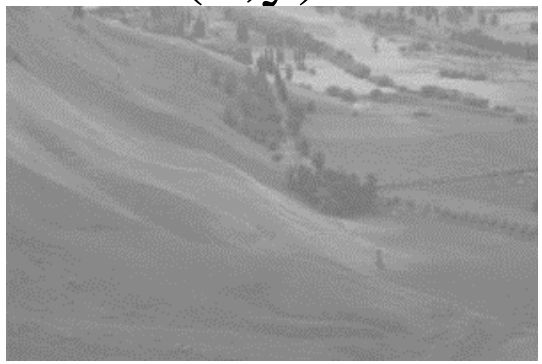
Image Filtering

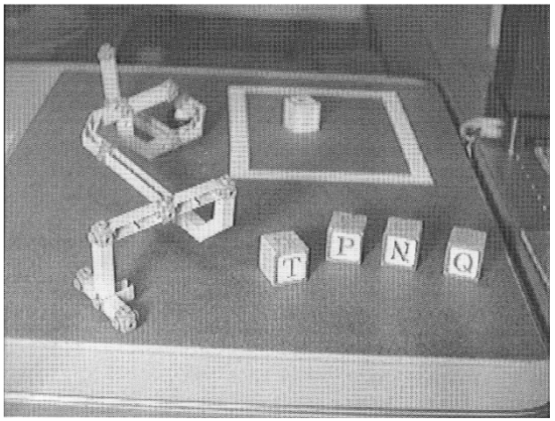
What is image filtering?

f(x,y)

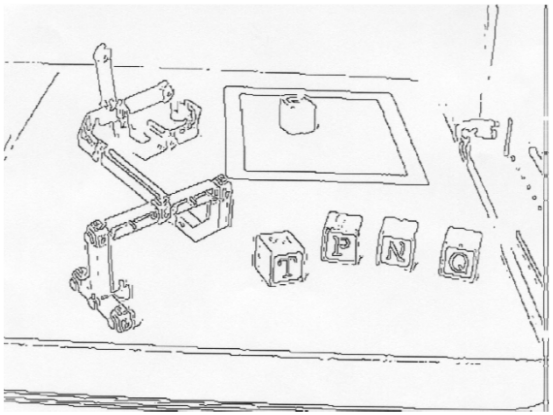


filtering

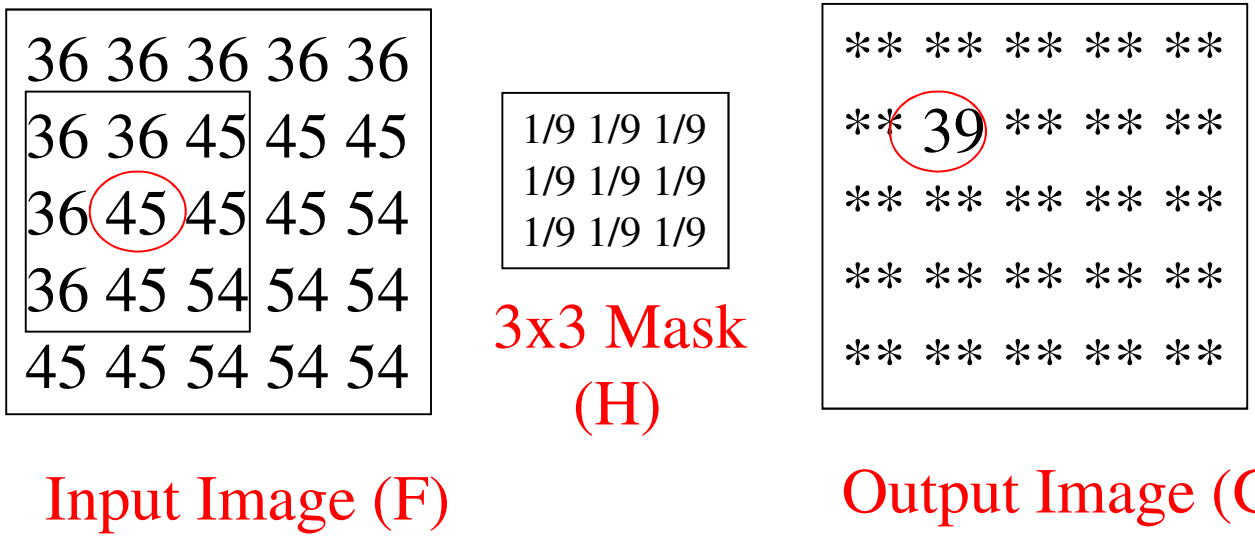
g(x,y)



