Video signals

IMAGE SPATIAL PROCESSING

Filtering examples

Original
Cameraman

Cameraman blurred vertically

Filter impulse response

$$\frac{1}{5}\begin{pmatrix} 1 \\ 1 \\ [1] \\ 1 \\ 1 \end{pmatrix}$$

Filtering examples

Original
Cameraman

Cameraman blurred by convolution
Filter impulse response

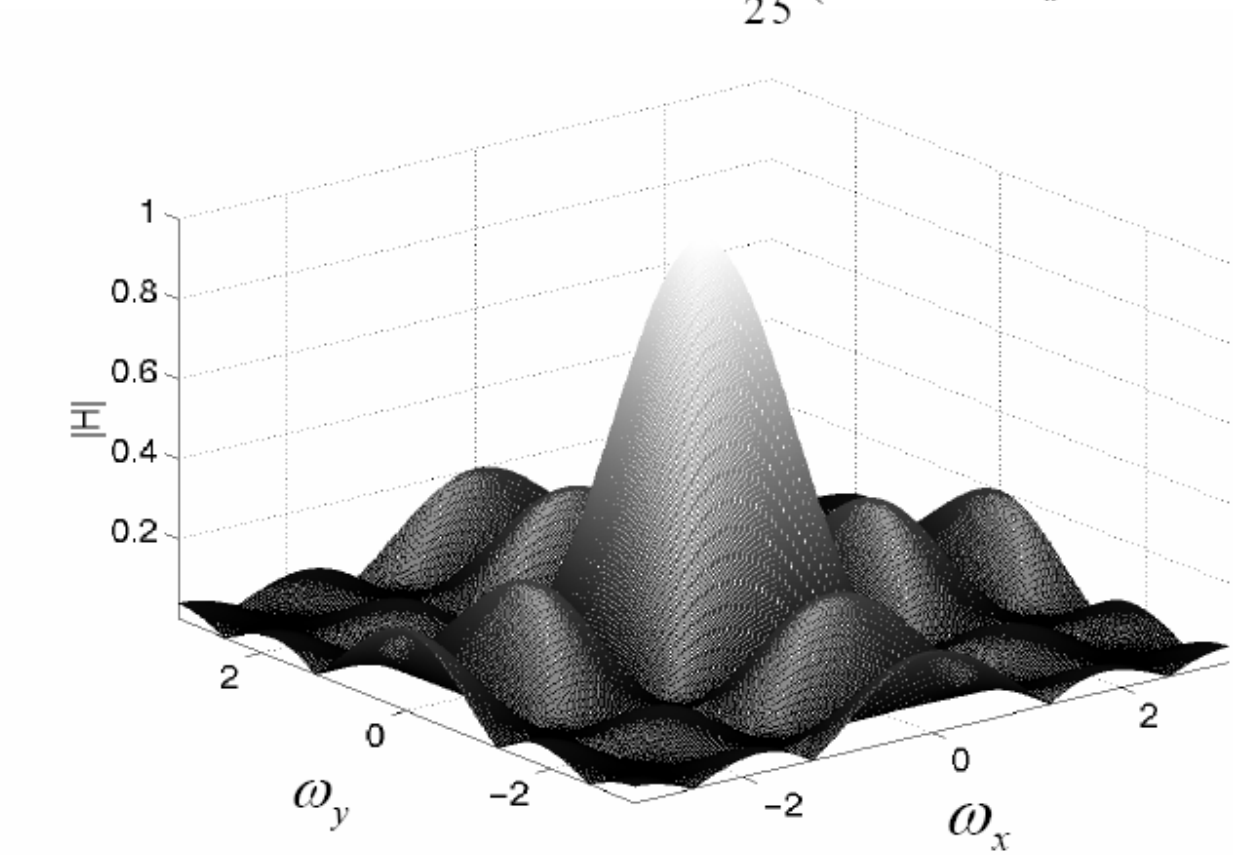
$$\frac{1}{25}\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & [1] & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Fourier interpretation

$$H\left(e^{j\omega_x}, e^{j\omega_y}\right) = \sum_{m-\infty}^{\infty} \sum_{n=-\infty}^{\infty} h[m,n] e^{-j\omega_x m - j\omega_y n}$$

$$= \frac{1}{25} \sum_{m=-2}^{2} \sum_{n=-2}^{2} e^{-j\omega_x m - j\omega_y n} = \frac{1}{25} \sum_{m=-2}^{2} e^{-j\omega_x m} \sum_{n=-2}^{2} e^{-j\omega_y n}$$

$$= \frac{1}{25}\left(1 + 2\cos\omega_x + 2\cos(2\omega_x)\right)\left(1 + 2\cos\omega_y + 2\cos(2\omega_y)\right)$$



Filtering Examples

Original
Cameraman

Cameraman blurred horizontally
Filter impulse response

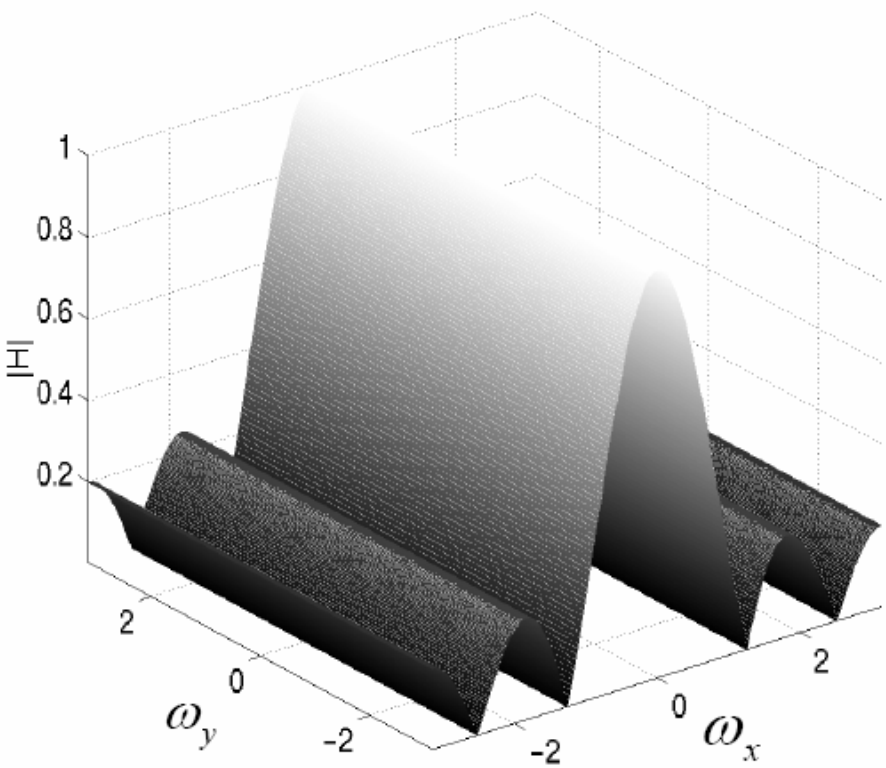
$$\frac{1}{5}\begin{pmatrix}1 & 1 & [1] & 1 & 1\end{pmatrix}$$

Fourier interpretation

Cameraman blurred horizontally

Filter impulse response

$$\frac{1}{5}\begin{pmatrix}1 & 1 & [1] & 1 & 1\end{pmatrix}$$



Filtering examples

Original
Cameraman

Cameraman sharpened

Filter impulse response

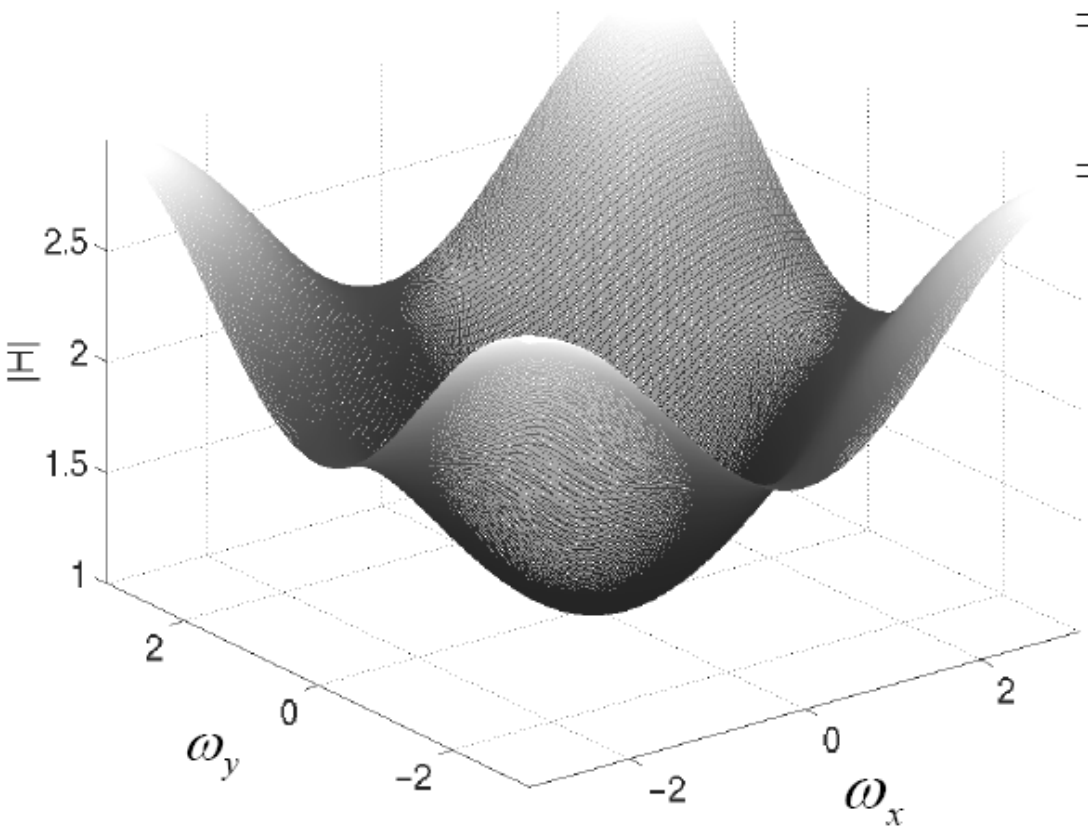
$$\frac{1}{4}\begin{pmatrix} 0 & -1 & 0 \\ -1 & [8] & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

Fourier interpretation

$$H\left(e^{j\omega_x}, e^{j\omega_y}\right) = \sum_{m-\infty}^{\infty} \sum_{n=-\infty}^{\infty} h[m,n] e^{-j\omega_x m - j\omega_y n}$$

$$= \frac{1}{4}\left(8 - e^{-j\omega_x} - e^{j\omega_x} - e^{-j\omega_y} - e^{j\omega_y}\right)$$

$$= 2 - \frac{1}{2}\cos\omega_x - \frac{1}{2}\cos\omega_y$$



Filtering examples

Original
Cameraman

Cameraman sharpened

Filter impulse response

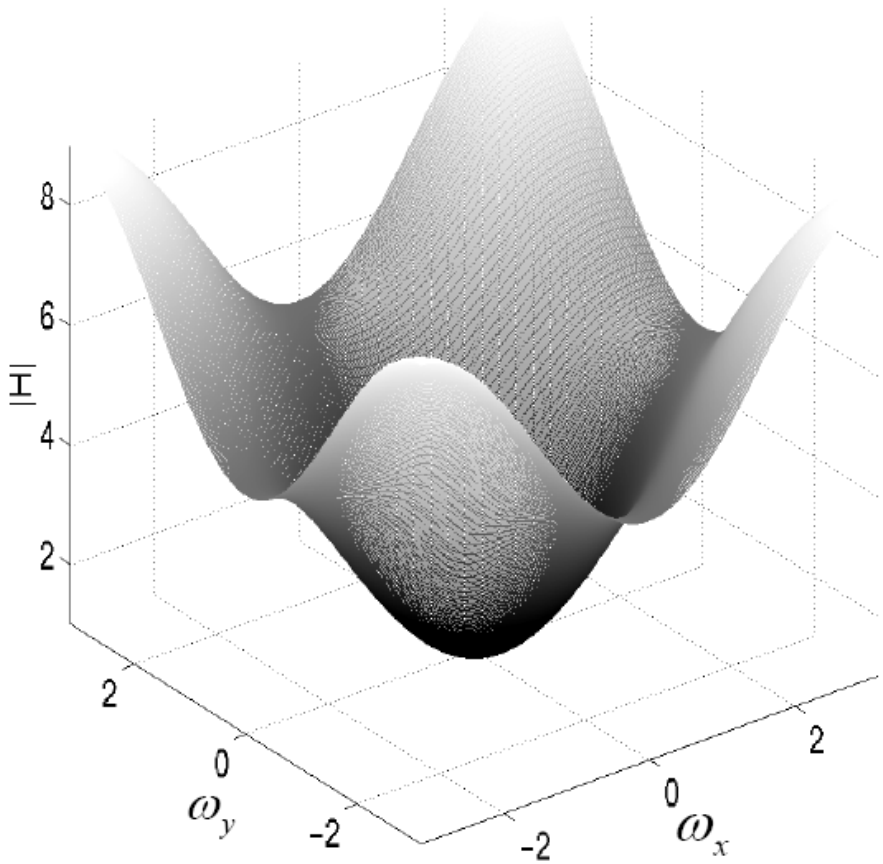
$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & [5] & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

Fourier interpretation

Cameraman sharpened

Filter impulse response

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & [5] & -1 \\ 0 & -1 & 0 \end{pmatrix}$$



Linear and non linear operations

10	13	9
12	8	9
15	11	6

Replace center pixel by:

Median Filter: (6, 8, 9, 9,**10**, 11, 12, 13, 15) = 10

Minimum = 6; Maximum: 15

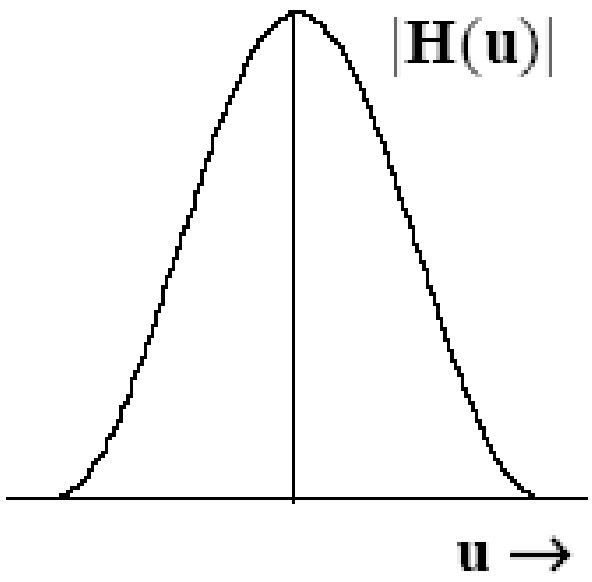
Average of nearest neighbours:

(10+13+9+12+8+9+15+11+6)/9 = 10.33 → **10**

Low pass gaussian filter

$$\frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} 1 & 2 & 1 \end{pmatrix} * \frac{1}{4}\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$

$$\frac{1}{441}\begin{pmatrix} 4 & 10 & 14 & 10 & 4 \\ 10 & 25 & 35 & 25 & 10 \\ 14 & 35 & 49 & 35 & 14 \\ 10 & 25 & 35 & 25 & 10 \\ 4 & 10 & 14 & 10 & 4 \end{pmatrix}$$



