

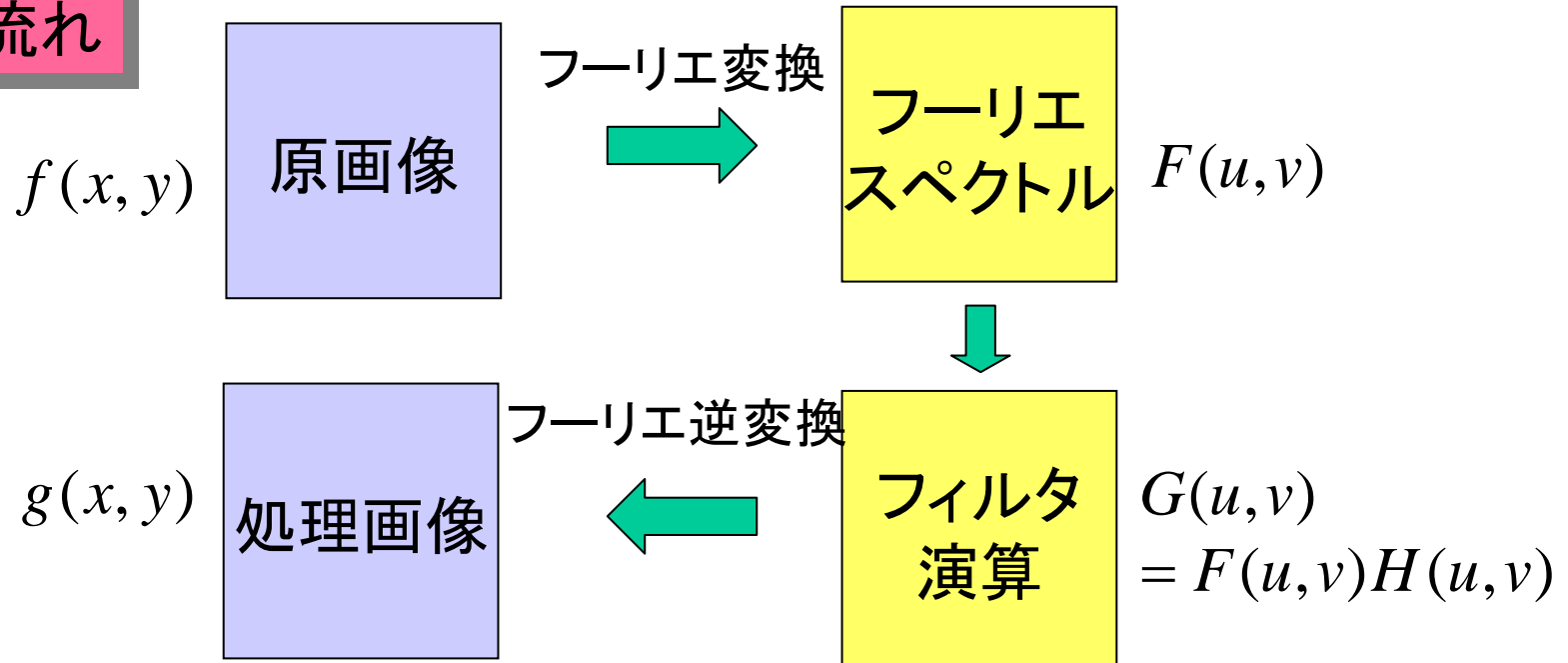
画像のフィルタリング処理

講義内容

- 実空間フィルタリング
 - 平滑化(LPF)
 - エッジ強調(HPF)
 - Laplacian of Gaussian(LOG)フィルタ(BPF)
- 周波数空間フィルタリング
 - LPF, HPF, BPF
 - 周波数選択的フィルタ
- 線形シフトインバリエントシステムと劣化画像復元
 - 線形システム
 - 劣化画像の復元
- MATLABを用いたデモ

フーリエ面での処理

処理の流れ



特徴

周波数成分に対する**自在な**フィルタリングが可能

例

LPF, BPF, HPF, 部分的なフィルタ

(特定周波数成分の除去, 周期構造をもつノイズの除去)

Wiener フィルタ (周波数ごとのSN比を考慮した復元フィルタ)

コンボリューション定理

実空間

フーリエ空間

$$f(x, y) \longleftrightarrow F(u, v)$$

$$h(x, y) \longleftrightarrow H(u, v)$$

$$g(x, y) \longleftrightarrow G(u, v)$$

$$g(x, y) = f(x, y) * h(x, y) \longleftrightarrow G(u, v) = F(u, v) \cdot H(u, v)$$

