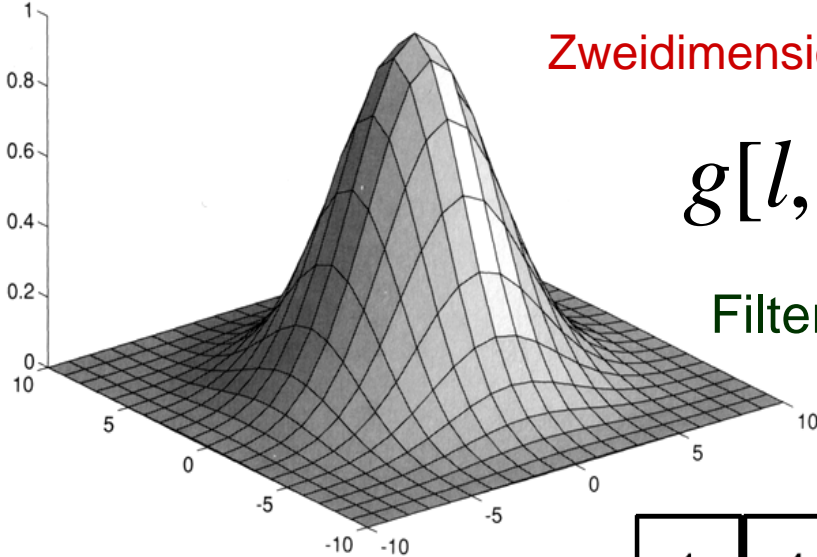


Filterungsergebnis  
mit  $3 \times 3$  Maske



Filterungsergebnis  
mit  $5 \times 5$  Maske

# Gaußscher Glättungsfilter (Gaussian smoothing filter)



Zweidimensionale diskrete Gaußfunktion mit Mittelwert 1:

$$g[l, k] = e^{-\frac{(l^2 + k^2)}{2\sigma^2}}$$

Filtermaske:

$$M[p, q] = \frac{g[p, q]}{\sum_p \sum_q g[p, q]}$$

1	2	1
2	4	2
1	2	1

1  
—  
256

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

$\sigma$  (sigma) =  
"Filterbreite"

# Gaußscher Glättungsfilter

## Beispiel

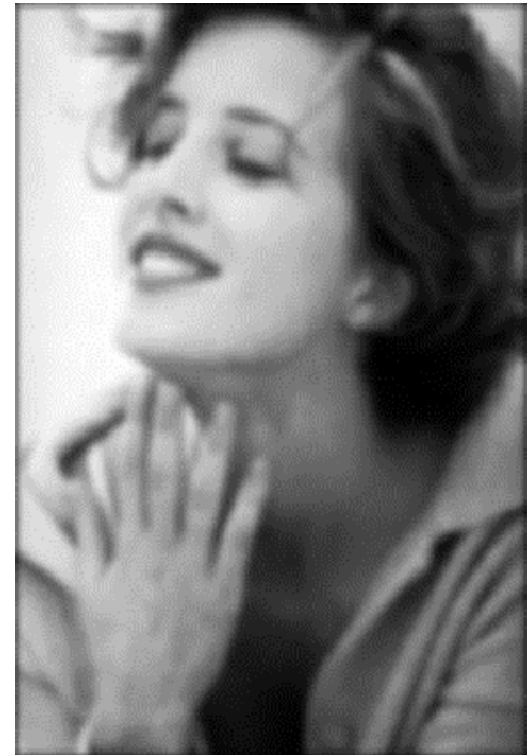
---



Original



$5 \times 5, \sigma = 1$



$9 \times 9, \sigma = 2$



## Separierbarer Gauss-Filter

- Auch der Gauss-Filter ist x/y-separierbar. Überlegen Sie, wie die beiden Filter  $M_1$  und  $M_2$  aussehen müssen, um

$$\frac{1}{16} \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

zu erzeugen

- Wie sieht's für den 5x5-Gauss-Filter aus?



# Inhalt

---

- Filter im Ortsraum: Die Faltung
- Tiefpass-Filter
- **Filter im Frequenzraum: Fouriertransformation**
  - **1D-Fouriertransformation**
  - **2D-Fouriertransformation**
- Anwendungen der Fouriertransformation (kurz)
  - Bandstop
  - Unsharp Masking

# Erzeugung einer Frequenzraumdarstellung (1)

## Diskrete 1D Fourier-Transformation

Die eindimensionale diskrete Fourier-Transformation (FT) eines Signals  $G(x)$  ist definiert als:

$$F(m) = \sum_{x=0}^{N-1} G(x) e^{-i \frac{2\pi m}{N} x}$$

Die Fourier-Transformation ist invertierbar (umkehrbar).  
Das Originalbild kann durch Rücktransformation in den Ortsraum wieder hergestellt werden.

$N$  ist die Signallänge und  $m$  ist die **Wellenzahl** (Wieviel volle Wellen passen in Signal hinein?)

$e$  ist die Basis des natürlichen Logarithmus (ca. 2,71828) und  $i = \sqrt{-1}$  ist die imaginäre Einheit für eine komplexe Zahl.

# Exkurs: Komplexe Zahlen

Allgemeine Form einer komplexen Zahl:

1. Komponentendarstellung  
(Realteil und Imaginärteil)
2. Darstellung durch Betrag und Winkel  
in der komplexen Ebene

$$a + bi = R \cdot e^{i\alpha}$$

Betrag:

$$R = \sqrt{(a^2 + b^2)}$$

Euler-Identität:

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

$$\Rightarrow e^{i0} = 1$$

$$e^{-i \frac{2\pi m}{N} x} = \cos\left(\frac{2\pi m}{N} x\right) - i \sin\left(\frac{2\pi m}{N} x\right)$$

