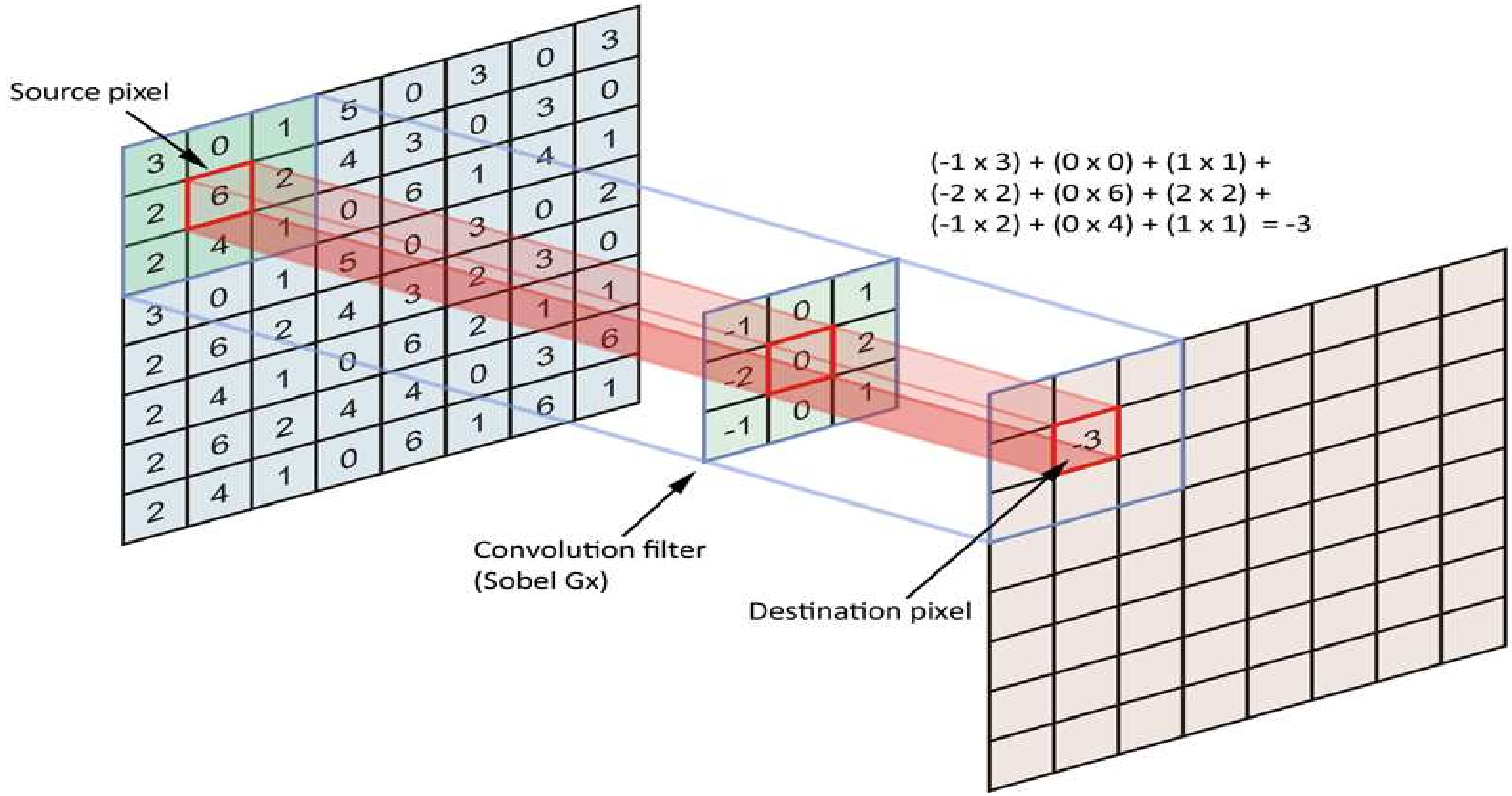
Neighborhood Processing

Yih-Lon Lin (林義隆)
Associate Professor,
Department of Computer Science and Information Engineering,
National Yunlin University of Science and Technology

Mapping



https://datascience.stackexchange.com/questions/23183/why-convolutions-always-use-odd-numbers-as-filter-size

Convolution

Image Matrix

0	0	0	0	0	0
0	105	102	100	97	96
0	103	99	103	101	102
0	101	98	104	102	100
0	99	101	106	104	99
0	104	104	104	100	98

