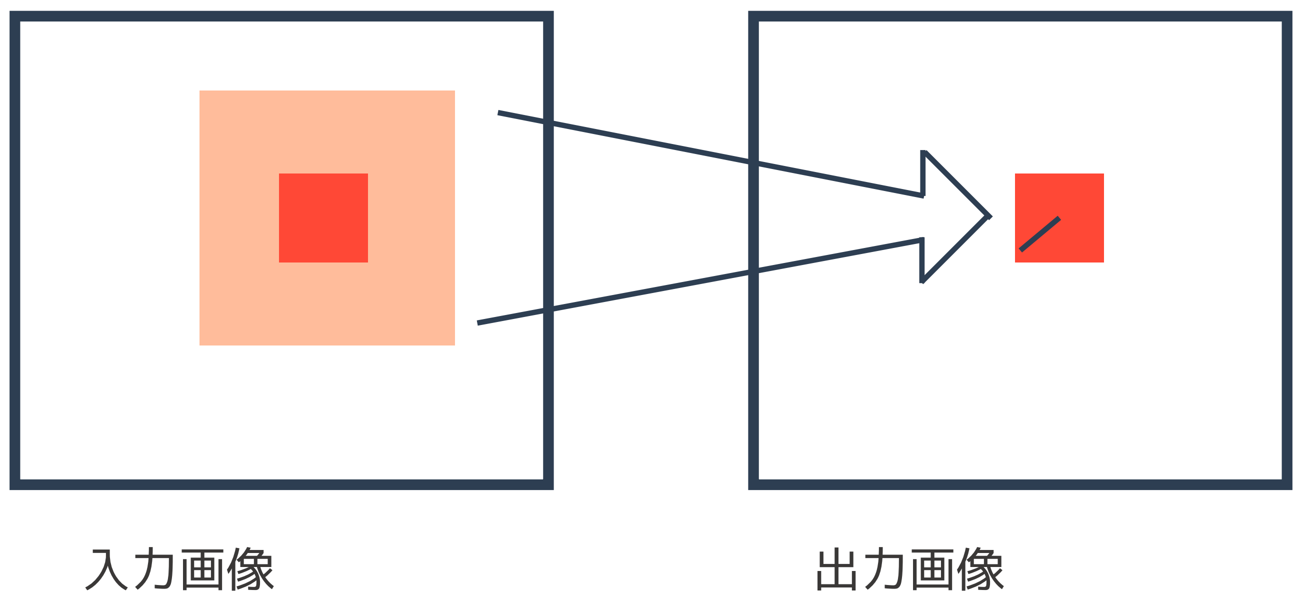
画像工学
空間フィルタ

藤田 一寿

空間フィルタ

- 画像の濃淡変換をするとき，対象の画素の画素値だけではなくその周囲の画素の画素値を考慮し変換する処理を空間フィルタリングと言い、その変換のルールを空間フィルタという。



線形フィルタ

- どのように値を変えるのか
- 積和の形で変換しよう（線形フィルタ）

積和の形でかけないもの
を非線形フィルタという

$$g(i,j) = \sum_{n=-w}^{w} \sum_{m=-w}^{w} f(i+m, j+n)h(m,n)$$

出力画像の
画素値

入力画像の
画素値

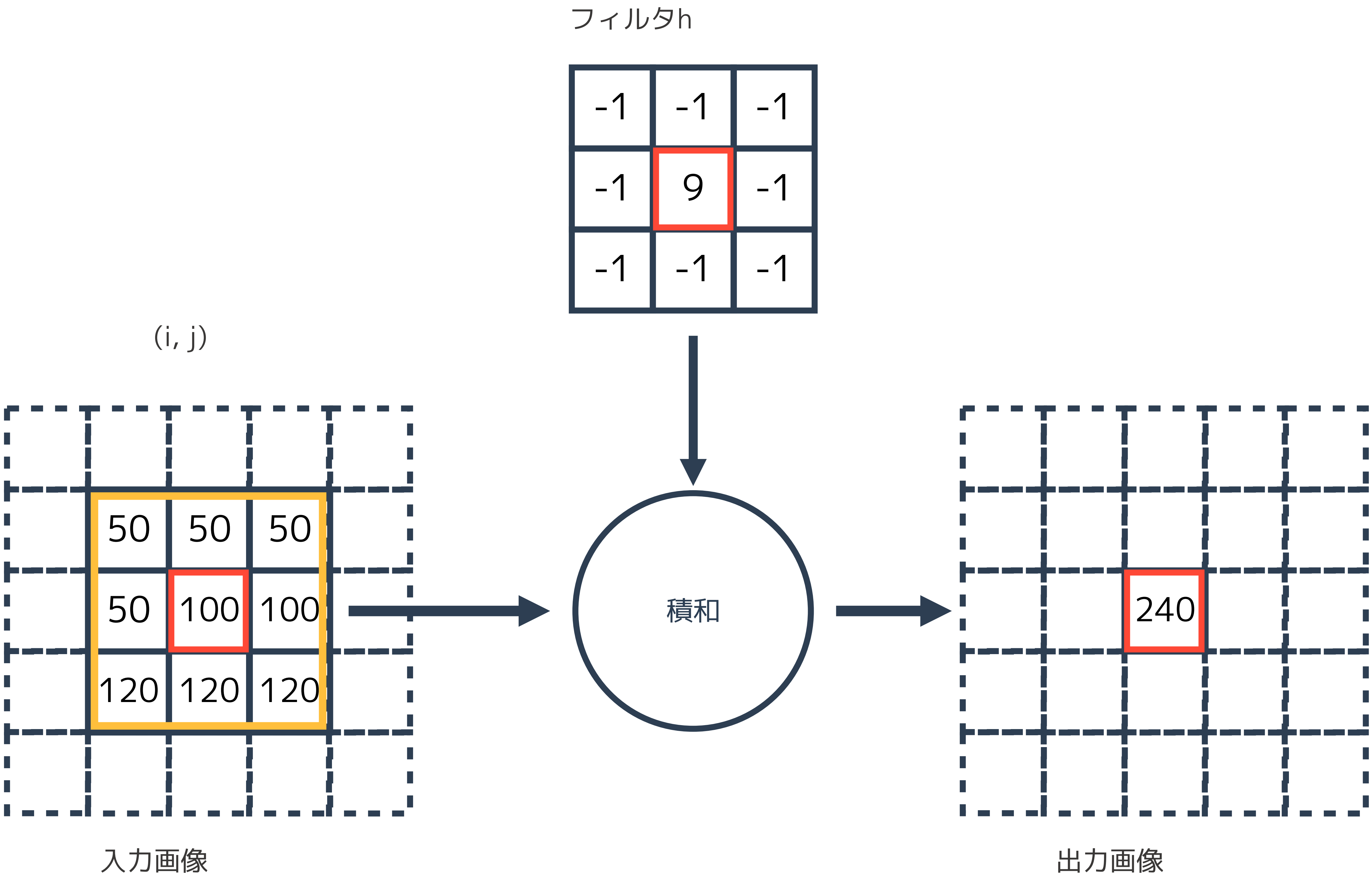
フィルタ
(カーネル)

フィルタのサイズは(2w+1)x(2w+1)

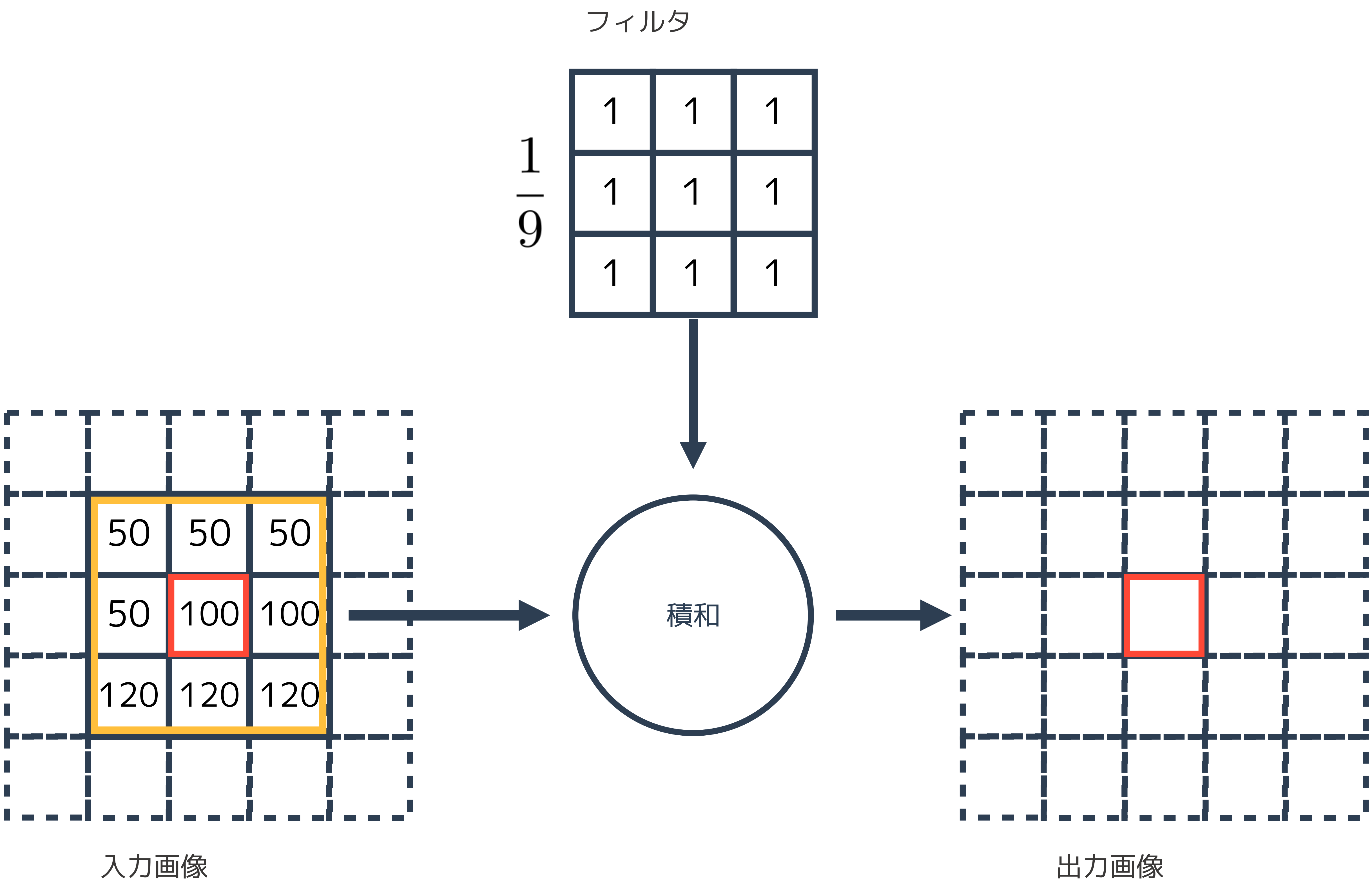
また、フィルタリングは畳み込み計算と同等

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

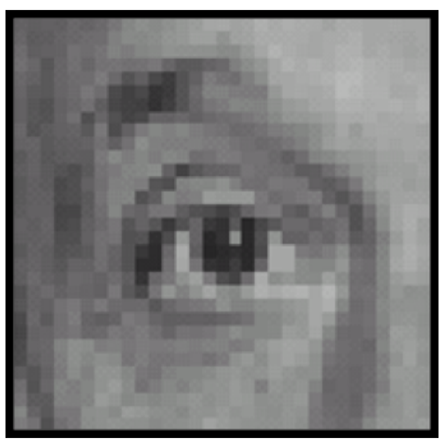
$$g(i,j) = \sum_{n=-w}^{w} \sum_{m=-w}^{w} f(i+m, j+n) h(m,n)$$



みんなも計算してみよう

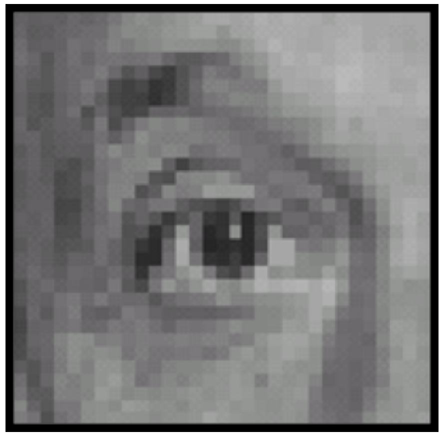


フィルタの例



$$* \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad =$$

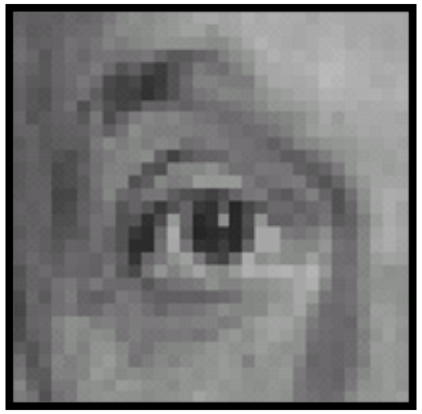
フィルタの例



*

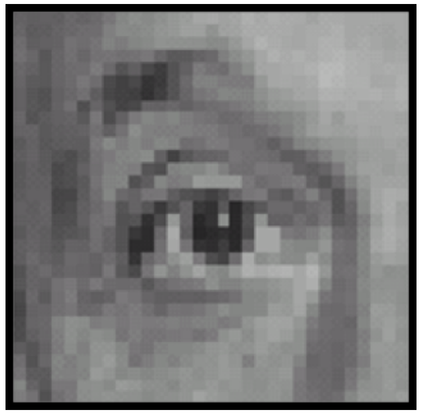
0	0	0
0	1	0
0	0	0

=



変化なし

フィルタの例

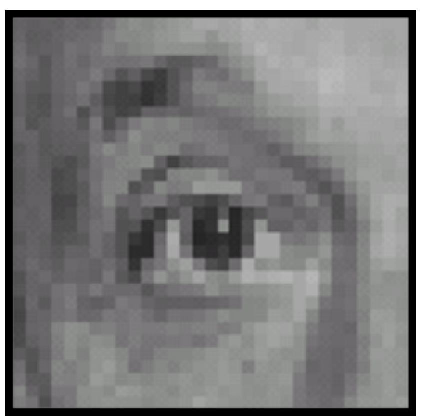


*

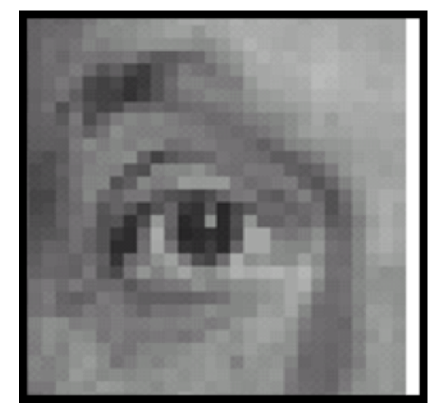
0	0	0
1	0	0
0	0	0

=

フィルタの例

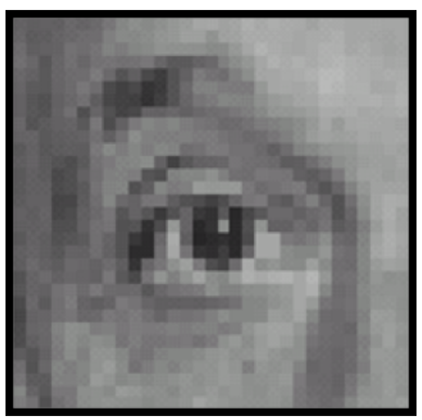


$$
* \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} =
$$



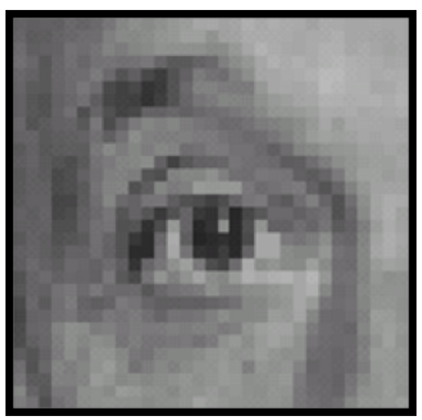
1ピクセルシフト

フィルタの例

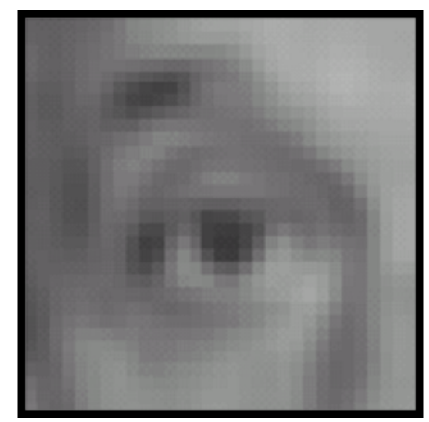


$$* \quad \frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} \quad =$$

フィルタの例

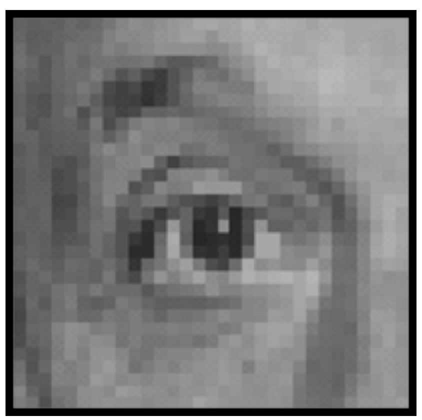


$$* \quad \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad =$$



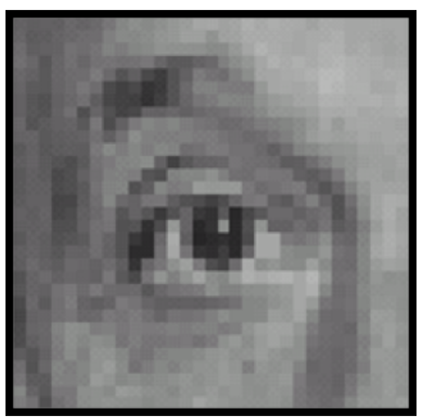
ぼやける

フィルタの例



$$* \left(\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 2 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} \; -1/9 \; \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \right) =$$

フィルタの例



$$* \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right) =$$

シャープになる

平滑化

- 輝度の凸凹をなくし、なめらかにすることを平滑化という。
- 平滑化することで画像はぼけてしまうので、平滑化のことをぼかしとも言う。
- 画像がぼけるとは、画像の高周波数成分が除去されたとも考えられる。この事については、画像のフーリエ変換で詳しく取り扱う。
- 結果的に、高周波ノイズを除去するフィルタとして用いることもできるが、ノイズ以外にも画像のdetailもボケてしまう。

平均化フィルタ

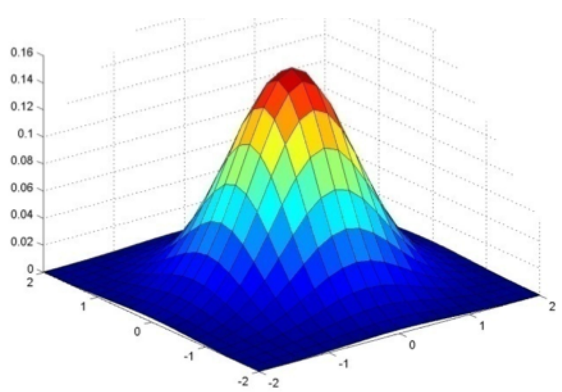
$$* \quad \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} \quad =$$

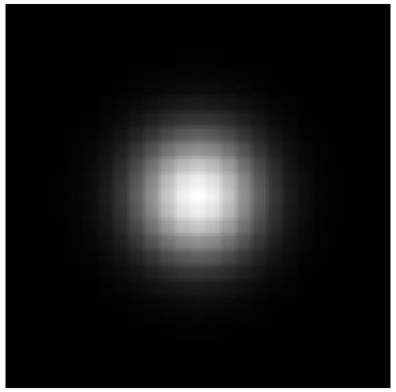
Gaussian filter

- 確率分布の一つであるガウス分布の式を用い平滑化を行うフィルタのことをGaussian filterという。

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

等方ガウス分布

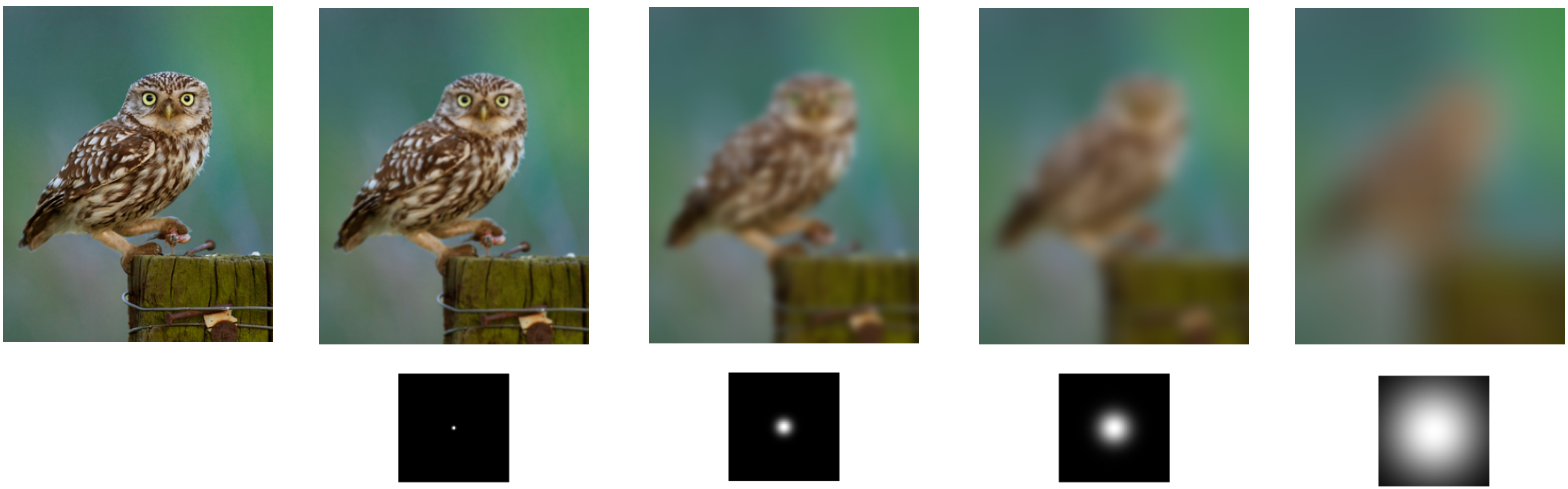




$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$
$\frac{2}{16}$	$\frac{4}{16}$	$\frac{2}{16}$
$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$