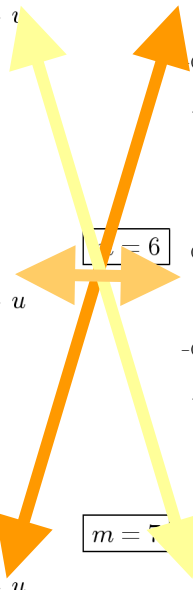
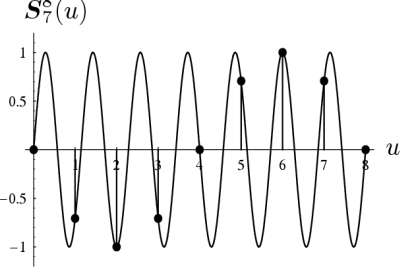
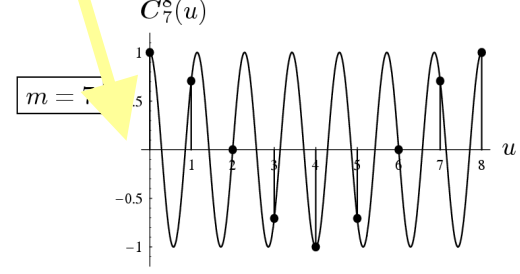
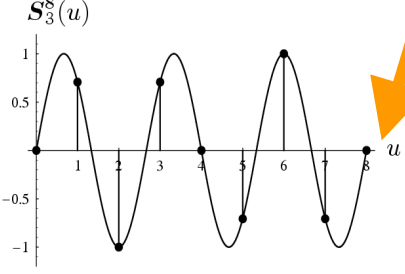
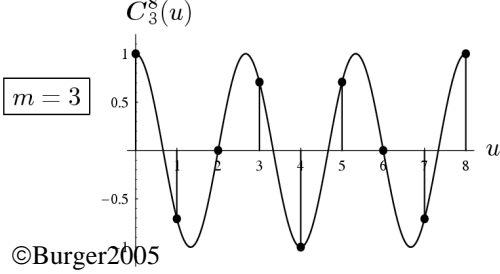
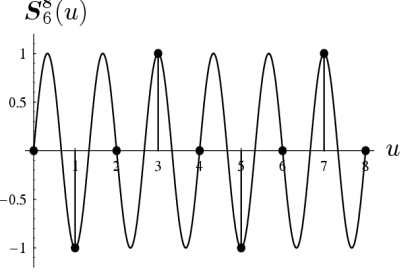
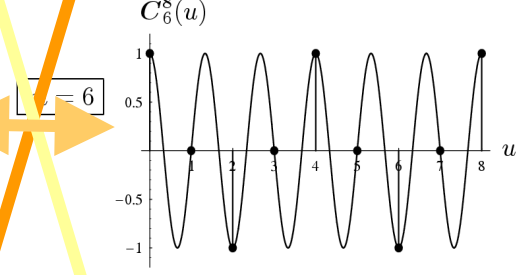
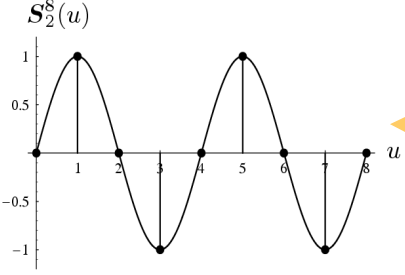
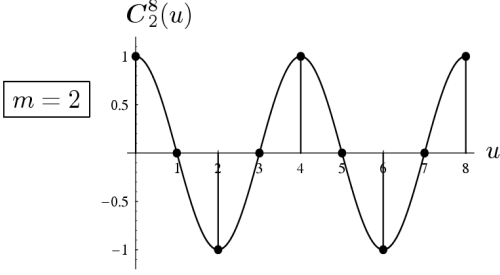
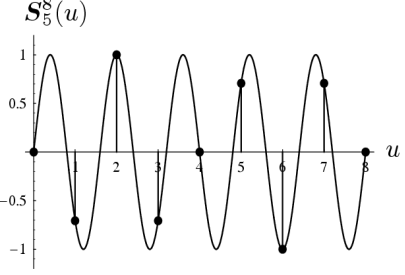
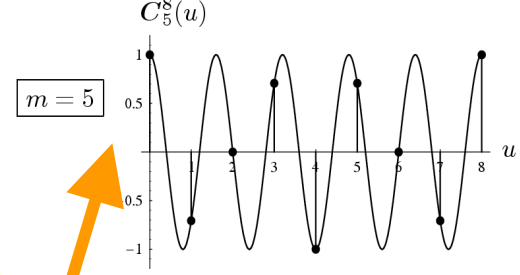
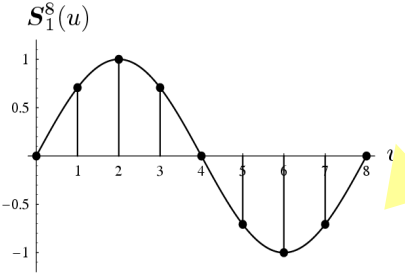
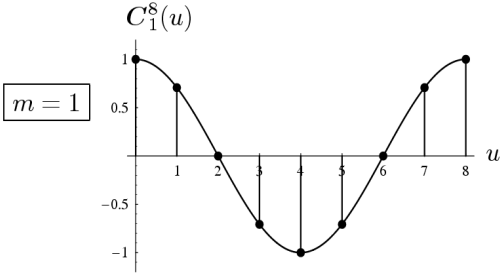
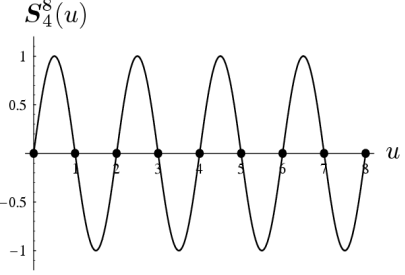
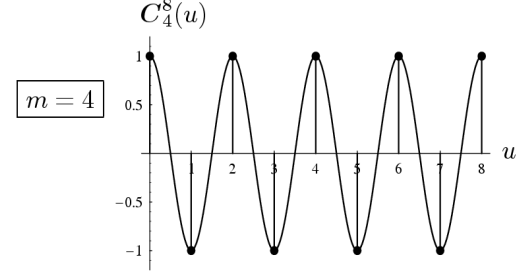
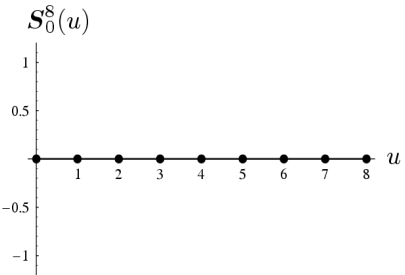
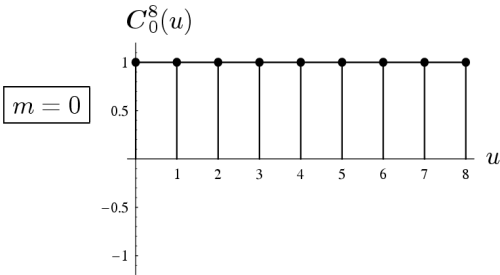
# Abtastung und Aliasing für N=8