Kernel Matrix

0	-1	0
-1	5	-1
0	-1	0

Output Matrix

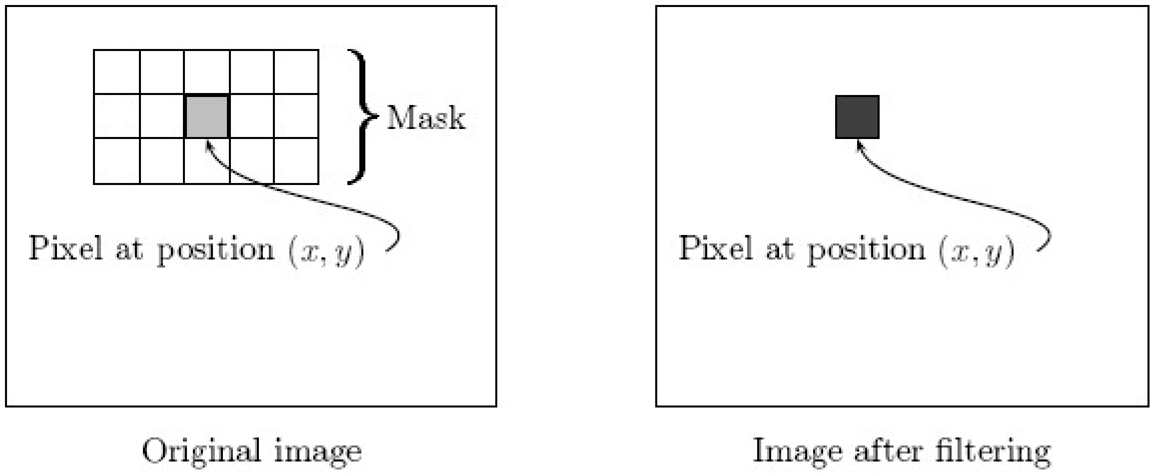
320				

$$
\begin{aligned}
&0*0+0*-1+0*0\\
&+0*-1+105*5+102*-1\\
&+0*0+103*-1+99*0 \quad = 320
\end{aligned}
$$

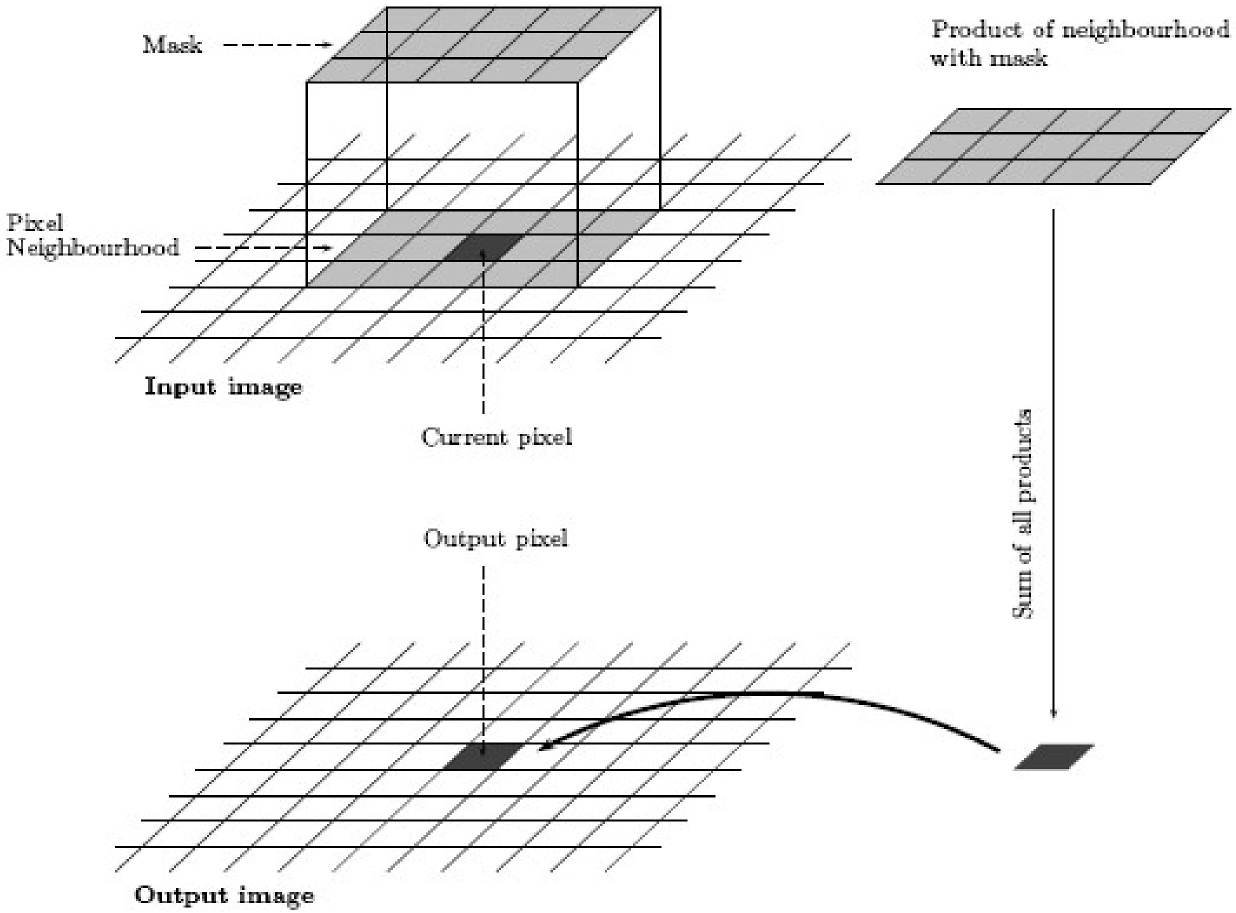
Convolution with horizontal and vertical strides = 1