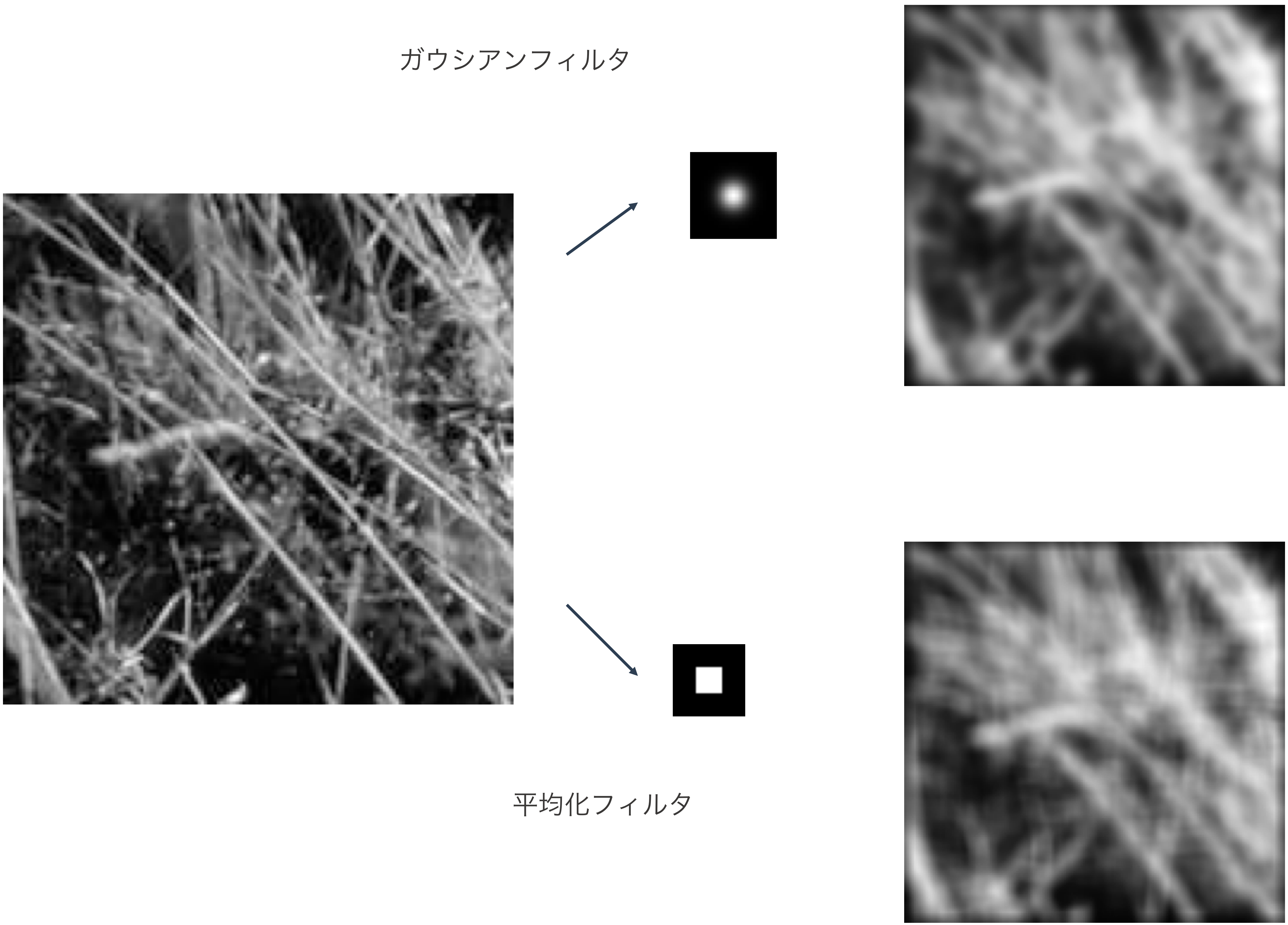
もちろん等方ガウス分布ではなく通常のガウス分布を用いることも可能である。その場合は、フィルタの性質が異なってくる。どのような性質か余裕があれば考えてみよ。

標準偏差とボケ具合



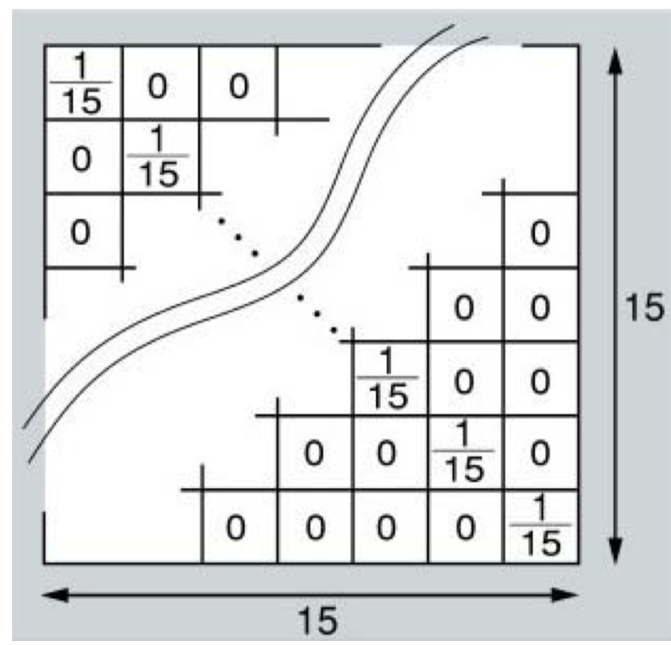
σ = 1 pixel　σ = 5 pixels　σ = 10 pixels　σ = 30 pixels

ガウシアンフィルタと平均化フィルタの比較



特定方向の平滑化

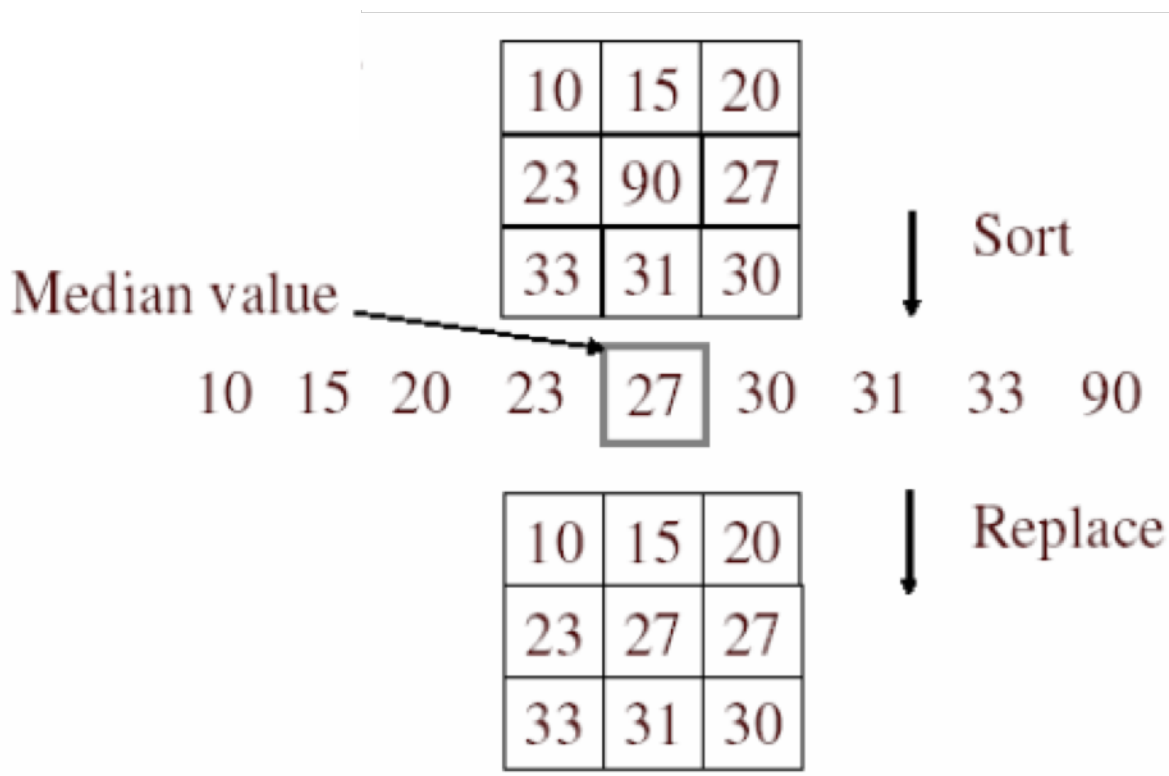
*



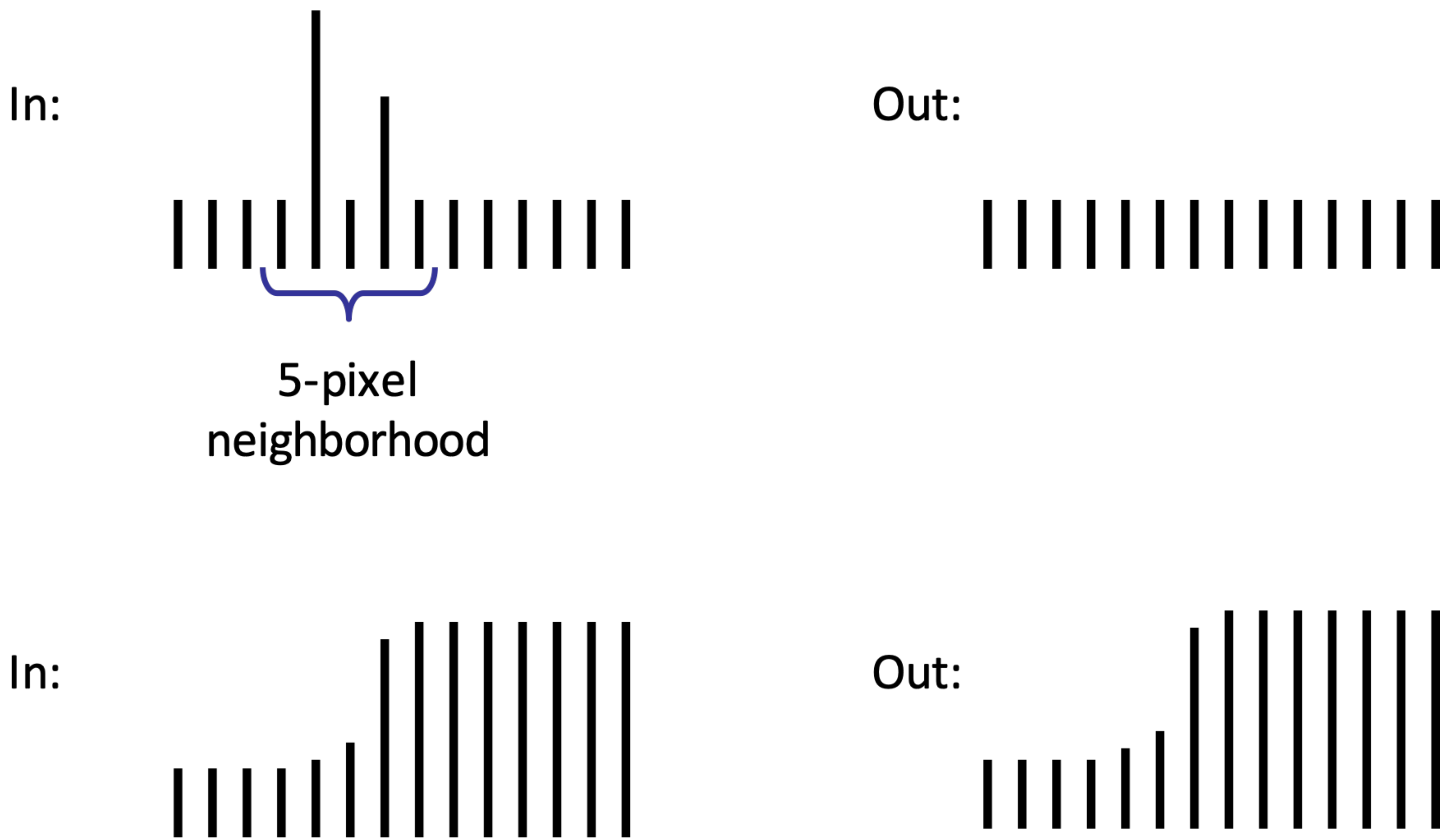
=

Median filter

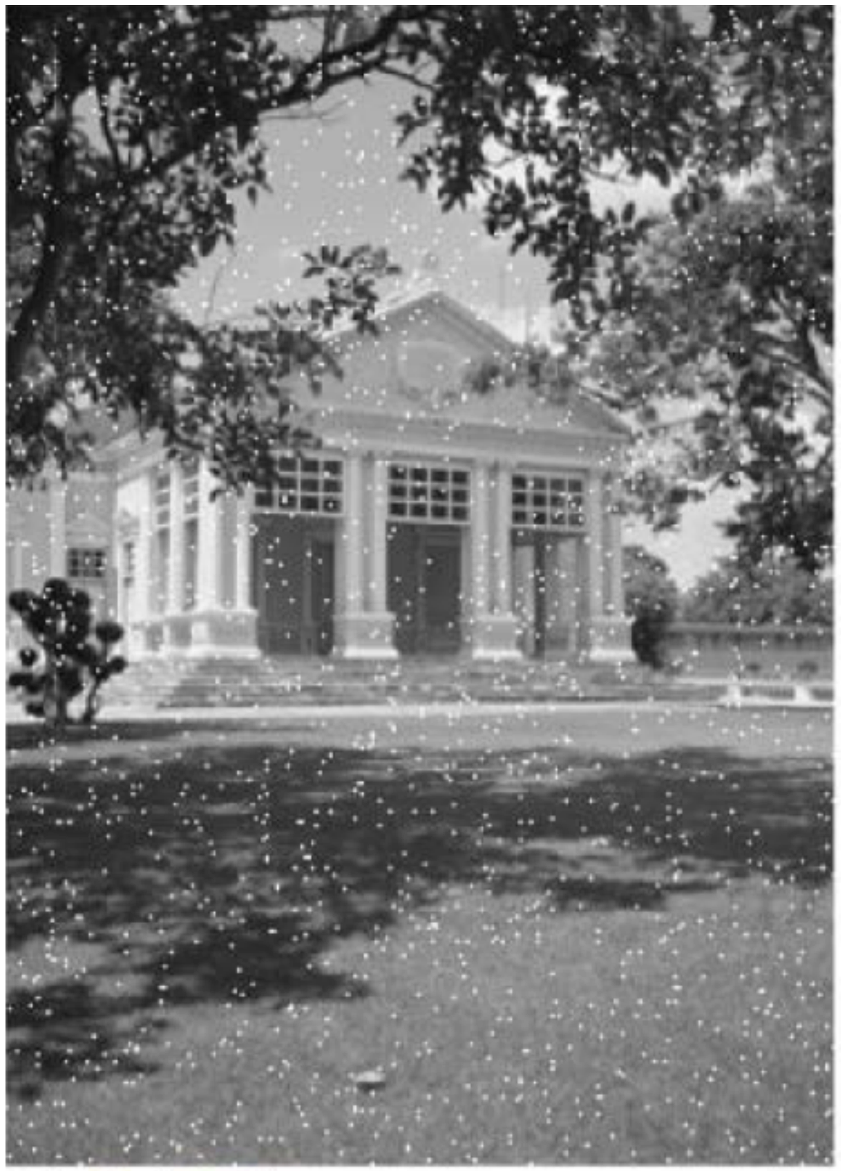
- 非線形フィルタの一種
- ごま塩ノイズ(スパイクノイズ)を除去するのが得意
- このフィルタは、任意のピクセルの輝度値を決定する際、ある領域の画素値の中央値を新たな輝度値として採用するものである。