Median filtering

Hi pass filtering for high frequencies

$$f(x)$$

$$\frac{df(x)}{dx} = \nabla_x f(x)$$

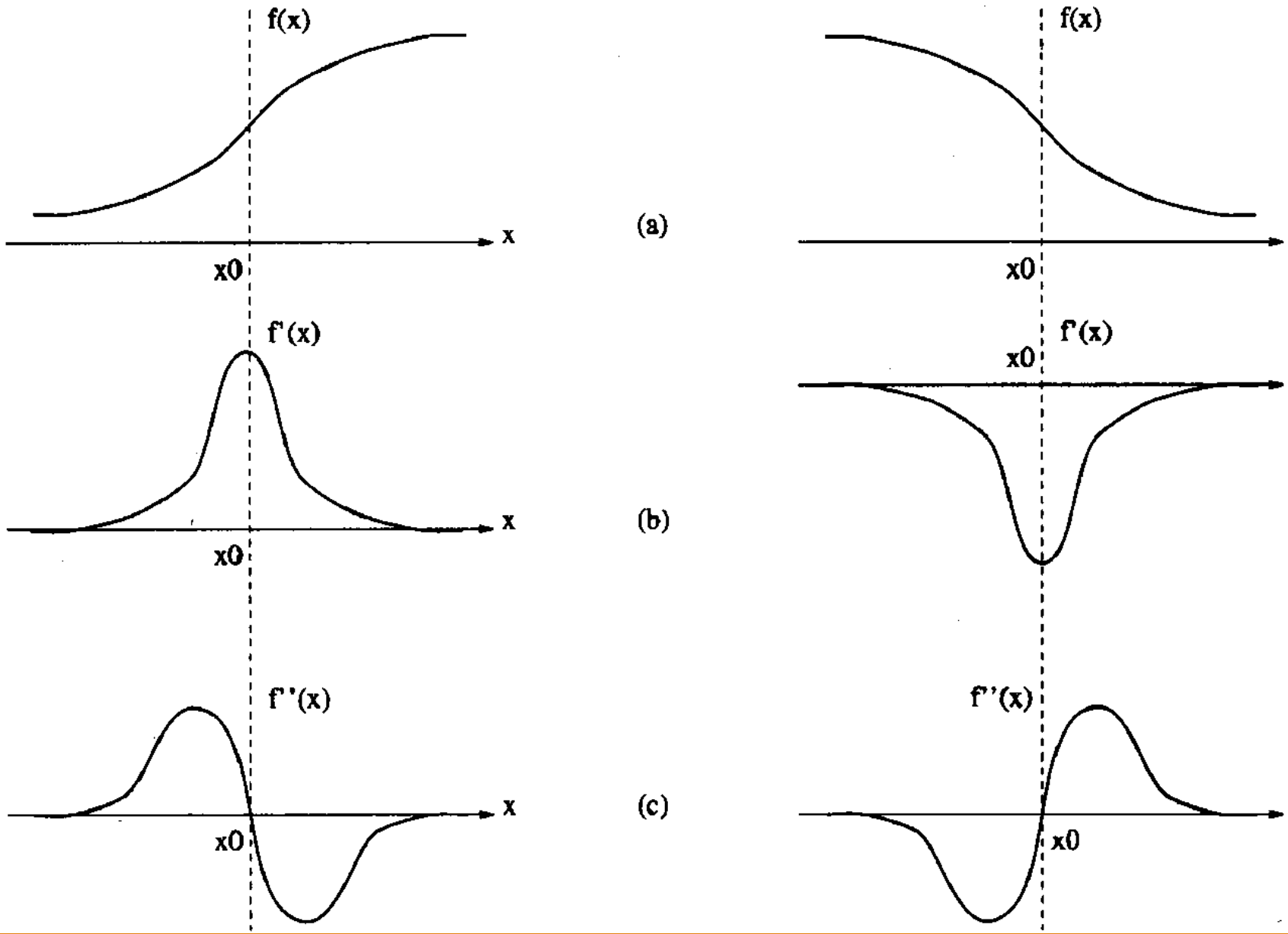
$$\frac{d^2 f(x)}{dx^2} = \nabla_x \begin{bmatrix} 0 & +1 & 0 \\ +1 & -4 & +1 \\ 0 & +1 & 0 \end{bmatrix}$$

$$= \Delta_x f(x)$$

$$(1 - \Delta_x) f(x)$$

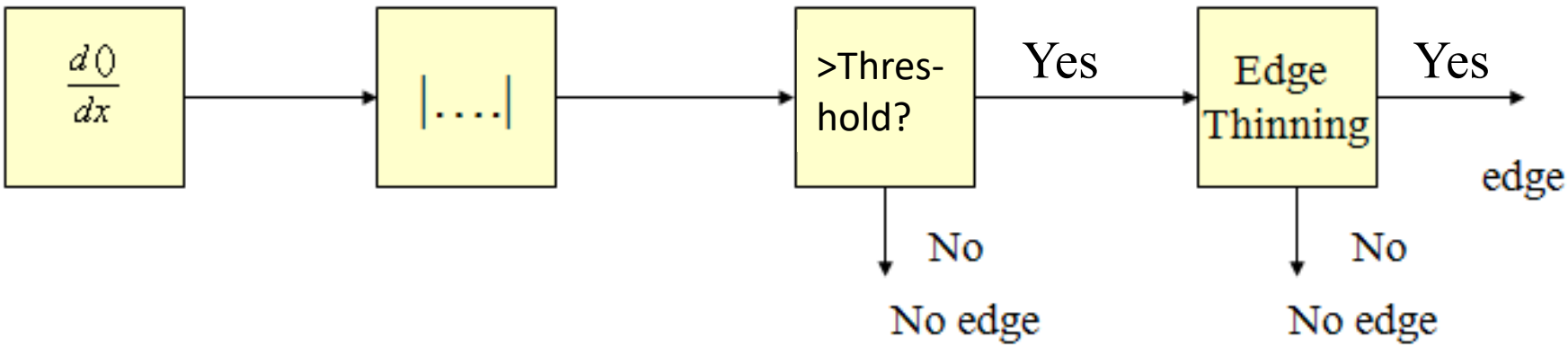
Example

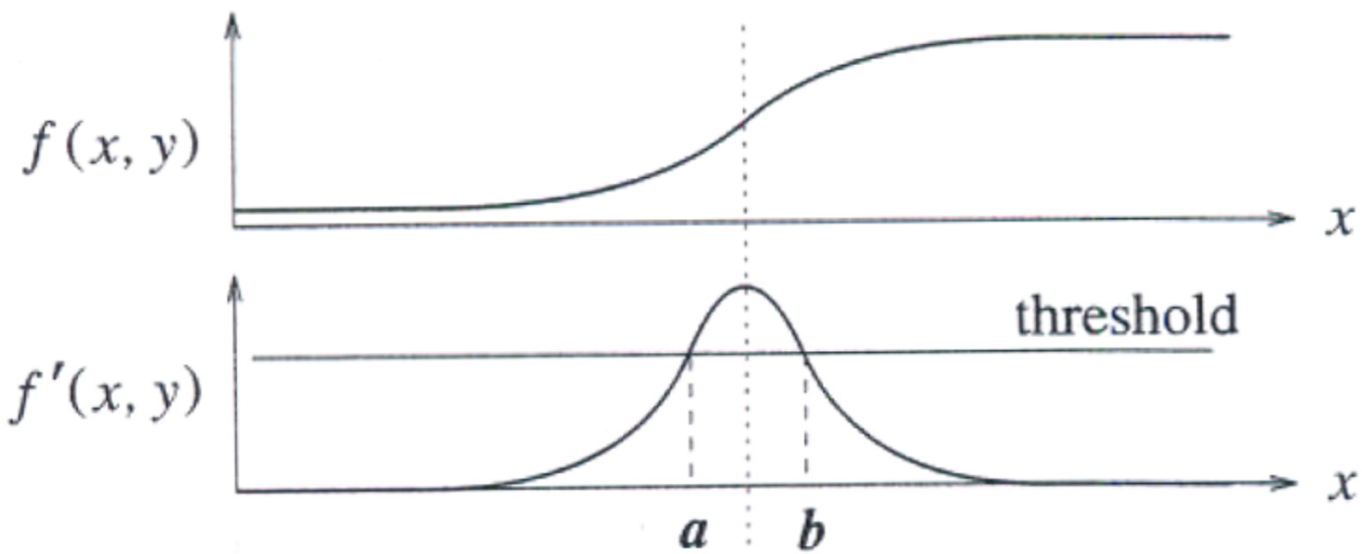
1D derivatives



Gradient methods

1D example





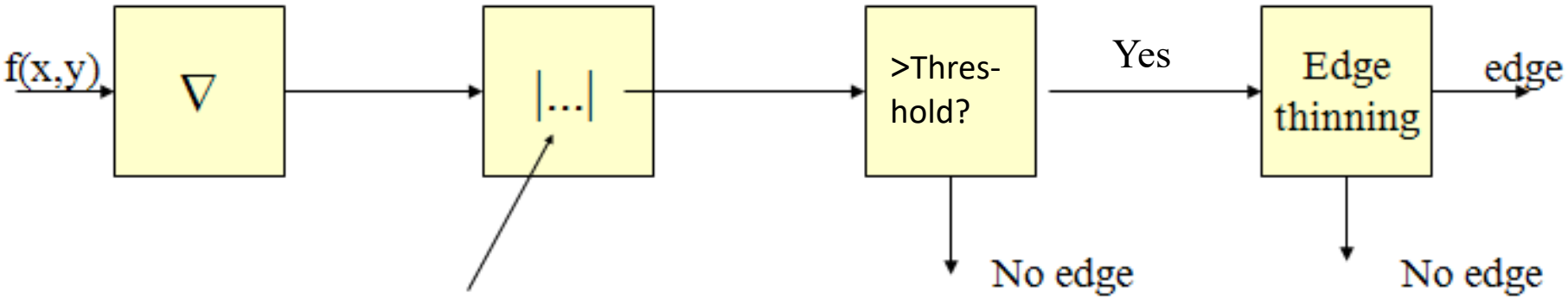
2D case

- The first derivative is substituted by the gradient

$$\nabla f(x,y) = \frac{\partial f(x,y)}{\partial x}\hat{i}_x + \frac{\partial f(x,y)}{\partial y}\hat{i}_y$$

- Omnidirectional detector
  - Based on $|\nabla f(x,y)|$: isotropic behaviour
- Directional detector
  - Based on an oriented derivative:
    - ex.: a possible horizontal edge detector is $|\partial f/\partial y|$

2D approach



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

Edge thinning

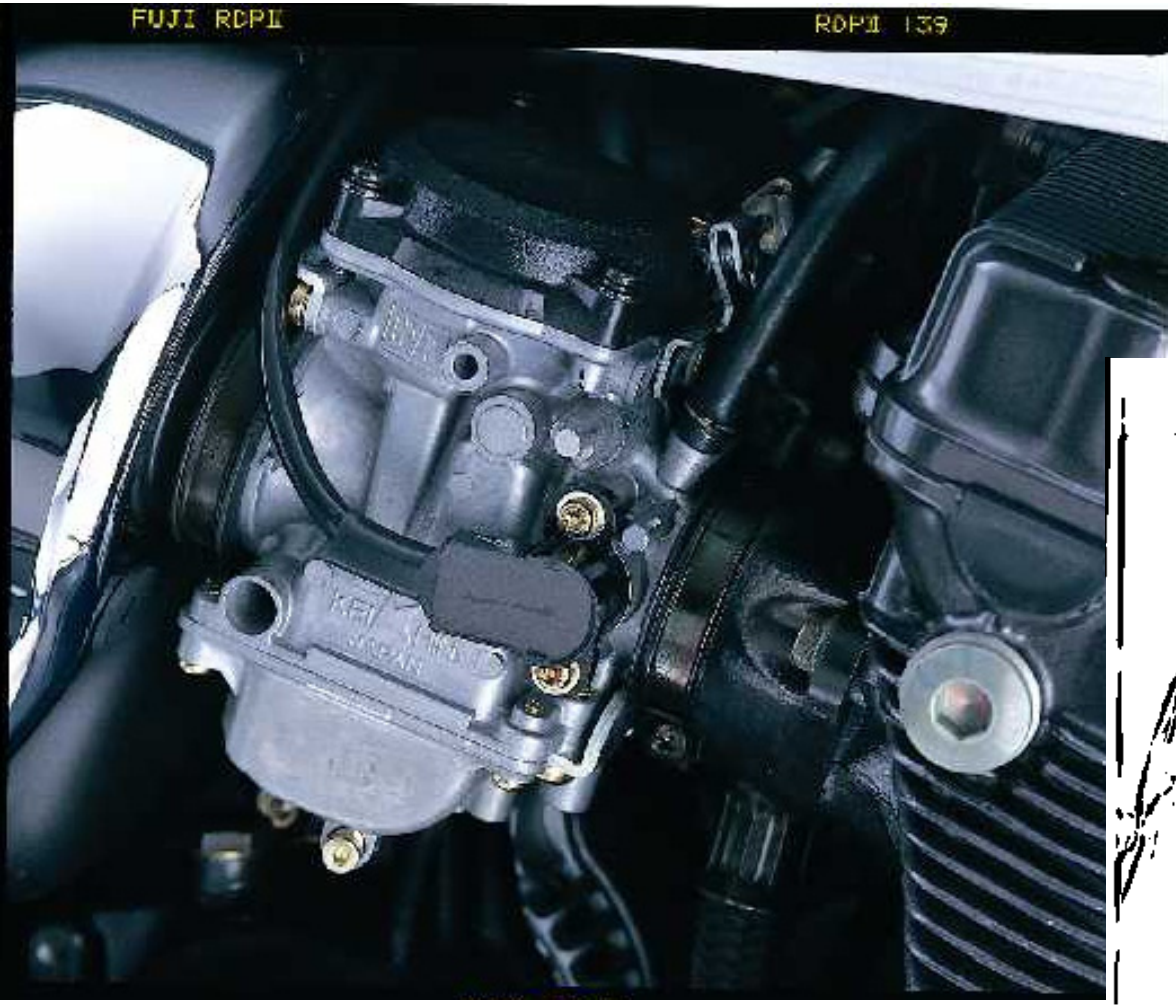
1) if $|\nabla f|$ has a local horizontal max but not a vertical one in (x_0,y_0), then, that point is an edge point if

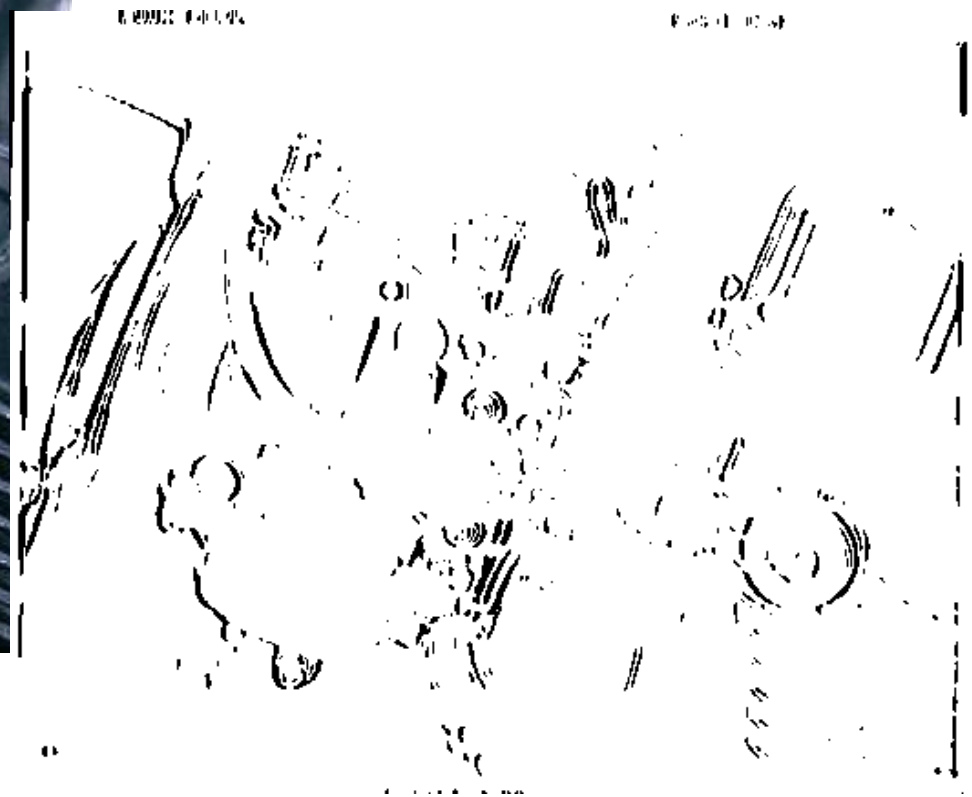
$$\left\|\frac{\partial f}{\partial x}\right\|_{(x_0,y_0)} > K\left\|\frac{\partial f}{\partial y}\right\|_{(x_0,y_0)} \qquad K = 2(\textit{typical}\ \text{value})$$

2) if $|\nabla f|$ has a local vertical max but not a horizontal one in (x_0,y_0), then, that point is an edge point if