filtering



Linear filter



- Cross correlation

$$G(i,j) = \sum_u \sum_v H(u,v)F(i+u, j+v)$$

- Convolution

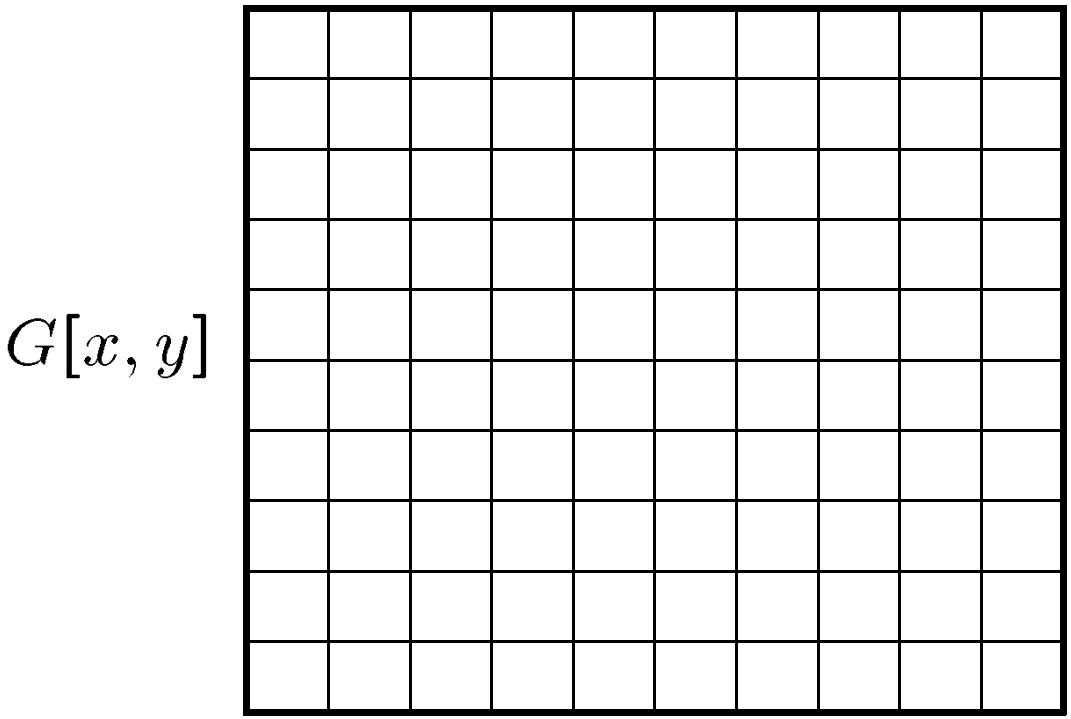
$$G(i,j) = \sum_u \sum_v H(u,v)F(i-u, j-v)$$

Example

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	100	130	110	120	110	0	0
0	0	0	110	90	100	90	100	0	0
0	0	0	130	100	90	130	110	0	0
0	0	0	120	100	130	110	120	0	0
0	0	0	90	110	80	120	100	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

*

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9


$G[x, y]$

Blurring Image

- Mean Filter
- Median Filter
- Gaussian

Mean Filter

- Sum the values in a kxk nbd. Divide by k^2. Replace center.