スパイクノイズの除去の様子



メディアンフィルタの適用例



[a] 入力画像

[b] メディアンフィルタの適用結果

[c] 平均化フィルタの適用結果

メディアンフィルタとガウシアンフィルタの比較

3x3 5x5 7x7

Gaussian

Median

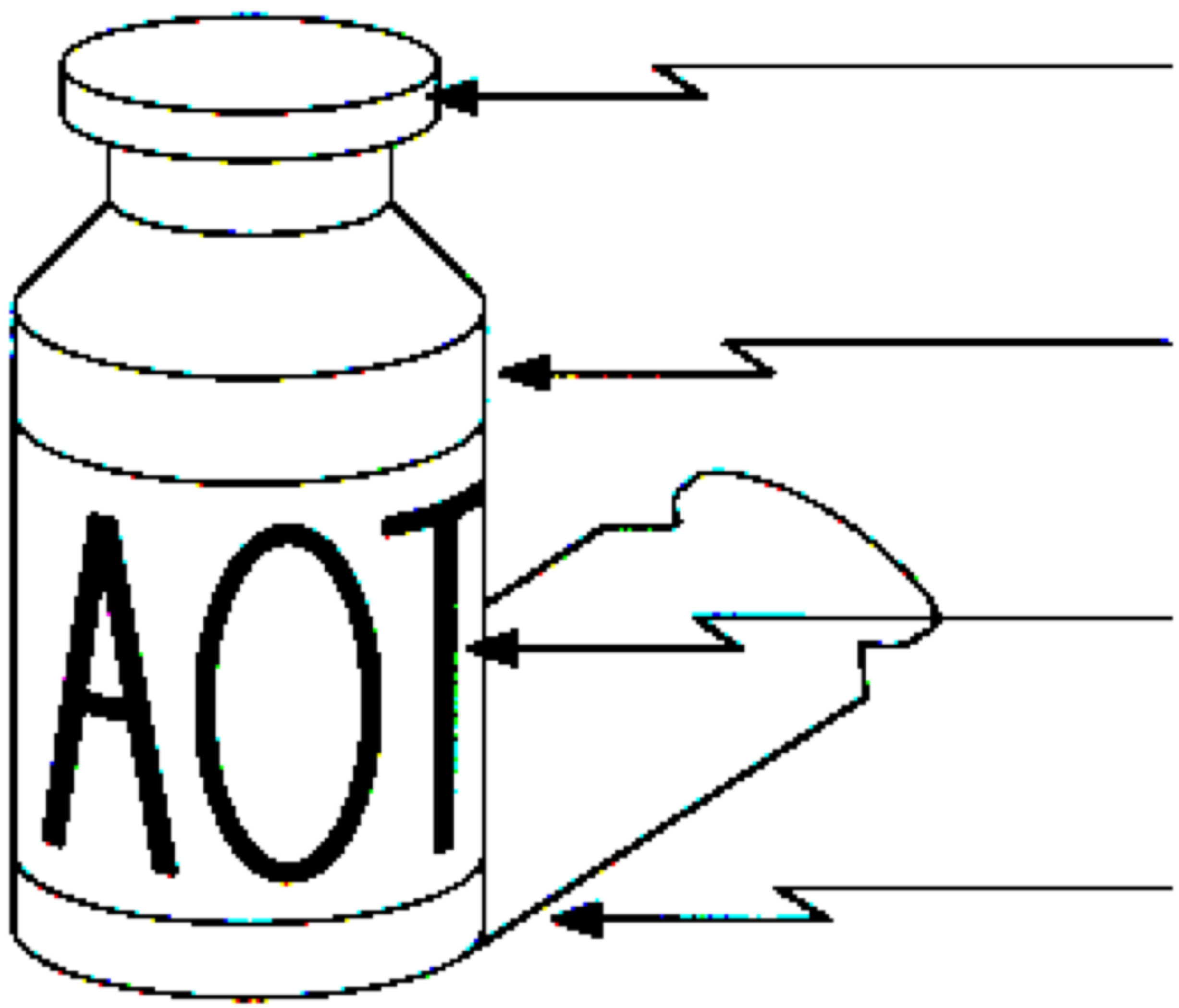
エッジ抽出

- エッジとは
  - 輝度値が急激に変わる場所、変化が不連続な場所
- エッジ抽出とは
  - 輝度値が急激に変わる場所を抽出する

エッジ抽出

白黒反転

エッジの生じる場所



表面の不連続

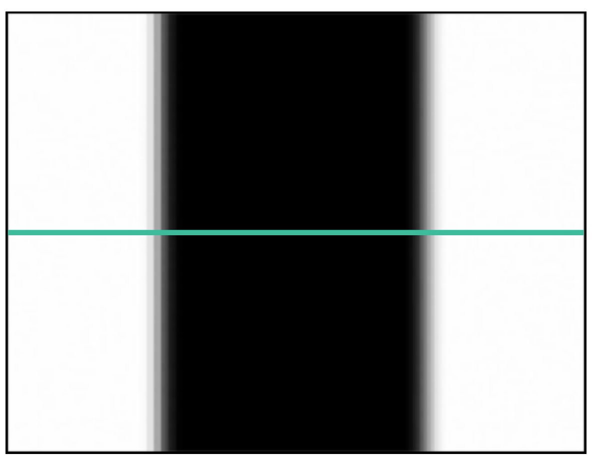
深度の不連続

色の不連続

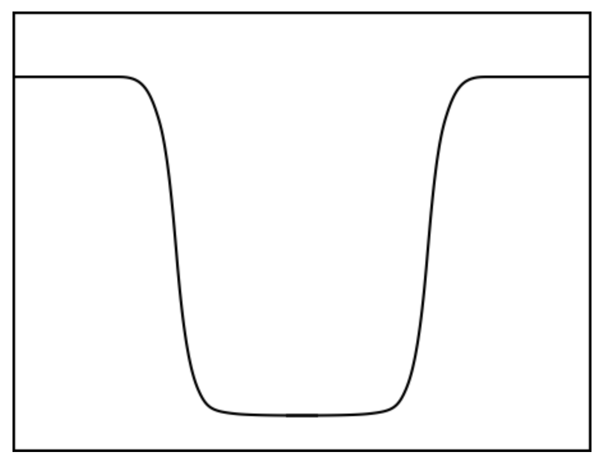
光の不連続

Source: Steve Seitz

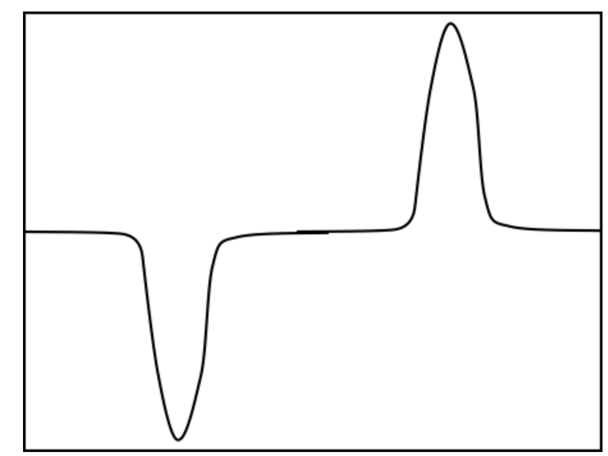
エッジと微分



輝度値



1階微分



極値がエッジに対応する

微分をどう計算するか

- 画像は不連続なため、どのように微分を計算するか？

$$\frac{\partial F[x,y]}{\partial x} \approx F(x+1,y) - F(x,y)$$

- 隣の値の差分で計算しよう！！
- この差分をフィルタで表すと

-1	1

微分フィルタ（横）

0	0	0
0	−1	1
0	0	0

[a]

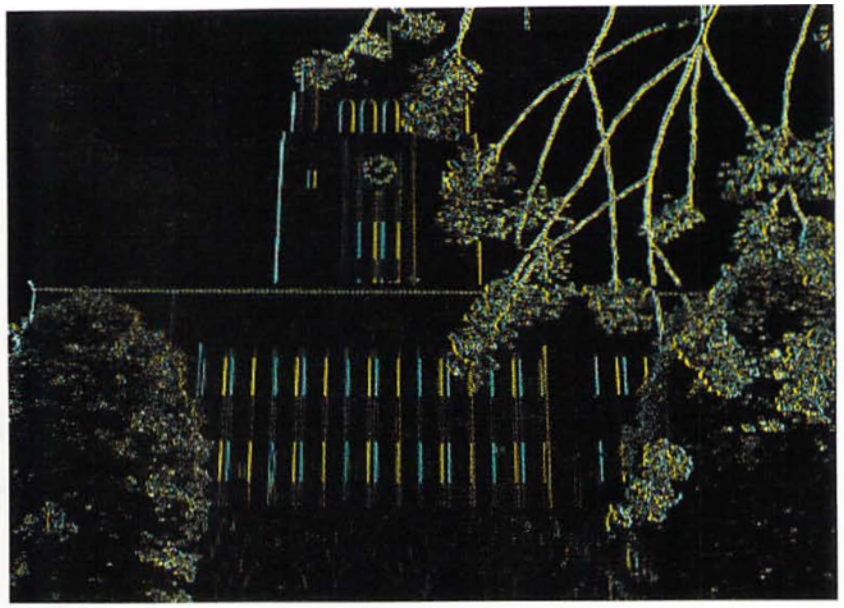
0	0	0
−1	1	0
0	0	0

[b]

0	0	0
$-\frac{1}{2}$	0	$\frac{1}{2}$
0	0	0

[c]

■図5.11——入力画像



■図5.12——微分フィルタ（横方向）の結果

微分フィルタ（縦）

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

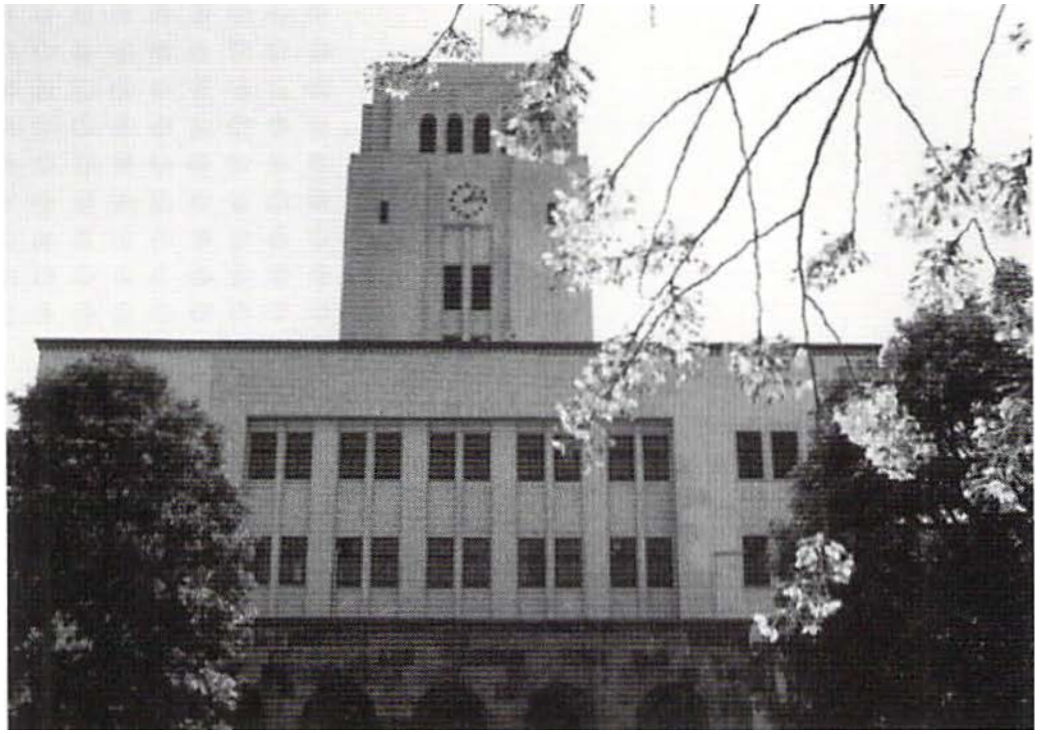
[a]

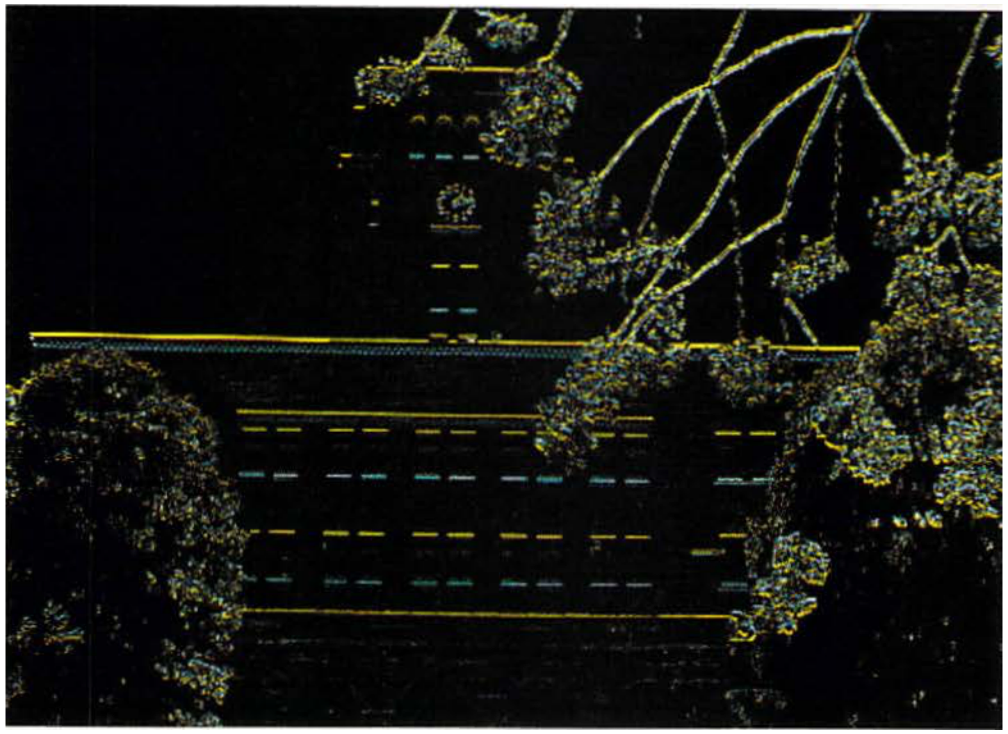
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

[b]

$$\begin{bmatrix} 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \\ 0 & -\frac{1}{2} & 0 \end{bmatrix}$$

[c]

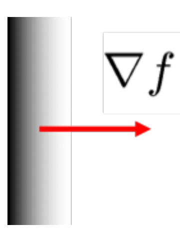




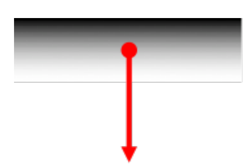
勾配

画像は２次元でありながら、先程までは１次元だと思って微分していた（偏微分）。本来は２次元で考えなければならない。２次元の輝度値の変化を取り扱うには勾配を計算する必要がある。

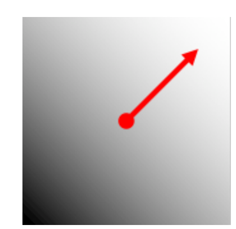
$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$



$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$



$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$



$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$

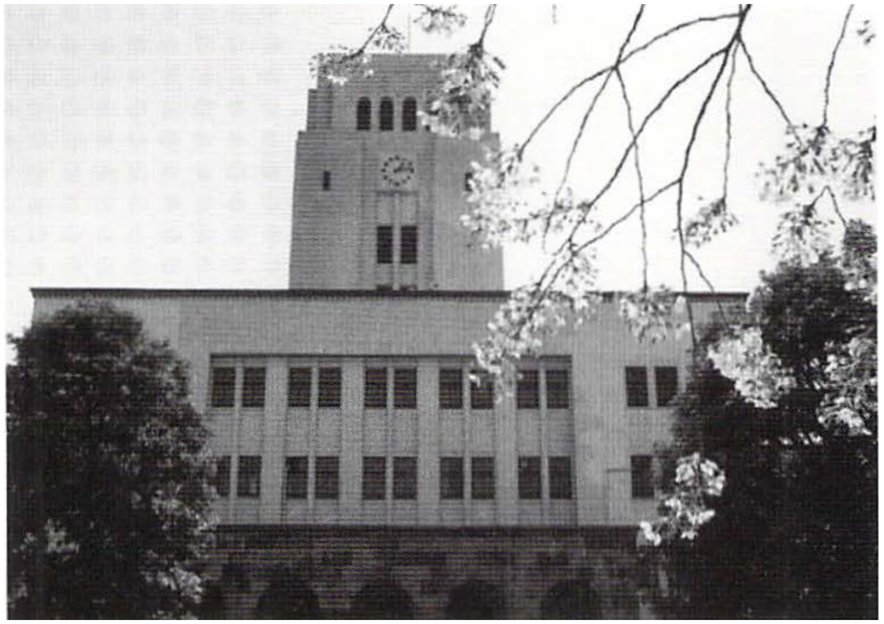
勾配の大きさと向き

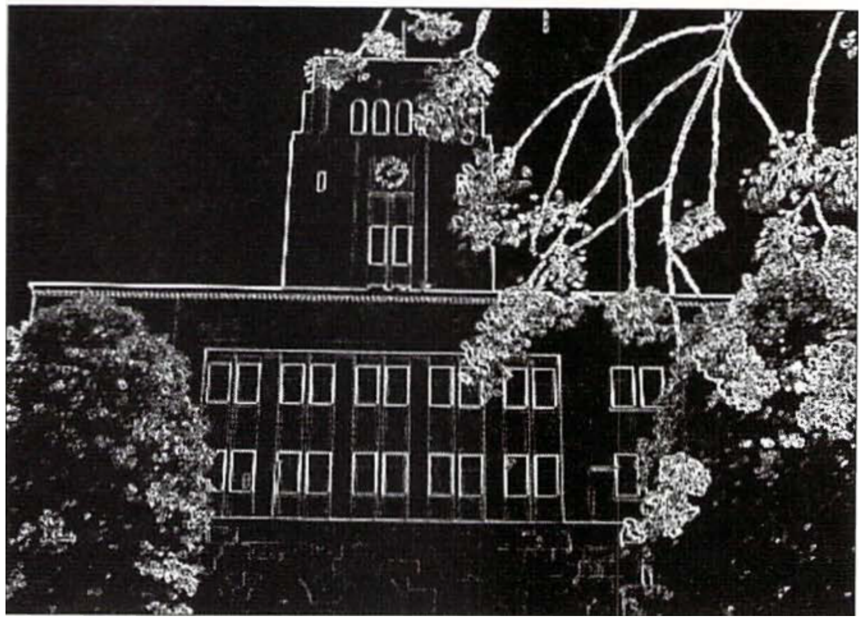
勾配の大きさ

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

勾配の向き

$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

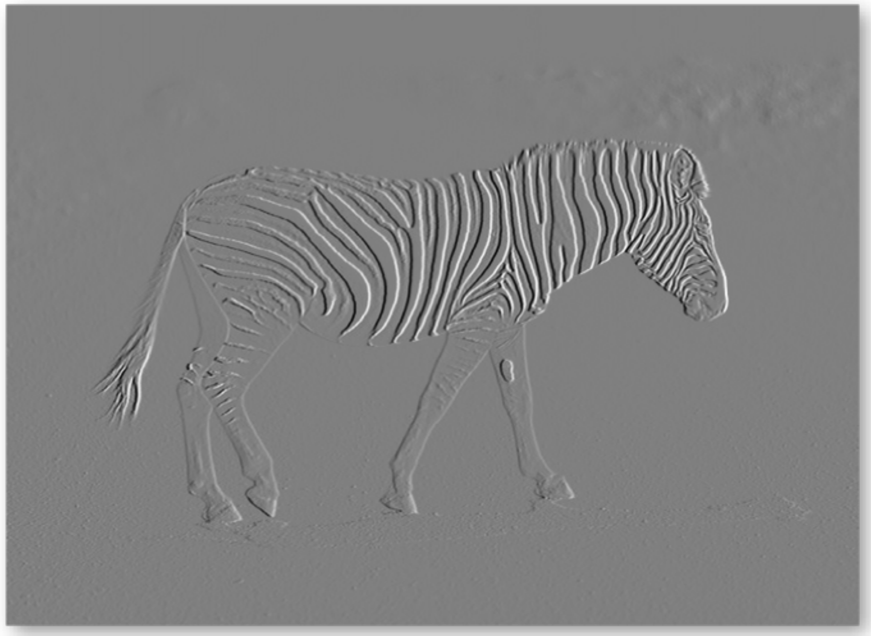




$$\|\nabla f\| = \sqrt{(\frac{\partial f}{\partial x})^2 + (\frac{\partial f}{\partial y})^2}$$

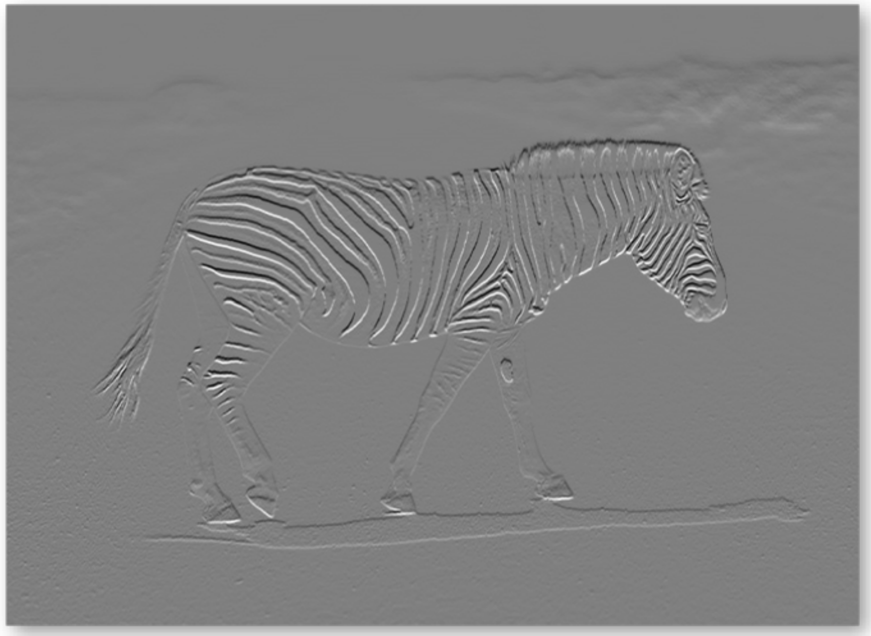
勾配計算の例

f



$\frac{\partial f}{\partial x}$

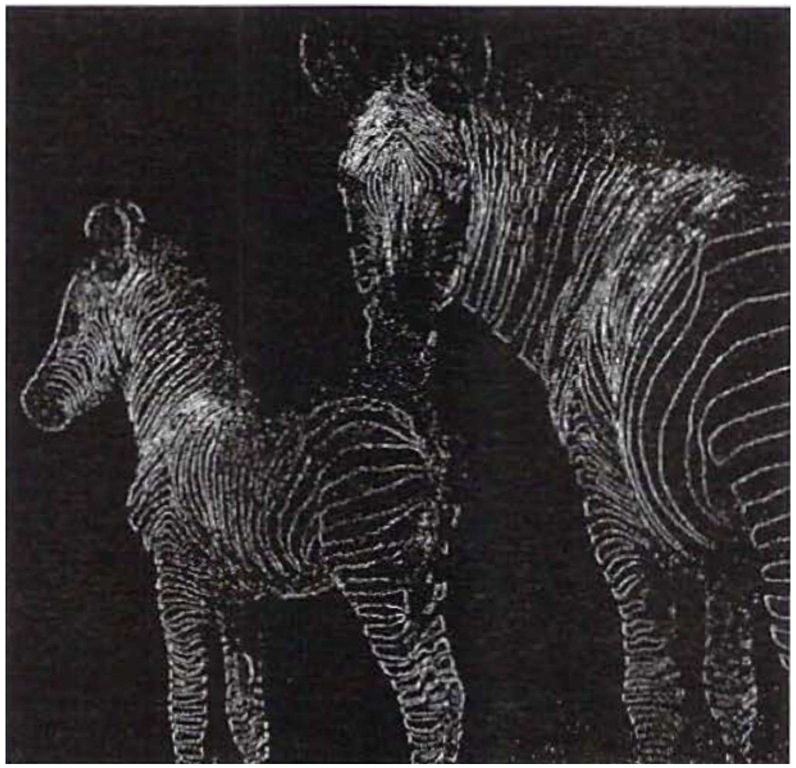
$\|\nabla f\| = \sqrt{(\frac{\partial f}{\partial x})^2 + (\frac{\partial f}{\partial y})^2}$



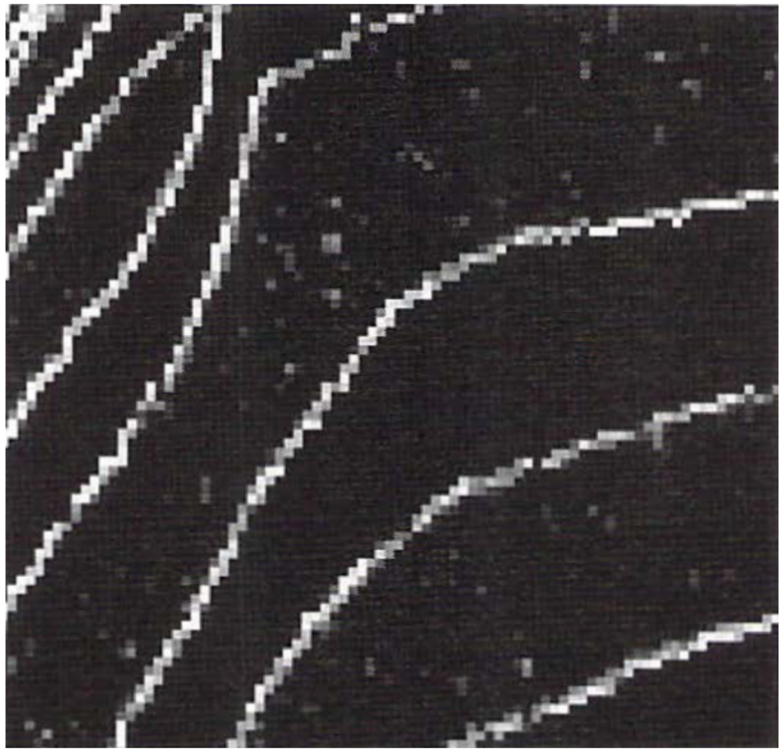
$\frac{\partial f}{\partial y}$

微分フィルタの問題

微分フィルタは変化が大きい場所を抽出する.
ノイズがある場所は変化が大きいため，微分フィルタはエッジだけではなくノイズも強調してしまう.