$$C_m^8(u) = \cos\left(\frac{2\pi m}{8}u\right)$$

$$S_m^8(u) = \sin\left(\frac{2\pi m}{8}u\right)$$

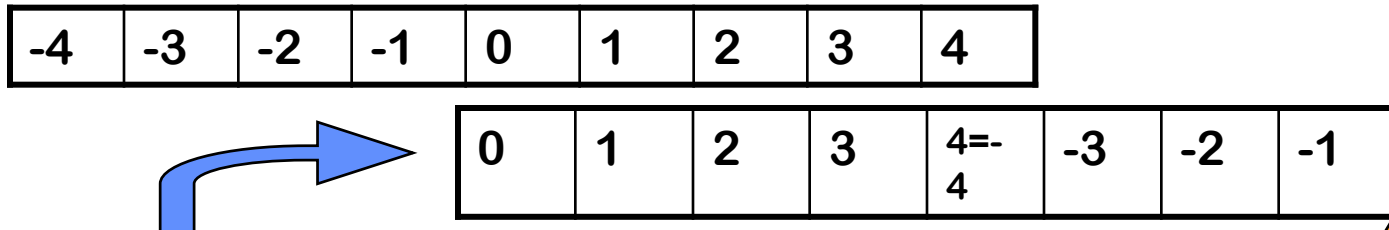
$$C_m^8(u) = \cos\left(\frac{2\pi m}{8}u\right)$$

$$S_m^8(u) = \sin\left(\frac{2\pi m}{8}u\right)$$



# Eigenschaften der diskreten 1D-FT (1)

- Nur Frequenzen bis  $m=-4, \dots, 0, \dots, +4$  erlaubt (bei  $N=8$ )
- Allgemein:  $m=-N/2, \dots, 0, \dots, +N/2$
- Ausserhalb liegende Frequenzen werden in  $-4, \dots, +4$  fälschlicherweise gemappt (**Aliasing, Nyquist-Theorem**): 5 auf 3, 6 auf 2 (genauer: -2)
- Anordnung Frequenzen physikalisch:



- Anordnung Frequenzen in FFT (genau 8 Pixel, wie in Signal). Man muss also mit Wrap-Around verschieben
- **$F(0)$  = Pixel-Summe des Signals  $G(x)$**
- **$F(-N/2) = F(N/2)$**

Aktivierung:  
Nachrechnen!

# Erzeugung einer Frequenzraumdarstellung (1)

## Diskrete 2D Fourier-Transformation

Die zweidimensionale diskrete Fourier-Transformation (FT) eines Bildes  $G(y,x)$  ist definiert als:

$$F(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} G(y, x) e^{-i\frac{2\pi u}{N}y} e^{-i\frac{2\pi v}{N}x}$$

Die Fourier-Transformation ist invertierbar (umkehrbar).  
Das Originalbild kann durch Rücktransformation in den Ortsraum wieder hergestellt werden.

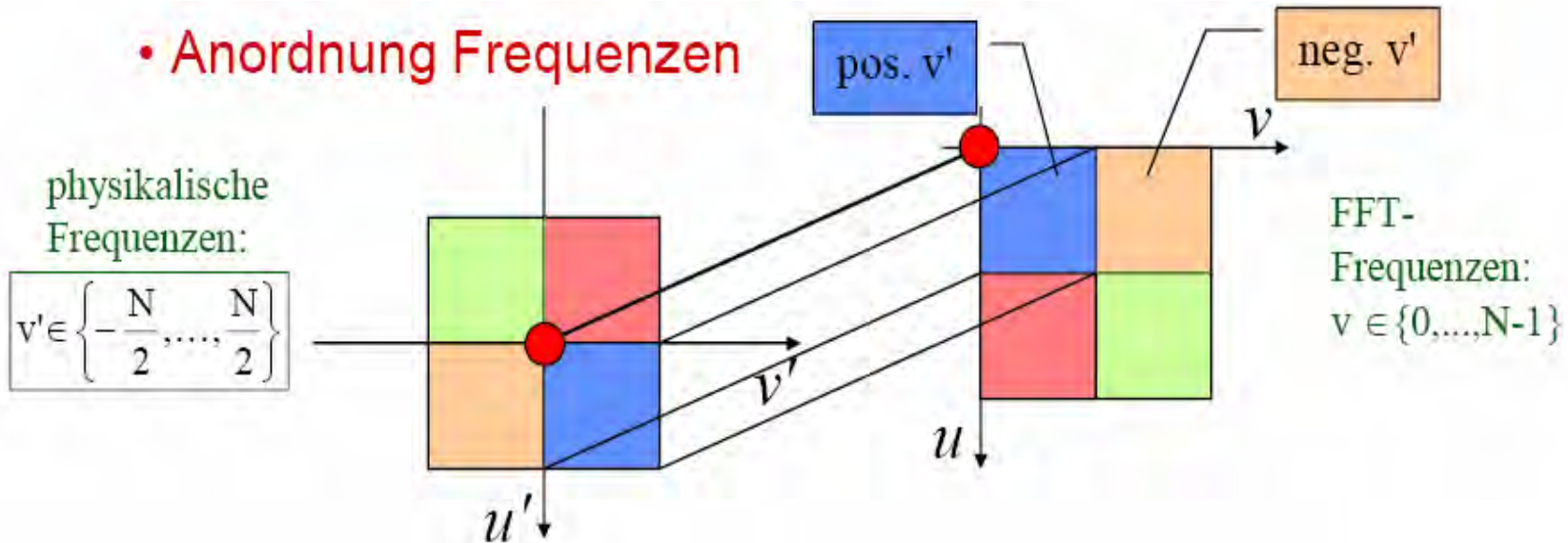
$u$  ist die **Wellenzahl** in  $y$ -Richtung und  $v$  die **Wellenzahl** in  $x$ -Richtung (Wieviel volle Wellen passen in Signal hinein?)