$F[x, y]$

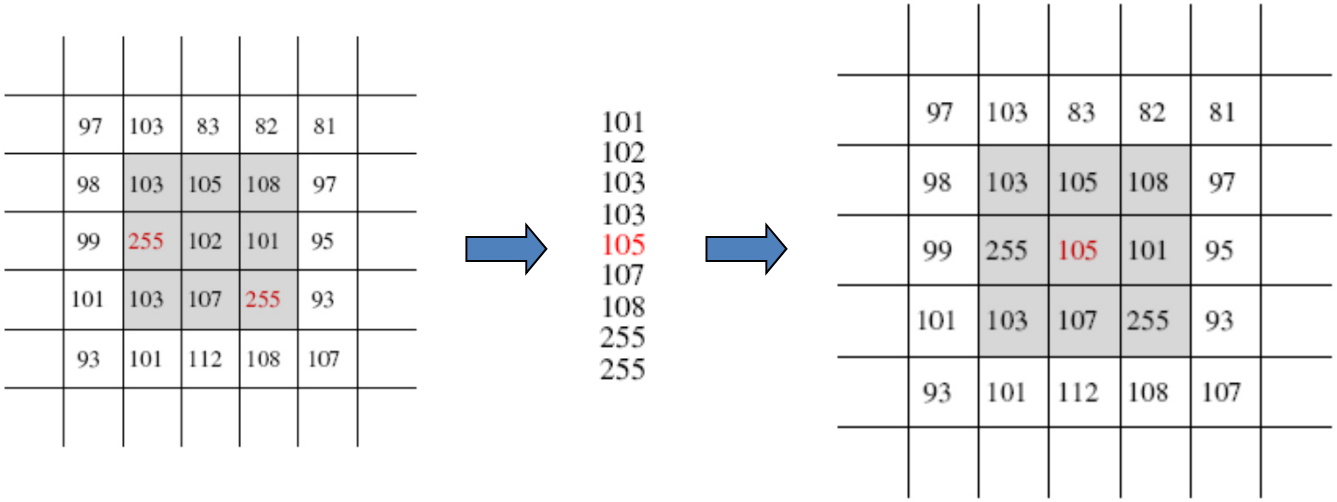
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

$G[x, y]$

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Median Filter

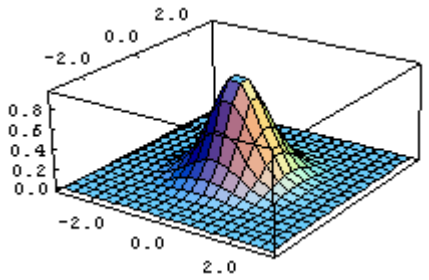
- Use the median of values in the mask to replace center.



Gaussian Blur

- A Gaussian kernel gives less weight to pixels further from the center of the window