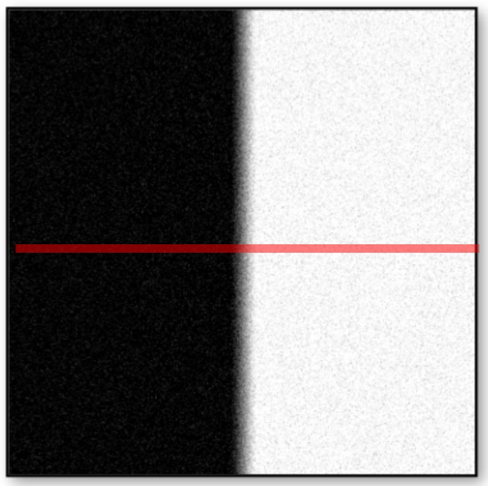


微分フィルタ



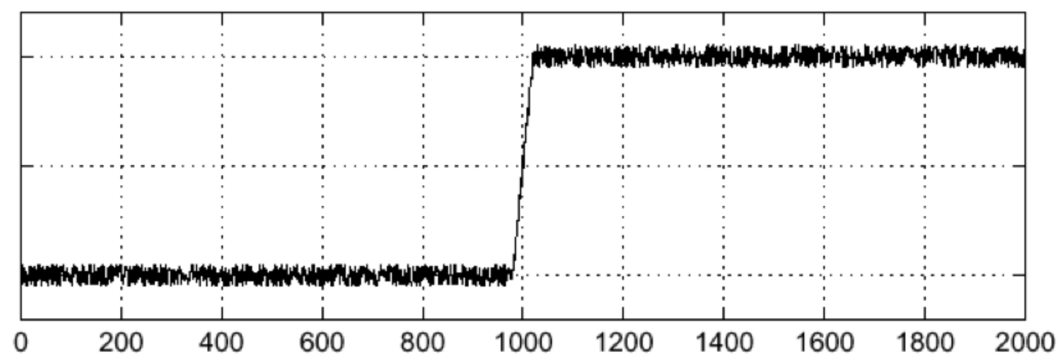
拡大図

ノイズへの微分フィルタリング

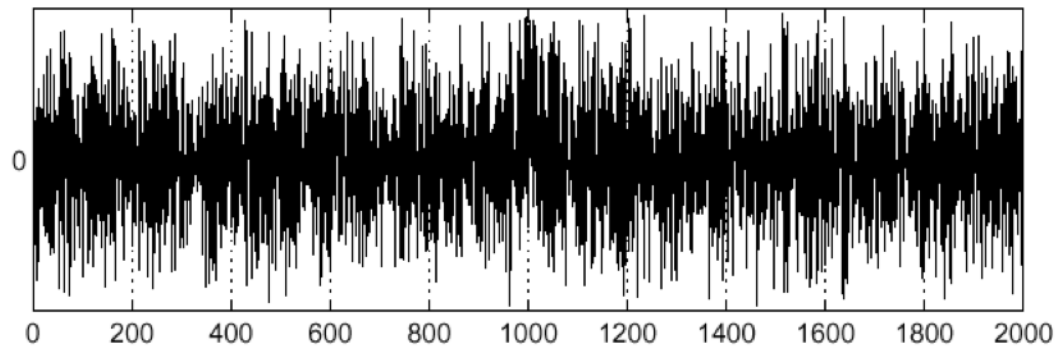


ノイズを含む画像

輝度値

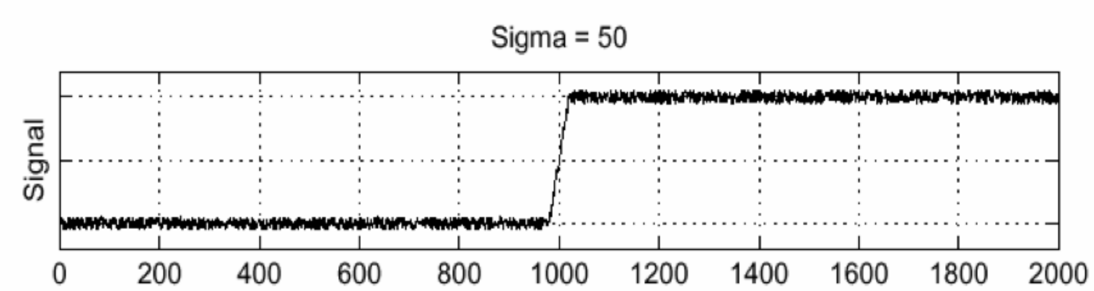


微分



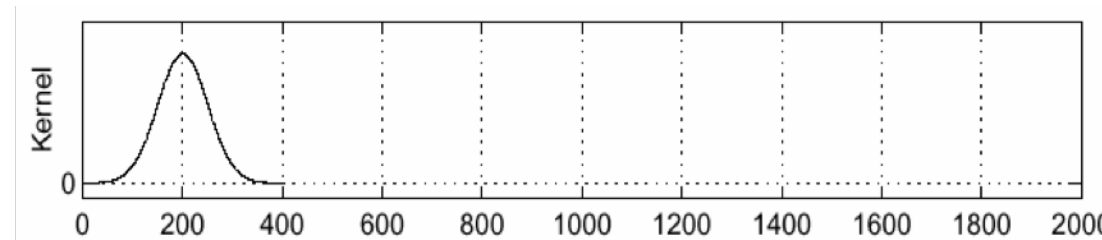
ノイズがあってもエッジ抽出したい

ノイズを含む画像



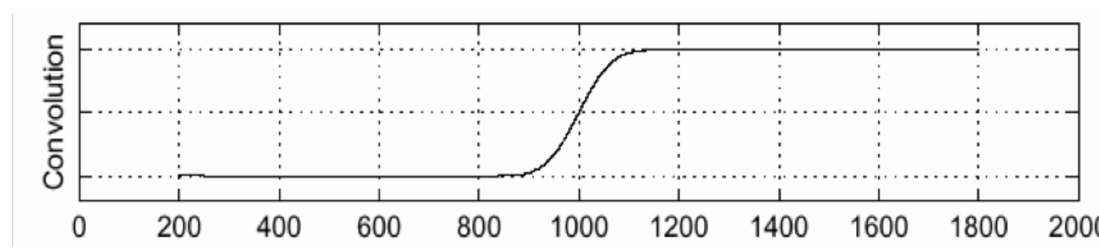
ガウシアンフィルタ

h



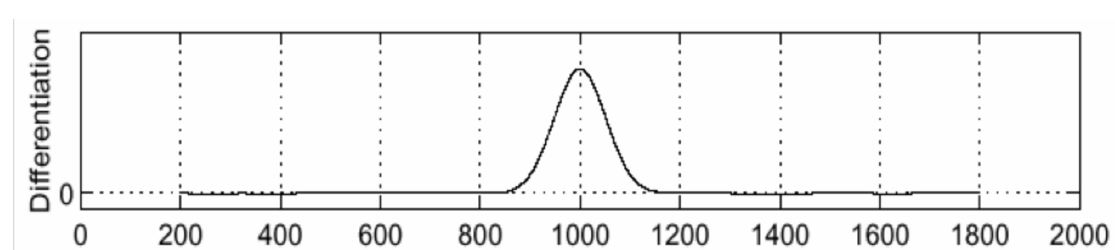
平滑化された画像

$f * h$



微分

$$\frac{d}{dx}(f * h)$$



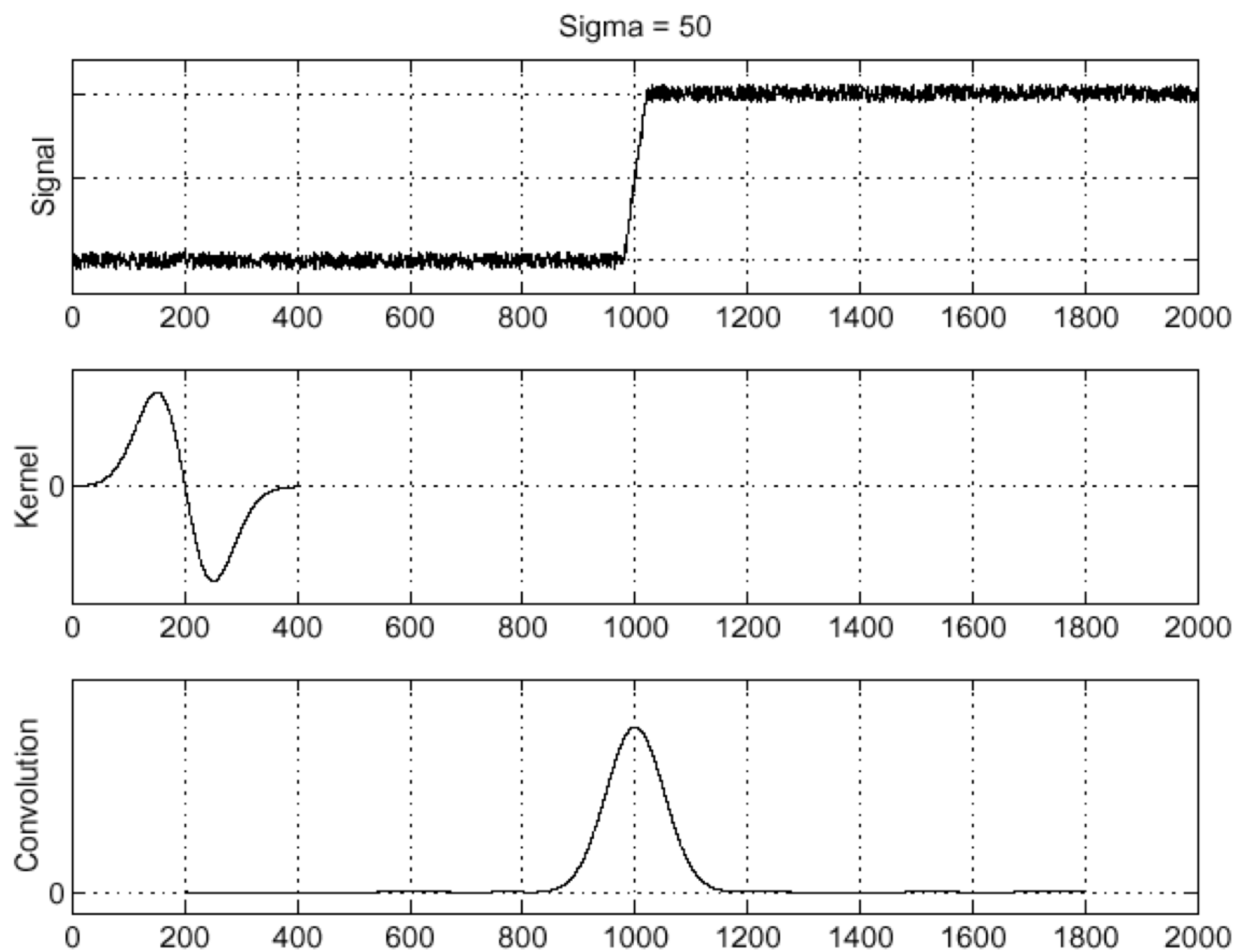
ガウシアンフィルタを微分しても同じ

$$\frac{d}{dx}(f*g)=f*\frac{d}{dx}g$$

ノイズを含む画像 f

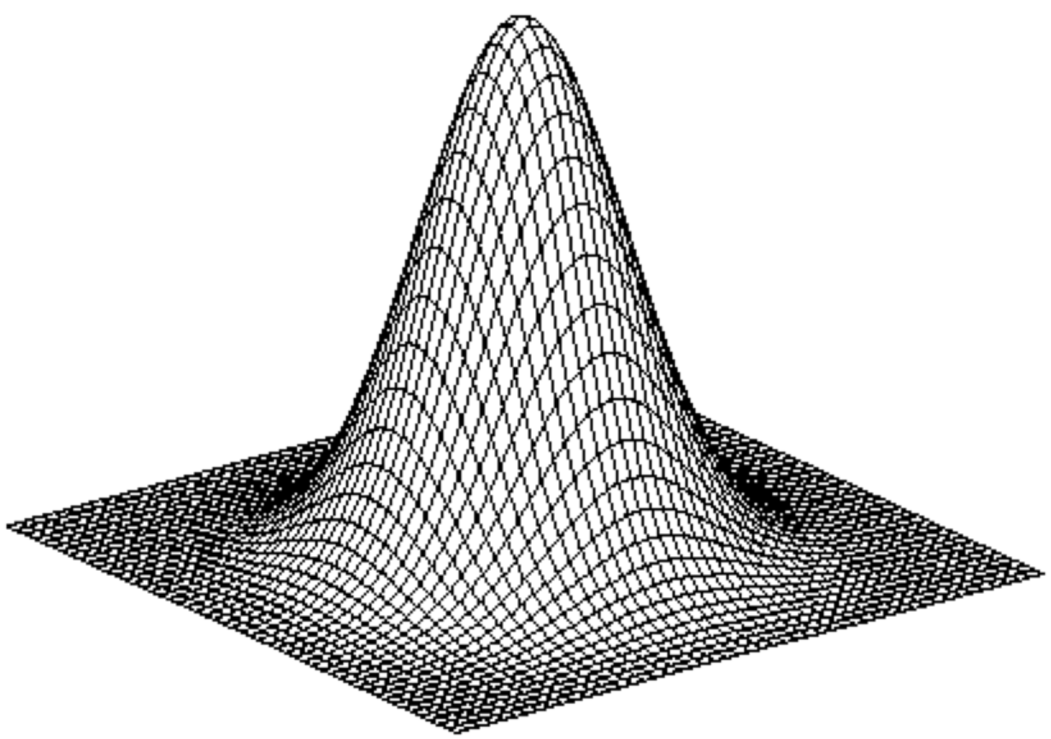
微分ガウシアンフィルタ $\frac{d}{dx}g$

フィルタリング結果 $f*\frac{d}{dx}g$

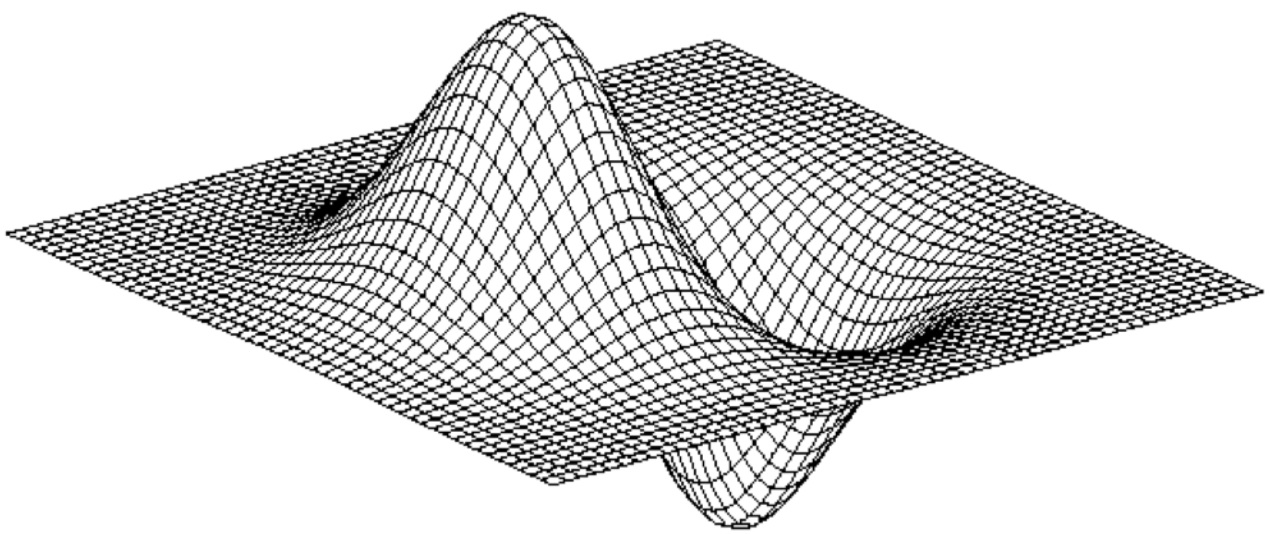


(注)畳み込み計算をしている

微分ガウシアン

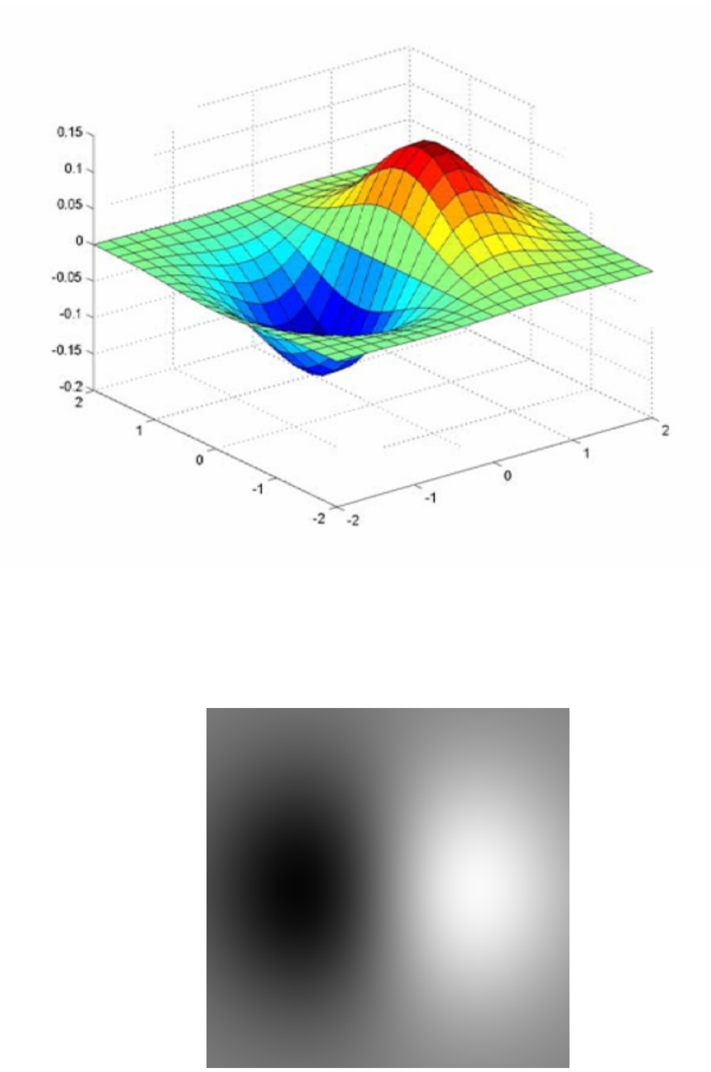


ガウシアン

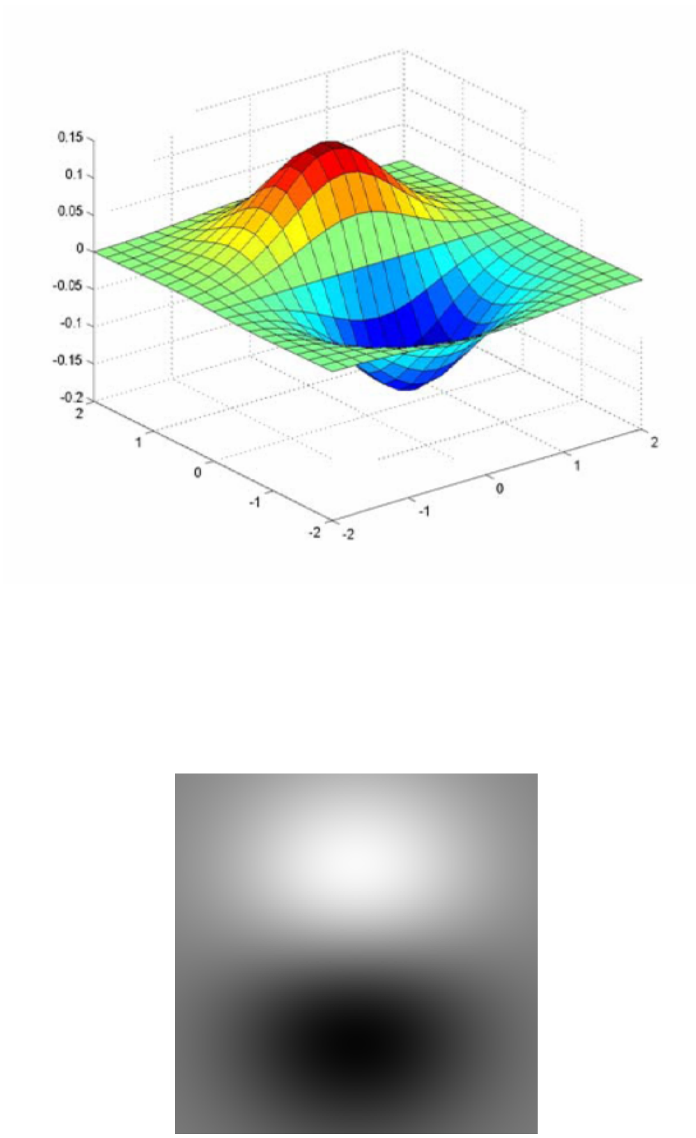


微分ガウシアン

微分ガウシアンフィルタ



微分ガウシアンフィルタ(横)