# Eigenschaften der diskreten 2D-FT (1)

- Es gilt  $F(-u, -v) = F(N - u, N - v)$  (nachrechnen, Periodizität)
- Inverse FT

$$G(y, x) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) e^{+i\frac{2\pi y}{N}u} e^{+i\frac{2\pi x}{N}v}$$

## • Anordnung Frequenzen



## Eigenschaften der diskreten 2D-FT (2)

---

- in  $F(0,0)$  steht Fourierkoeff. zu 0-Frequenz ("Gleichstromanteil", DC)
- in Spalte  $v=10$  stehen Fourierkoeffizienten, deren Wellenlängen in x-Richtung genau 10x ins Bild passen (**Wellenzahl**)
- oberhalb von  $v=N/2$  beginnen die negativen Frequenzen
- Übergang zu physikalischen Frequenzen durch "Wrap-around-Shift" um  $(u_0, v_0) = (N/2, N/2)$  (**fftshift** in MATLAB, "**Origin at image center**" in ImageJ's Plugin FFTJ)



# Erzeugung einer Frequenzraumdarstellung

## Fourierspektrum (Spektrum der Ortsfrequenzen)

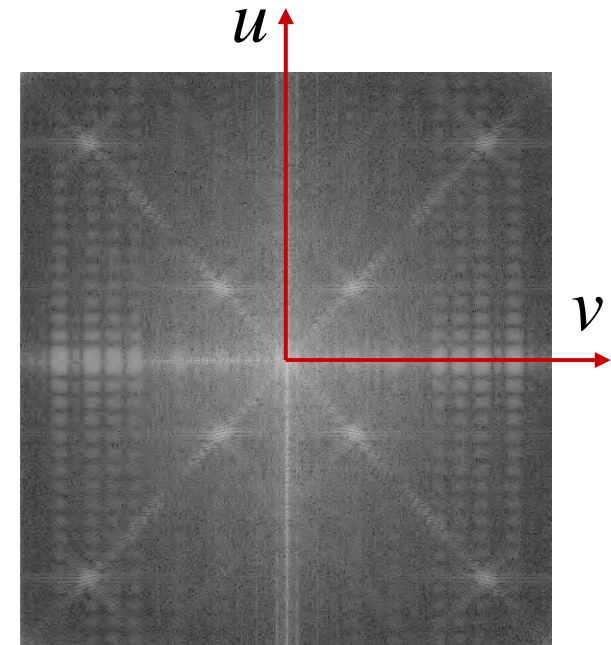
Den Betrag der komplexen Fourier-Transformierten bezeichnet man als Frequenzspektrum (kurz "Spektrum"):

Leistungsspektrum (power spectrum):  $P(u, v) = [R(u, v)]^2 + [I(u, v)]^2$

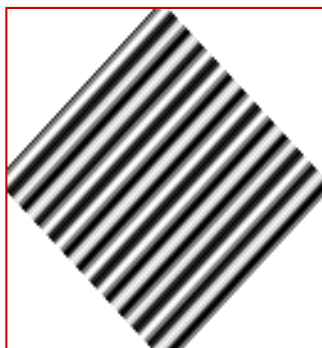
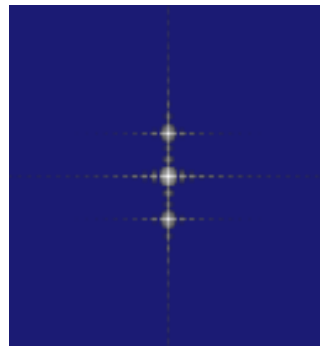
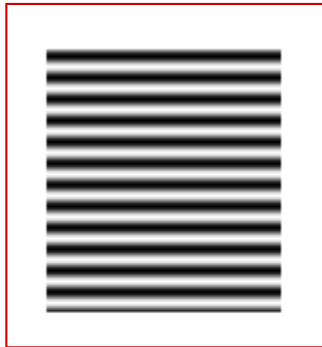
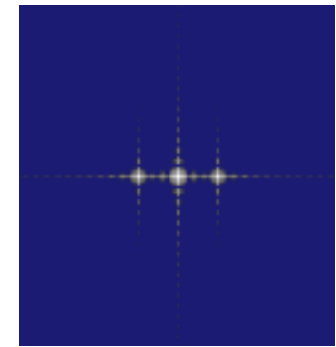
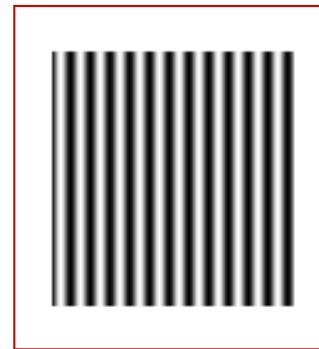
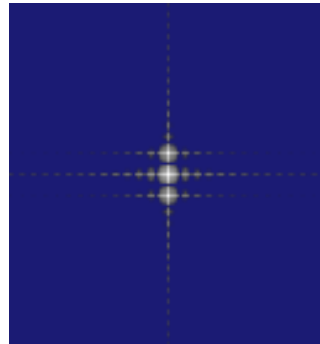
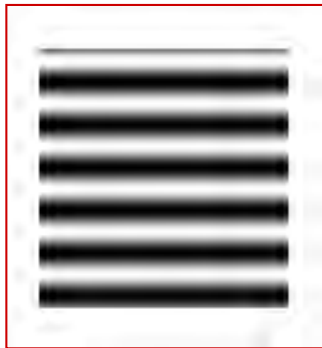
$$|F(u, v)| = \sqrt{[R(u, v)]^2 + [I(u, v)]^2}$$



512 × 512



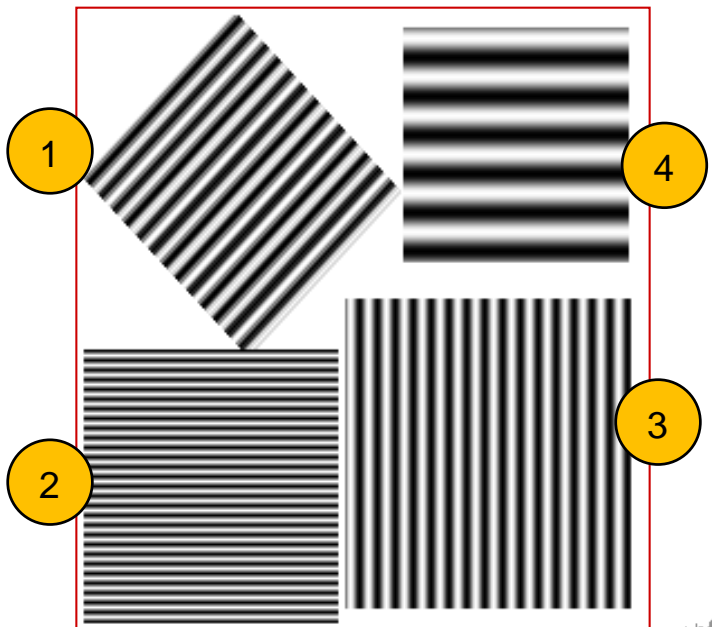
# Verschiedene Sinusgitter und ihre Frequenzbilddarstellung



- Der Betrag der Fourier-Transformierten kann als ein "Frequenz-Bild" dargestellt werden.
- Das Frequenzbild ist punktsymmetrisch bzgl. seines Ursprungs, des Null-Frequenz-Punkts ("DC"-Komponente).
- Horizontale (vertikale) Frequenzen im Ortsraum werden im Frequenzbild entlang der horizontalen (vertikalen) Achse aufgetragen.