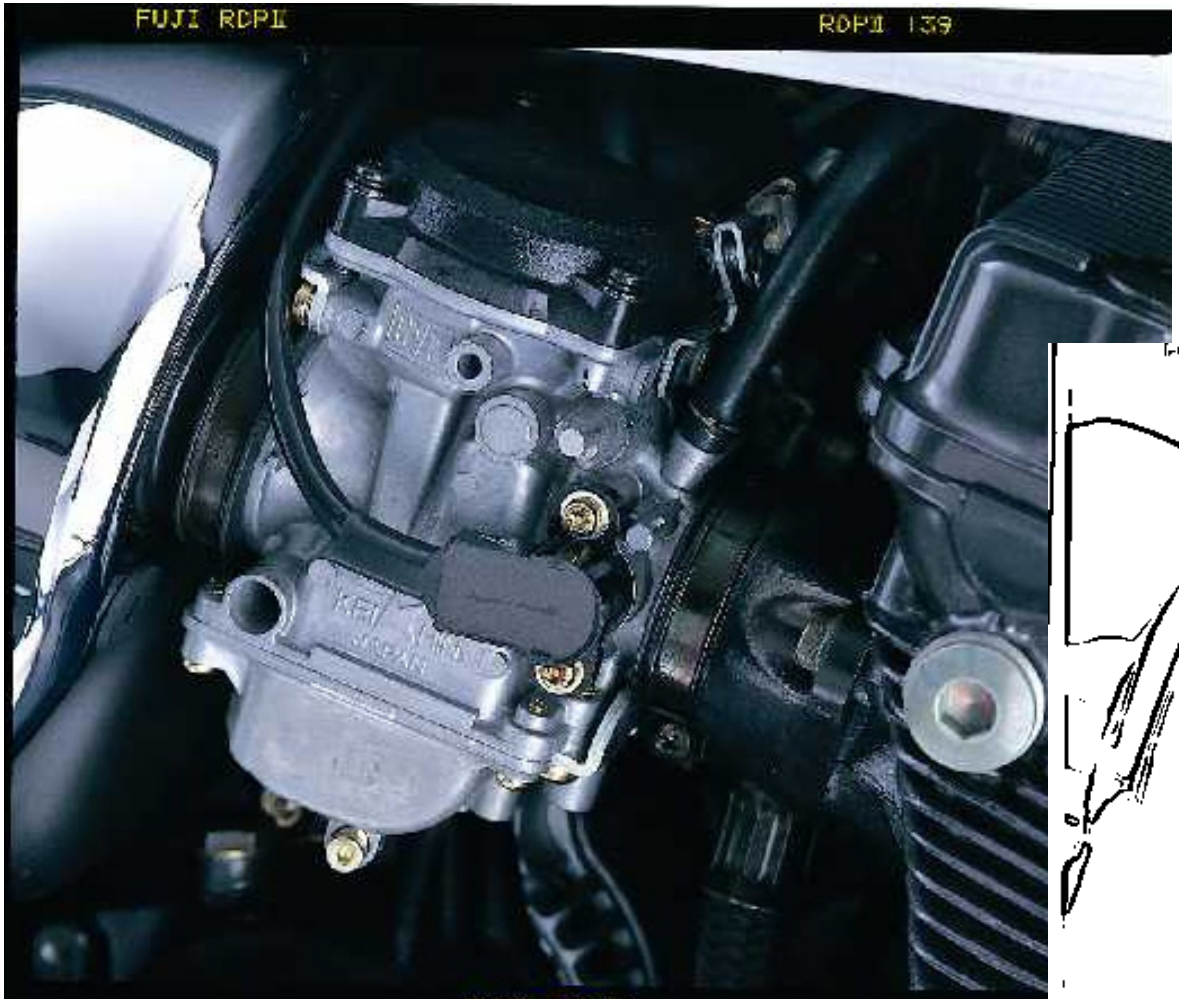
$$\left\|\frac{\partial f}{\partial y}\right\|_{(x_0,y_0)} > K\left\|\frac{\partial f}{\partial x}\right\|_{(x_0,y_o)} \qquad K = 2(\text{typical value})$$

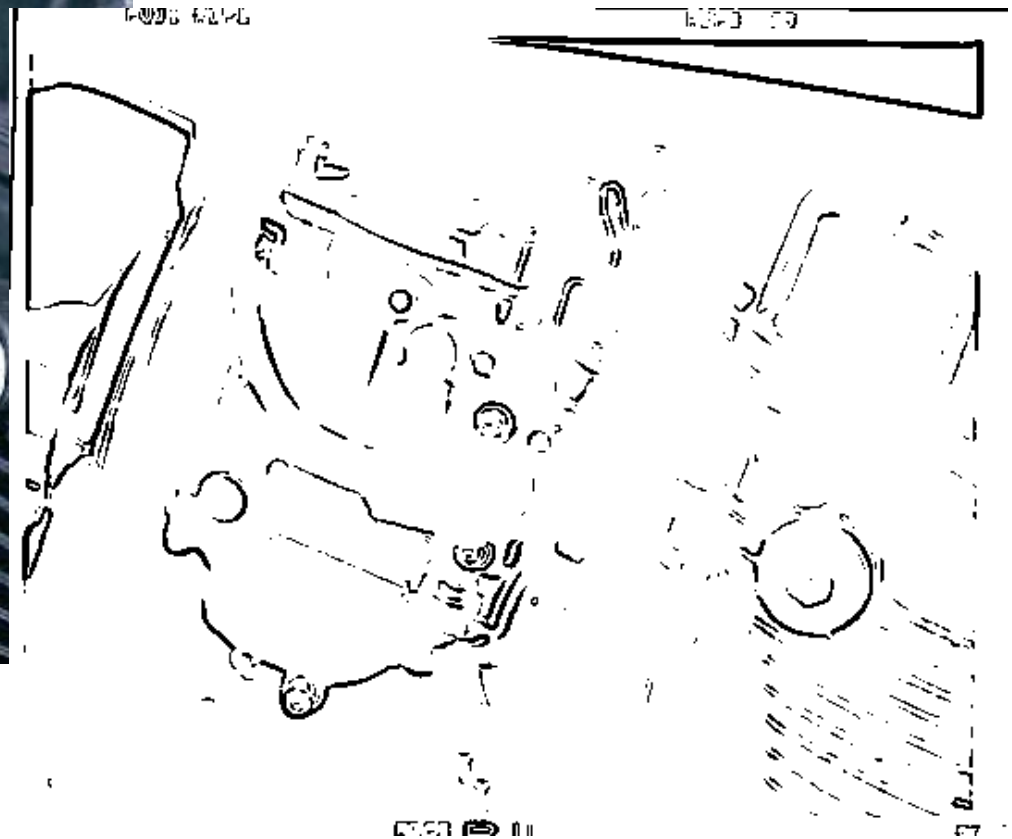
Directional case





Example: isotropic case





Discretization

The gradient operator can be discretized as:

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

$$\|\nabla f(n_1,n_2)\| = \sqrt{f_x(n_1,n_2)^2 + f_y(n_1,n_2)^2}$$

Which is based on a discretization of directional derivatives:

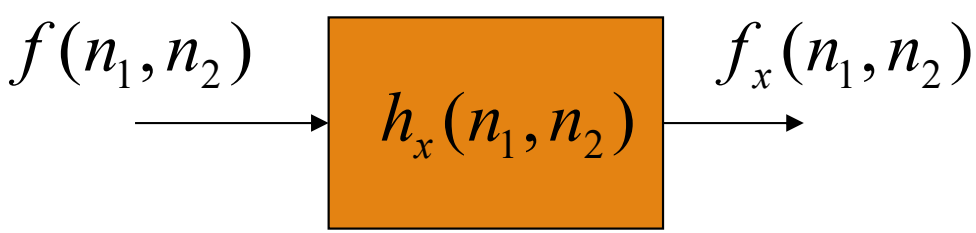
$$\|\nabla f(n_1,n_2)\| = \|f_x(n_1,n_2) + f_y(n_1,n_2)\|$$

Finite Impulse Response Model

Discrete operators for derivative estimation can be estimated as FIR filters.

$$f_y(n_1,n_2) = f(n_1,n_2) * h_y(n_1,n_2)$$

$$f_x(n_1,n_2) = f(n_1,n_2) * h_x(n_1,n_2)$$



Discrete differential operators

- Pixel difference: luminance difference between to neighbour pixels along orthogonal directions.

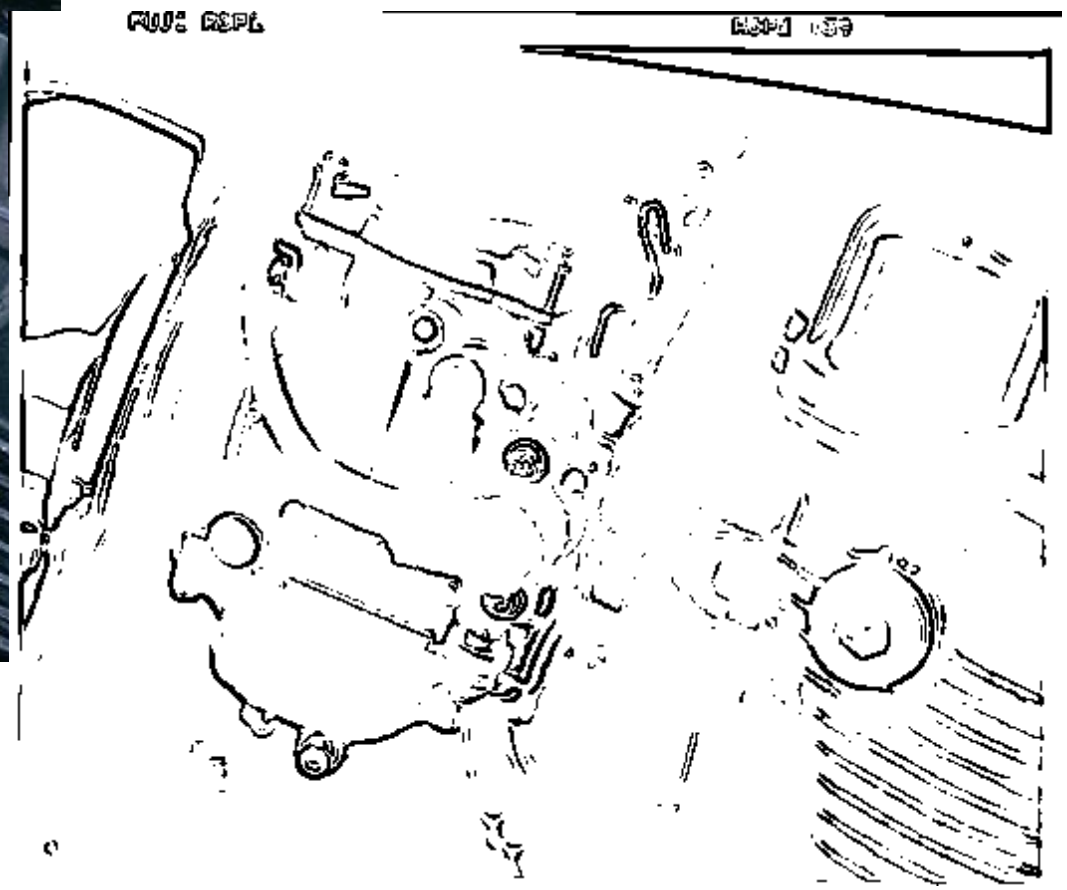
$$f_x(j,k) = f(j,k) - f(j,k-1)$$

$$f_y(j,k) = f(j,k) - f(j+1,k)$$

Separable filters

$$h_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \qquad h_y = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Example: pixel difference



Separated pixel difference

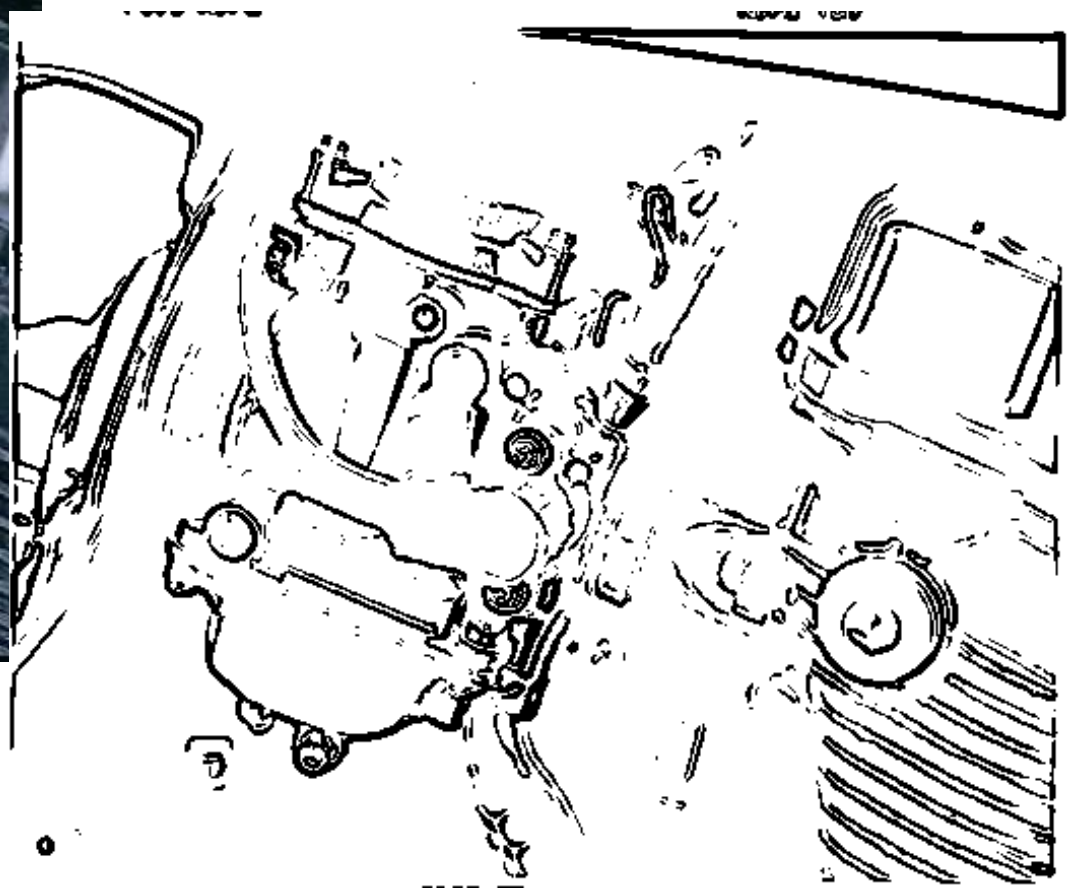
- If farther pixels are chosen there is a higher noise rejection, and there is no phase translation in edge definition.

$$f_x(j,k) = f(j,k+1) - f(j,k-1)$$

$$f_y(j,k) = f(j-1,k) - f(j+1,k)$$

$$h_x = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad h_y = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Ex.: separated pixel difference



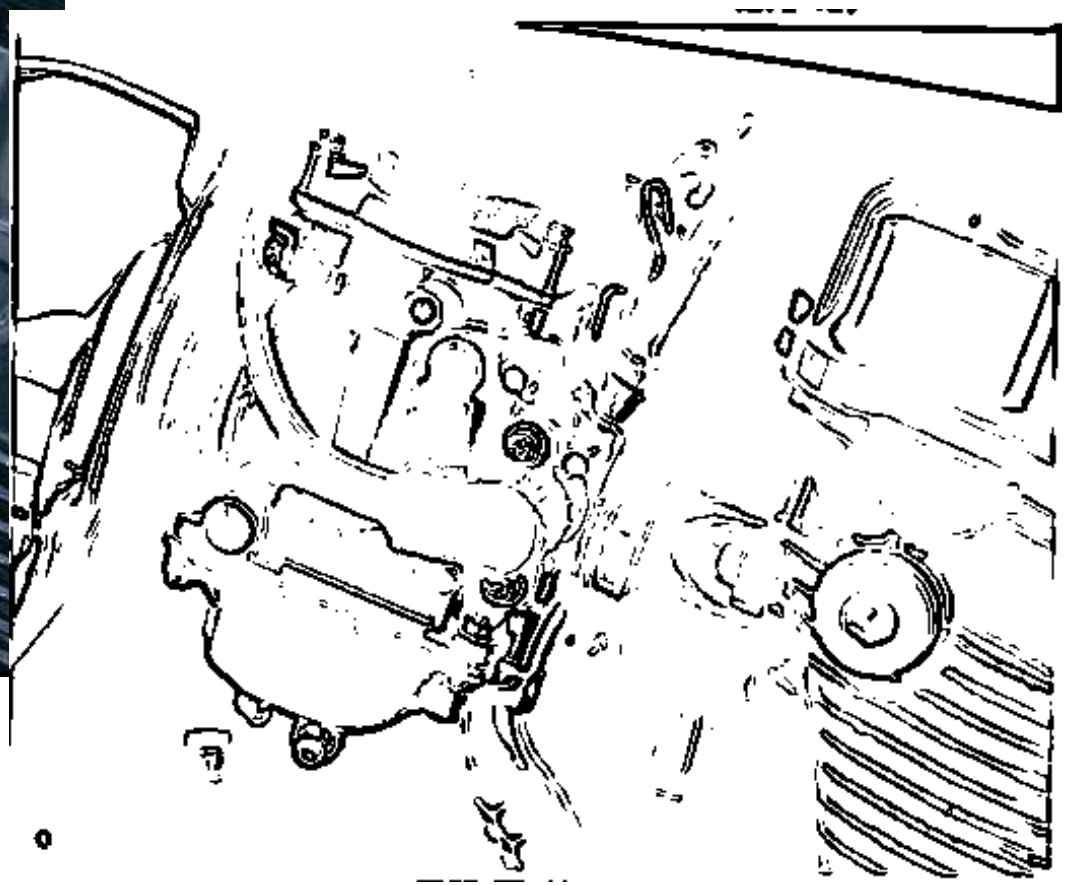
Roberts edge extraction

$$f_x(j,k) = f(j,k) - f(j+1,k+1)$$

$$f_y(j,k) = f(j,k+1) - f(j+1,k)$$

$$h_x = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad h_y = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Ex.: Roberts method



Prewitt method

Estimation can be improved involving more samples for te gradient operator 3x3

$$K*\left(\frac{\partial f}{\partial x}\right)\cong\left[f\left(n_1+1,n_2+1\right)-f\left(n_1-1,n_2+1\right)\right]+$$
$$+\left[f\left(n_1+1,n_2\right)-f\left(n_1-1,n_2\right)\right]+\left[f\left(n_1+1,n_2-1\right)-f\left(n_1-1,n_2-1\right)\right]$$