コンボリューション

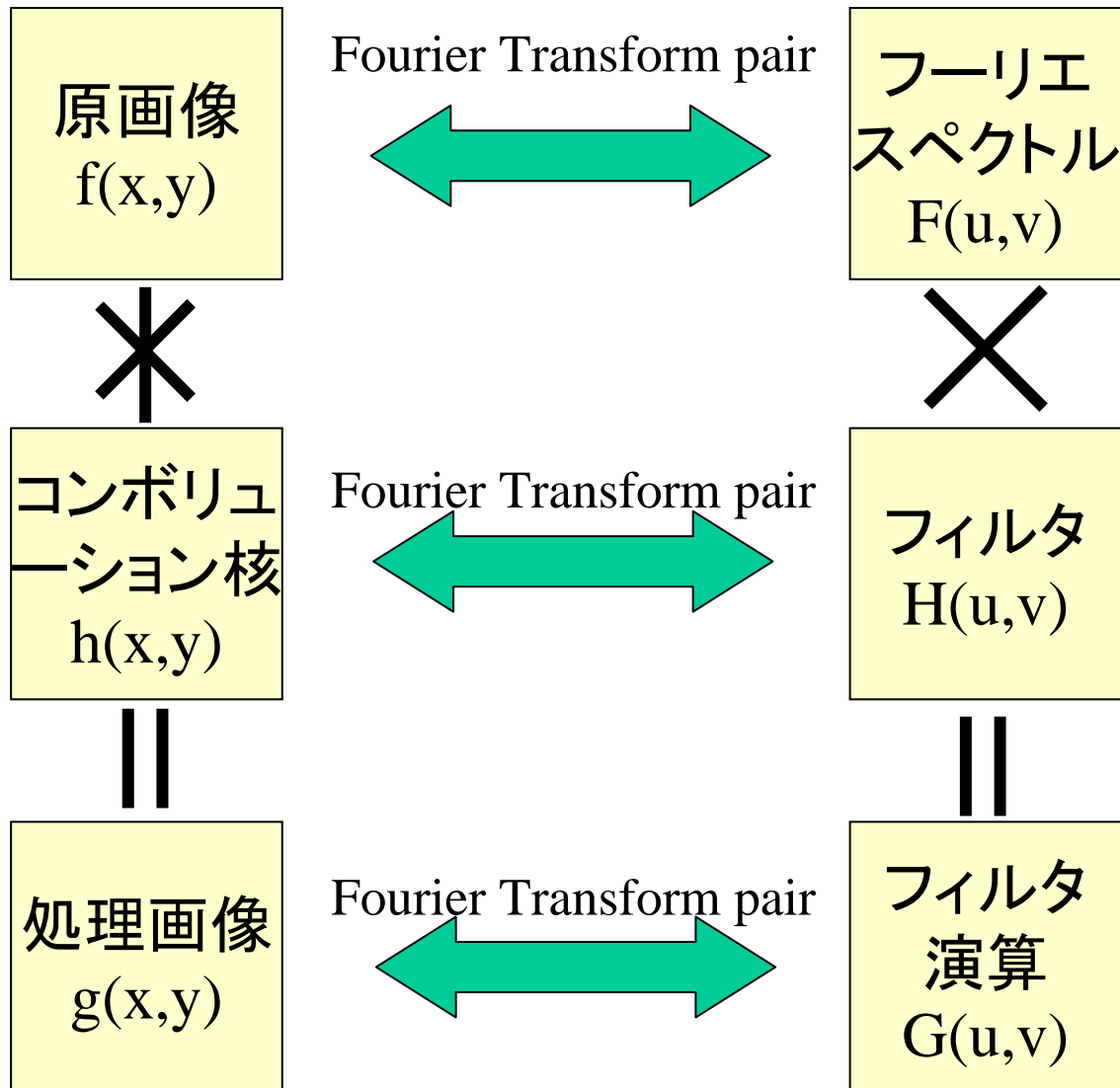
積

$$g(x, y) = f(x, y) \cdot h(x, y) \longleftrightarrow G(u, v) = F(u, v) * H(u, v)$$

積

コンボリューション

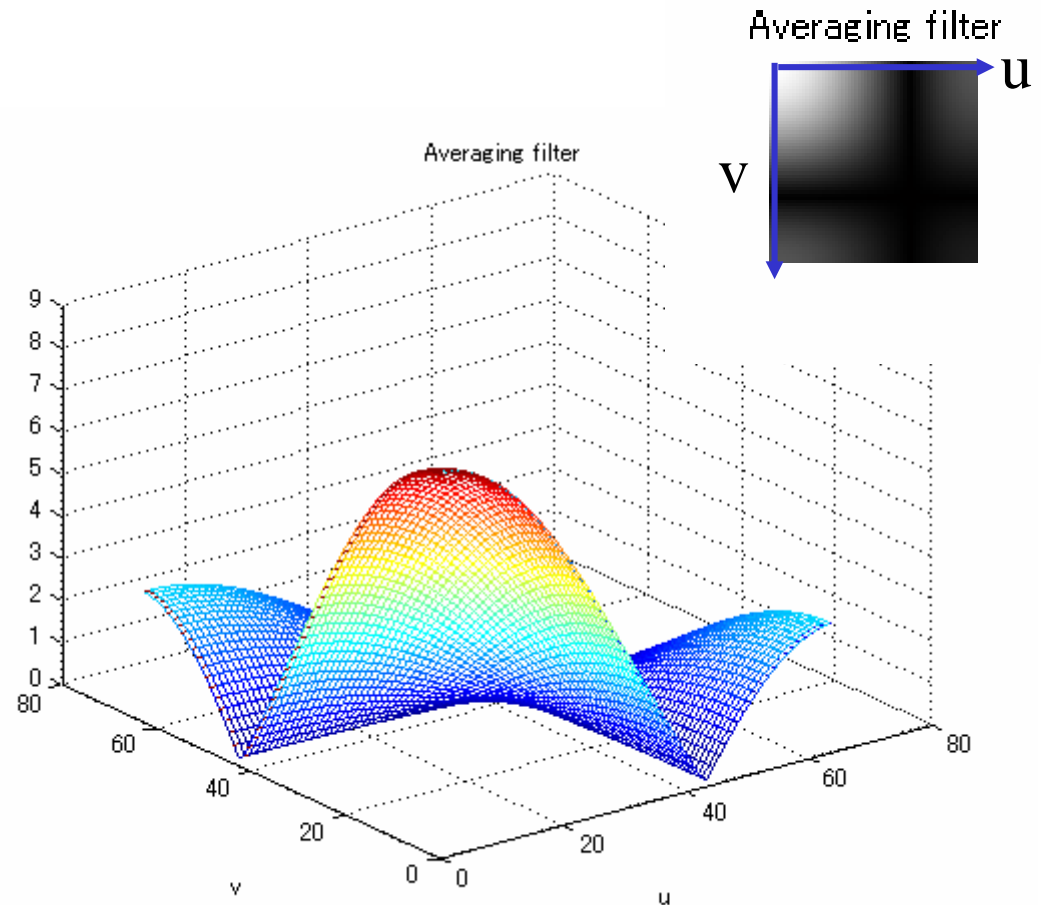
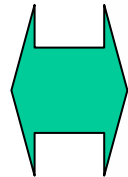
処理の等価性