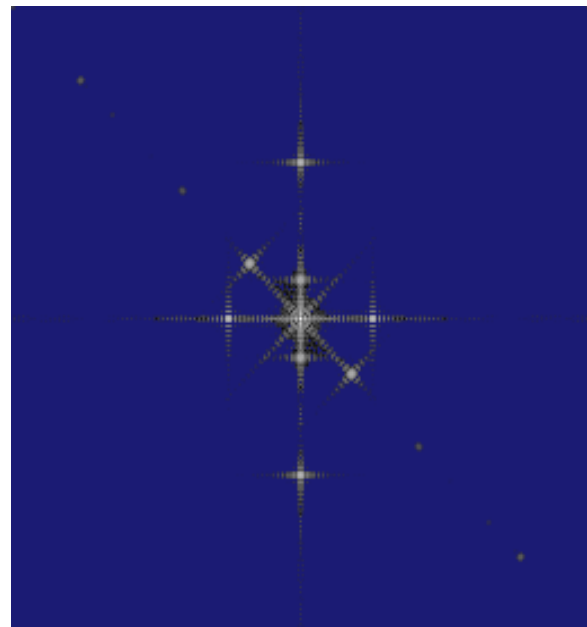
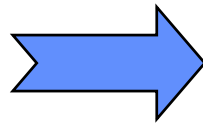
Aktivierung

# Frequenzbilddarstellung eines Bildes aus unterschiedlichen lokalen Sinusmustern



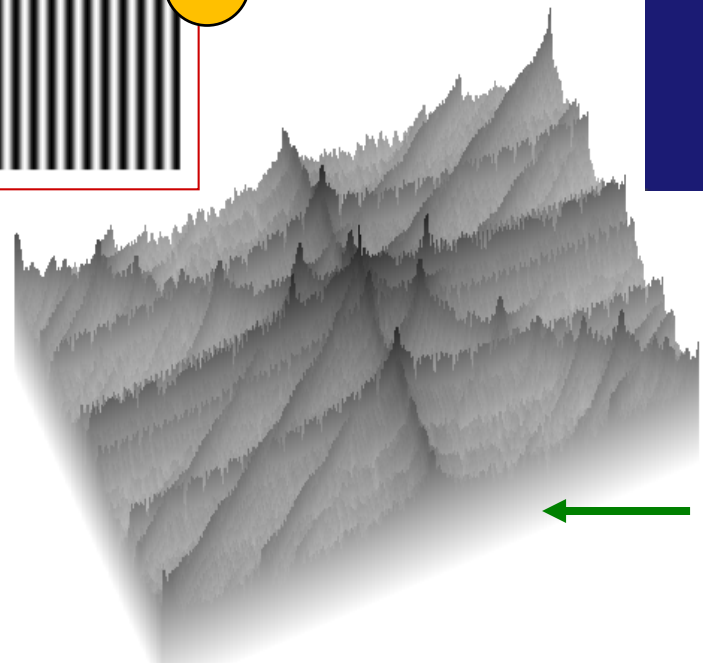
Originalbild

Fourier-Transformation und anschließende Betragsbildung



Betrag der Fourier-Transformierten

Ordnen Sie den Bildelementen die Peaks in der FFT zu!



---

jetzt Übung FFT /  
Sinus Gratings,  
Teil A

# Inhalt

---

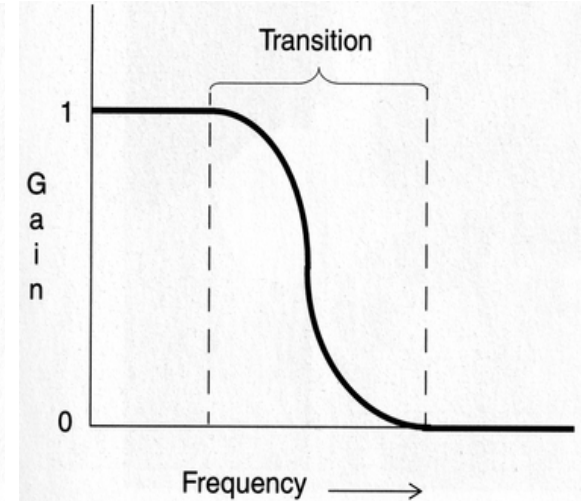
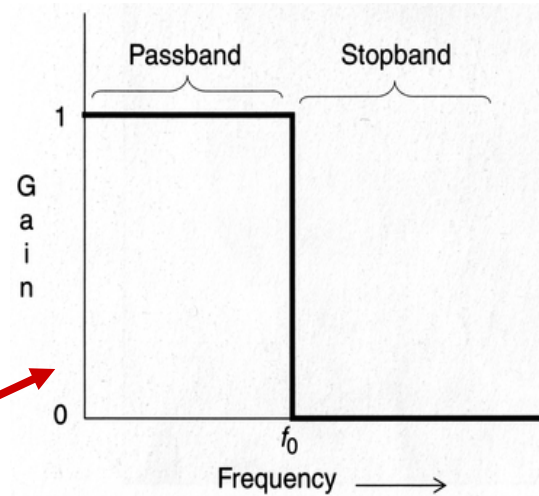
- Filter im Ortsraum: Die Faltung
- Tiefpass-Filter
- Filter im Frequenzraum: Fouriertransformation
  - 1D-Fouriertransformation
  - 2D-Fouriertransformation
- **Anwendungen der Fouriertransformation (kurz)**
  - **Bandstop**
  - **Unsharp Masking**