= vertical low pass* horizontal high pass

$$K*\left(\frac{\partial f}{\partial y}\right)\cong\left[f\left(n_1+1,n_2+1\right)-f\left(n_1+1,n_2-1\right)\right]+$$
$$+\left[f\left(n_1,n_2+1\right)-f\left(n_1,n_2-1\right)\right]+\left[f\left(n_1-1,n_2+1\right)-f\left(n_1-1,n_2-1\right)\right]$$

= vertical high pass* horizontal low pass

Gradient estimation

- Gradient modulus = the value of the higher directional derivative
- Gradient phase = orientation of the higher directional derivative

Squared lattice =eight possible directions

$$h(n_1,n_2)=\begin{bmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{bmatrix}$$

$\rightarrow$ E

$$h(n_1,n_2)=\begin{bmatrix}0 & 1 & 1\\ -1 & 0 & 1\\ -1 & -1 & 0\end{bmatrix}$$

$\nearrow$ NE

$$h(n_1,n_2)=\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix}$$

$\uparrow$ N

$$h(n_1,n_2)=\begin{bmatrix}1 & 1 & 0\\ 1 & 0 & -1\\ 0 & -1 & -1\end{bmatrix}$$

$\nwarrow$ NW

Gradient estimation

$$h(n_1,n_2)=\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix}$$

$\leftarrow$ W

$$h(n_1,n_2)=\begin{bmatrix}0 & -1 & -1\\ 1 & 0 & -1\\ 1 & 1 & 0\end{bmatrix}$$

$\swarrow$ SW

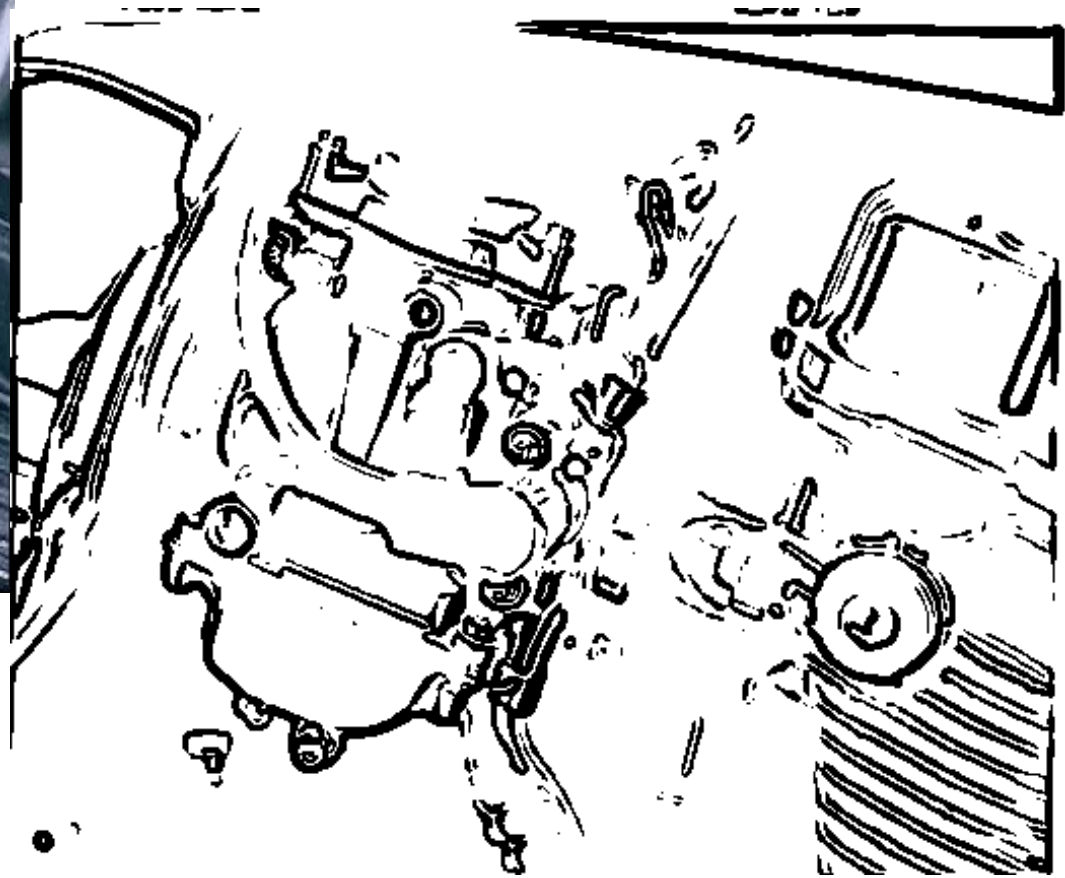
$$h(n_1,n_2)=\begin{bmatrix}-1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{bmatrix}$$

$\downarrow$ S

$$h(n_1,n_2)=\begin{bmatrix}-1 & -1 & 0\\ -1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix}$$

$\searrow$ SE

Ex.: Prewitt method 3x3



Sobel method

Same dimensions of Prewitt filter

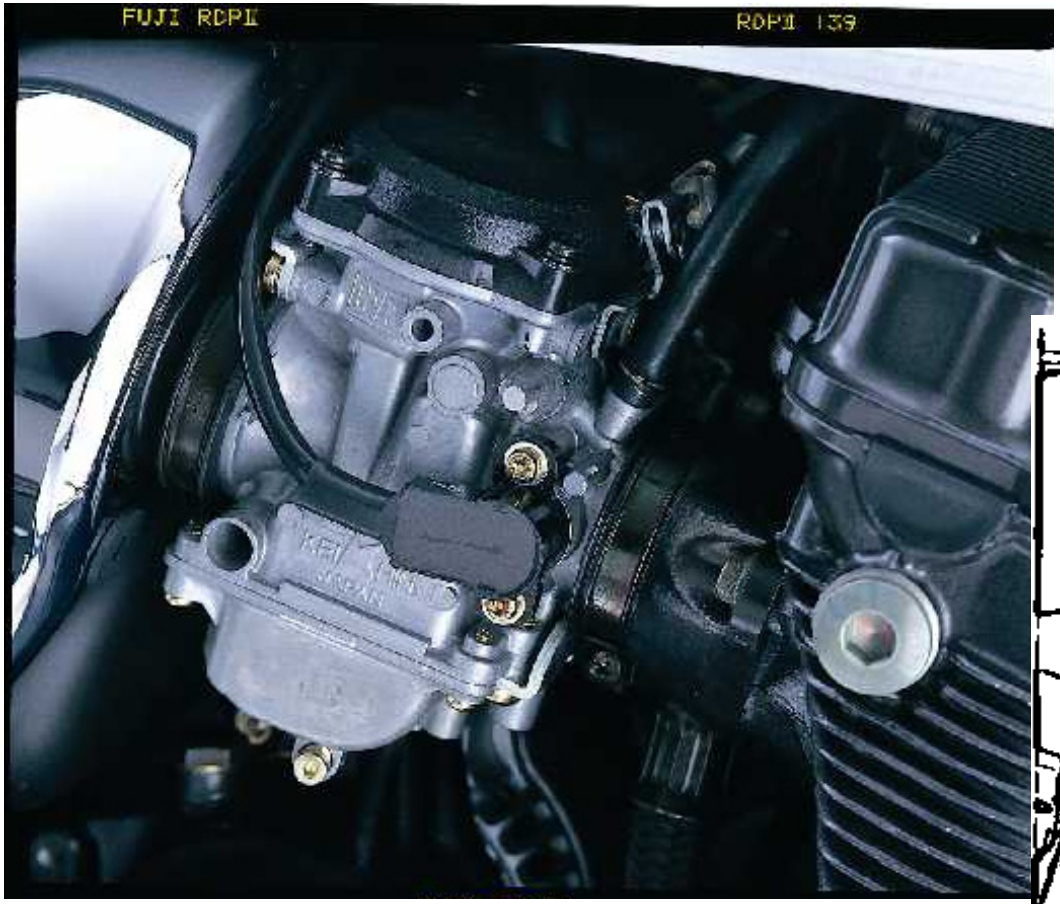
Different weight for central point in different directions.

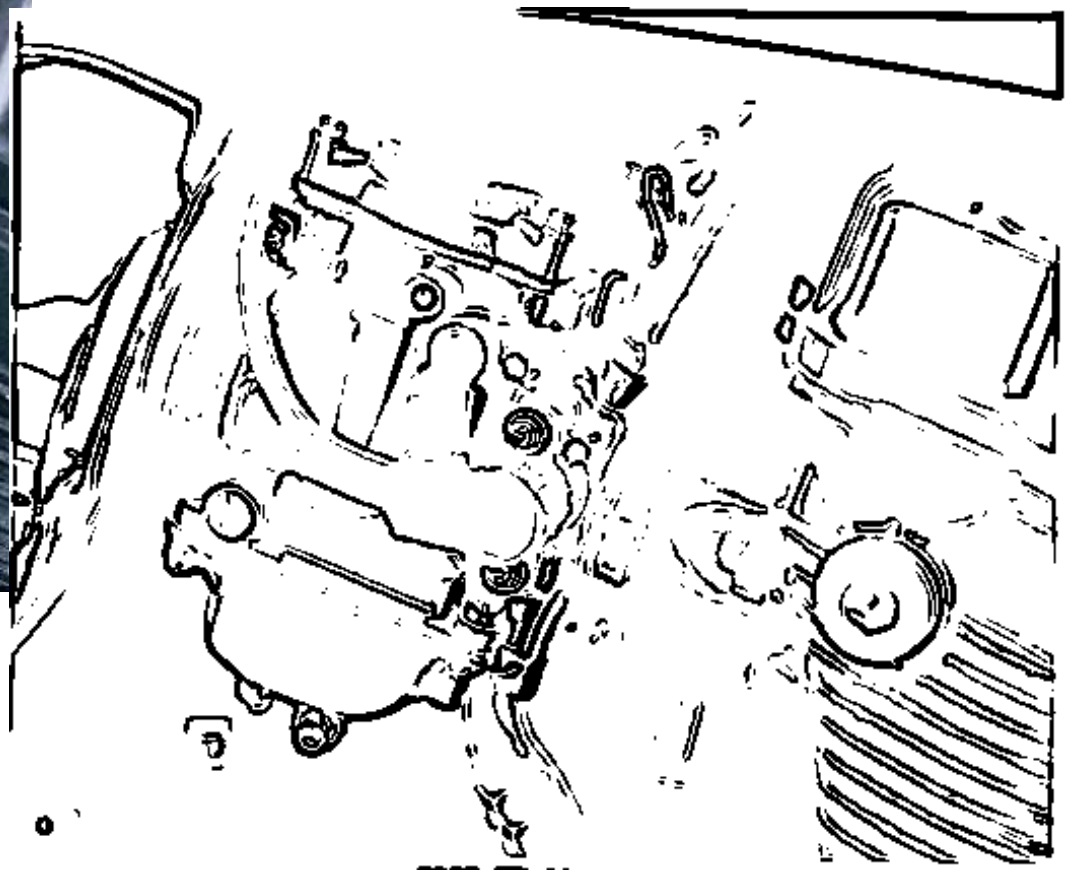
$$h_r = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad h_c = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Sobel for exagonal grids.

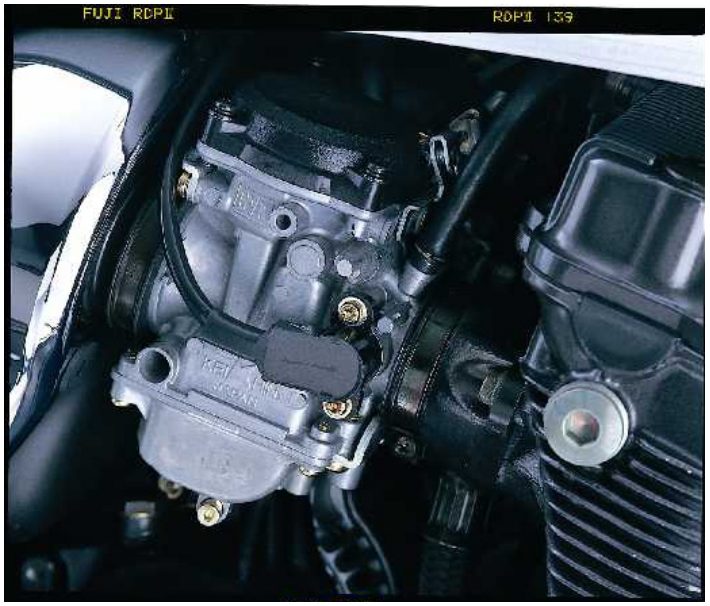
$$h = \begin{bmatrix} & -1 & 1 & \\ -2 & 0 & 2 \\ & -1 & 1 & \end{bmatrix}$$