平滑化フィルタ

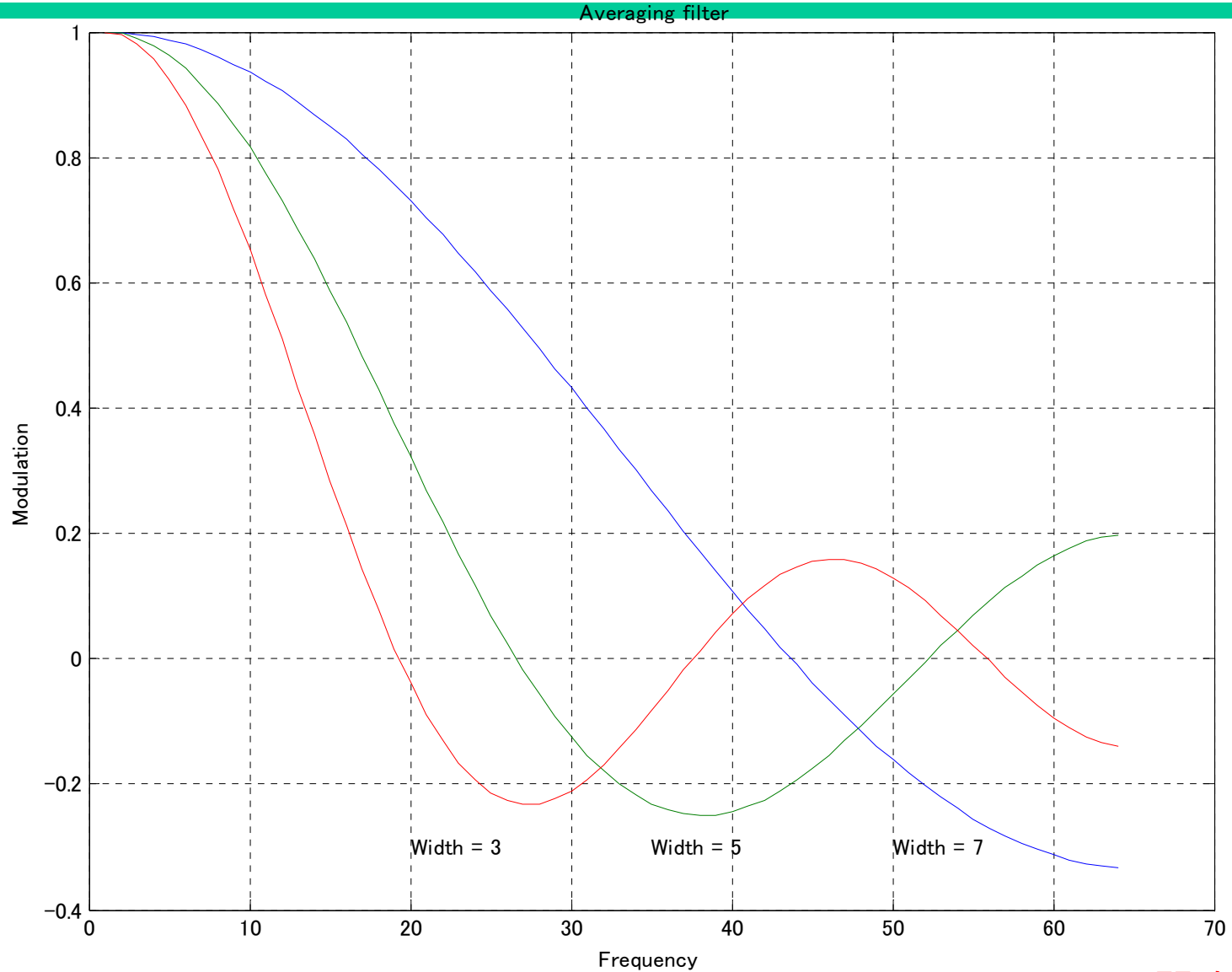
空間周波数フィルタ

実空間でのフィルタ
(コンボリューション核)

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


(フィルタ特性の絶対値をとって表示)

平滑化フィルタの周波数特性



Low pass filter

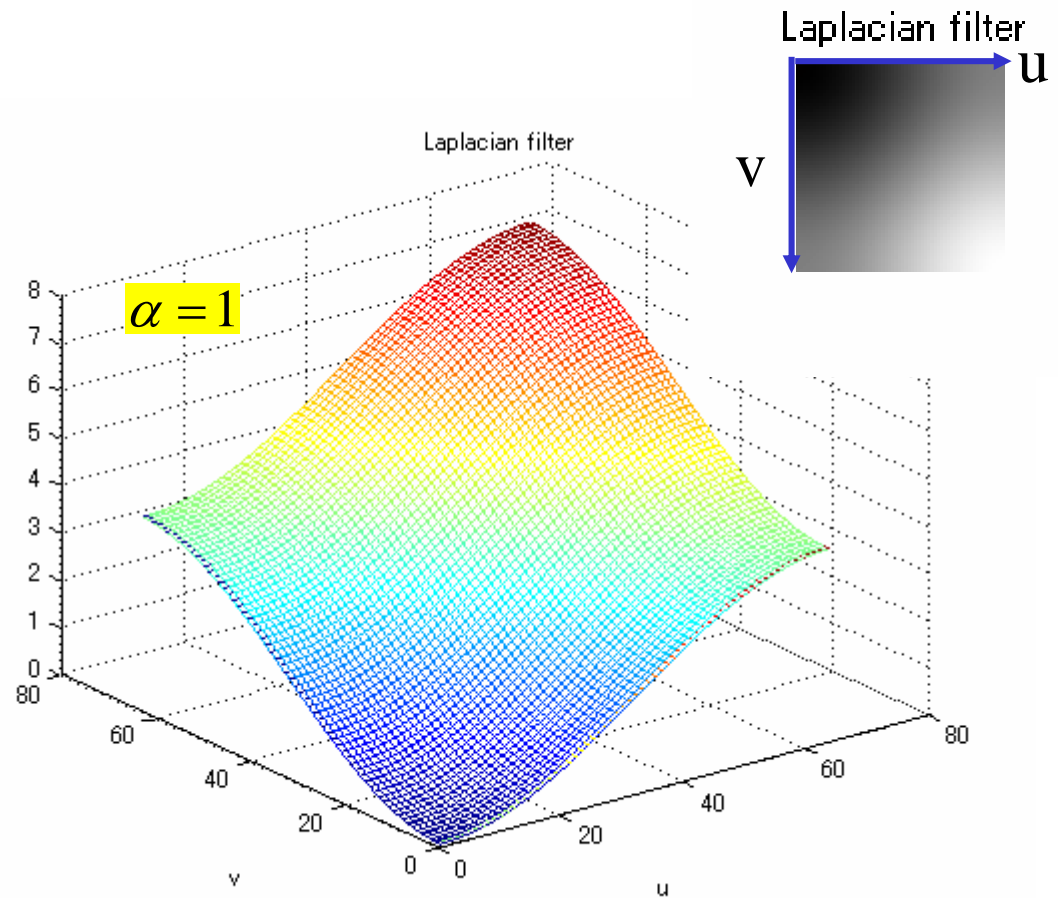
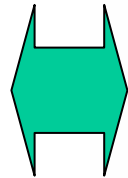
間違い！
3, 7が逆.