# Lineare Filter (1)

## □ Tiefpass

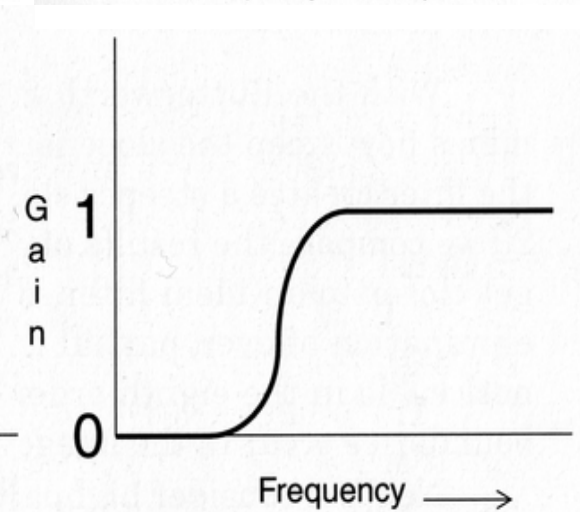
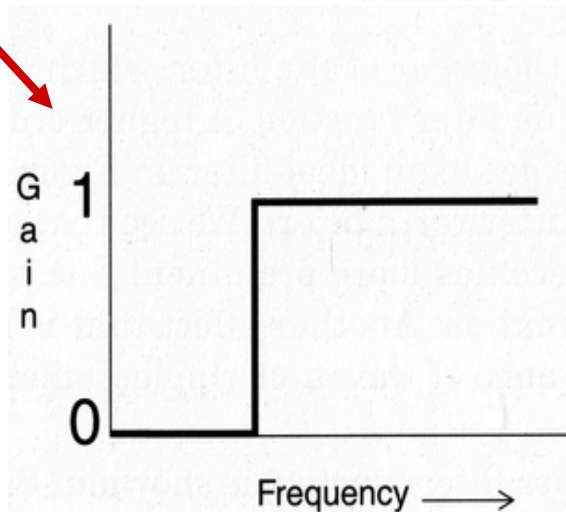
- Unterdrückung hoher Frequenzen (feiner Details)

Ideale und nicht-ideale 1D-Filterfunktion im Frequenzbereich



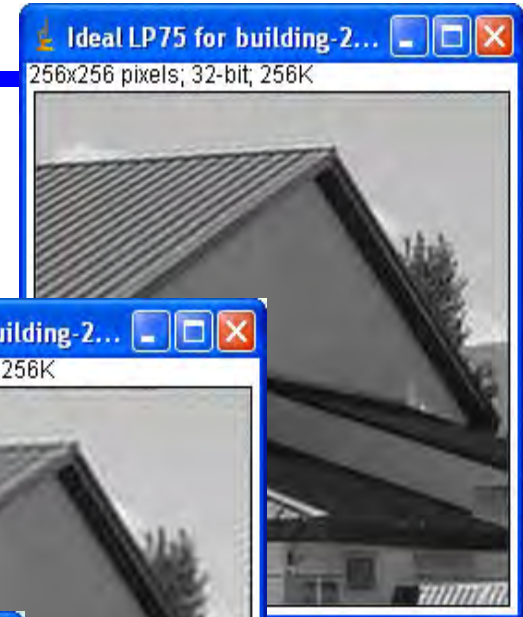
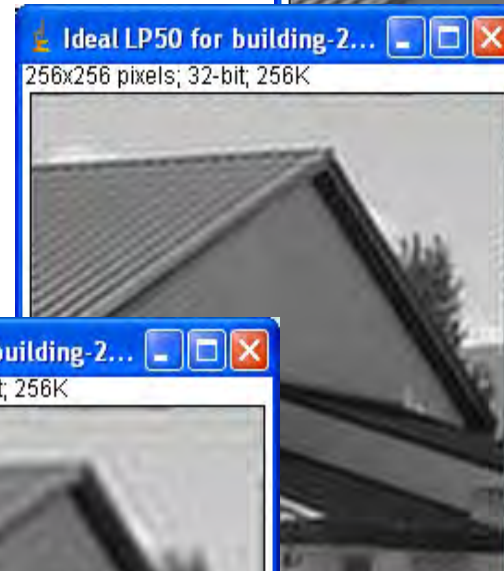
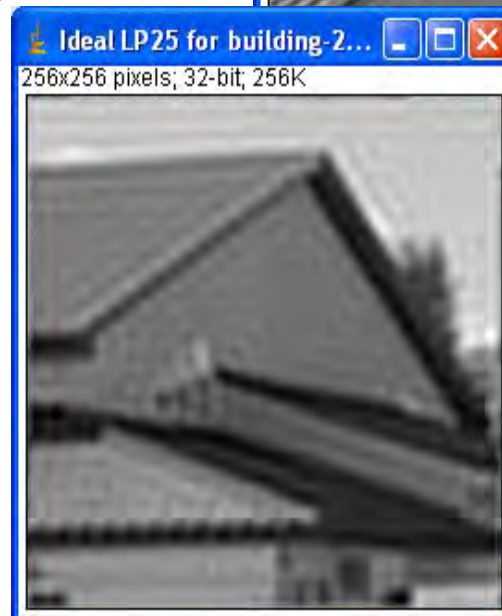
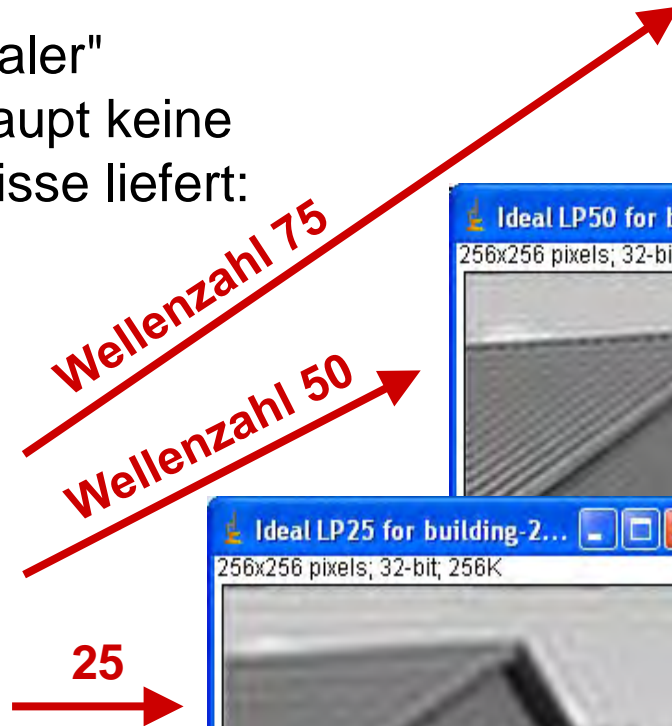
## □ Hochpass

- Unterdrückung niedriger Frequenzen (grober Strukturen)



# Lineare Filter (2): Ringing-Artefakte

- Warum ein "idealer" Tiefpass überhaupt keine idealen Ergebnisse liefert:



**"Ringing"**: eine scharfe Kante umgibt sich mit (mehreren) Ringen

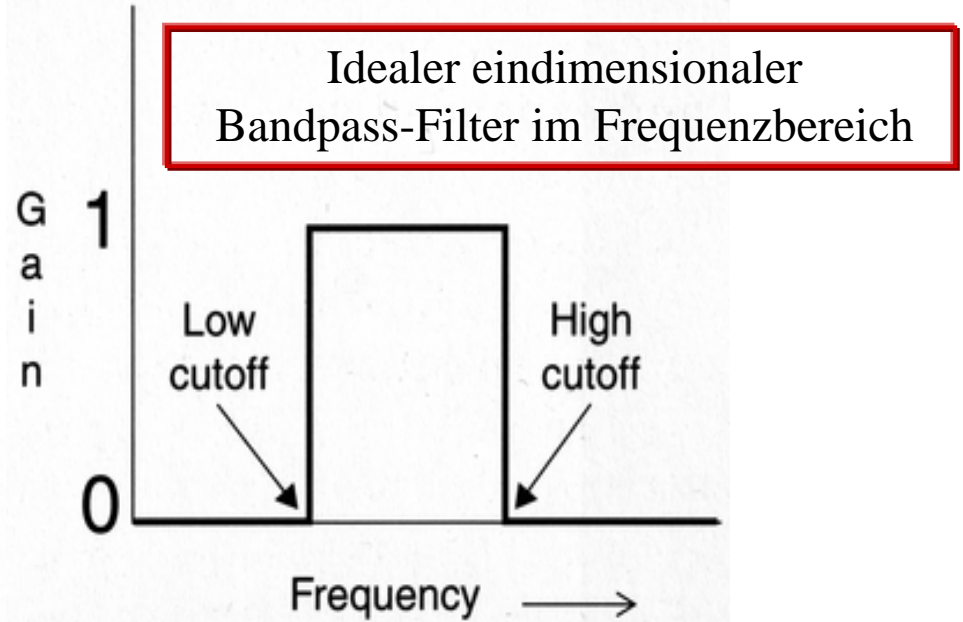
[ImageJ\plugins\FFTJ\deal\\_LP\\_FFTJ.java](#)