Introduction

- We create a new image whose pixels have gray values calculated from the grey values under the **mask**, as shown in figure.
- The combination of mask and function is called **filter**.



Performing spatial filtering



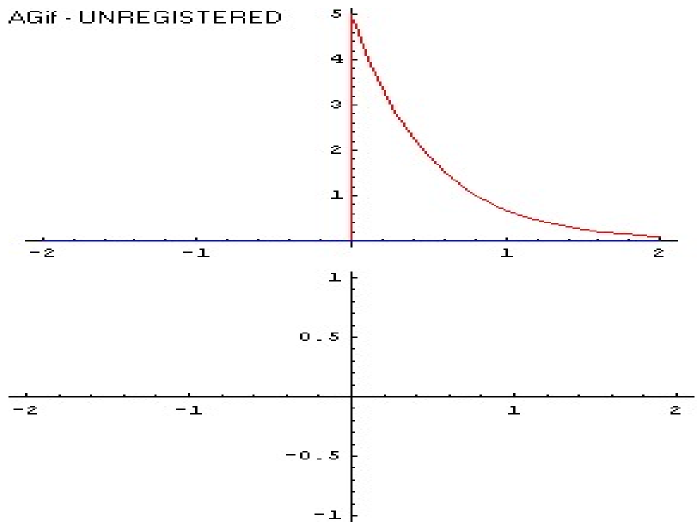
Spatial filtering

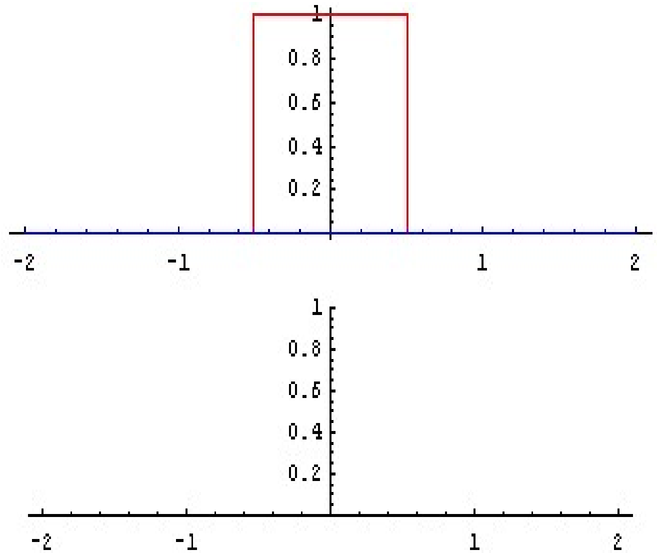
- Spatial filtering thus requires three steps:
  - position the mask over the current pixel,
  - form all products of filter elements with the corresponding elements of the neighborhood,
  - add up all the products.

This must be repeated for every pixel in the image.

Convolution

- Convolution is a mathematical operation on two functions *f* and *g*, producing a third function that is typically viewed as a modified version of one of the original functions.