Laplacianフィルタ

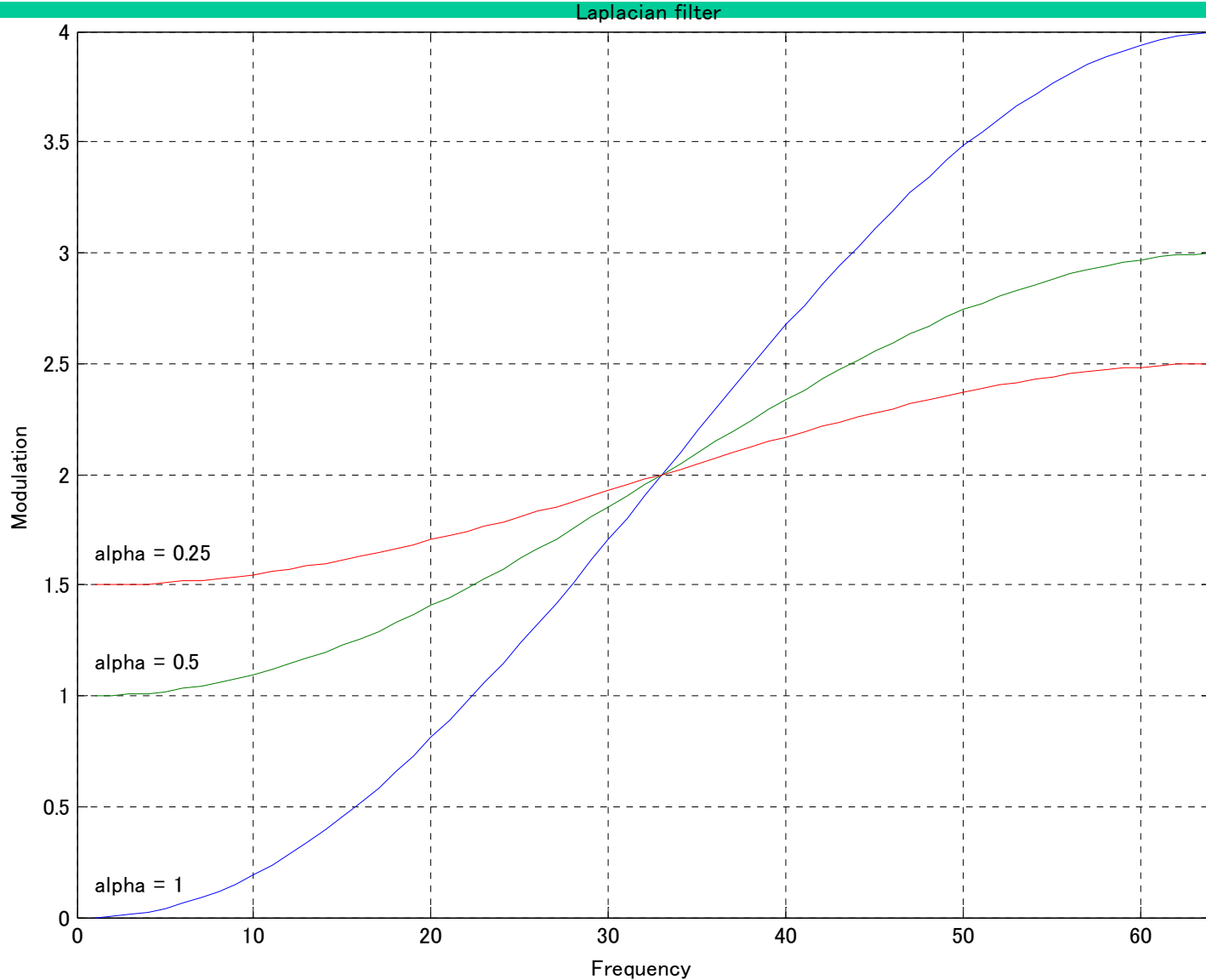
空間周波数フィルタ

実空間でのフィルタ
(コンボリューション核)

0	$-\alpha$	0
$-\alpha$	4	$-\alpha$
0	$-\alpha$	0



ラプラシアンフィルタの周波数特性



High pass filter

Sobel フィルタ

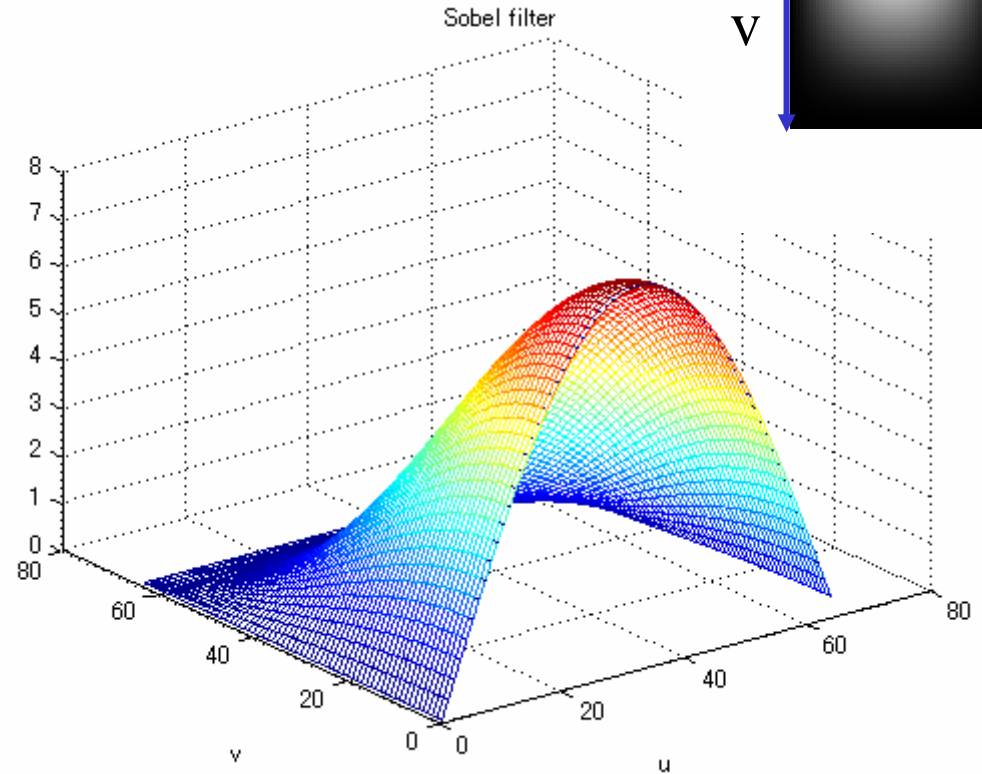
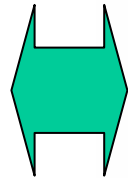
空間周波数フィルタ

実空間でのフィルタ
(コンボリューション核)