微分ガウシアンフィルタ(縦)

ソーベルフィルタ

- ソーベルフィルタは微分ガウシアンフィルタの近似的なフィルタ

$$\frac{1}{8}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

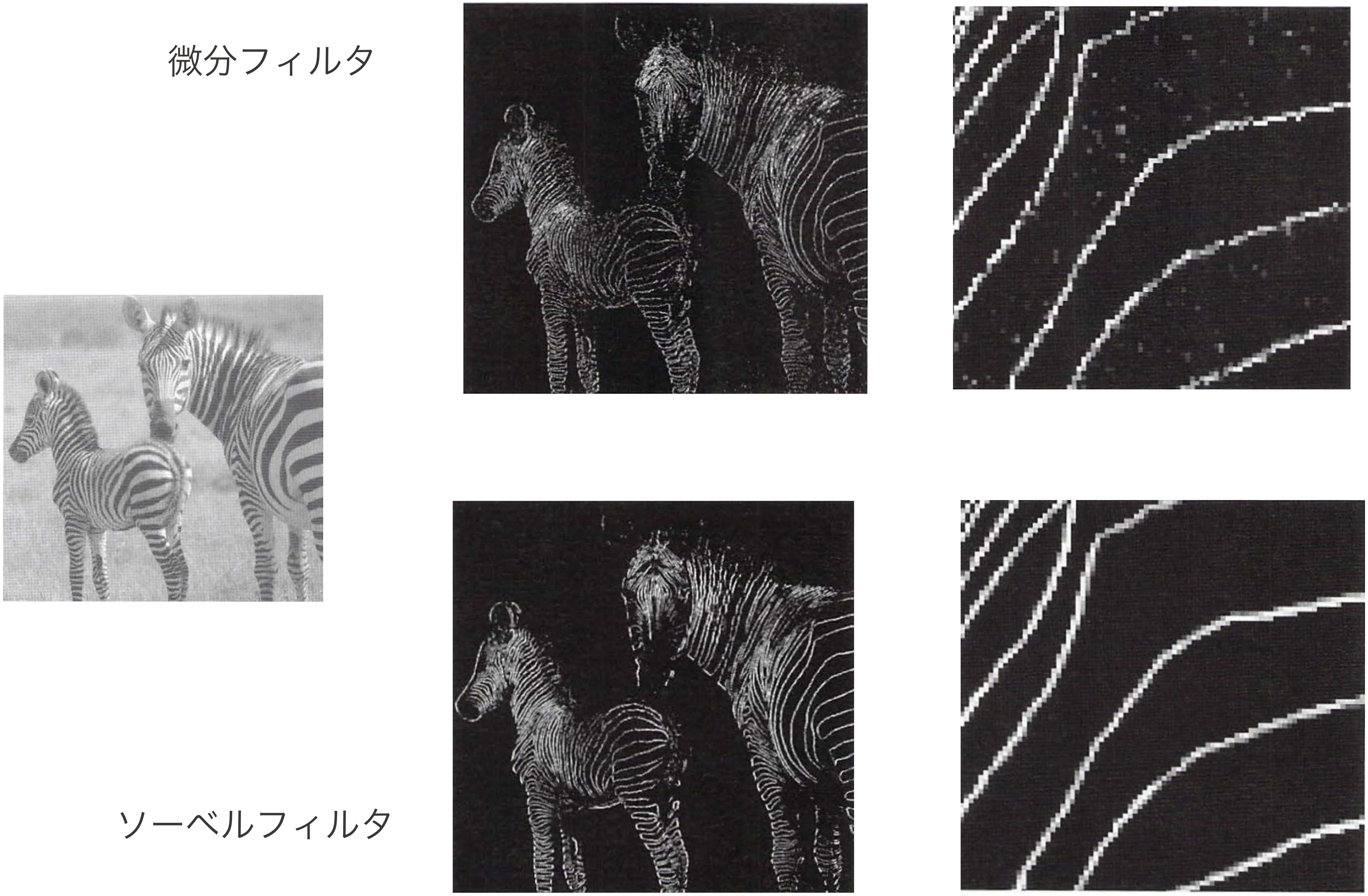
横方向

$$\frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

縦方向

ソーベルフィルタの例

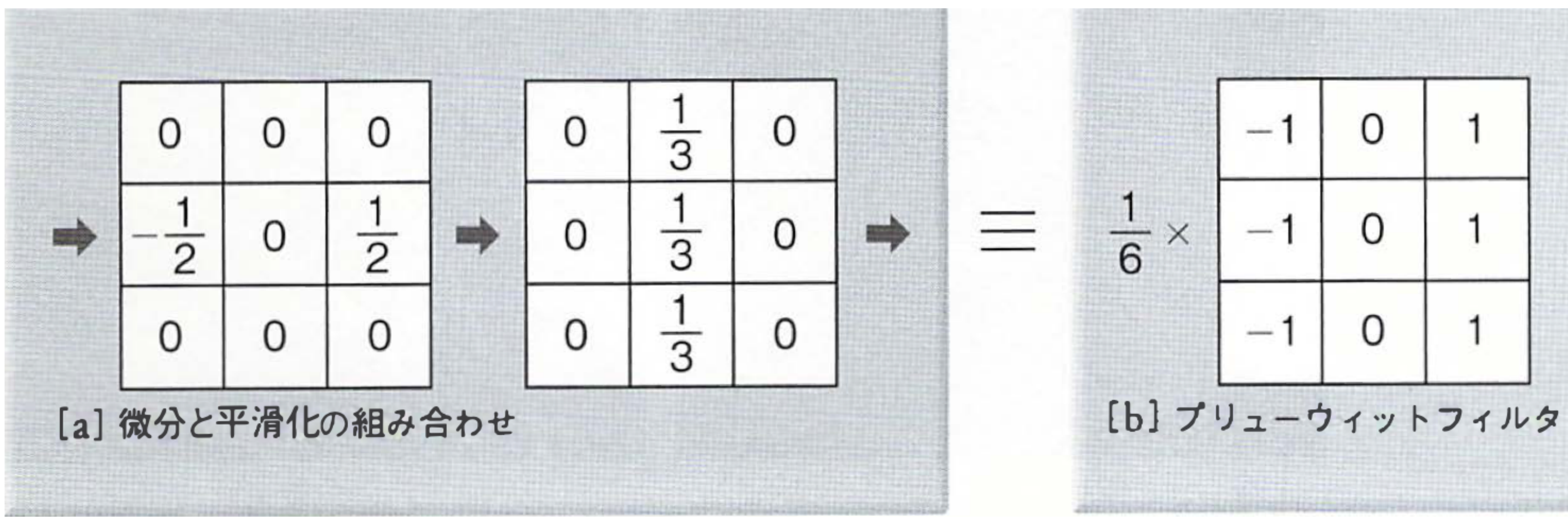
微分フィルタ



ソーベルフィルタ

Prewitt filter

微分フィルタ＋平均化フィルタ



ラプラシアンフィルタ

ラプラシアンを思い出そう

$$\nabla^2 = \nabla \cdot \nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

∇の内積でした（2階微分）.

$$\frac{\partial^2 f}{\partial x^2} = f(i, j+1) - 2f(i, j) + f(i, j-1)$$

$$\frac{\partial^2 f}{\partial y^2} = f(i+1, j) - 2f(i, j) + f(i-1, j)$$

$$\nabla^2 f = -4f(i, j) + f(i, j+1) + f(i, j-1) + f(i+1, j) + f(i-1, j)$$

2階微分の近似式

ラプラシアンフィルタ

$$\frac{\partial^2 f}{\partial x^2} = f(i, j+1) - 2f(i, j) + f(i, j-1)$$

$$\frac{\partial^2 f}{\partial y^2} = f(i+1, j) - 2f(i, j) + f(i-1, j)$$

$$\nabla^2 f = -4f(i, j) + f(i, j+1) + f(i, j-1) + f(i+1, j) + f(i-1, j)$$

0	0	0
1	−2	1
0	0	0

2次微分フィルタ(横)

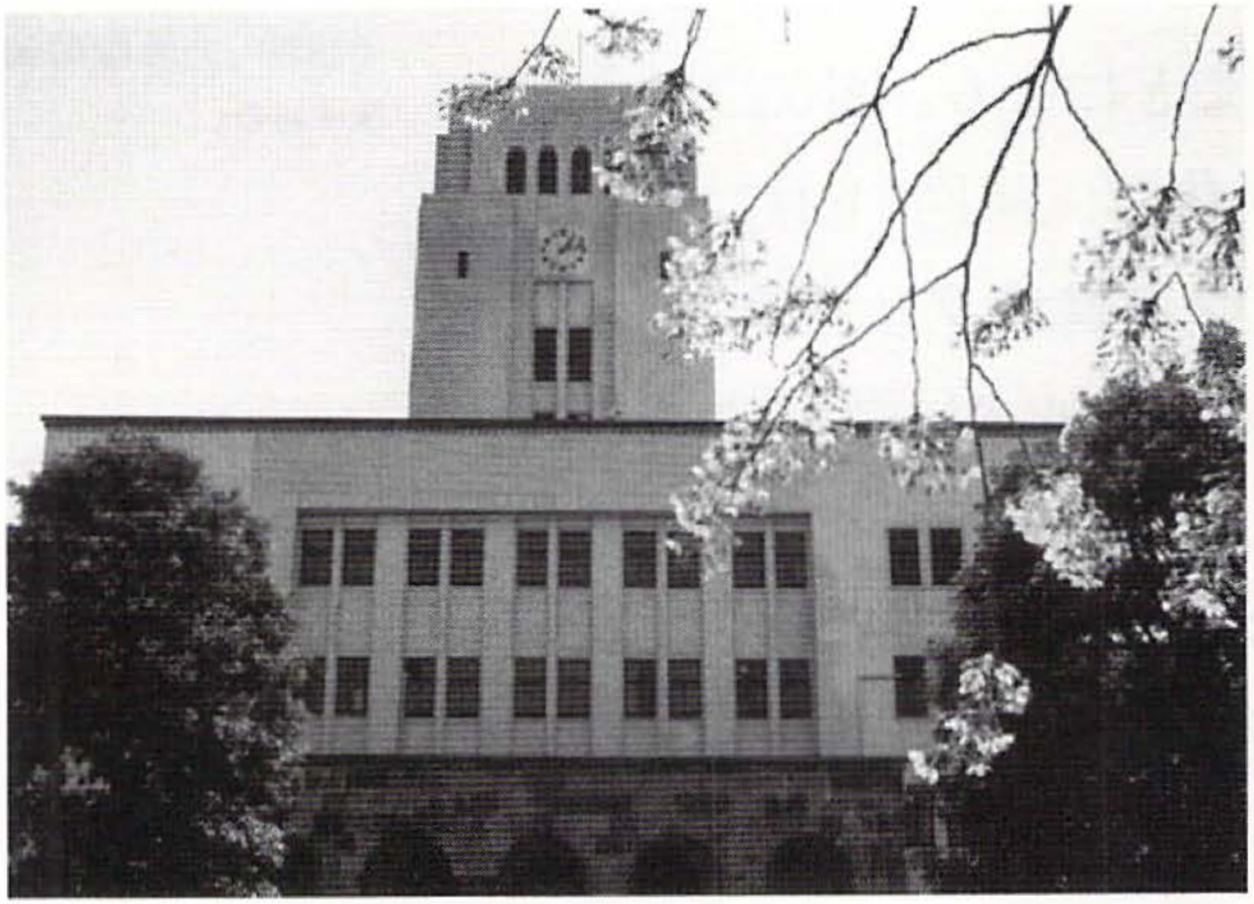
0	1	0
0	−2	0
0	1	0

2次微分フィルタ(縦)

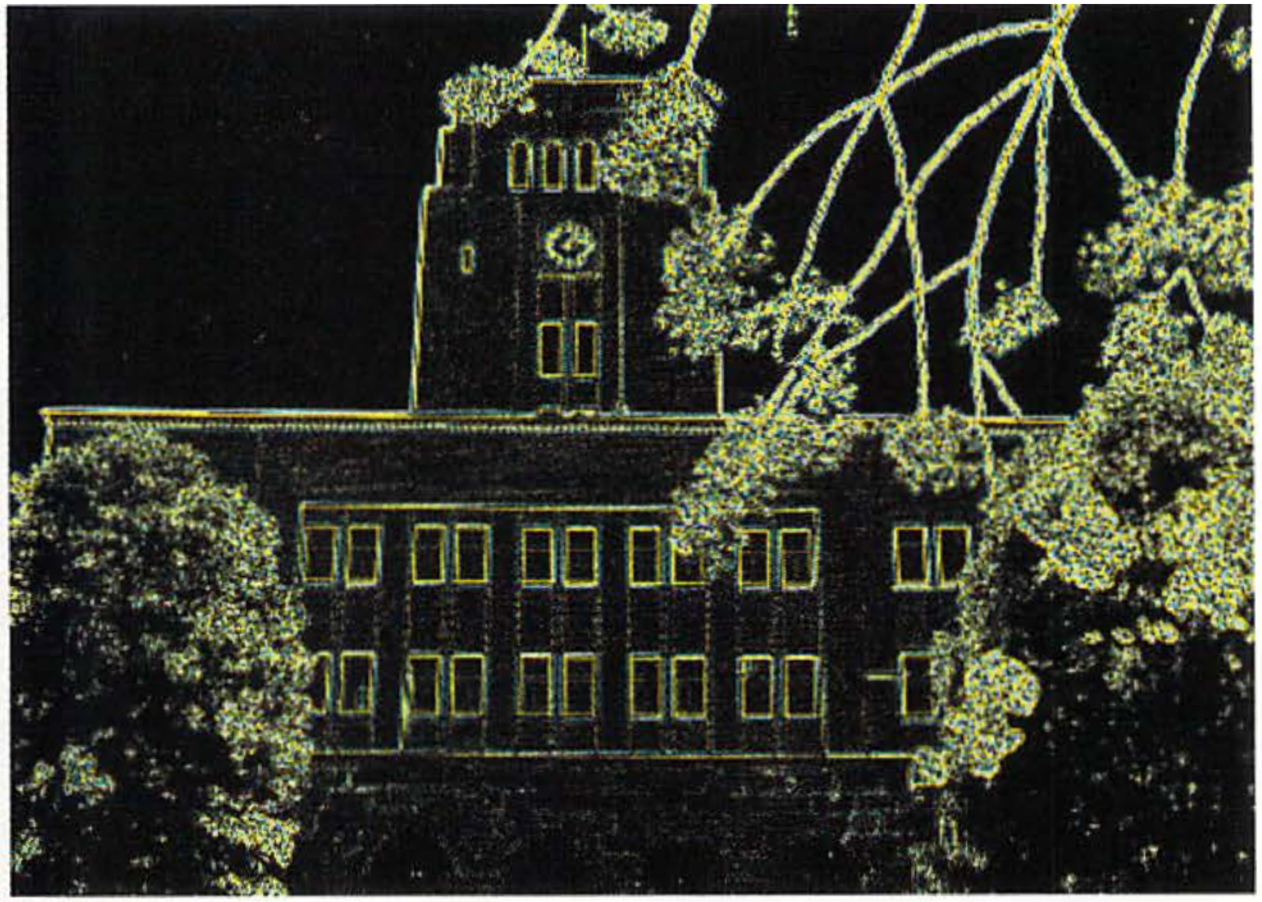
0	1	0
1	−4	1
0	1	0

ラプラシアンフィルタ

ラプラシアンフィルタの結果

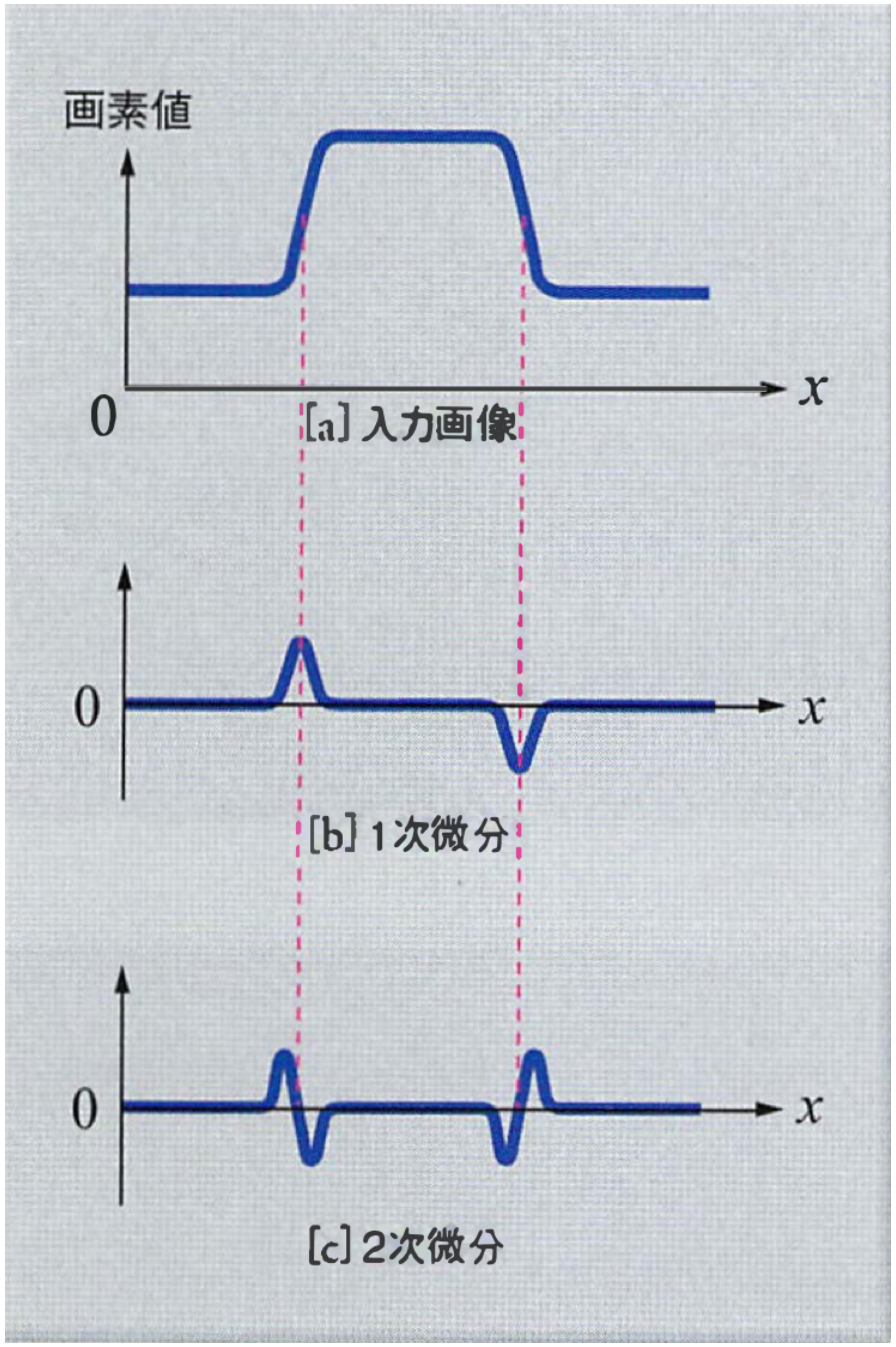


[a] 入力画像



[b] ラプラシアンフィルタの結果

画像の微分



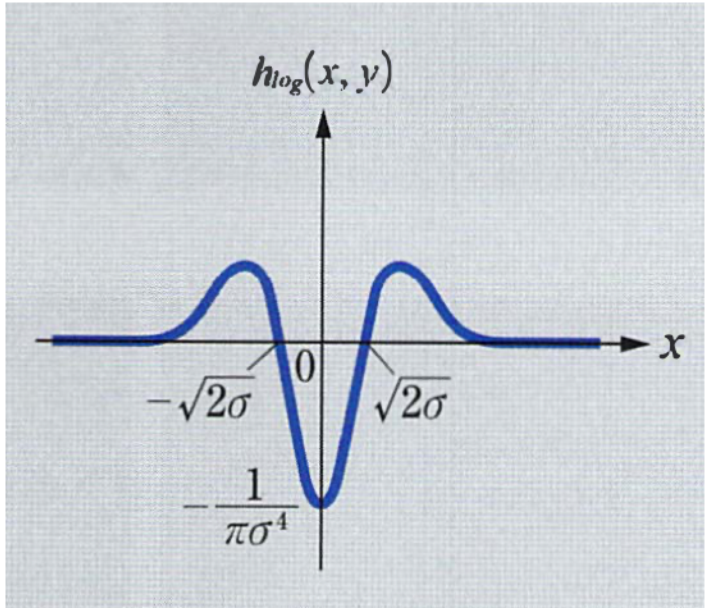
1階微分は変化している場所を捉えている.