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



Gaussian Blur

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

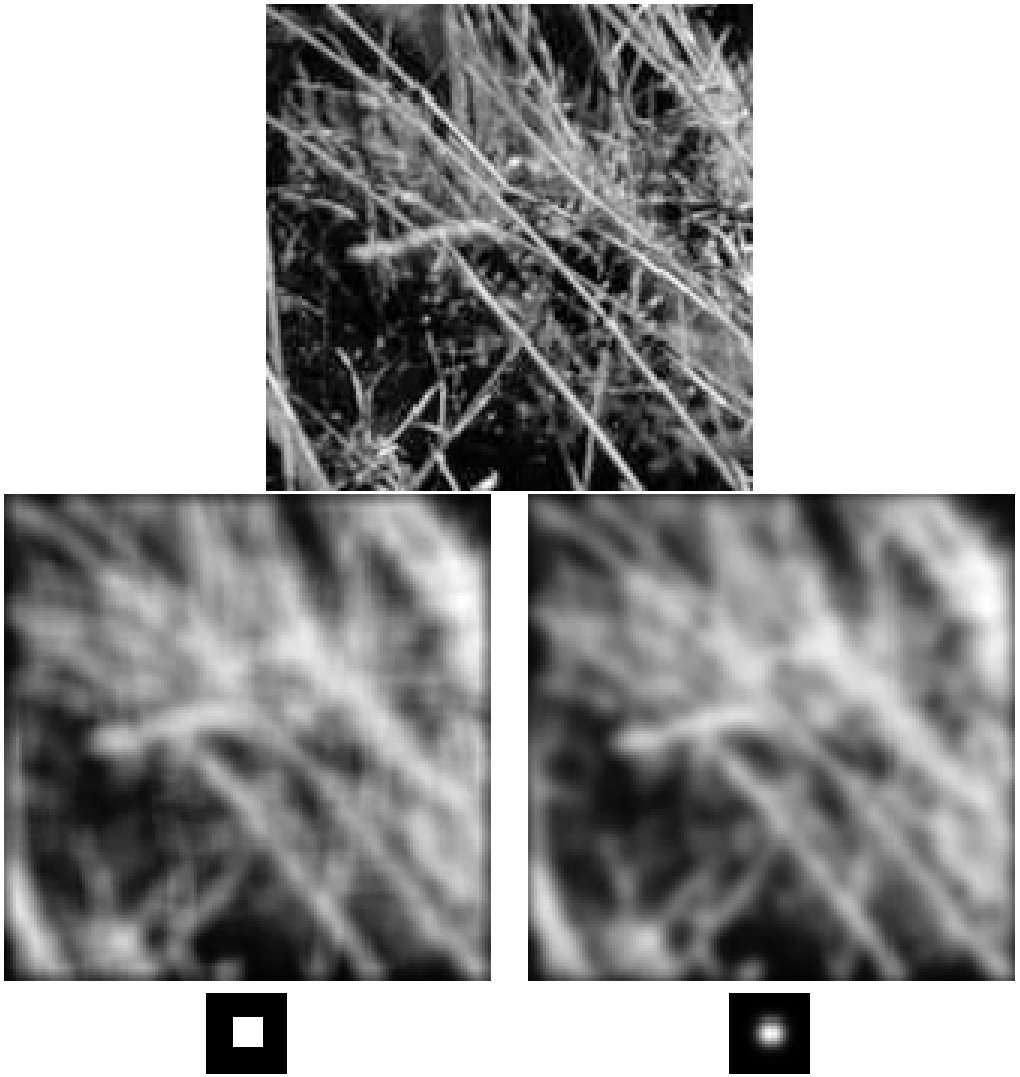
$F[x, y]$

$$\frac{1}{16}$$

1	2	1
2	4	2
1	2	1

$H[u, v]$

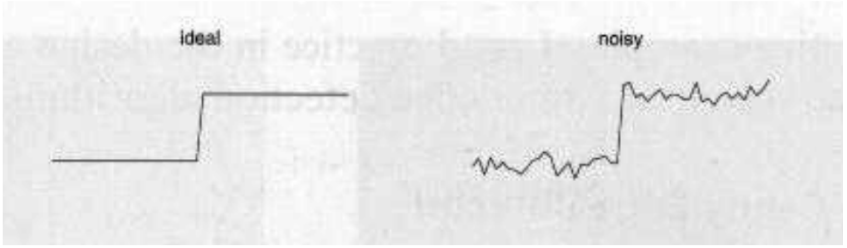
Mean vs. Gaussian filtering

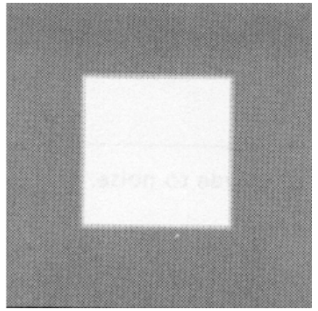


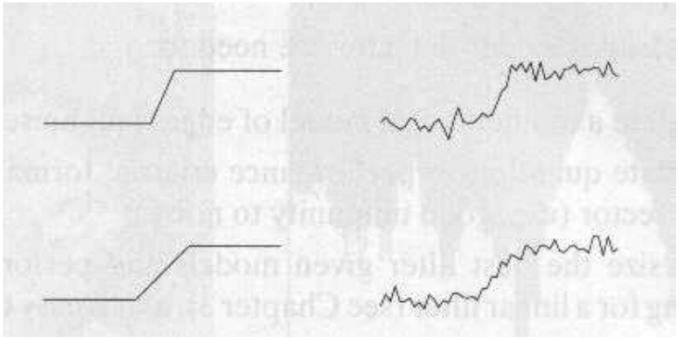
Edge Detection

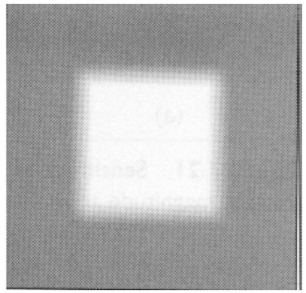
- Vertical, Horizontal edge detection
- Prewitt
- Sobel