-1	0	1
-2	0	2
-1	0	1

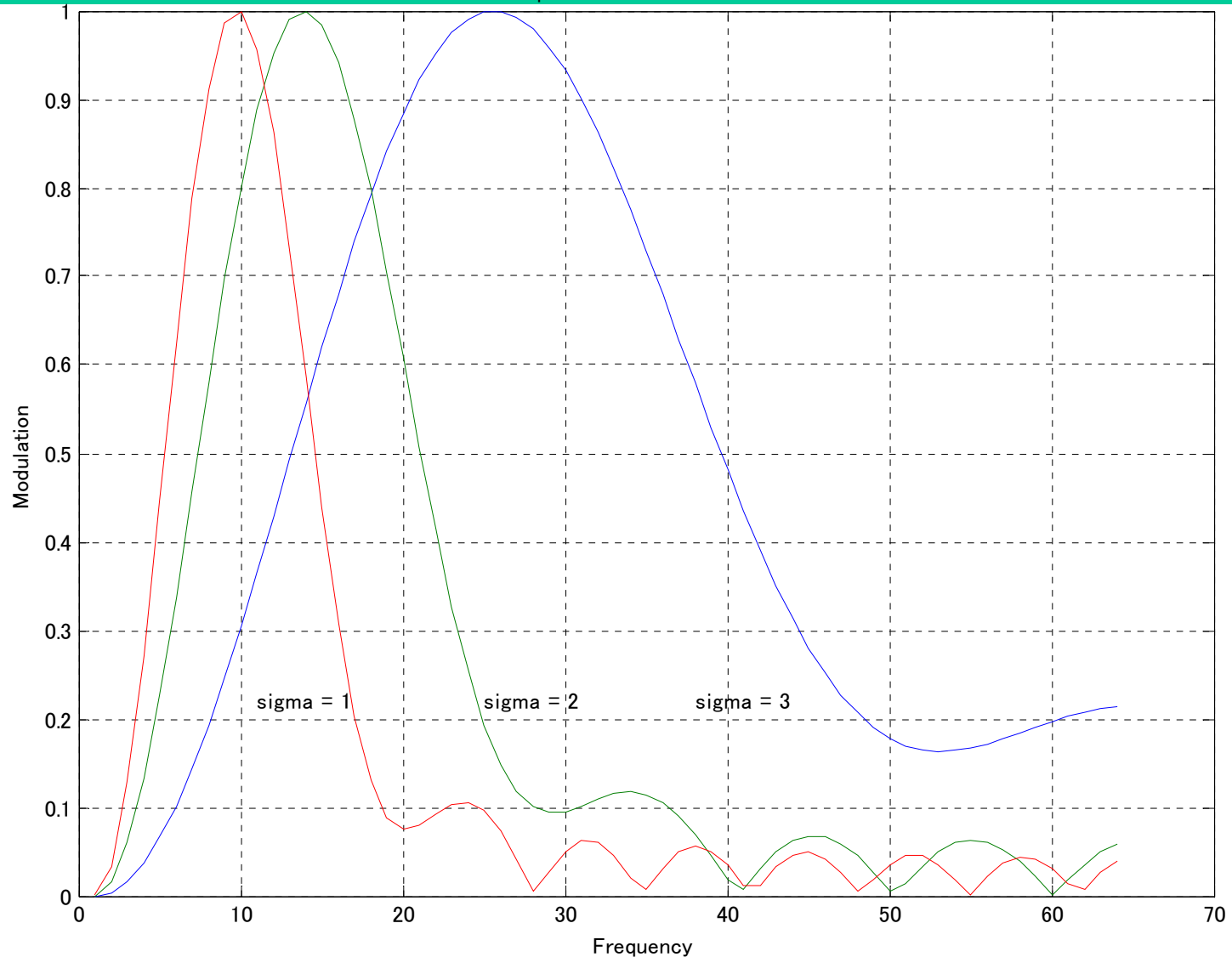
x

y



LOGフィルタの周波数特性

Laplacian of Gaussian filter



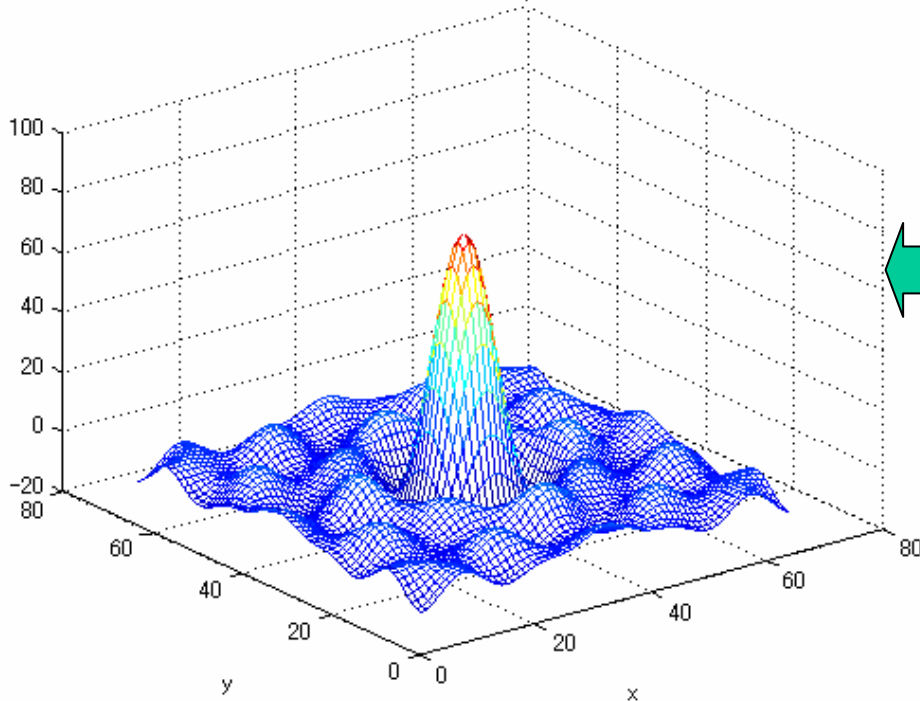
Band pass filter

間違い！
Sigmaの並びが逆.

空間周波数フィルタとコンボリューション核の例

実空間

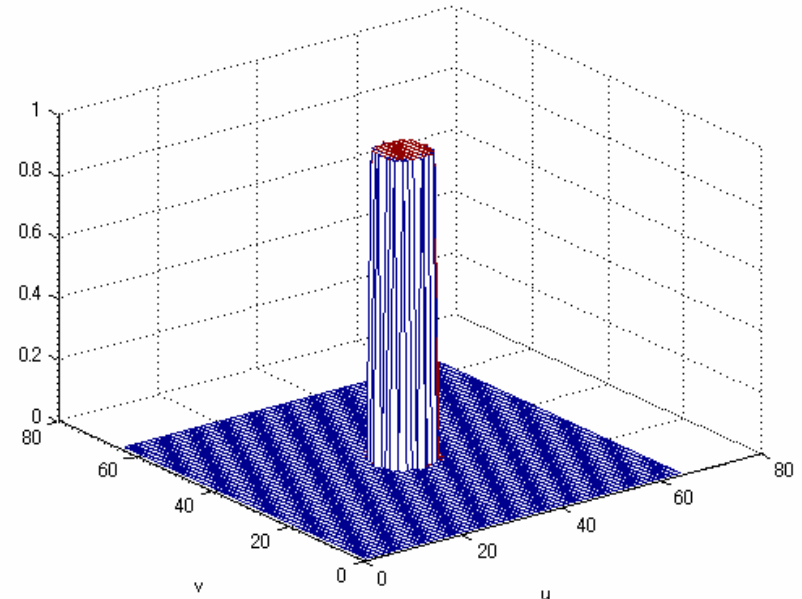
Convolution kernel in spatial domain



コンボリューション核

フーリエ空間

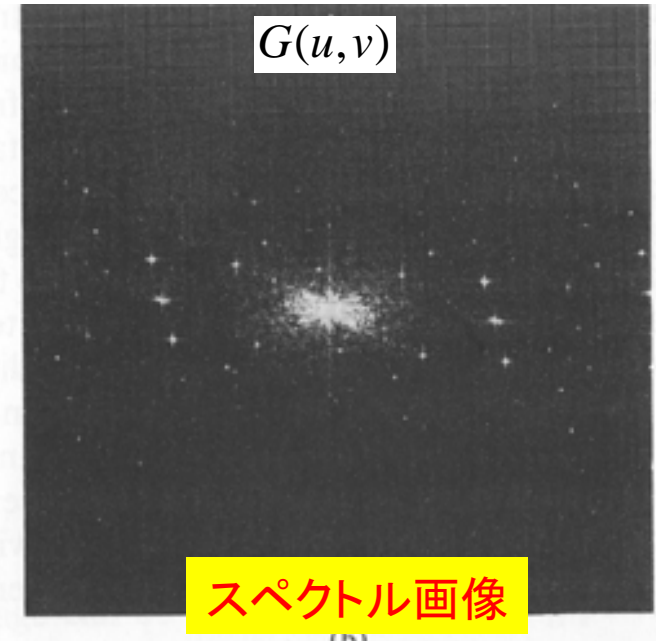
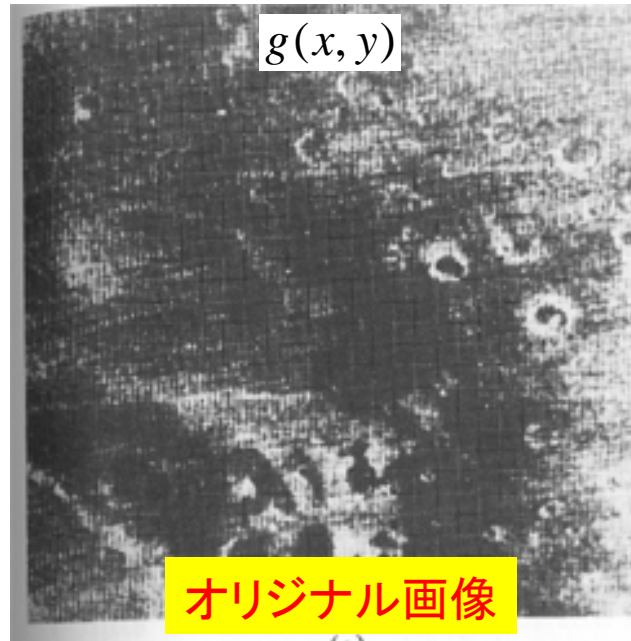
Filter in Fourier domain