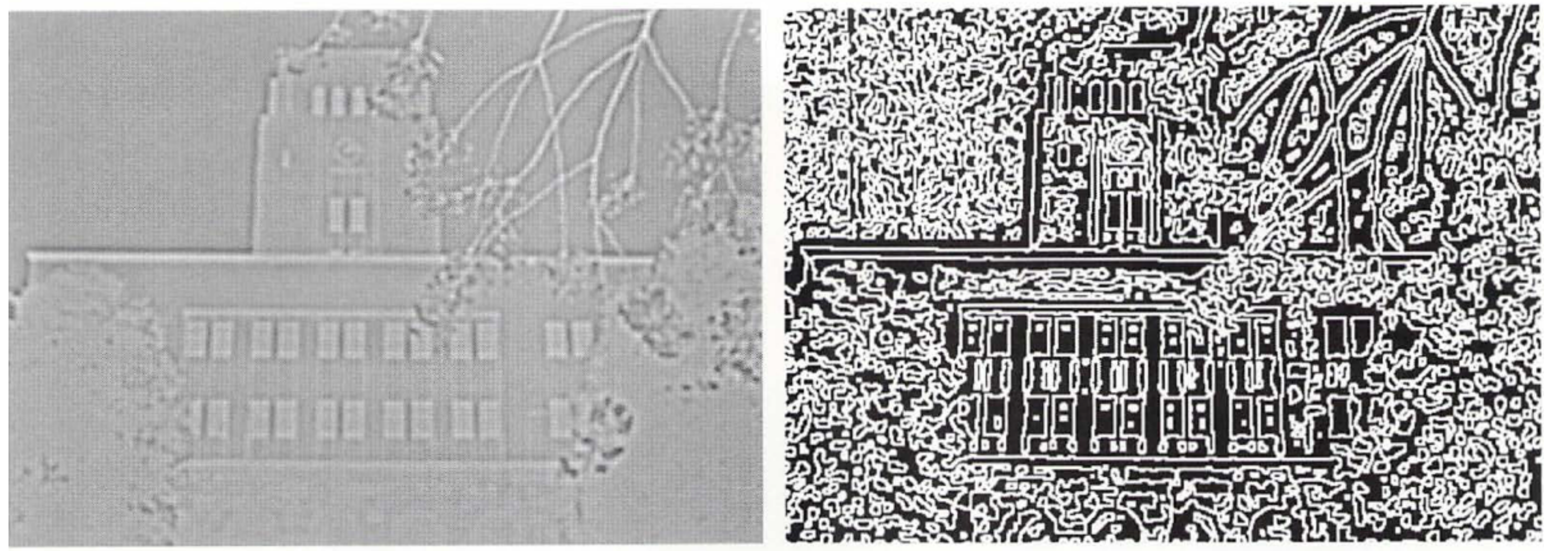
2階微分のゼロ交差点をエッジと考えることが出来る.

LoGフィルタ

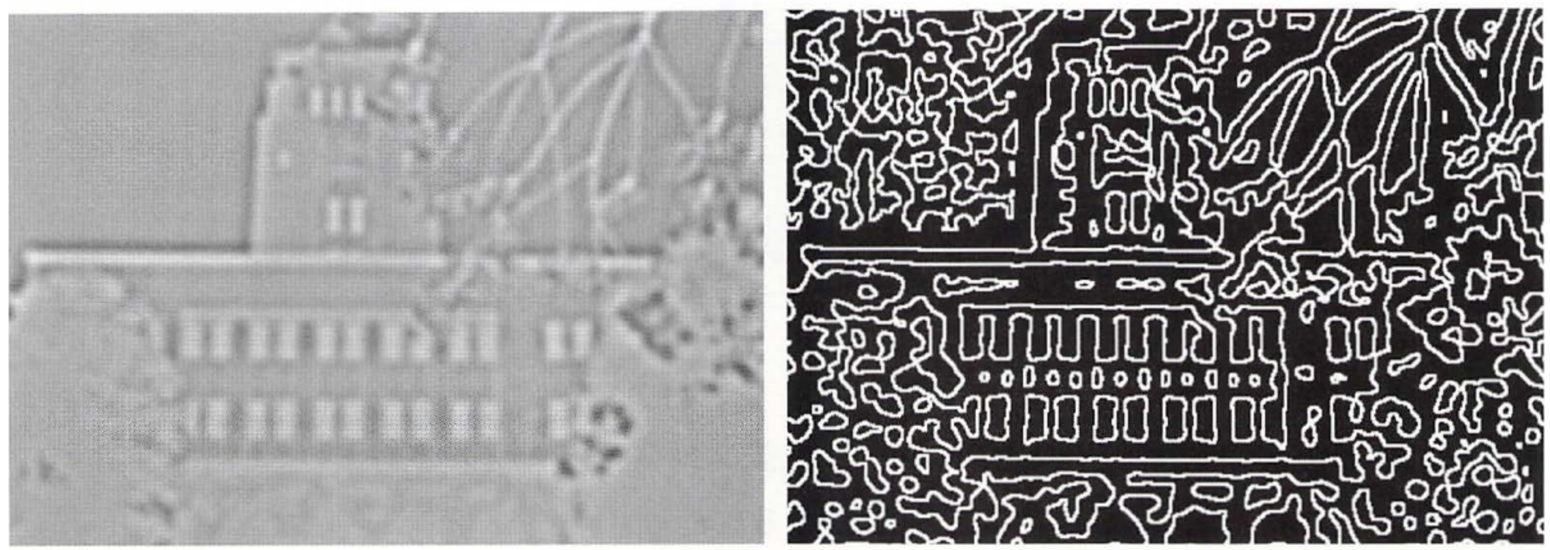
- ラプラシアンフィルタも微分しているだけなので,
- 微分フィルタ同様ノイズも捉えてしまう.
- ガウシアンフィルタを用いた平滑化によりノイズ除去をして，ラプラシアンフィルタをかけたほうが良い.
- それは，ガウシアンフィルタを2階微分することと同様である.

$$h_{log}(x, y)=\frac{x^2+y^2-2\sigma^2}{2\pi\sigma^6}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

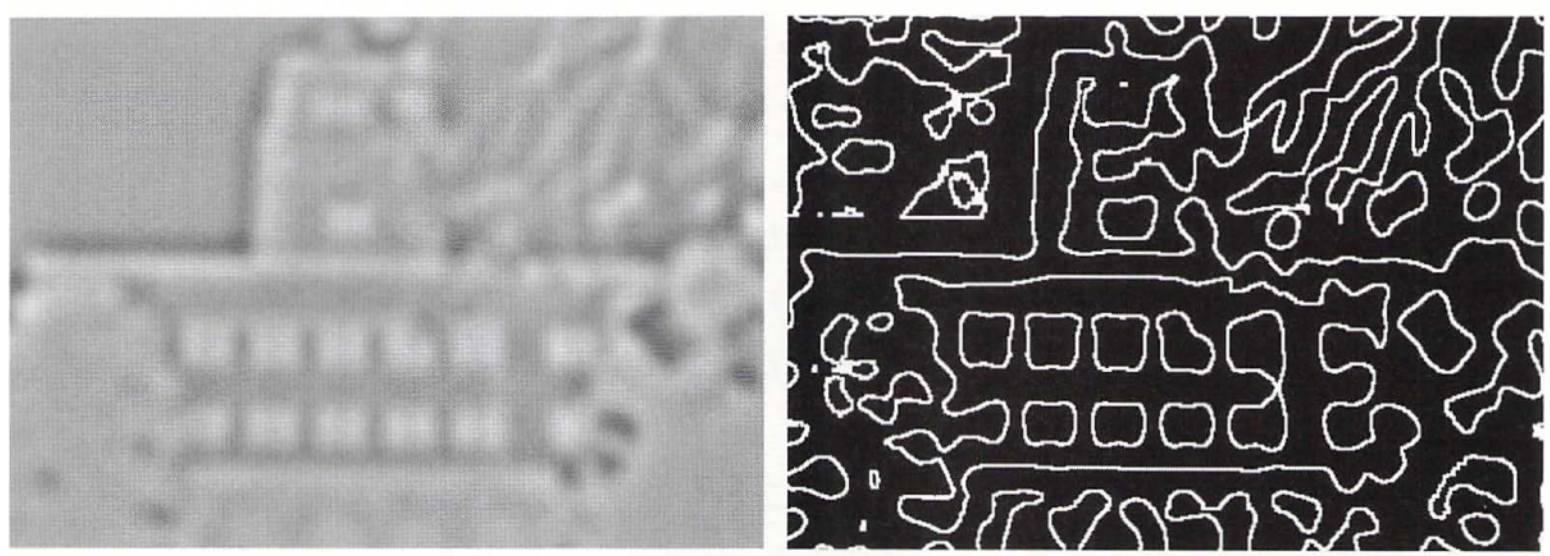




[a] σ=2



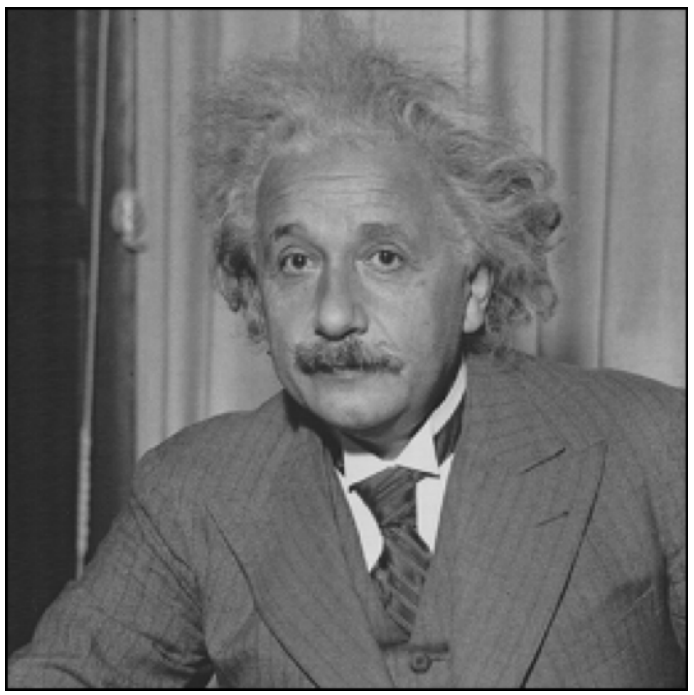
[b] σ=4



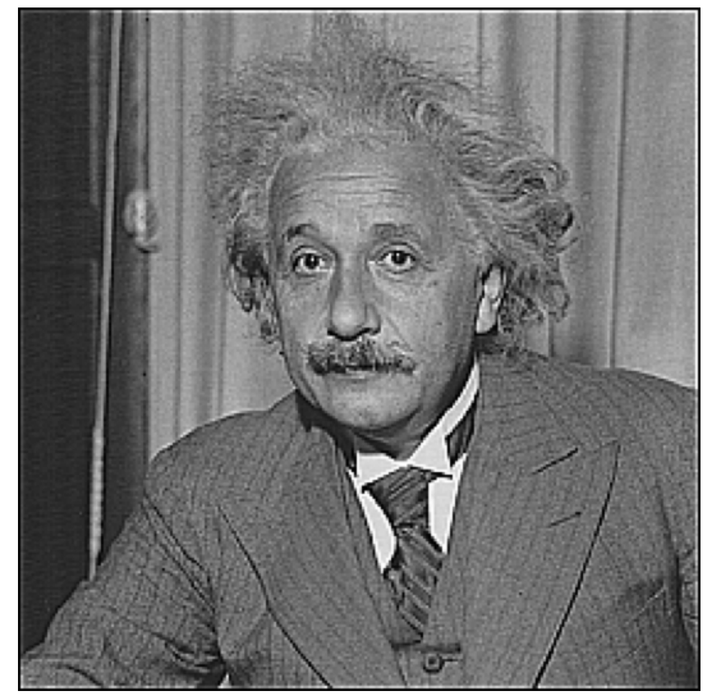
[b] σ=8

LoGフィルタの結果　　　　　　ゼロ交差

鮮鋭化



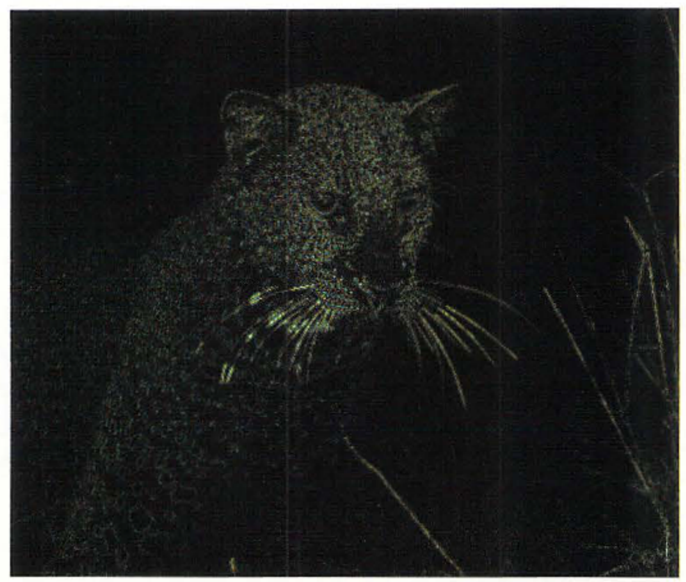
元画像



エッジが強調された画像

鮮鋭化

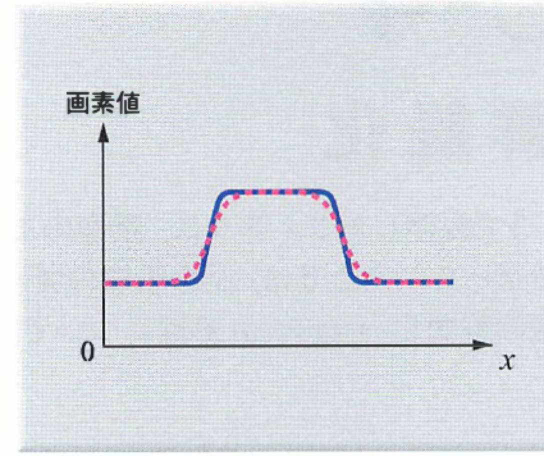
[a] 入力画像



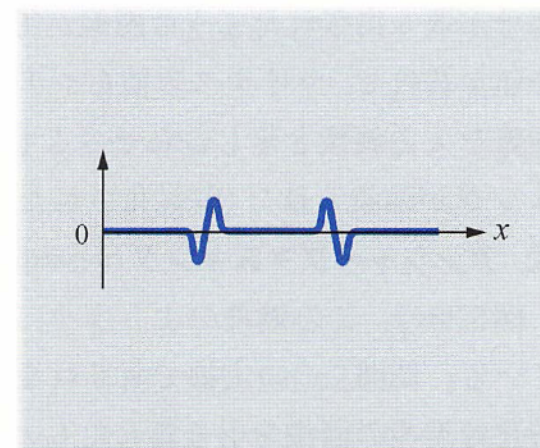
[b] 入力画像から平滑化した画像を引き算した画像

[c] 鮮鋭化された画像

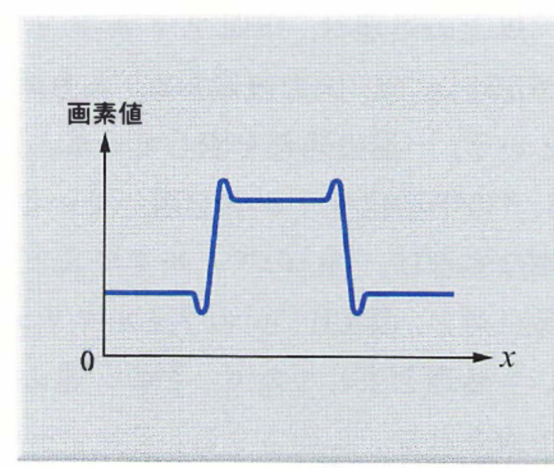
■図5.29——アンシャープマスキングによる画像の鮮鋭化



[a] 入力画像（実線）とその平滑化（点線）

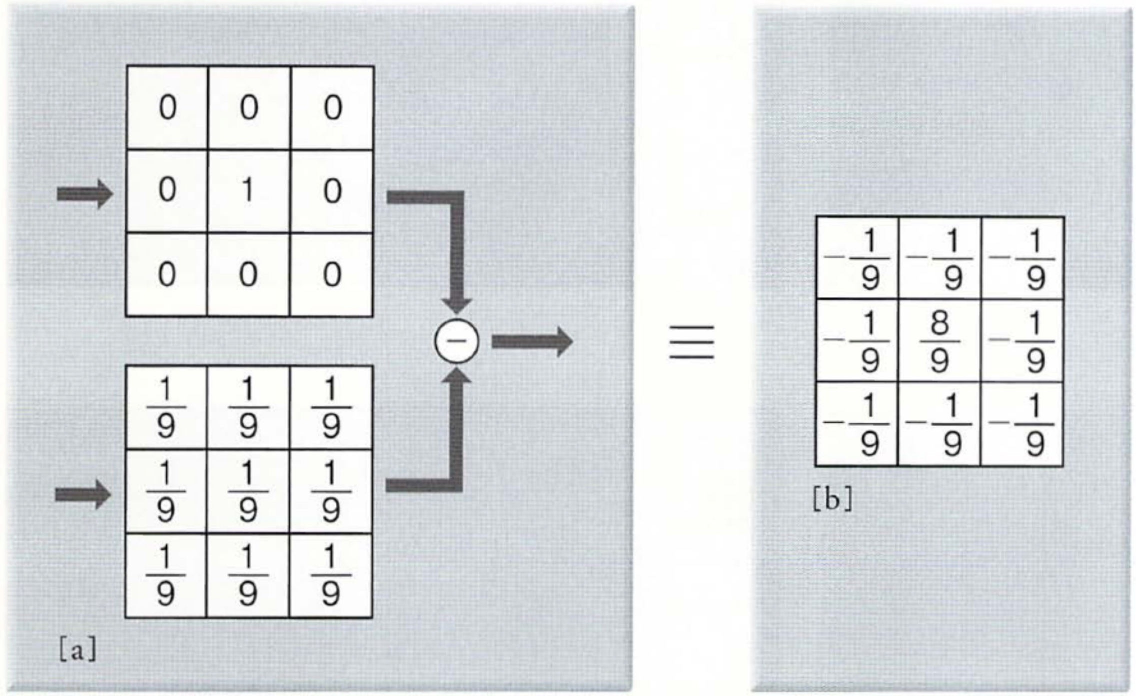


[b] [a] の実線から点線を引いたもの

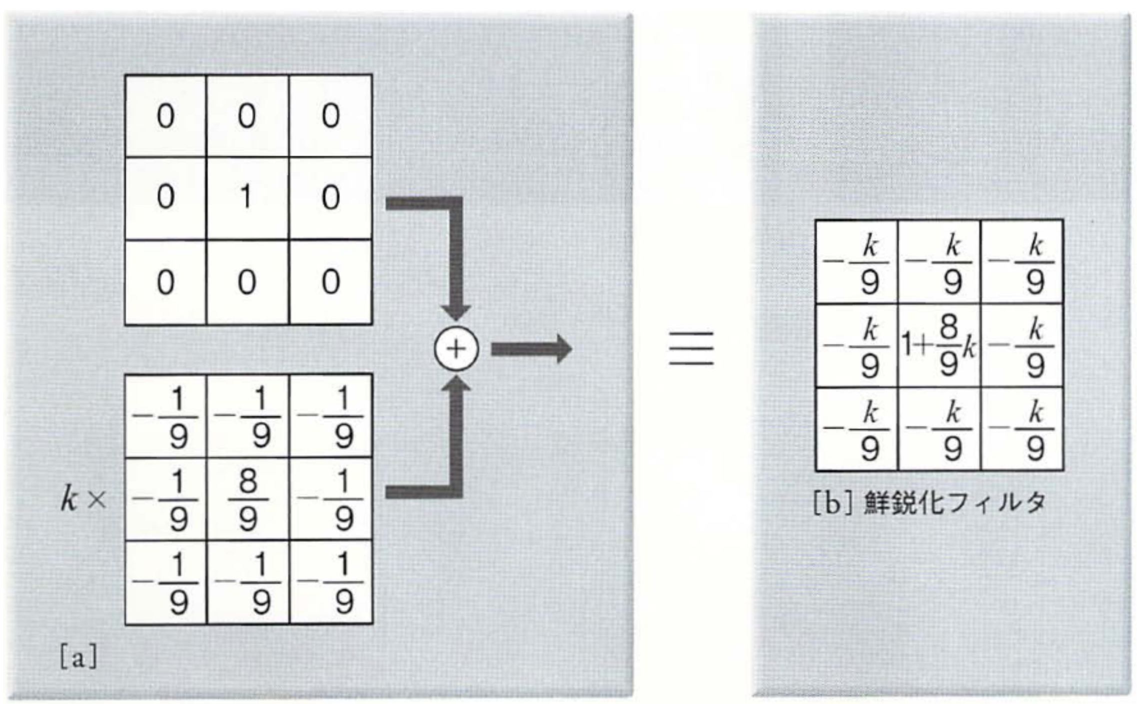


[c] [a] の実線に対し，[b] を定数倍して足したもの

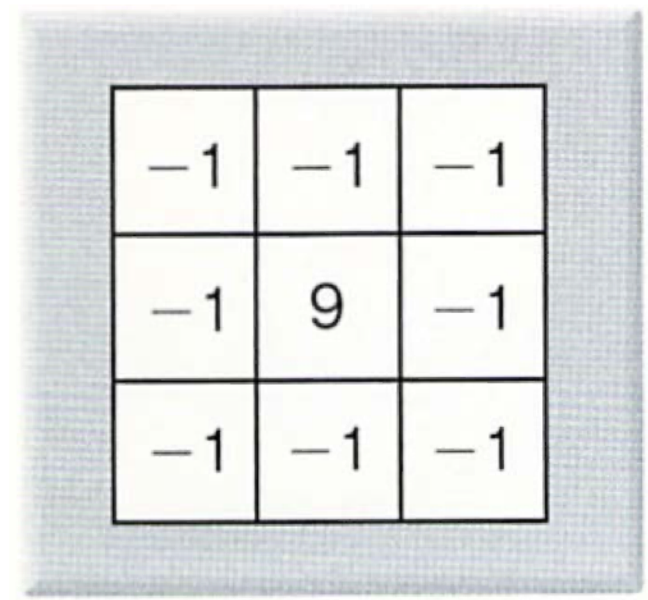
■図5.30——1次元画像を用いたアンシャープマスキングのようす



■図5.31――入力画像から，入力画像を平滑化した画像を引き算する



■図5.32――入力画像に対し，図5.31［b］でフィルタリングした画像をk倍してから足し算する



■図5.33――$k=9$のときの鮮鋭化フィルタ

鮮鋭化フィルタは中心興奮周辺抑制型の神経ネットワークと同じ構造になっている。

Bilateral filter

- Gaussian filterなどのぼかしフィルタでノイズ除去を行おうとすると画像全体がぼけた画像になる
- ノイズを除きつつはっきりとした画像にしたい場合どうするか？
  - bilateral filterを使う