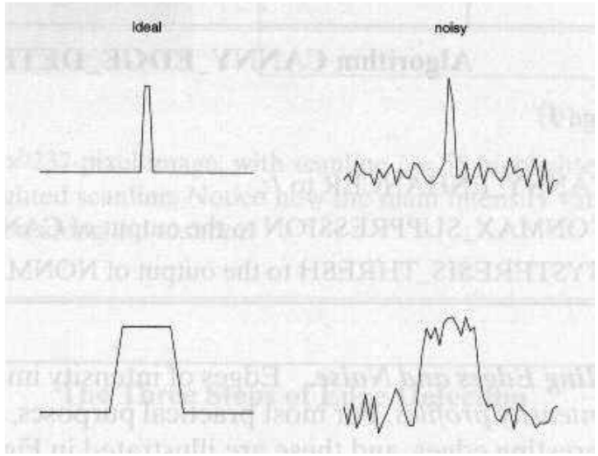
Edge Detection











Edge Detection

- Edge Detection Using First Derivative

$$f'(x) = \lim_{h->0} \frac{f(x+h) - f(x)}{h} \approx f(x+1) - f(x) \ \ (h{=}1)$$ ⇒ mask: $[-1 \quad 1]$

Edge Detection

Edge Detection Using second Derivative

$$f''(x) = \lim_{h->0} \frac{f'(x+h) - f'(x)}{h} \approx f'(x+1) - f'(x) =$$

(centered at x+1)

$$f(x+2) - 2f(x+1) + f(x) \quad (h=1)$$

Replace x+1 with x (i.e., centered at x):

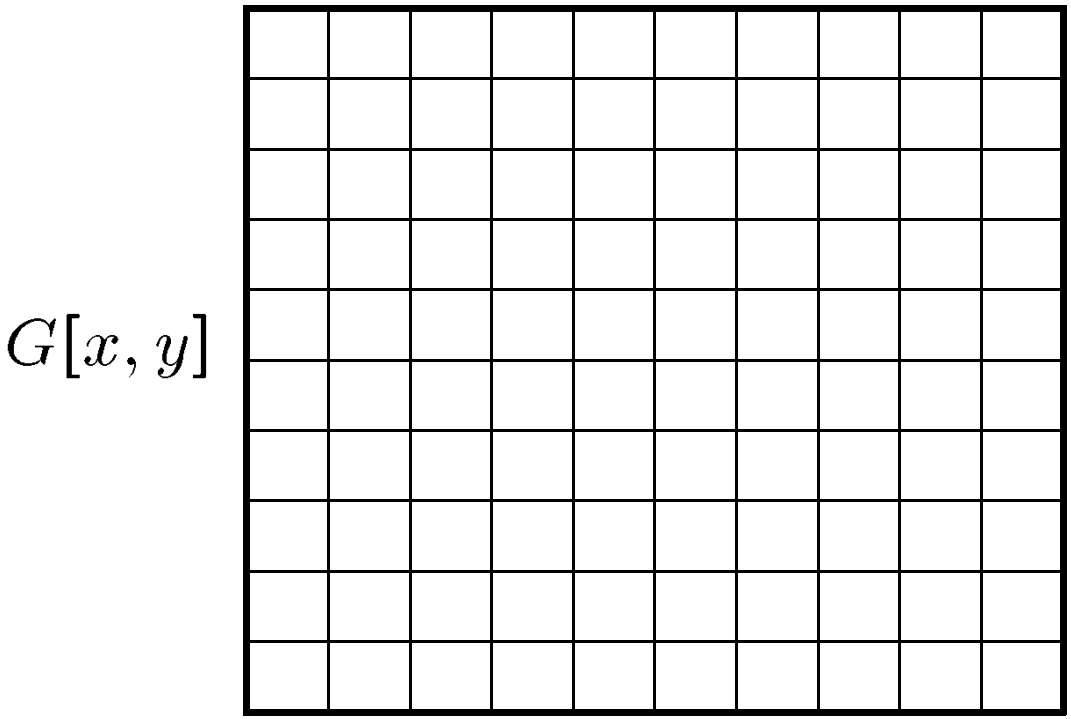
$$f''(x) \approx f(x+1) - 2f(x) + f(x-1)$$

mask: [1 −2 1]

Vertical, Horizontal edge detection

10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0
10	10	10	10	10	0	0	0	0	0

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


$G[x, y]$

10	10	10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10	10	10
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

What should be a 3x3 mask for detecting horizontal edge?

Prewitt edge detection

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{G_x} = \begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G_y} = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * \mathbf{A}$$

38	66	65
14	35	64
12	15	42

Gx=

Gy= 38+66+65-12-15-42

G=

Sobel Edge detection

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$