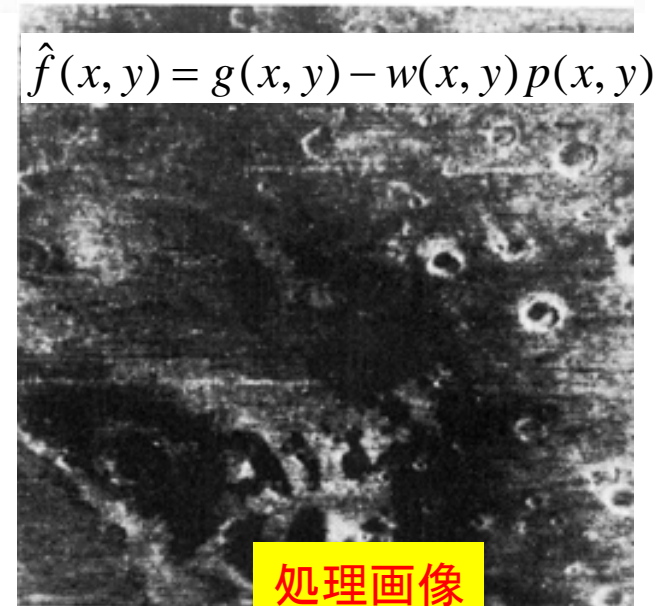
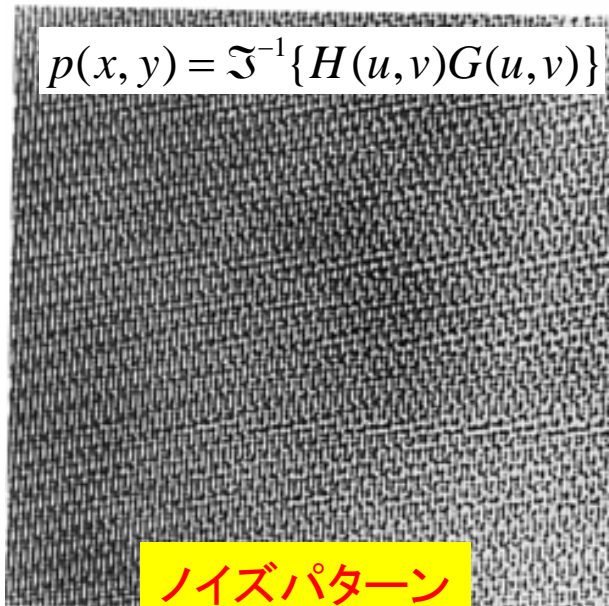
空間周波数フィルタ
Sharp-cut LPF

周期性のあるノイズの低減

周波数空間の一部にノイズのパワーが集中しているようなとき



重み $w(x, y)$ は (x, y) の近傍で推定画像の分散が最小になるように決定.



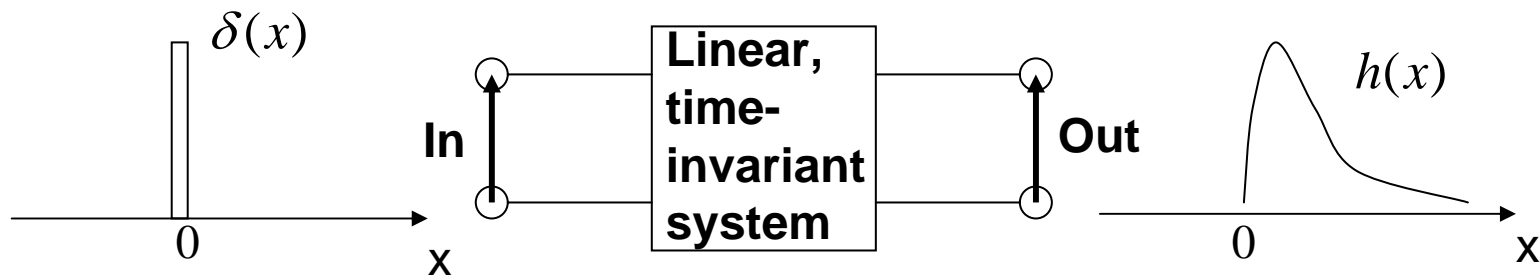
Digital Image Processing, R. C. Gonzalez and R. E. Woodsから引用

画像のフィルタリング処理

講義内容

- 実空間フィルタリング
 - 平滑化(LPF)
 - エッジ強調(HPF)
 - Laplacian of Gaussian(LOG)フィルタ(BPF)
- 周波数空間フィルタリング
 - LPF, HPF, BPF
 - 周波数選択的フィルタ
- 線形シフトインバリエントシステムと劣化画像復元
 - 線形システム
 - 劣化画像の復元
- MATLABを用いたデモ