Bilateral filter

$$g(i, j) = \frac{\sum_{n=-w}^{w} \sum_{m=-w}^{w} w(i, j, m, n) f(i+m, j+n)}{\sum_{n=-w}^{w} \sum_{m=-w}^{w} w(i, j, m, n)}$$

正規化項

$$w(i, j, m, n) = \exp\left(-\frac{m^2+n^2}{2\sigma_1^2}\right) \exp\left(-\frac{(f(i, j) - f(i+m, j+n))^2}{2\sigma_2^2}\right)$$

フィルタの意味

$$w(i, j, m, n) = \boxed{\exp\left(-\frac{m^2+n^2}{2\sigma_1^2}\right)} \boxed{\exp\left(-\frac{(f(i, j)-f(i+m, j+n))^2}{2\sigma_2^2}\right)}$$

ガウシアンフィルタ
ぼかす効果(ノイズ除去)

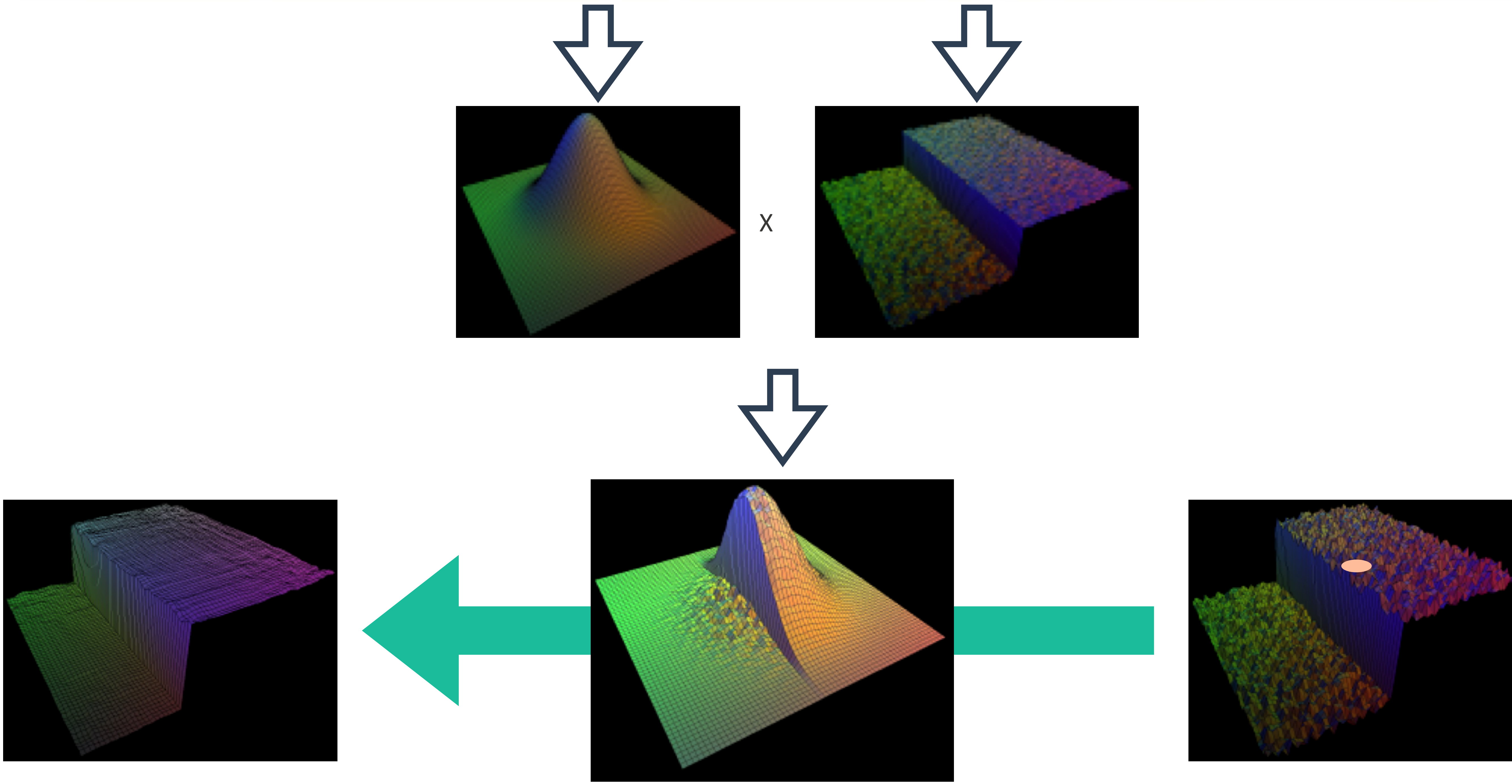
周辺画素の画素値の差が小さければ
大きな値

画素値の差が小さい場所ではガウシアンフィルタが強くかかり、画素値の差が大きい場所ではガウシアンフィルタが弱くかかる

コントラストが維持されたままノイズ除去できる

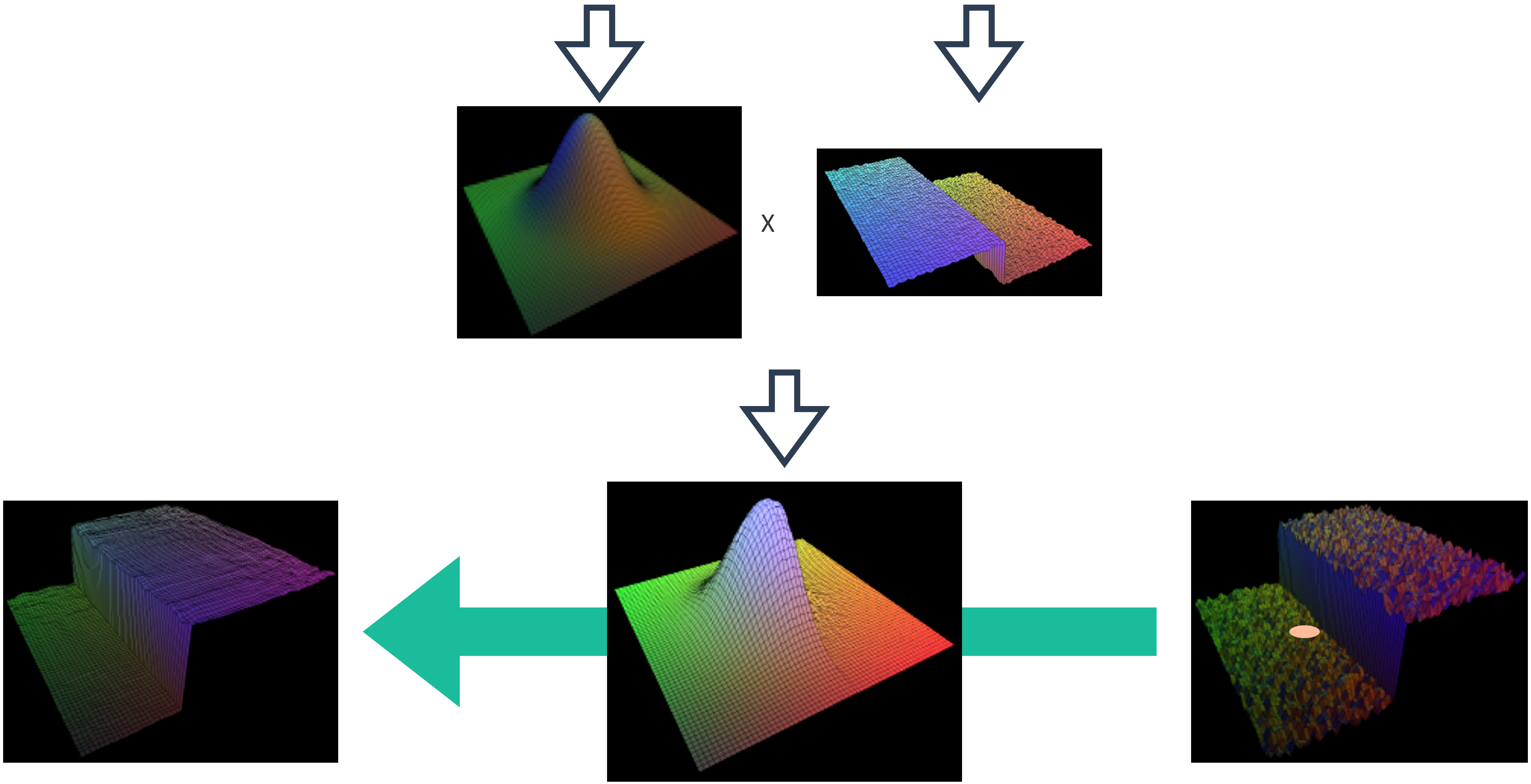
フィルタの適用の様子

$$w(i, j, m, n) = \exp\left(-\frac{m^2+n^2}{2\sigma_1^2}\right)\exp\left(-\frac{(f(i, j)-f(i+m, j+n))^2}{2\sigma_2^2}\right)$$

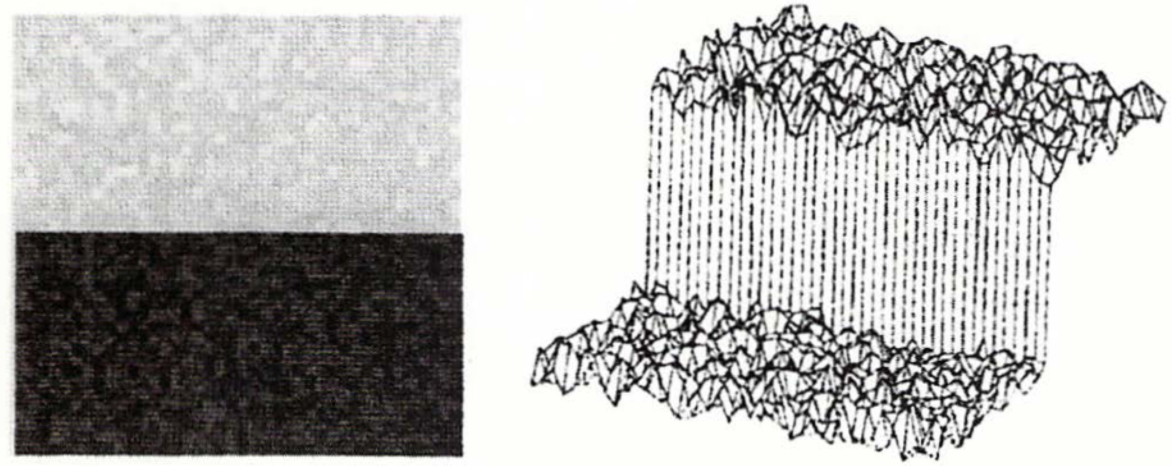


フィルタの適用の様子

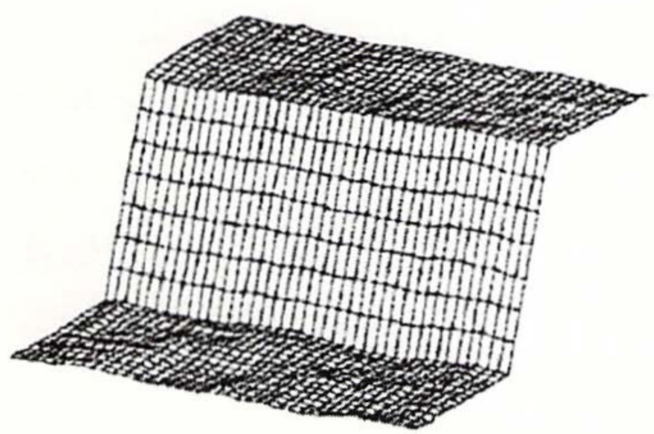
$$w(i,j,m,n)=\exp\left(-\frac{m^2+n^2}{2\sigma_1^2}\right)\exp\left(-\frac{\left(f(i,j)-f(i+m,j+n)\right)^2}{2\sigma_2^2}\right)$$



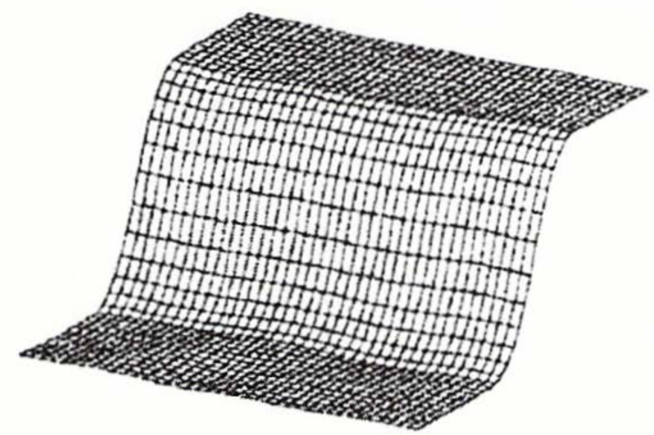
平均化フィルタとの比較



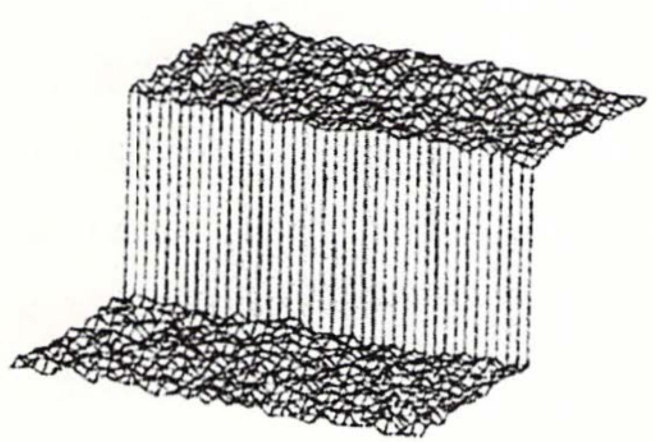
[a] 入力画像



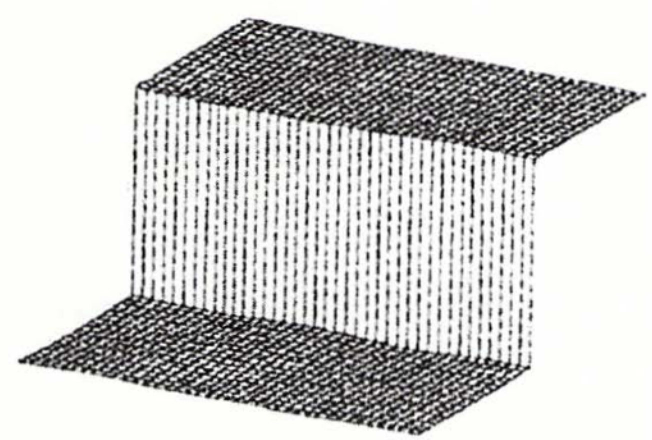
[b] 平均化フィルタ（1回）



[c] 平均化フィルタ（2回）



[d] バイラテラルフィルタ（1回）



[e] バイラテラルフィルタ（2回）

bilateral filter適用例

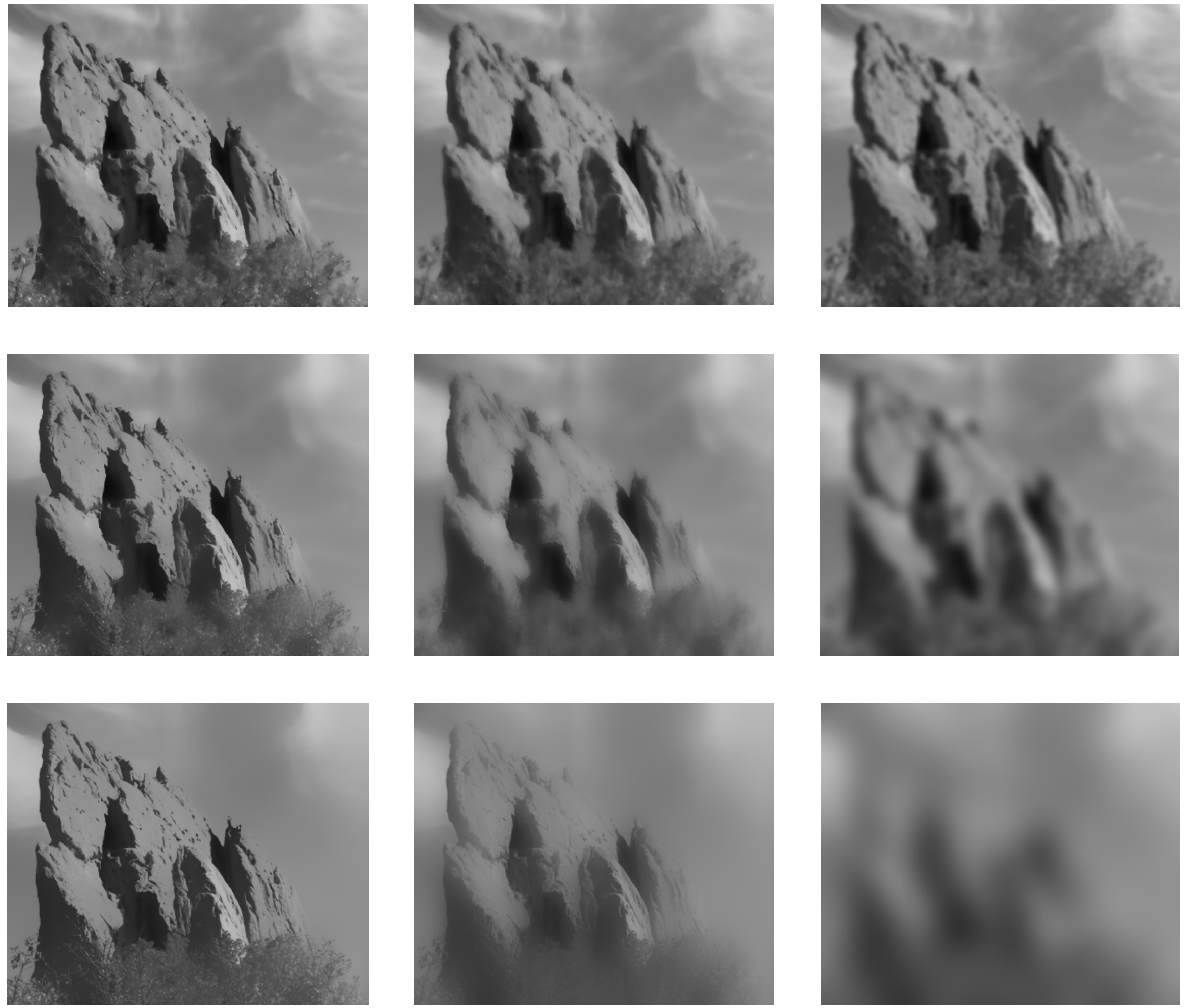
original

σ2 = 0.1 | σ2 = 0.25 | σ2 = ∞

σ1 = 2

σ1 = 6

σ1 = 18



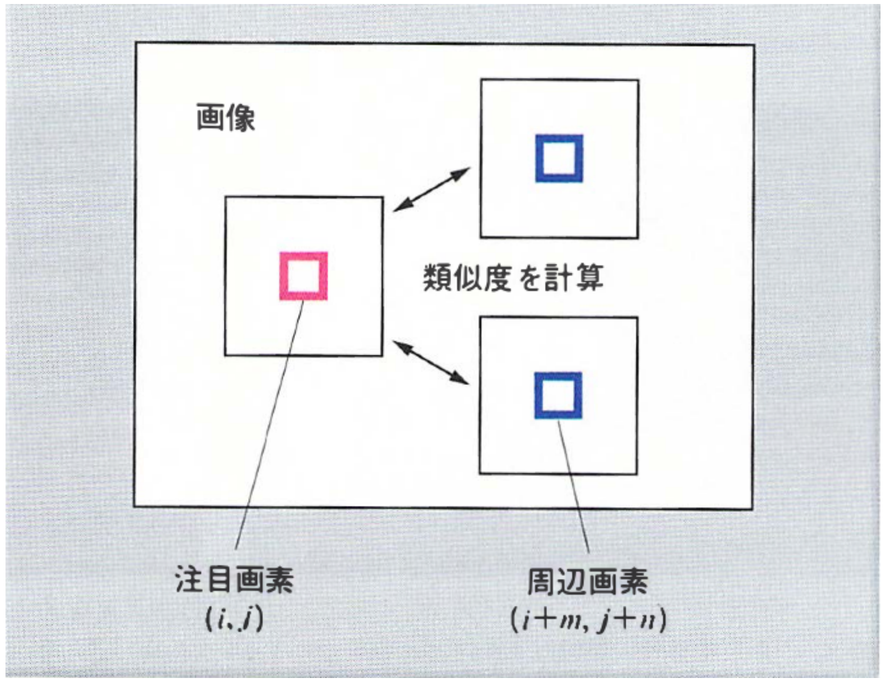
non-local mean filter

ある大きさのブロック間の2乗誤差(L2 norm)

$$w(i,j,m,n)=\exp\left(-\frac{\boxed{\sum_{t=-w'}^{w'}\sum_{s=-w'}^{w'}\left(f(i+s,j+t)-f(i+m+s,j+n+t)\right)^2}}{2\sigma_3^2}\right)$$