Ex.: Sobel method

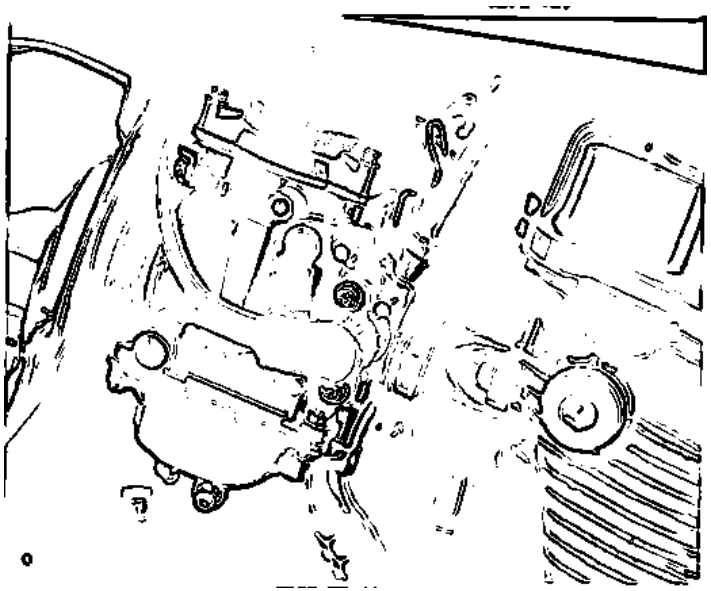




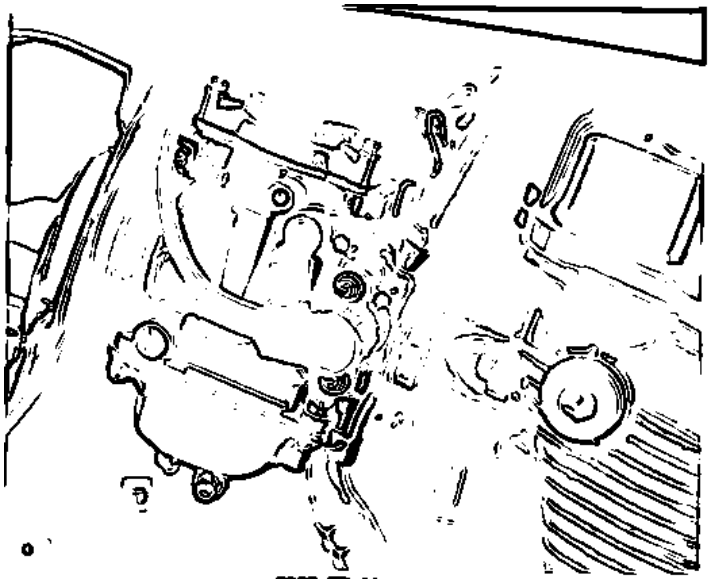
Comparison



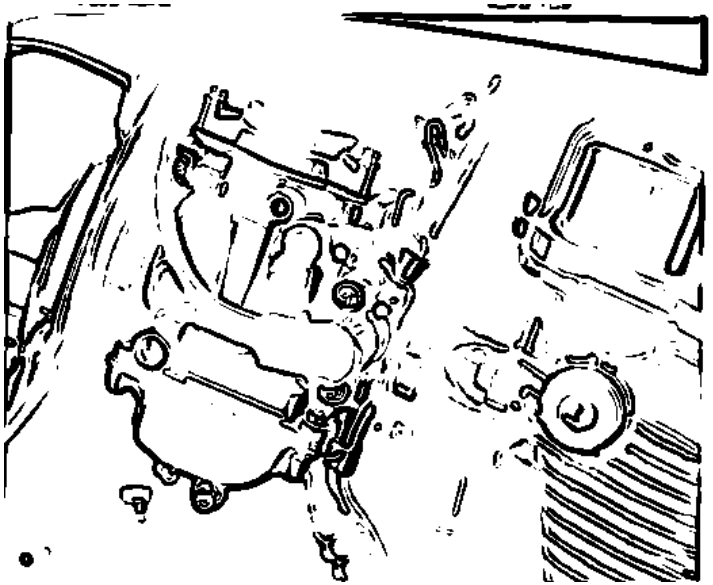
Roberts



Sobel



Prewitt



Frei-Chen operator

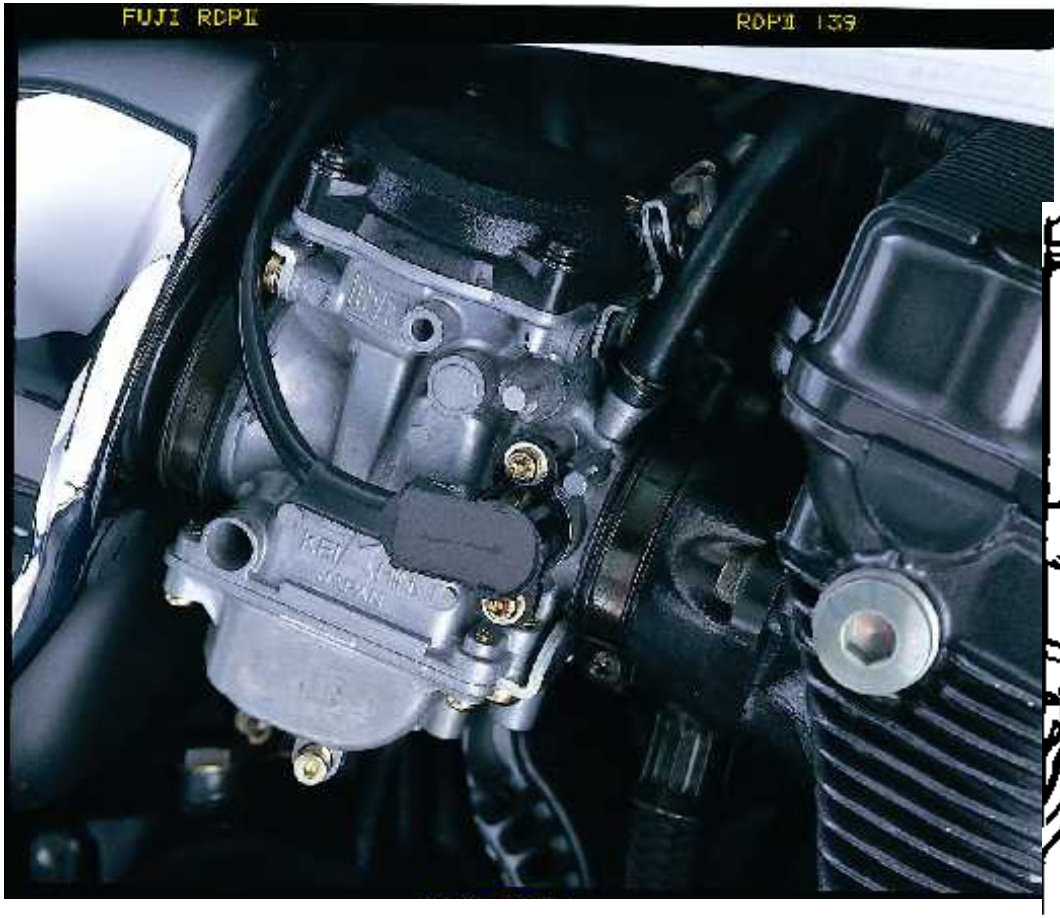
Isotropic operator similar to Prewitt

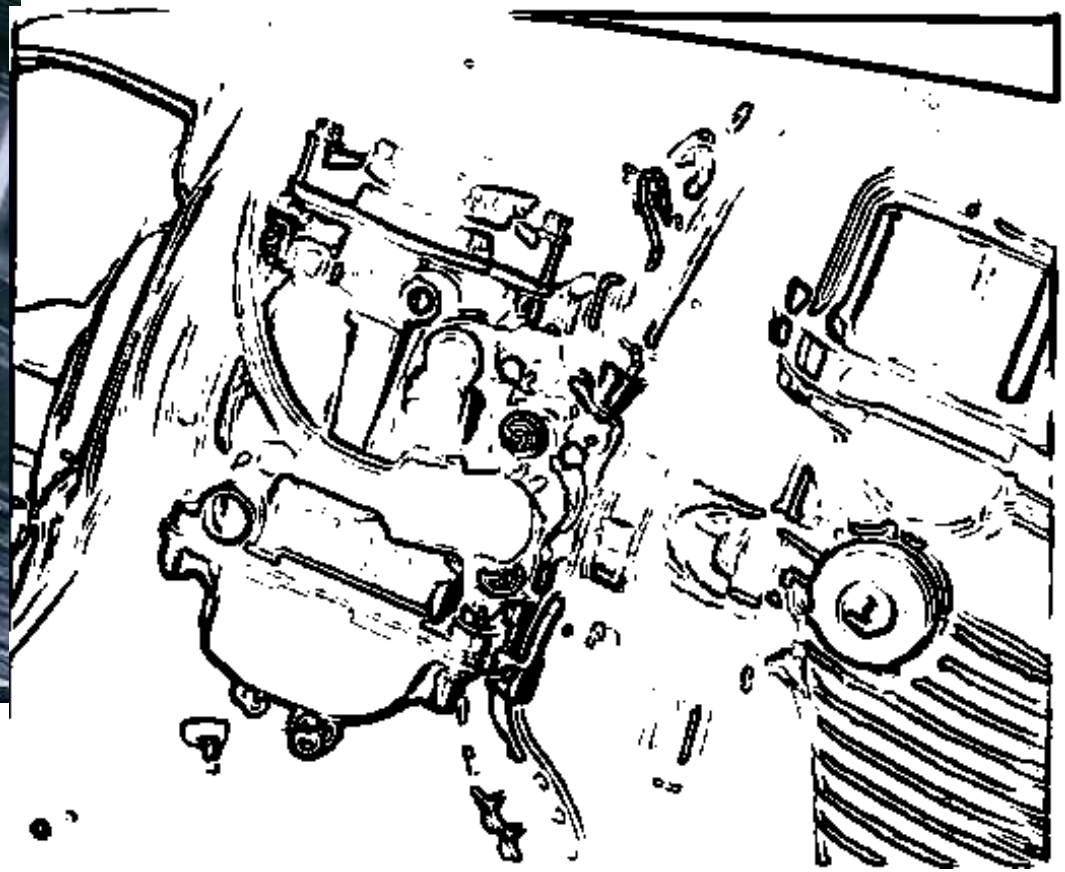
different weight for the central point in the 4 directions.

The gradient has the same value for horizontal, vertical and diagonal edges.

$$h_r = \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix} \qquad h_c = \begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$$

Ex.: Frei-Chen method





Extended operators

A limit for the aforementioned methods is their weakness in accurate edge detection when SNR is very low.

A possible solution in to extend their size: the result will be a less accurate edge positioning but noise rejection will be higher.

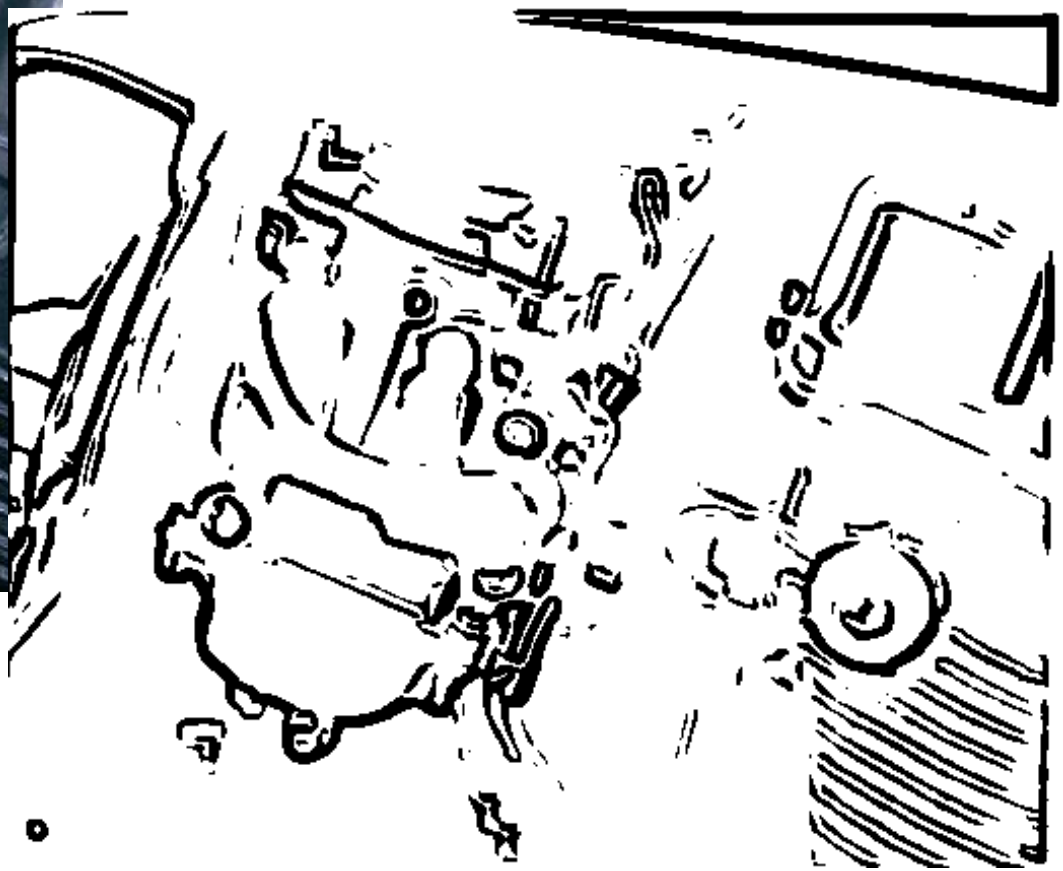
PREWITT METHOD 7X7

Extension of Prewitt 3X3

- Normalized impulse response:

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

Ex.: Prewitt 7x7 method



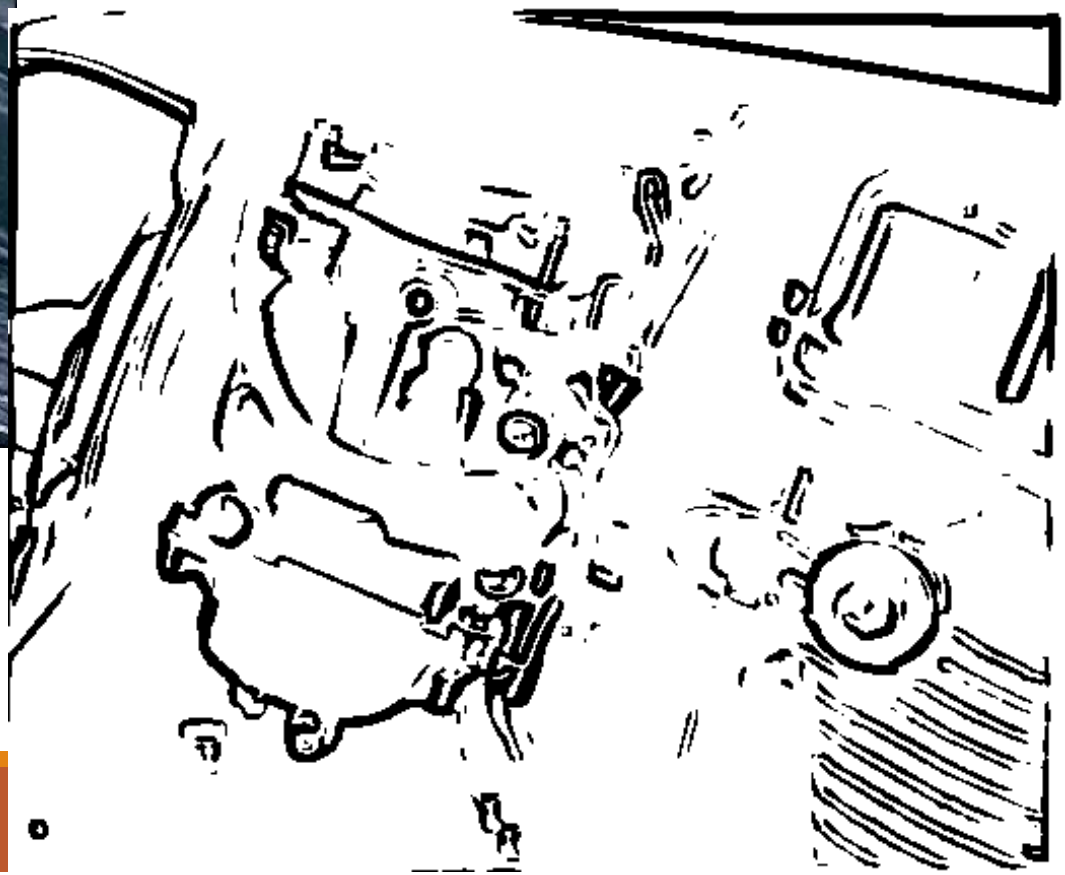
Abdou 7X7 Method

Is a filter mask that gives a linear decreasing sample weight as they are farther from the edge. Its behaviour is close to a truncated pyramid.

The normalized impulse response is:

$$h_r = \begin{bmatrix} 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 2 & 2 & 0 & -2 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 2 & 0 & -2 & -2 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix}$$

Ex.: Abdou 7x7 Method



Further extended operators

It is possible to obtain extended gradient filters for low SNR conditions convolving a 3x3 operator with a low-pass filter.

$$h(j,k) = h_G(j,k) * h_{PG}(j,k)$$

H_G (j,k) is one of the previously considered filters, H_{PB} (j,k) is the impulse response for a low-pass filter.

Example

Prewitt 3X3 convoled with: $$h = \frac{1}{9} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

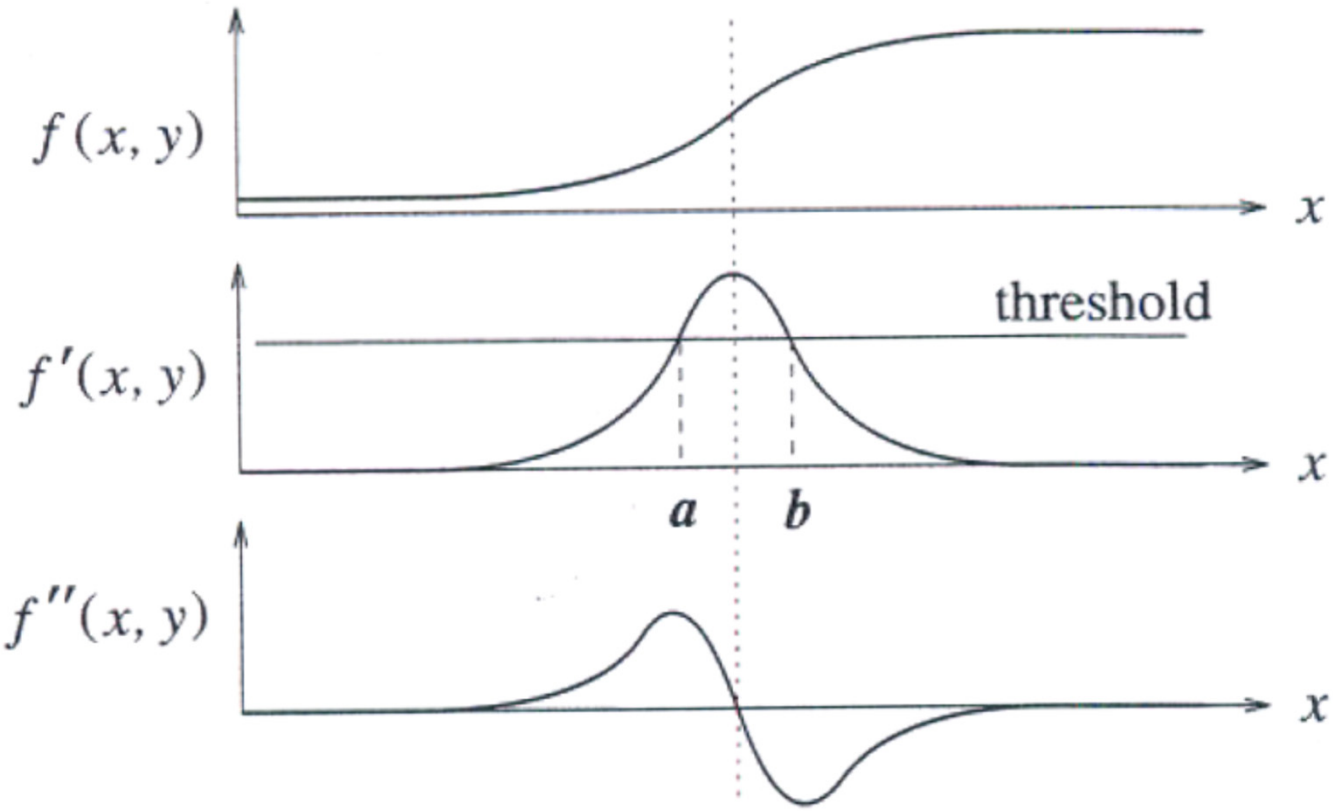
...and we get the ***Smoothed Prewitt* 5X5**

$$h = \begin{bmatrix} 1 & 1 & 0 & -1 & -1 \\ 2 & 2 & 0 & -2 & -2 \\ 3 & 3 & 0 & -3 & -3 \\ 2 & 2 & 0 & -2 & -2 \\ 1 & 1 & 0 & -1 & -1 \end{bmatrix}$$

Laplacian based methods

1D Case

- Find zero-crossing of the second derivative, corresponding to inflection points.