# Lineare Filter (3)

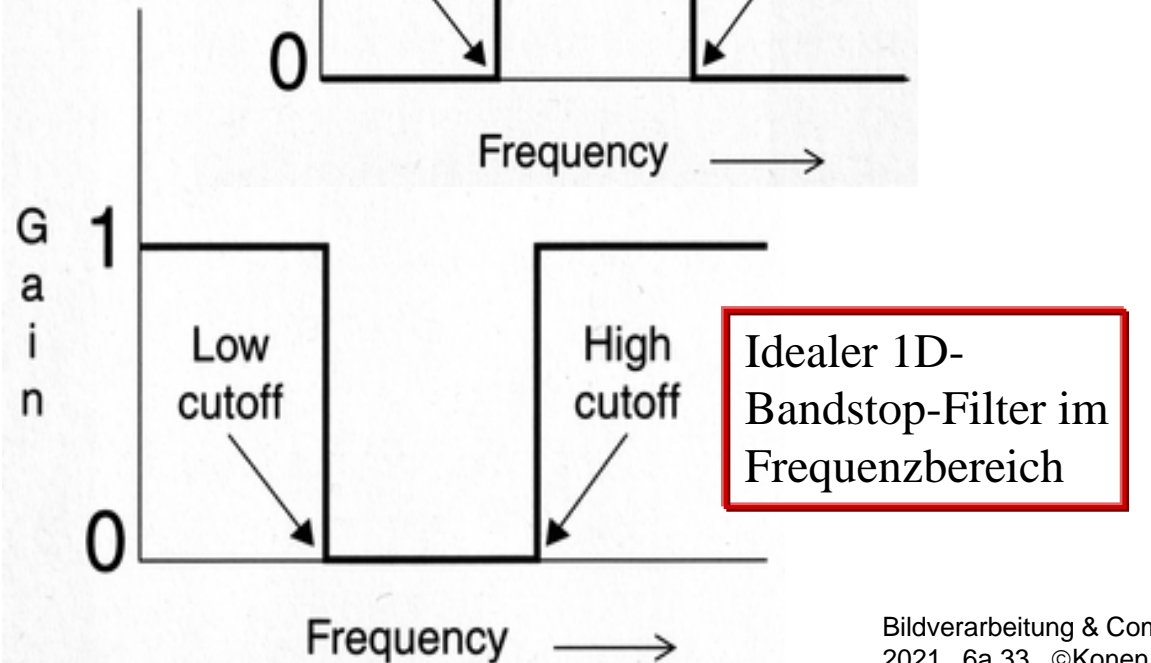
## □ Bandpass

- Unterdrückung von hohen und niedrigen Frequenzen



## □ Bandstop

- Unterdrückung von Mittenfrequenzen

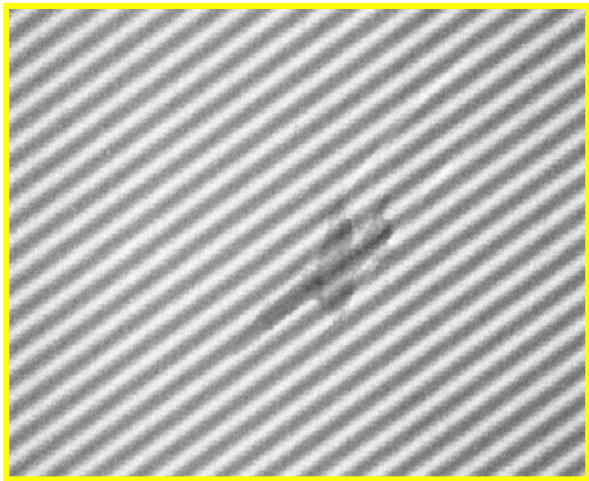




# Anwendung eines Bandstop-Filters

## Beispiel: Bildrestauration

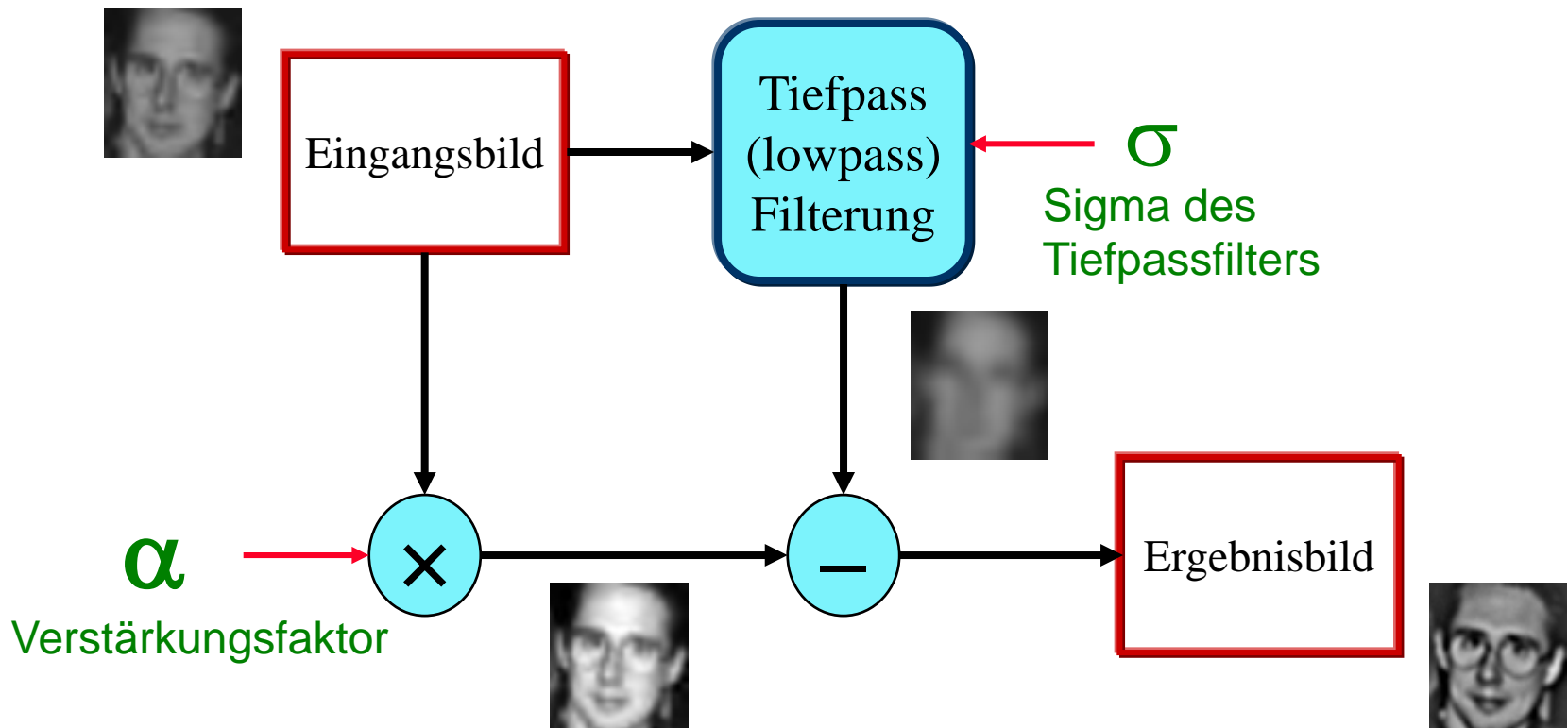
---



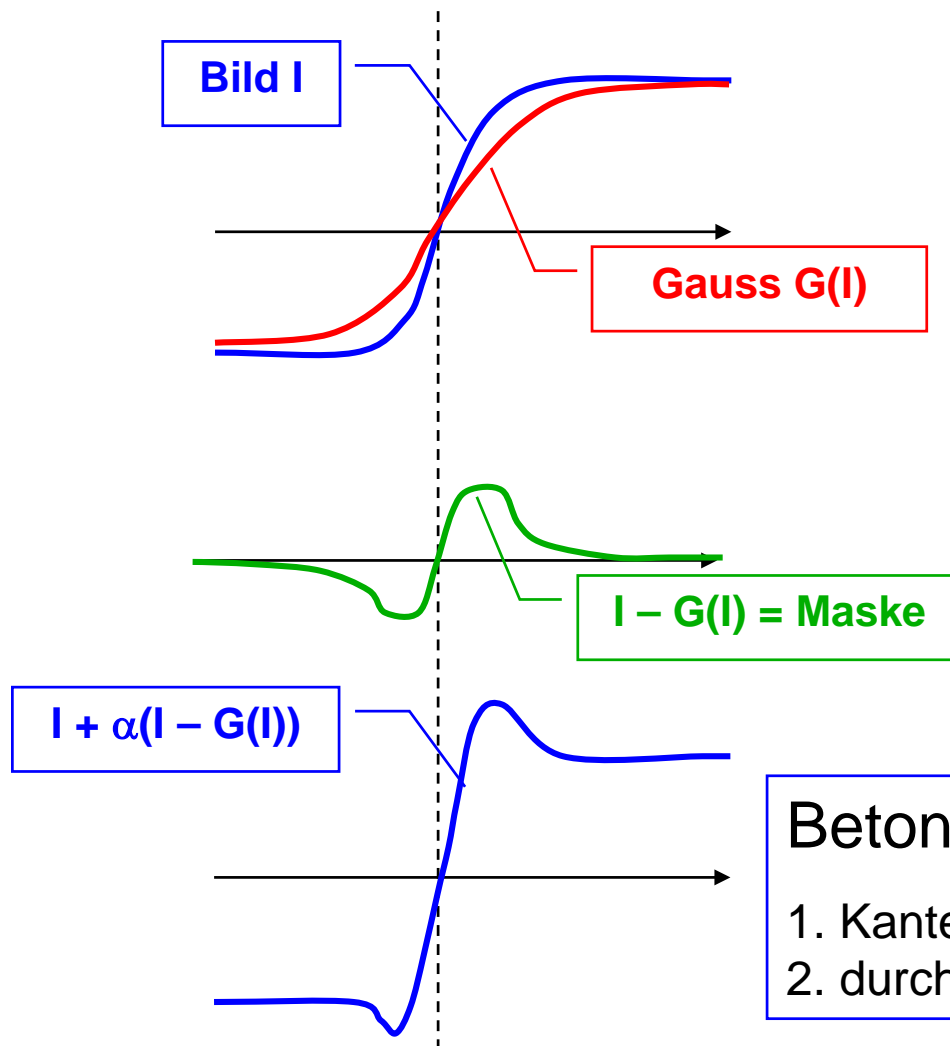
**Unterdrückung der  
Störfrequenz**

# Anwendung des Tiefpassfilters für das Schärfen (sharpening) eines Bildes

## "Unsharp Masking" - Verfahren



# Unsharp-Masking Verfahren



Betonung der Kanten:

1. Kanten werden steiler
2. durch Überschwinger stärkerer Kontrast

# Unschärfe Maske (Unsharp Masking)

## Bildbeispiele

---