線形時不変システム または線形シフトインバリエントシステム

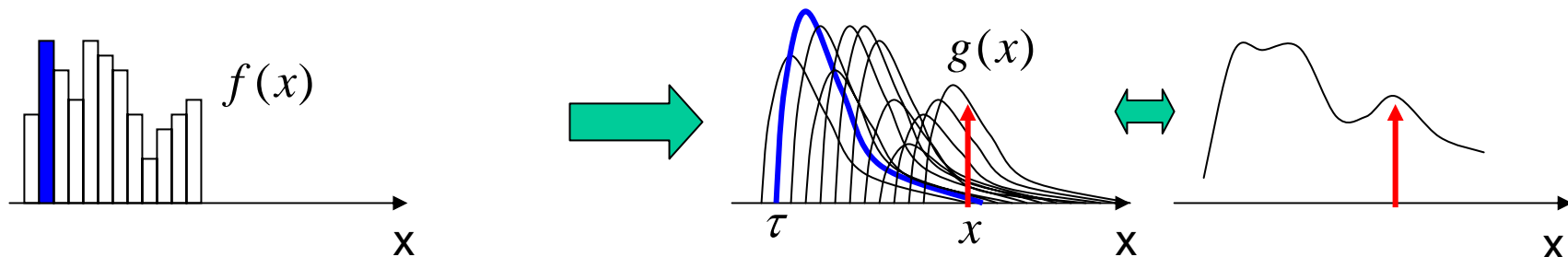


ディラックのデルタ関数
：インパルス関数

デルタ関数入力に対する応答：
インパルス応答

入力信号

出力信号



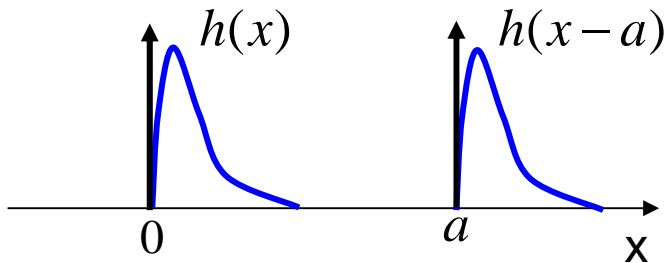
出力信号は入力信号と
インパルス応答との
コンボリューションで
表される。

$$g(x) = \int_{-\infty}^{\infty} h(x-\tau)f(\tau)d\tau$$

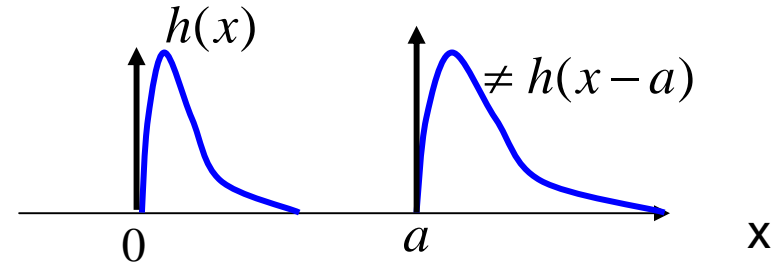
$$= h(x) * f(x)$$

シフトインバリエントシステム

シフトインバリエント：インパルス応答が，シフトによらないこと。

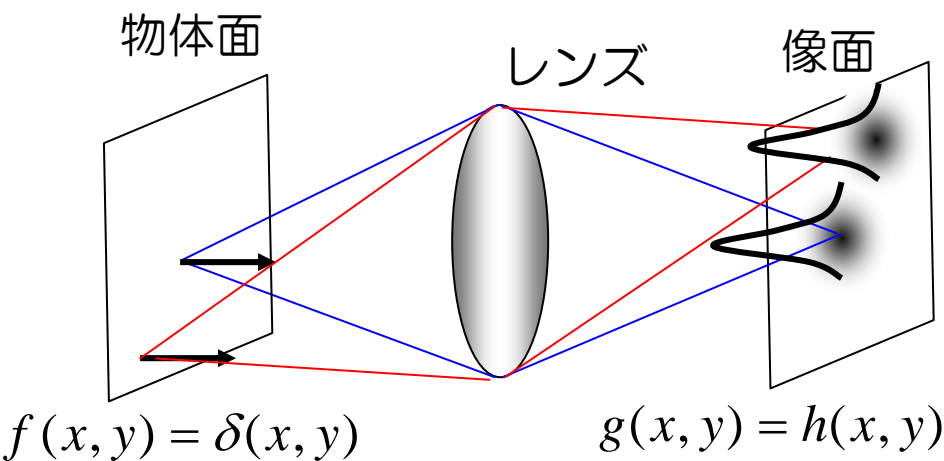


シフトインバリエント

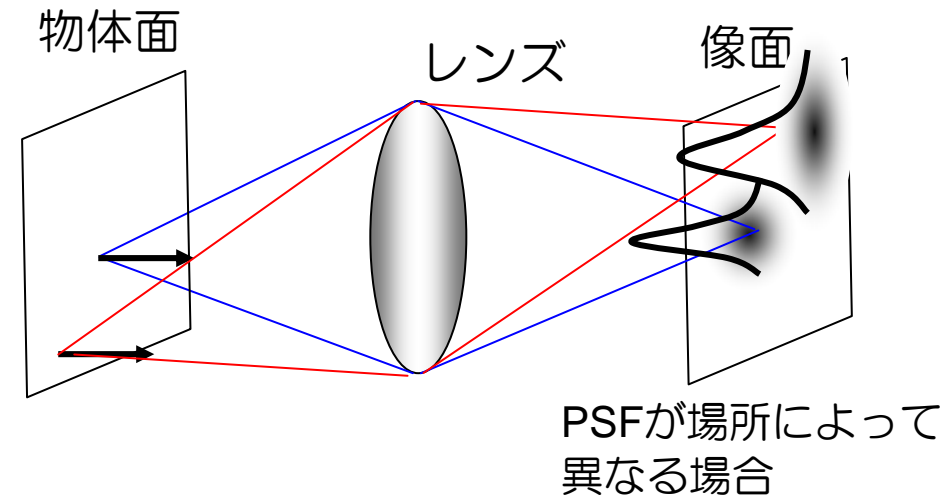


シフトバリエント

2次元（画像）の場合 インパルス応答＝点光源に対するレンズによる像
（点像分布関数point spread functionとよぶ）



シフトインバリエント



シフトバリエント

線形システム

線形システム：重ね合わせの原理が成り立つこと

入力 $f(x)$ に対して、 $g(x)$ を出力するシステムを以下のように定義する。

$$g(x) = S\{f(x)\}$$

このシステムが線形であるとは、以下の関係が成り立つことである。

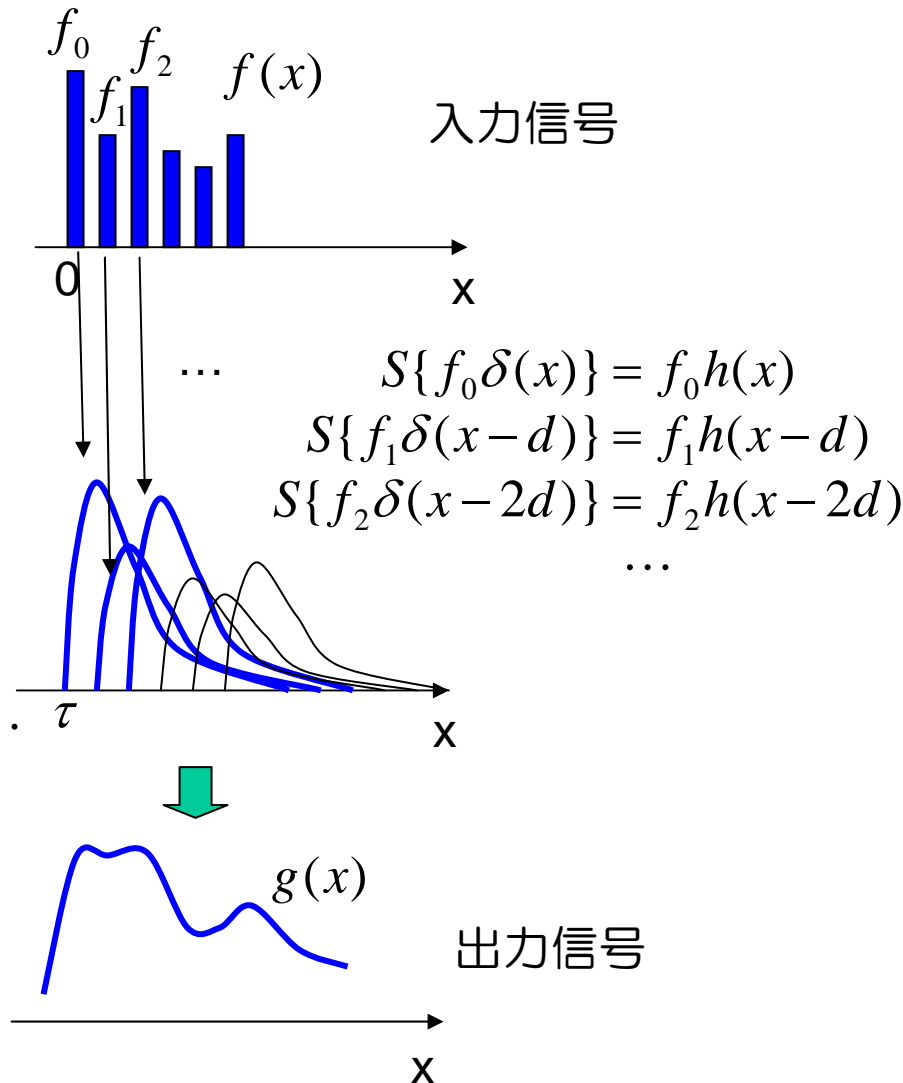
$$\begin{aligned} g(x) &= S\{a_1 f_1(x) + a_2 f_2(x)\} \\ &= a_1 S\{f_1(x)\} + a_2 S\{f_2(x)\} \end{aligned}$$

入力関数：

$$f(x) = f_0 \delta(x) + f_1 \delta(x-d) + f_2 \delta(x-2d) + \dots$$

出力関数：

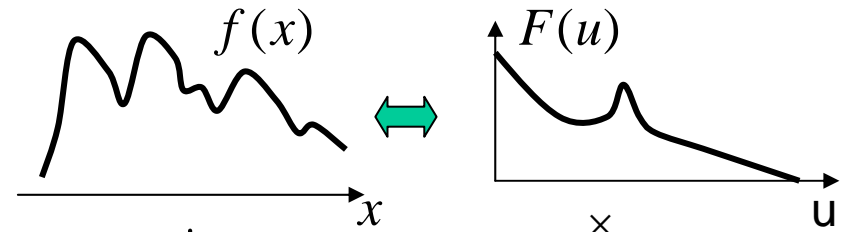