Laplacian

- For the 2D case the 2nd order differential operator is the Laplacian

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

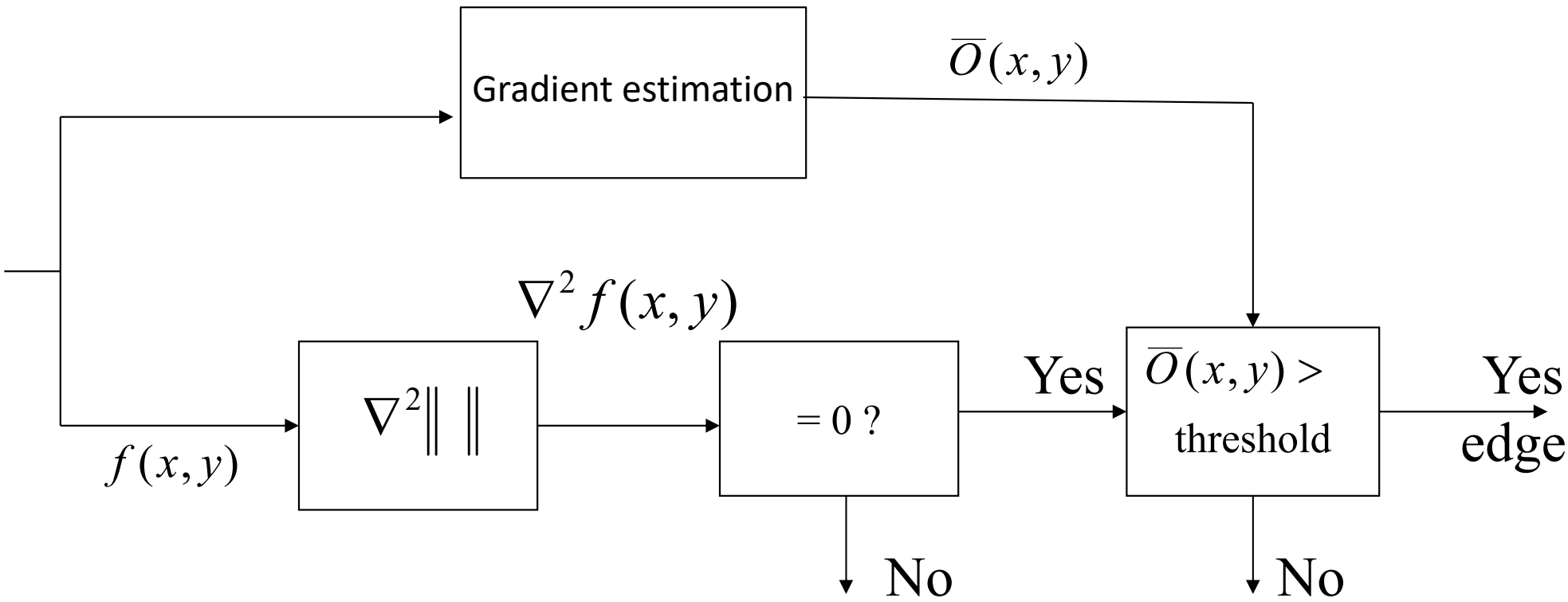
Isotropic operator

More sensible to noise with respect to gradient

False edges can be generated due to noise.

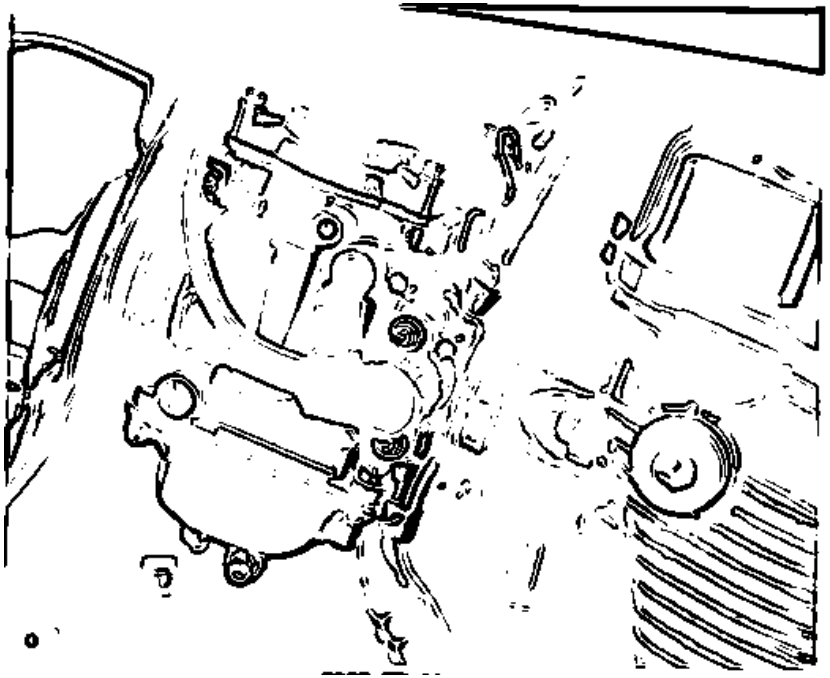
Thinner edges are produced.

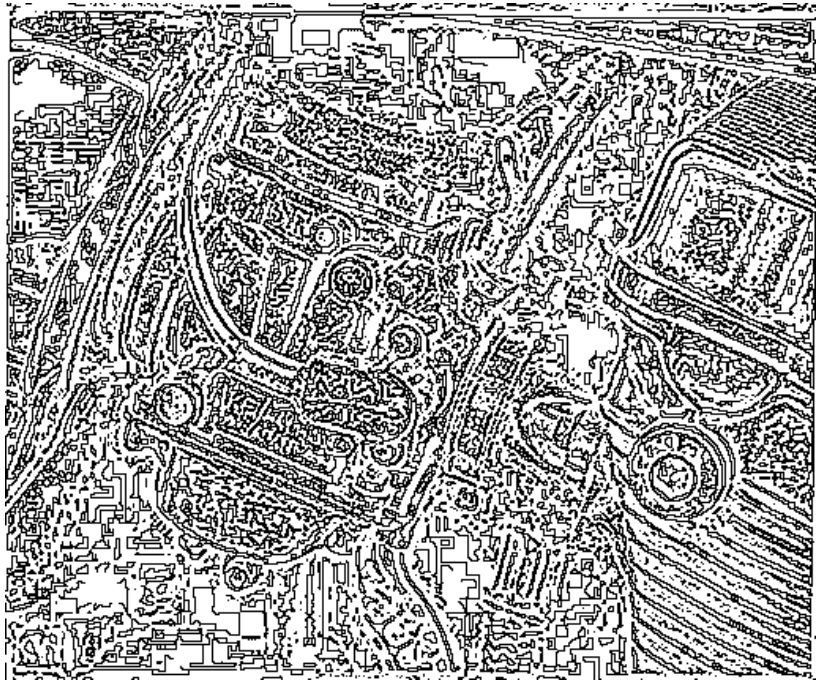
Algorithm: case 2D



Zero-crossing without threshold

Sobel vs. Laplacian





Laplacian Discretization

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

Can be seen as the convolution of $f(n_1,n_2)$ with the impulse response $h(n_1,n_2)$ of a linear system.

$$\nabla^2 f(n_1,n_2) = f(n_1,n_2) * h(n_1,n_2)$$

4 neighbours method

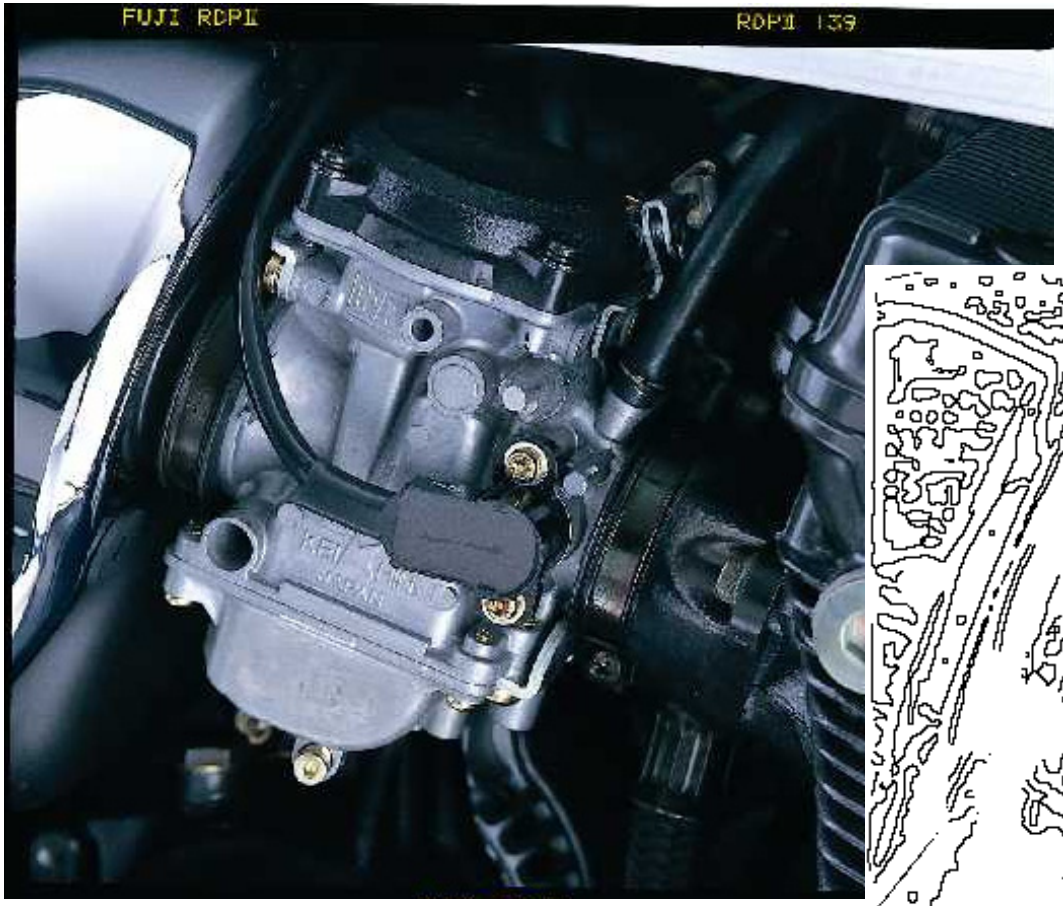
Separable normalized filter

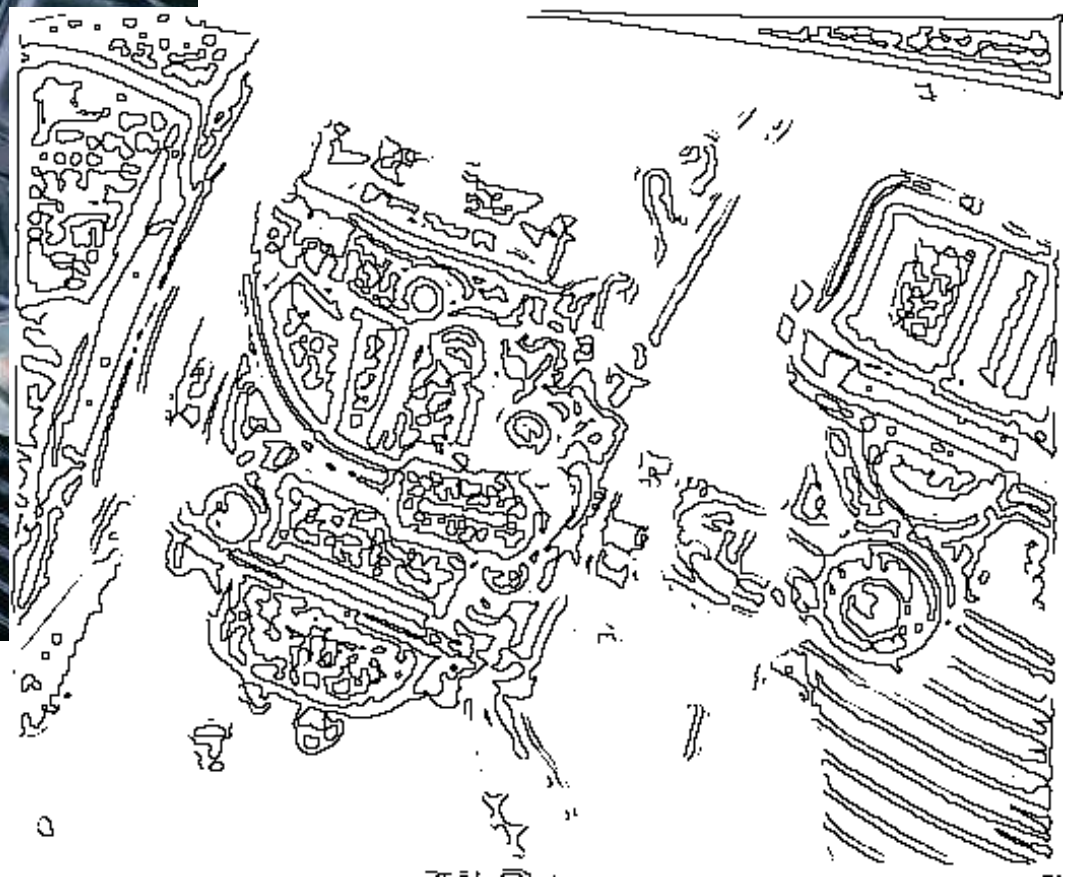
- Unit gain for continuous component
- The sign of **$h(n_1,n_2)$** can be changed without changes in the final result (since we are looking for zeros of laplacian*)*

$$h(n_1,n_2)=\frac{1}{4}\begin{bmatrix}0 & 0 & 0\\ -1 & 2 & -1\\ 0 & 0 & 0\end{bmatrix}+\frac{1}{4}\begin{bmatrix}0 & -1 & 0\\ 0 & 2 & 0\\ 0 & -1 & 0\end{bmatrix}$$

$$=\frac{1}{4}\begin{bmatrix}0 & -1 & 0\\ -1 & 4 & -1\\ 0 & -1 & 0\end{bmatrix}$$

Ex.: 4 neighbours method





Laplacian Discretization

The laplacian can be approximated with finite differences

$$\frac{\partial f(x,y)}{\partial x} \Rightarrow f_x(x,y) = f(j+1,k) - f(j,k)$$

$$\frac{\partial^2 f(x,y)}{\partial x^2} \Rightarrow f_{xx}(j,k) = f_x(j,k) - f_x(j-1,k) =$$

$$= f(j+1,k) - 2f(j,k) + f(j-1,k)$$

Discretization examples

Prewitt method

Not separable filter

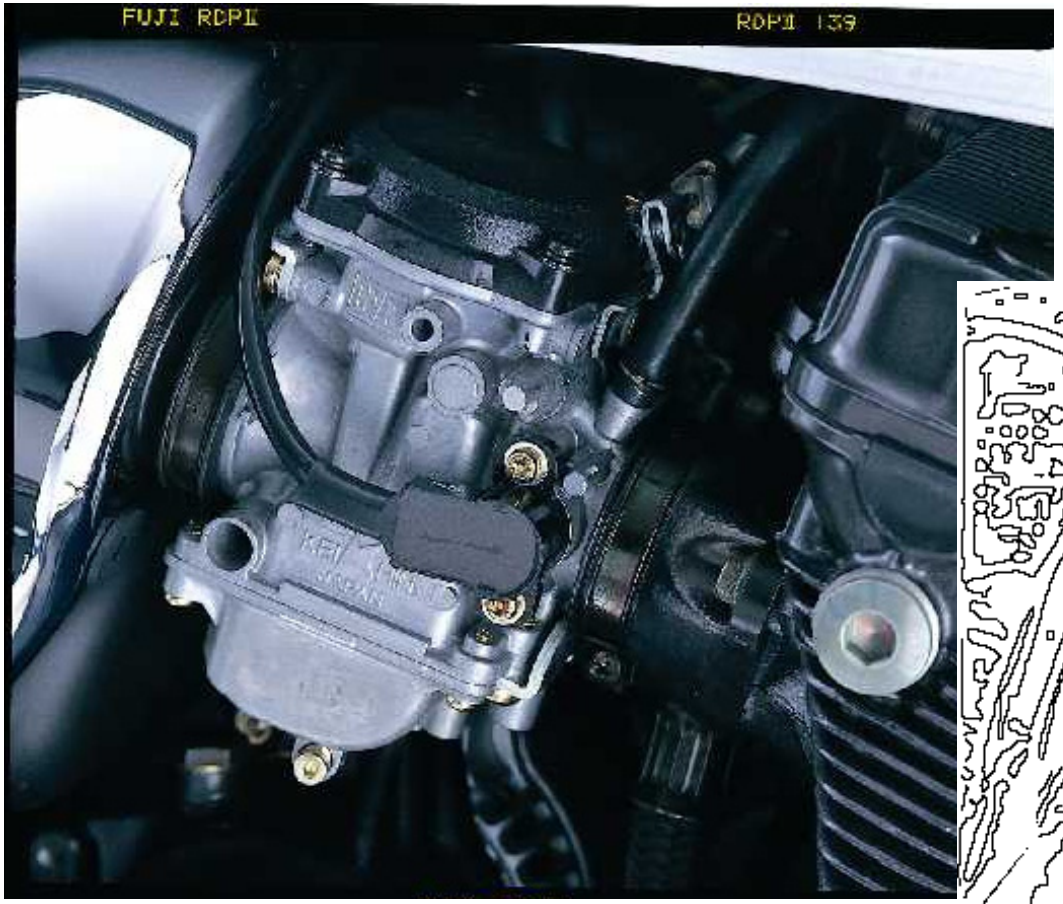
$$h(n_1,n_2)=\frac{1}{8}\begin{bmatrix}-1 & -1 & -1\\ -1 & 8 & -1\\ -1 & -1 & -1\end{bmatrix}$$