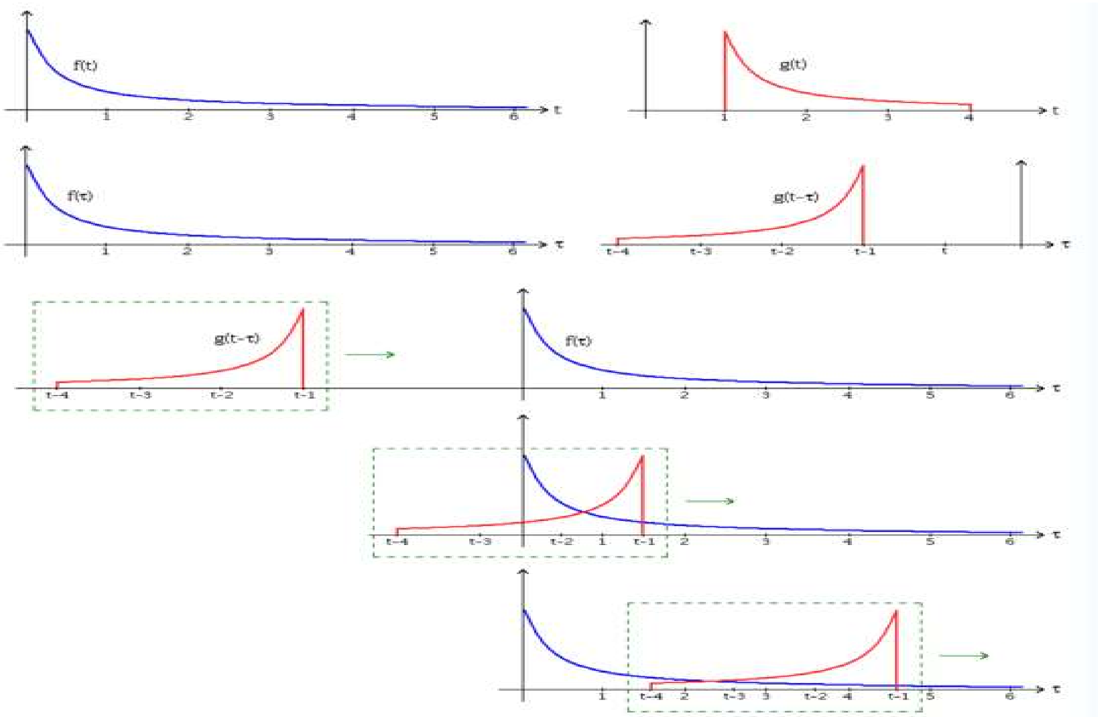




http://www.jhu.edu/~signals/convolve/index.html

Convolution

$$(f * g)(t) \stackrel{\mathrm{def}}{=} \int_{-\infty}^{\infty} f(\tau) \cdot g(t-\tau)\, d\tau$$
$$= \int_{-\infty}^{\infty} f(t-\tau) \cdot g(\tau)\, d\tau.$$



Special convolution

- Allied to spatial filtering is spatial convolution.

$$\sum_{s=-1}^{1} \sum_{t=-2}^{2} m(-s,-t)p(i+s,j+t).$$

$$\sum_{s=-1}^{1} \sum_{t=-2}^{2} m(s,t)p(i-s+j-t).$$

Spatial filtering and spatial convolution will produce the same output.

Linear filter

$m(-1,-2)$	$m(-1,-1)$	$m(-1,0)$	$m(-1,1)$	$m(-1,2)$
$m(0,-2)$	$m(0,-1)$	$m(0,0)$	$m(0,1)$	$m(0,2)$
$m(1,-2)$	$m(1,-1)$	$m(1,0)$	$m(1,1)$	$m(1,2)$

$\oplus$

$p(i-1,j-2)$	$p(i-1,j-1)$	$p(i-1,j)$	$p(i-1,j+1)$	$p(i-1,j+2)$
$p(i,j-2)$	$p(i,j-1)$	$p(i,j)$	$p(i,j+1)$	$p(i,j+2)$
$p(i+1,j-2)$	$p(i+1,j-1)$	$p(i+1,j)$	$p(i+1,j+1)$	$p(i+1,j+2)$

$$\sum_{s=-1}^{1}\sum_{t=-2}^{2} m(s,t)p(i+s,j+t).$$

Averaging Filter

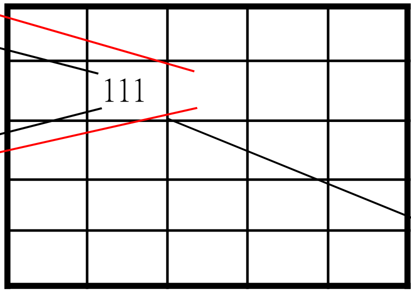
a	b	c
d	e	f
g	h	i

$$\longrightarrow \frac{1}{9}(a+b+c+d+e+f+g+h+i)$$

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

170	240	10	80	150
230	50	70	140	160
40	60	130	200	220
100	120	190	210	30
110	180	250	20	90