$$g(x) = f_0 h(x) + f_1 h(x-d) + f_2 h(x-2d) + \dots$$



周波数空間で考える

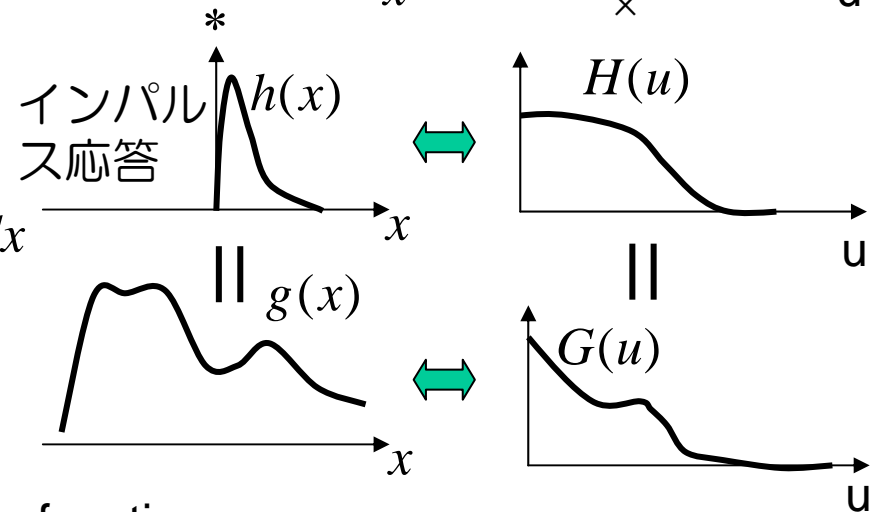
入力信号のスペクトル：

$$F(u) = \int_{-\infty}^{\infty} f(x) \exp(-j2\pi ux) dx$$



出力信号のスペクトル：

$$\begin{aligned} G(u) &= \int_{-\infty}^{\infty} g(x) \exp(-j2\pi ux) dx \\ &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} h(x-\tau) f(\tau) d\tau \right] \exp(-j2\pi ux) dx \\ &= H(u)F(u) \end{aligned}$$



$$H(u) = \frac{G(u)}{F(u)} = \frac{\text{output}}{\text{Input}} \quad \text{： 伝達関数 Transfer function}$$

実空間

$$\begin{aligned} g(x) &= \int_{-\infty}^{\infty} h(x-\tau) f(\tau) d\tau \\ &= h(x) * f(x) \end{aligned}$$

コンボリューション

フーリエ空間

$$G(u) = H(u)F(u)$$

掛け算



Wiener Filter

劣化画像の復元などに用いられる

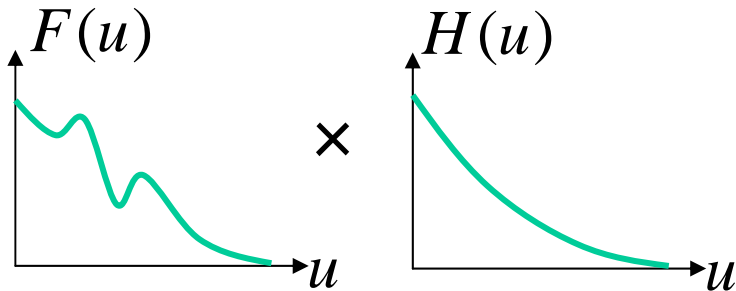
理想画像: $f(x, y)$

劣化の点像分布関数: $h(x, y)$

劣化画像:

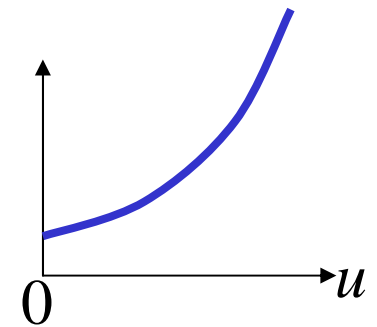
$$g(x, y) = f(x, y) * h(x, y) + n(x, y)$$

$$G(u, v) = F(u, v)H(u, v) + N(u, v)$$



Inverse filter:

$$\frac{1}{H(u, v)}$$



Wiener filter:

$$\frac{1}{H(u, v) + \frac{P_N(u, v)}{P_S(u, v)}}$$

ノイズパワー 信号パワー