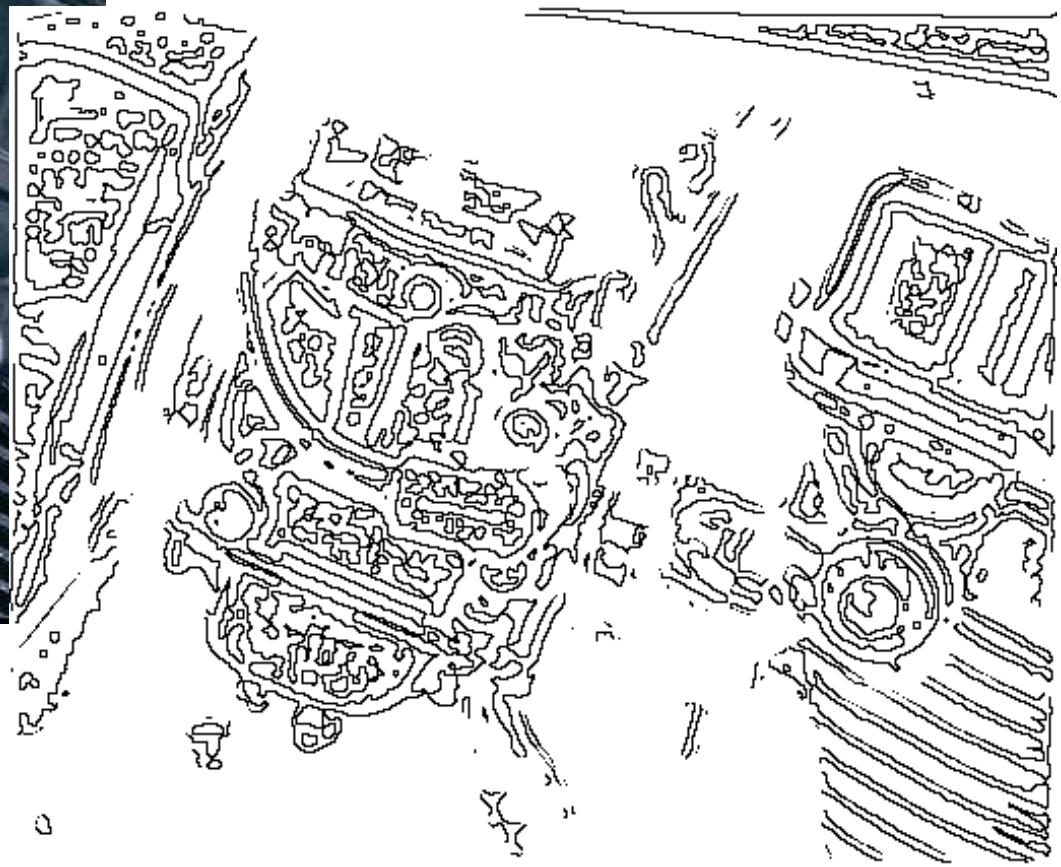
8 Neighbours method

Similar to Prewitt but with a separable formulation

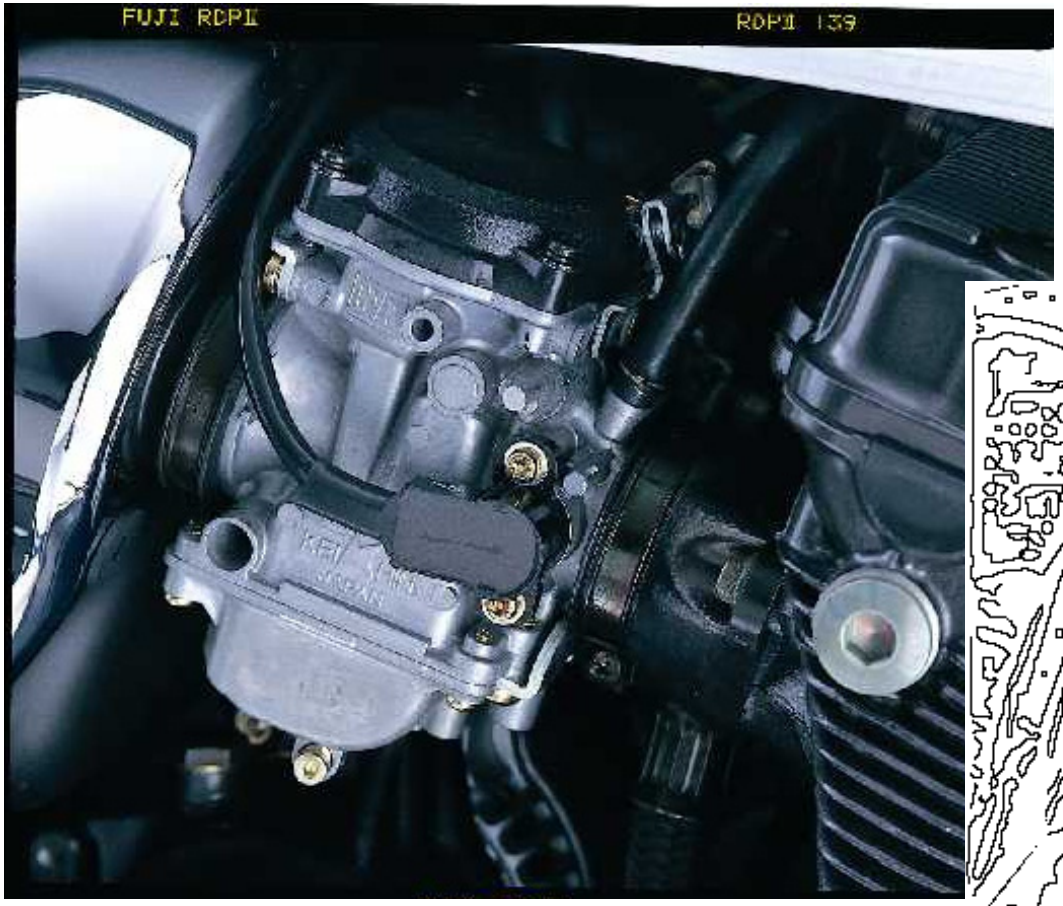
$$h(n_1,n_2)=\frac{1}{8}\begin{bmatrix}-1 & 2 & -1\\ -1 & 2 & -1\\ -1 & 2 & -1\end{bmatrix}+\frac{1}{8}\begin{bmatrix}-1 & -1 & -1\\ 2 & 2 & 2\\ -1 & -1 & -1\end{bmatrix}=\frac{1}{8}\begin{bmatrix}-2 & 1 & -2\\ 1 & 4 & 1\\ -2 & 1 & -2\end{bmatrix}$$

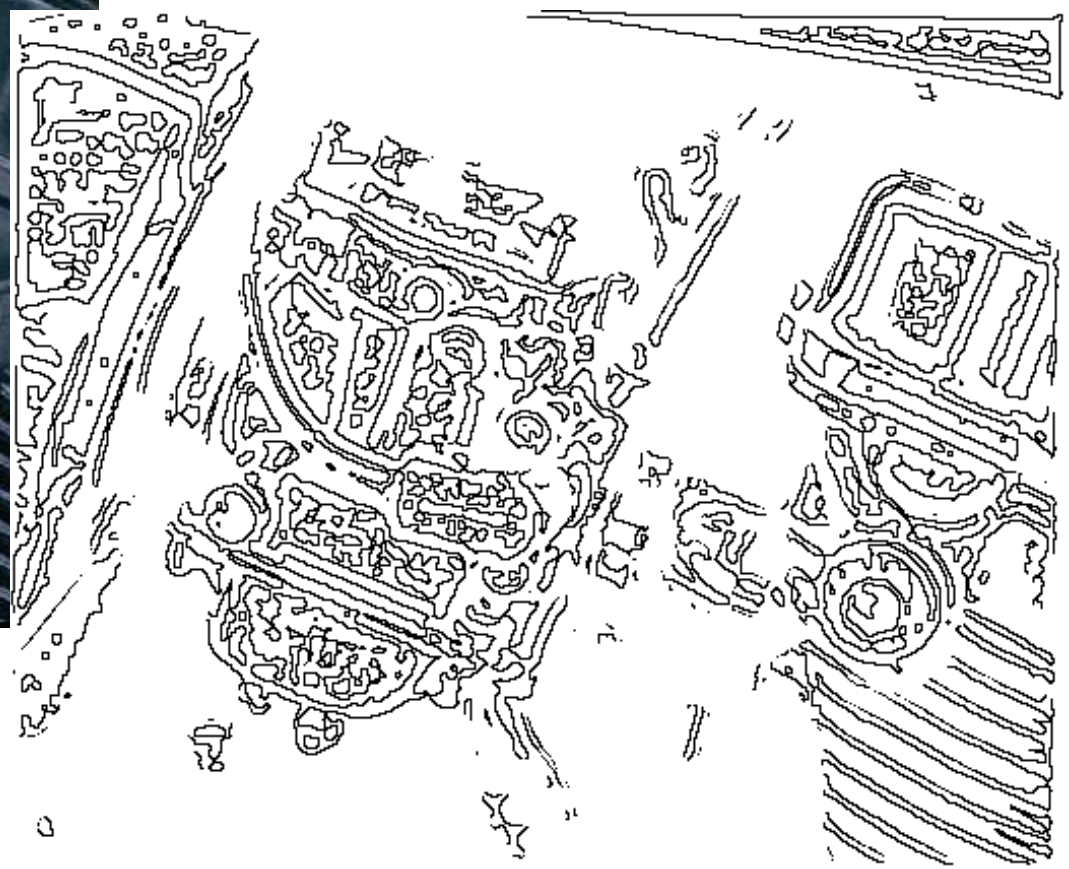
Ex.: 8 neighbors method





Ex.: Prewitt not separable





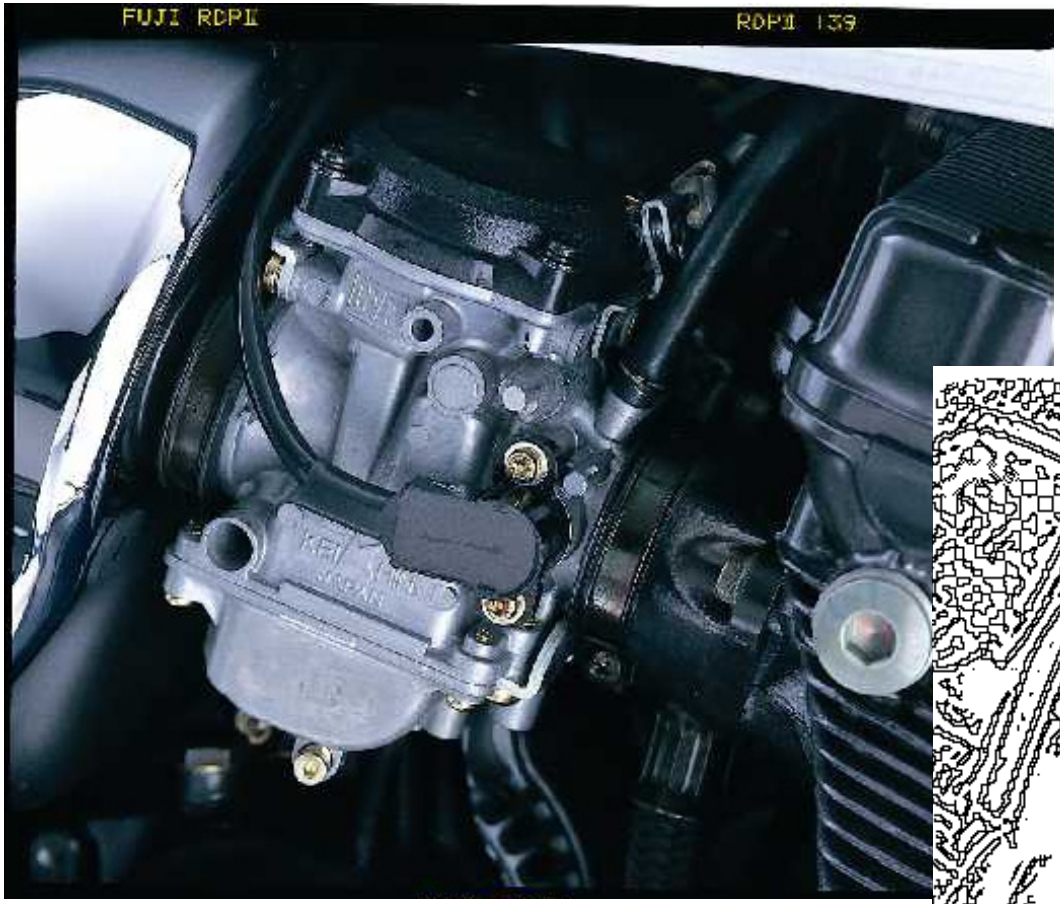
Noise presence

- When noise is significant these filters could not be accurate for diagonal edges. The Prewitt filter can work even in regions with high density of edges.

$$h(n_1, n_2) = \frac{1}{8}\begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

- Since ege are directional and noise can generate luminance variations, zero-crossing for laplacian could find non-correct edges.

Ex.: Laplacian for diagonal edges



Super-resolution (Laplacian)

First method.

Given two neighbour pixels, mark as possible edge point the intrapixels points if the laplacian values in the two pixels have different signs.

Assume as effective edge the point, among them, with the largest gradient.

Apply this analysis to all the pixels couples.

$$\begin{bmatrix} I_4 & I_3 & I_2 \\ I_5 & I_0 & I_1 \\ I_6 & I_7 & I_8 \end{bmatrix} \qquad \begin{bmatrix} + & + & + \\ - & - & + \\ - & + & + \end{bmatrix}$$

superresolution

Second method: analytical approach

Approximate the continuous form of function $f(n_1,n_2)$ with a 2D polynomial in order to describe the laplacian in an analytical way.

Polynomial example:

$$\hat{F}(r,c) = K_1 + K_2 r + K_3 c + K_4 r^2 + K_5 rc + K_6 c^2 + K_7 r^2 c + K_8 rc^2 + K_9 r^2 c^2$$

where K_i are the weights obtained from the discrete image.

r and c then become continuous variables associated to a discrete image matrix.