ブロック内の図形が似ていれば大きくなる

似ていれば混ぜましょう！！



ノイズ除去の比較

[a] 入力画像


[b] 平均化フィルタの結果

[c] バイラテラルフィルタの結果

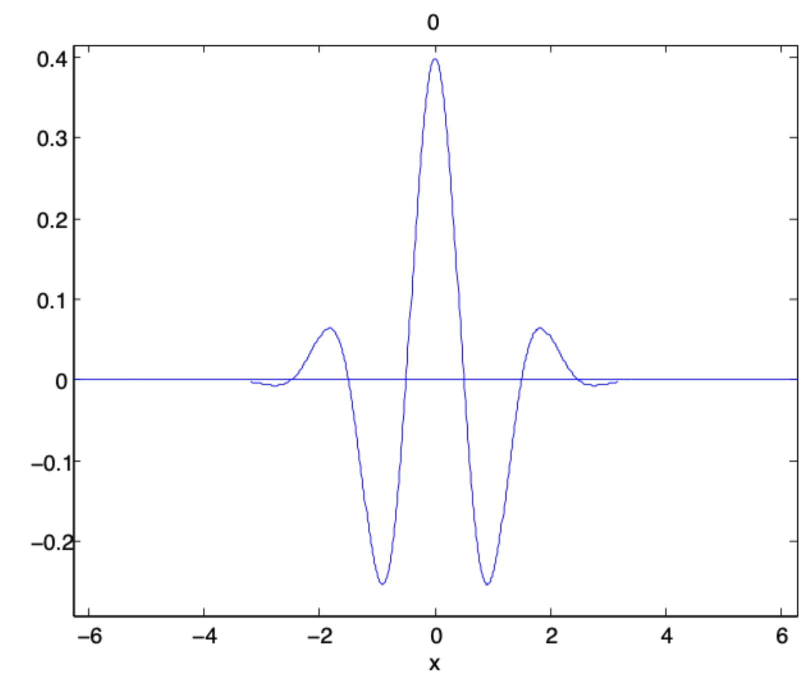
[d] ノンローカルミーンフィルタの結果

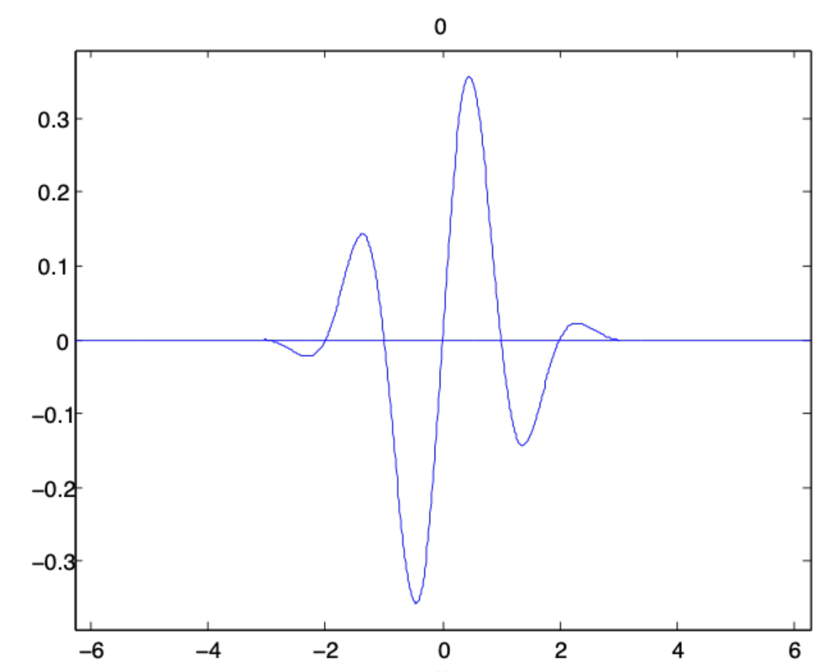
Gabor filter

1次元ガボールフィルタ

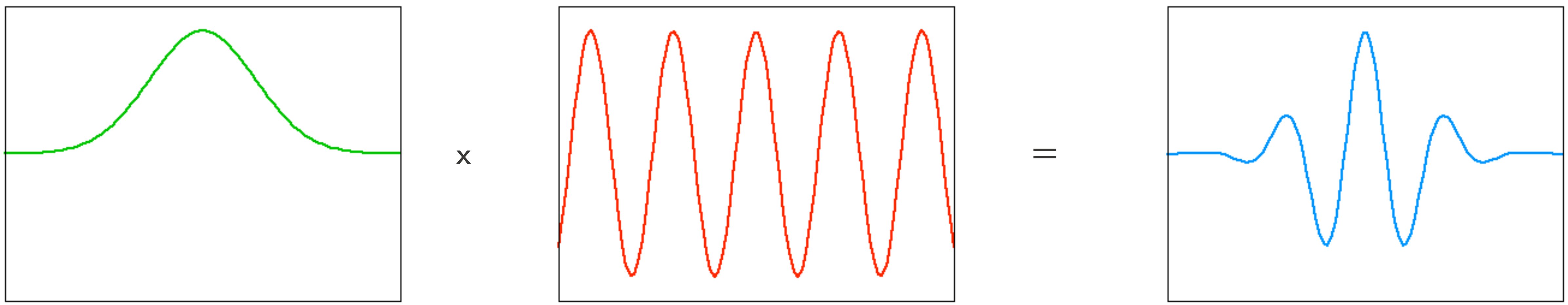
$$g_e(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \cos(2\pi\omega_0 x)$$

$$g_o(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \sin(2\pi\omega_0 x)$$

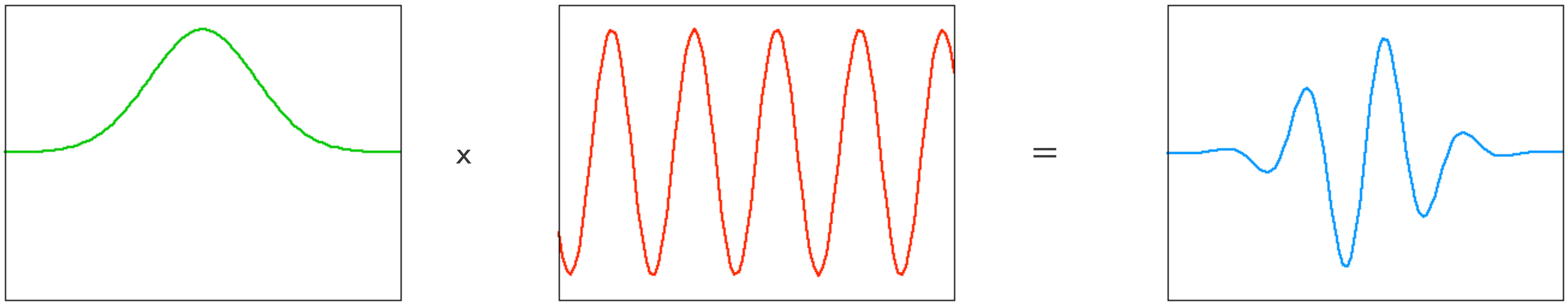




$$g_e(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \cos(2\pi\omega_0 x)$$



$$g_o(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \sin(2\pi\omega_0 x)$$

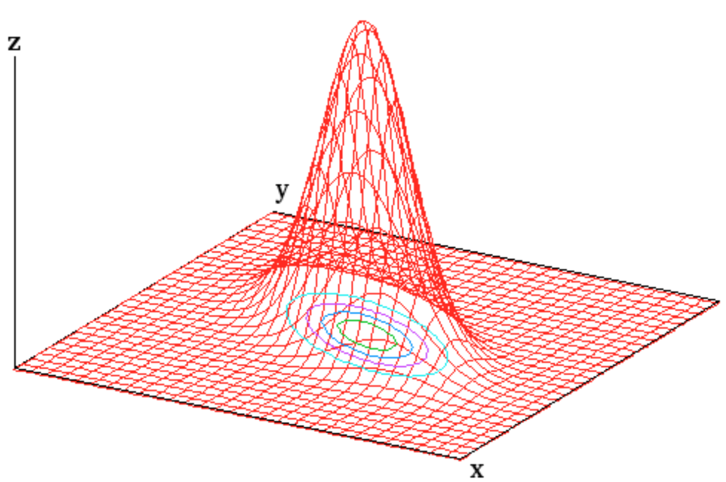


ガウシアンフィルタ　　正弦波

2次元ガボールフィルタ

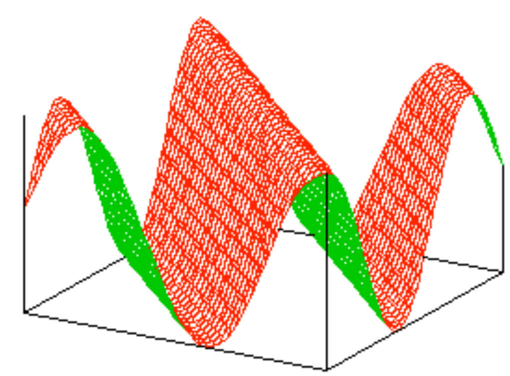
$$g_e(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{\sigma x} + \frac{y^2}{\sigma y}\right)} \cos(2\pi\omega_{x_0} x + 2\pi\omega_{y_0} y)$$

$$g_o(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{\sigma x} + \frac{y^2}{\sigma y}\right)} \sin(2\pi\omega_{x_0} x + 2\pi\omega_{y_0} y)$$



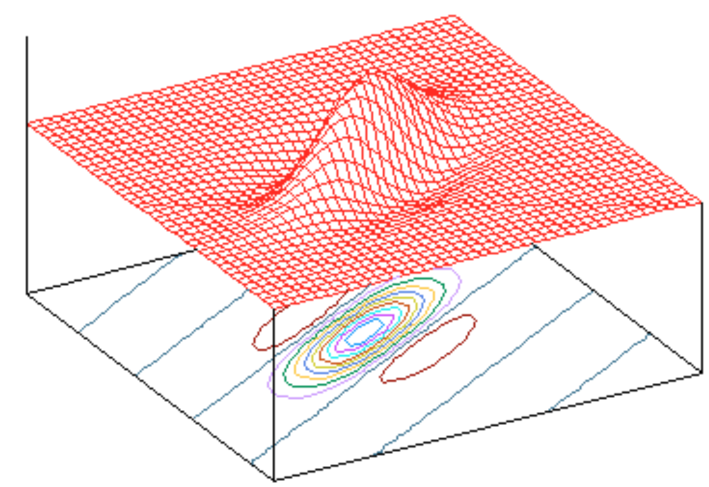
2次元ガウシアンフィルタ

x



正弦波

=



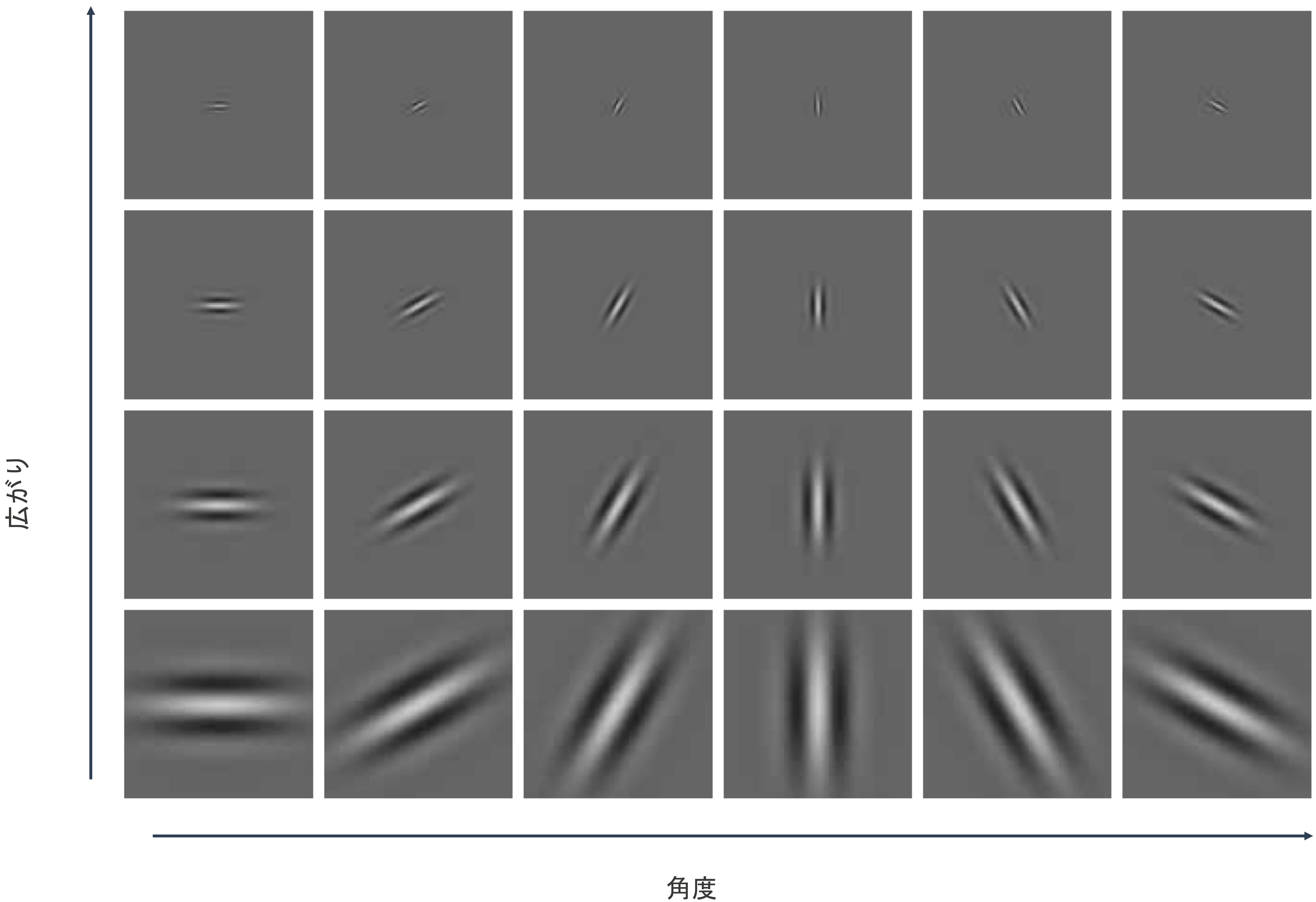
$$g_e(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x}+\frac{y^2}{\sigma_y}\right)} \cos(2\pi\omega_{x_0}x + 2\pi\omega_{y_0}y)$$

ガウシアンフィルタ
分散を調整することで広がりを調整できる。

正弦波のωを調整することで、正弦波の空間周波数と方向を調整できる。

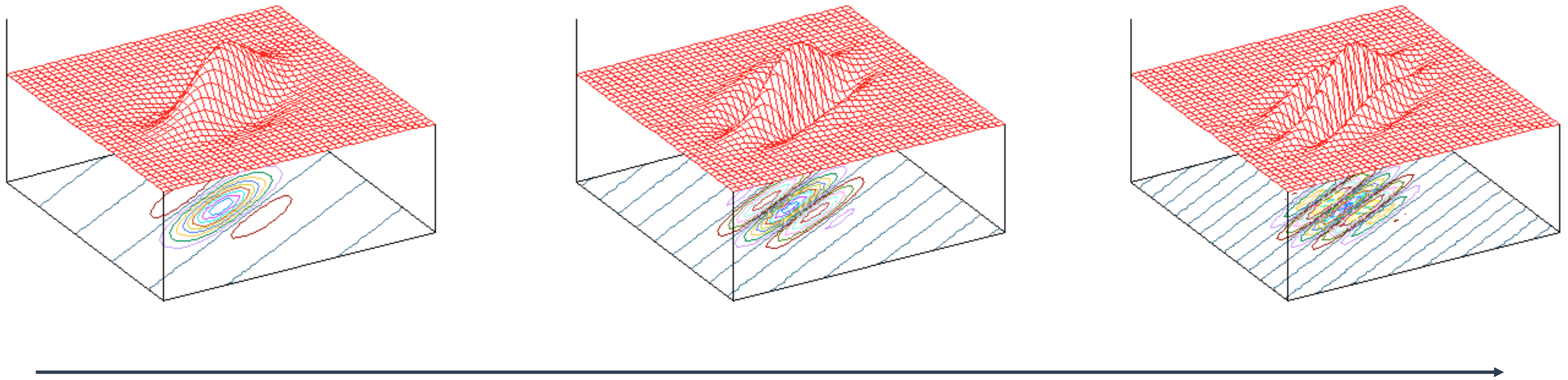
ωy=0ならば、縦縞になる。

ωx=ωyならば、斜め45度の縞模様になる。



広がり
角度

周波数の違いとGabor filter

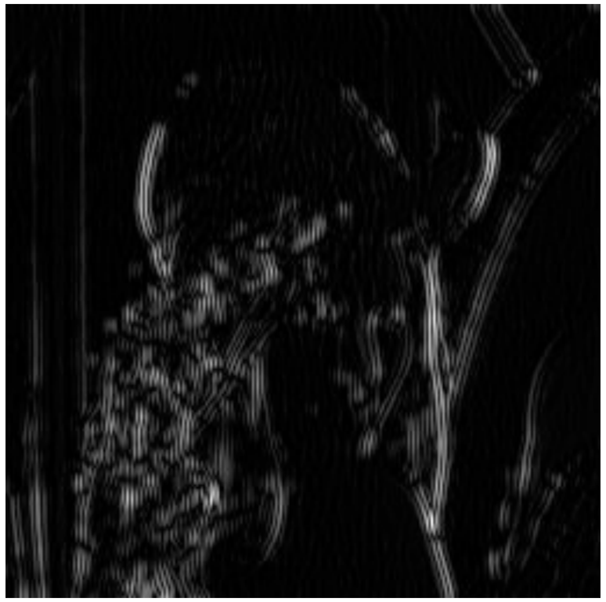


低周波数

高周波数

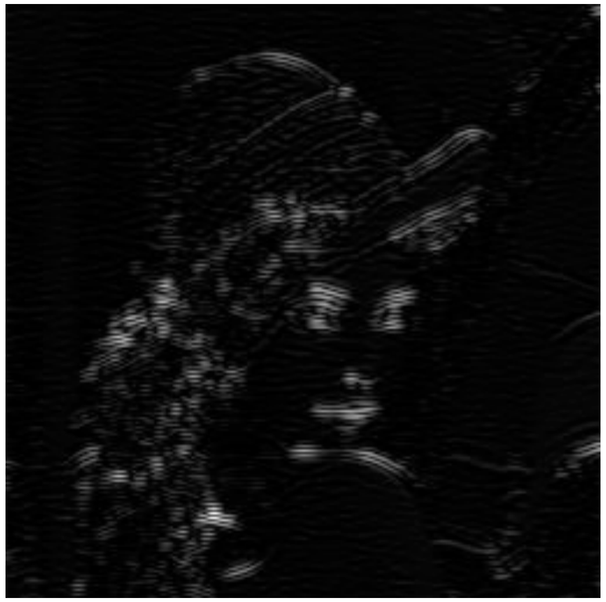
Gabor filterの適用例

θ=0



θ=45

θ=90



θ=135

結局何の役に立つの？

- 画像に含まれる方向成分を抽出できる

- 薀蓄
    - 脳の第１次視覚野の細胞の応答特性はGabor filterで近似できる。
    - 自己組織化マップやTopographic ICAを用い画像を処理したところ、画像の構成要素としてGabor filterのような要素が抽出された。