Polynomial formulation can be found with small efforts.

$$\frac{-(W-1)}{2} \leq r,c \leq \frac{(W-1)}{2}$$

Comparing Edge Operators

Gradient:

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

Roberts (2 x 2):

0	1
-1	0

1	0
0	-1

Prewitt (3 x 3):

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	1

Prewitt (5 x 5):

-1	-2	0	2	1
-2	-3	0	3	2
-3	-5	0	5	3
-2	-3	0	3	2
-1	-2	0	2	1

1	2	3	2	1
2	3	5	3	2
0	0	0	0	0
-2	-3	-5	-3	-2
-1	-2	-3	-2	-1

Good Localization
Noise Sensitive
Poor Detection

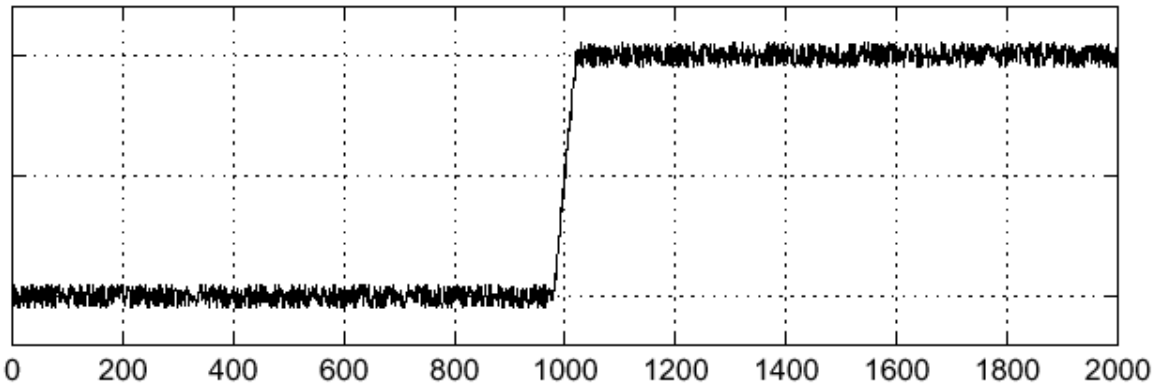
Poor Localization
Less Noise Sensitive
Good Detection

Effects of Noise

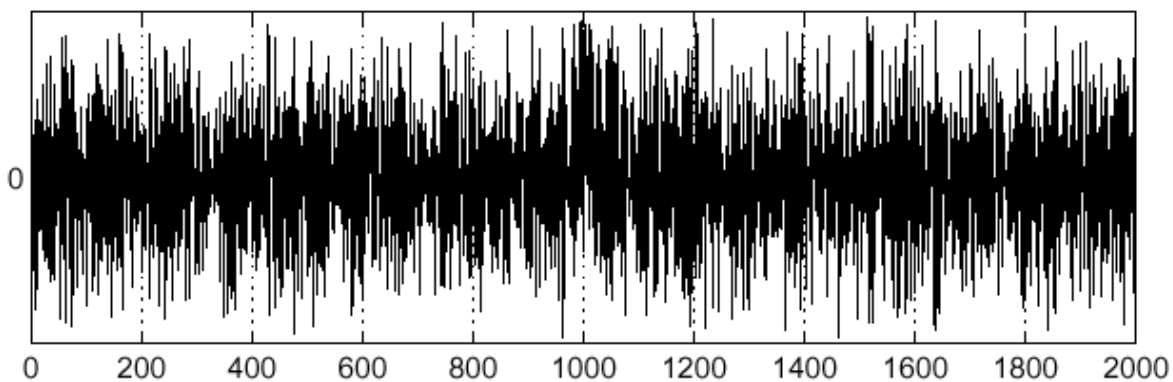
Consider a single row or column of the image

- Plotting intensity as a function of position gives a signal

$f(x)$



$\frac{d}{dx}f(x)$



Where is the edge??

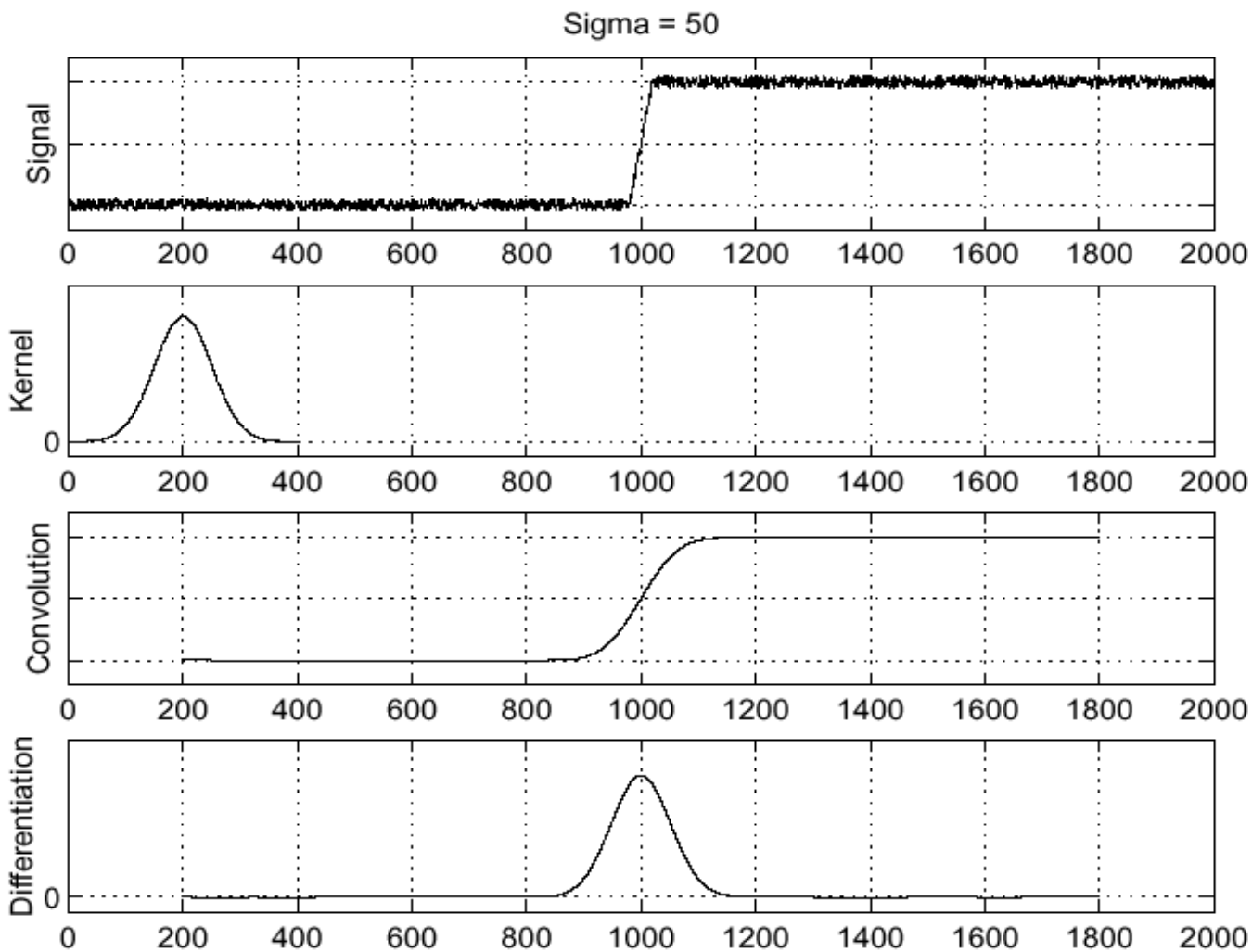
Solution: Smooth First

f

h

$h \star f$

$\frac{\partial}{\partial x}(h \star f)$



Where is the edge?

Look for peaks in $\frac{\partial}{\partial x}(h \star f)$

Derivative Theorem of Convolution

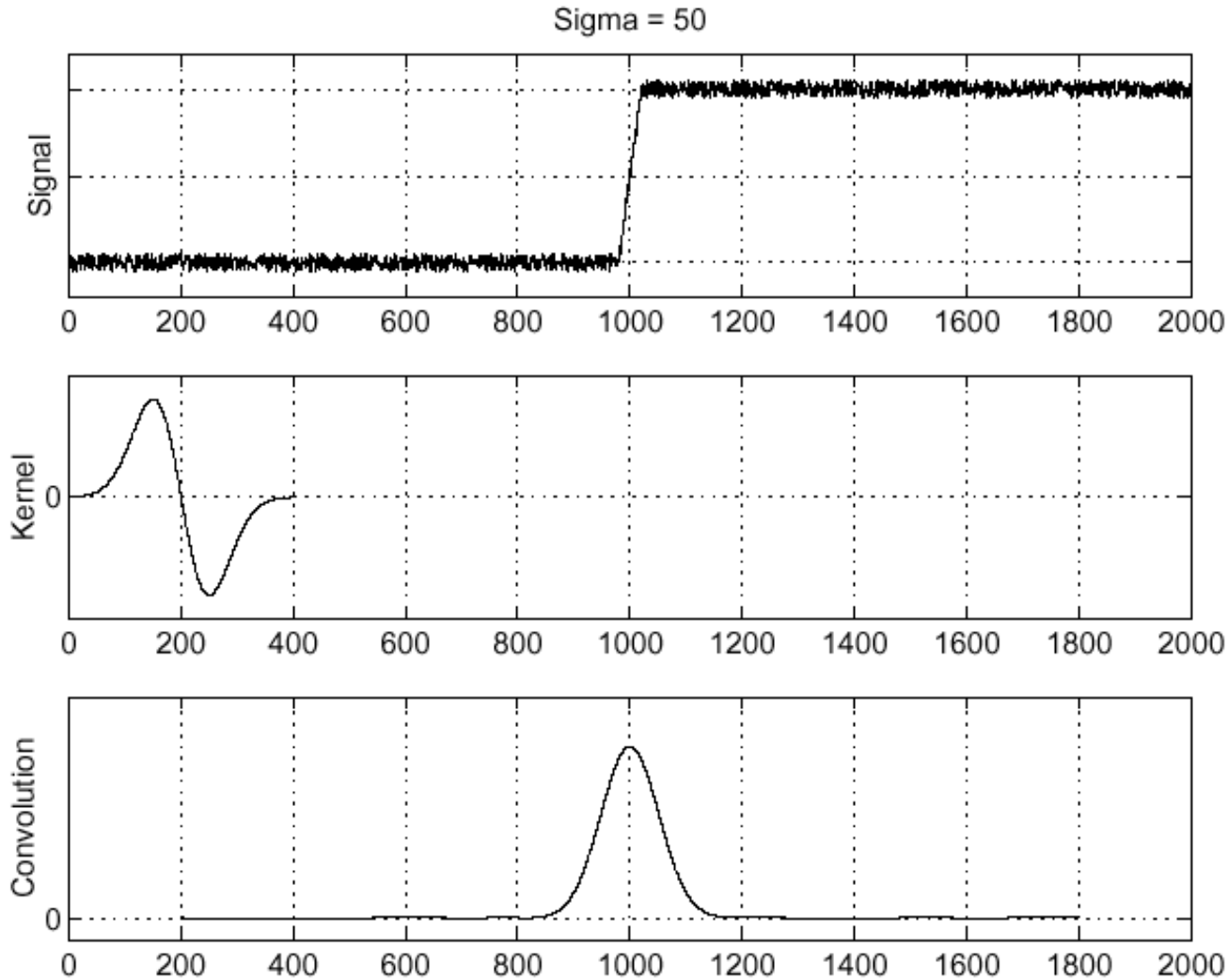
$$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$

…saves us one operation.

f

$\frac{\partial}{\partial x}h$

$(\frac{\partial}{\partial x}h) \star f$



Laplacian of Gaussian (LoG)

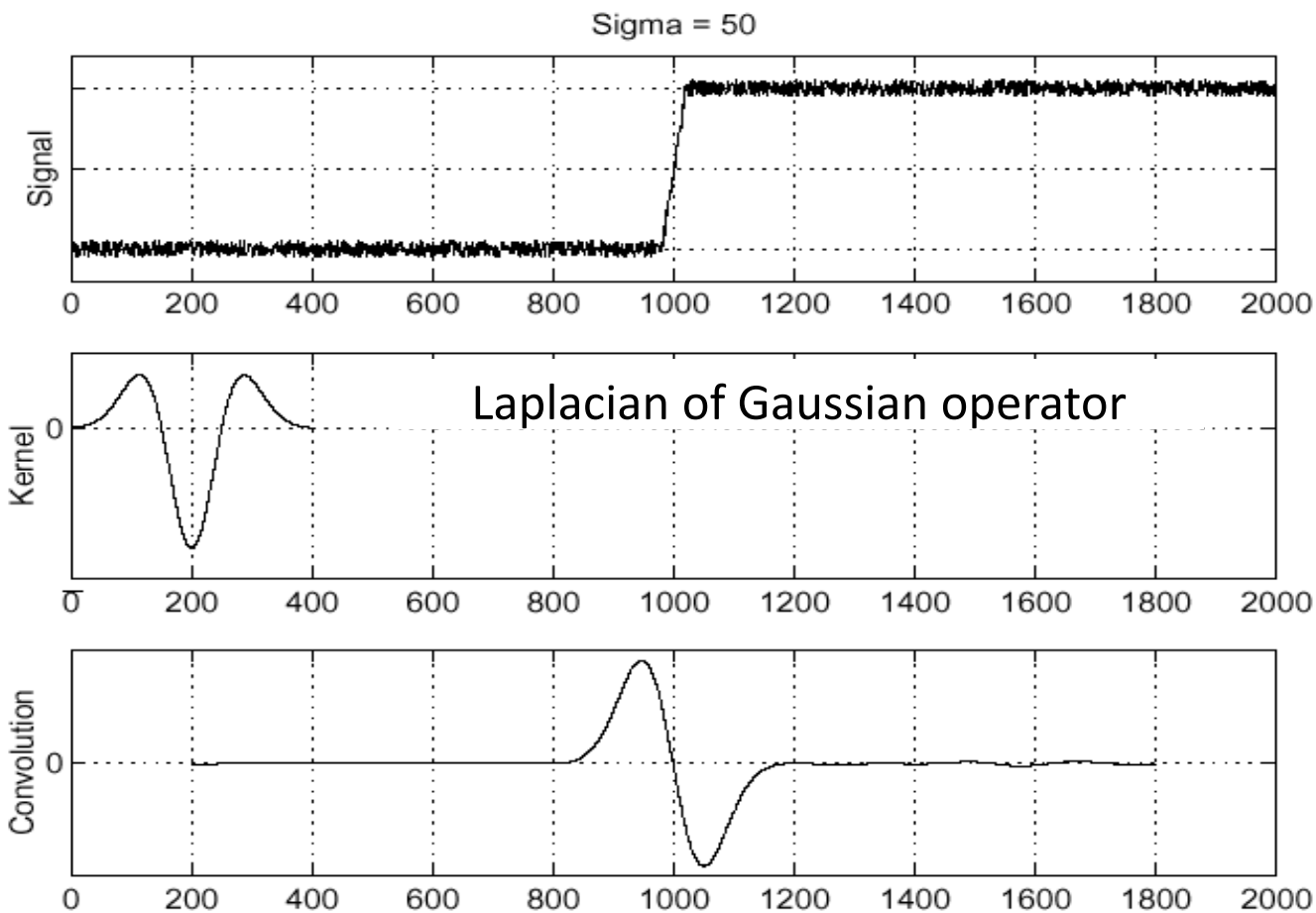
$$\frac{\partial^2}{\partial x^2}(h * f) = \boxed{\left(\frac{\partial^2}{\partial x^2} h\right)} * f$$

Laplacian of Gaussian

f

$\frac{\partial^2}{\partial x^2} h$

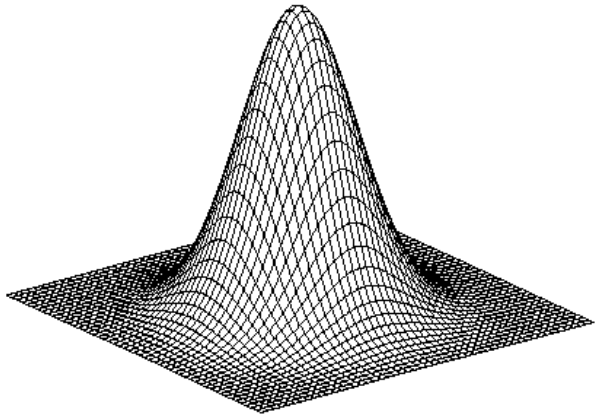
$(\frac{\partial^2}{\partial x^2} h) \star f$



Where is the edge?

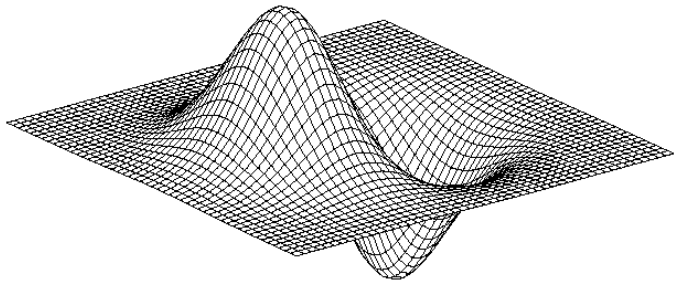
Zero-crossings of bottom graph !

2D Gaussian Edge Operators



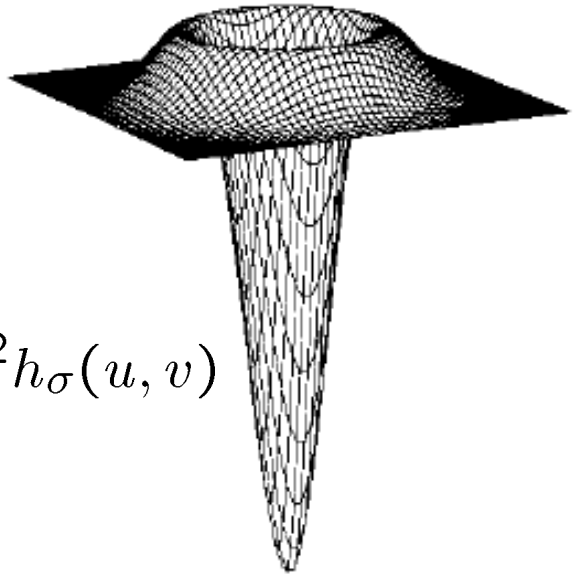
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$

Gaussian



$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

Derivative of Gaussian (DoG)



$$\nabla^2 h_\sigma(u,v)$$

Laplacian of Gaussian

Mexican Hat (Sombrero)

- ∇^2 is the **Laplacian** operator:

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