170	240	10	80	150
230	50	70	140	160
40	60	130	200	220
100	120	190	210	30
110	180	250	20	90



(1/9)*(170+240+10+230+50+70+40+60+130)=111

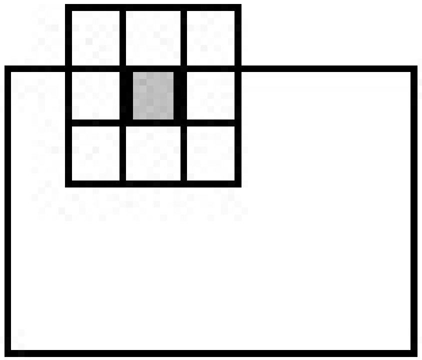
Example

$$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

$$\begin{array}{|c|c|c|} \hline a & b & c \\ \hline d & e & f \\ \hline g & h & i \\ \hline \end{array} \longrightarrow a - 2b + c - 2d + 4e - 2d + g - 2h + i$$

Edges of the image

- Ignore the edges
- Pad with zeros



170	240	10	80	150
230	50	70	140	160
40	60	130	200	220
100	120	190	210	30
110	180	250	20	90

0	0	0	0	0	0	0
0	170	240	10	80	150	0
0	230	50	70	140	160	0
0	40	60	130	200	220	0
0	100	120	190	210	30	0
0	110	180	250	20	90	0
0	0	0	0	0	0	0

0	0	0	0	0	0	0
0	170	240	10	80	150	0
0	230	50	70	140	160	0
0	40	60	130	200	220	0
0	100	120	190	210	30	0
0	110	180	250	20	90	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	170	240	10	80	150	0
0	0	230	50	70	140	160	0
0	0	40	60	130	200	220	0
0	0	100	120	190	210	30	0
0	0	110	180	250	20	90	0
0	0	0	0	0	0	0	0

Filtering

170	240	10	80	150
230	50	70	140	160
40	60	130	200	220
100	120	190	210	30
110	180	250	20	90

76.6667	85.5556	65.5556	67.7778	58.8889
87.7778	111.1111	108.8889	128.8889	105.5556
66.6667	110.0000	130.0000	150.0000	106.6667
67.7778	131.1111	151.1111	148.8889	85.5556
56.6667	105.5556	107.7778	87.7778	38.8889

170	240	10	80	150
230	50	70	140	160
40	60	130	200	220
100	120	190	210	30
110	180	250	20	90

111.1111	108.8889	128.8889
110.0000	130.0000	150.0000
131.1111	151.1111	148.8889

0	0	0	0	0	0	0
0	170	240	10	80	150	0
0	230	50	70	140	160	0
0	40	60	130	200	220	0
0	100	120	190	210	30	0
0	110	180	250	20	90	0
0	0	0	0	0	0	0

18.8889	45.5556	46.6667	36.6667	26.6667	25.5556	16.6667
44.4444	76.6667	85.5556	65.5556	67.7778	58.8889	34.4444
48.8889	87.7778	111.1111	108.8889	128.8889	105.5556	58.8889
41.1111	66.6667	110.0000	130.0000	150.0000	106.6667	45.5556
27.7778	67.7778	131.1111	151.1111	148.8889	85.5556	37.7778
23.3333	56.6667	105.5556	107.7778	87.7778	38.8889	13.3333
12.2222	32.2222	60.0000	50.0000	40.0000	12.2222	10.0000

Filtering

(a) Original image

(b) Average filtering

(c) Using a 9×9 filter

(d) Using a 25×25 filter

Separable Filters

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \frac{1}{3}\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}.$$

Frequencies

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- **Low-pass filters**
  - Low frequency components are parts of the image characterized by little change in the gray values (pass over the low-frequency components and reduces or eliminates high-frequency components)
- **High-pass filters**
  - High frequency components are characterized by large changes in gray values over small distances (pass over the high-frequency components and reduces or eliminates low-frequency components)

$$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

150	152	148	149
147	152	151	150
152	148	149	151
151	149	150	148

→

11	6
−13	−5

High-pass filters

(a) Laplacian filter

(b) Laplacian of Gaussian ("log") filtering

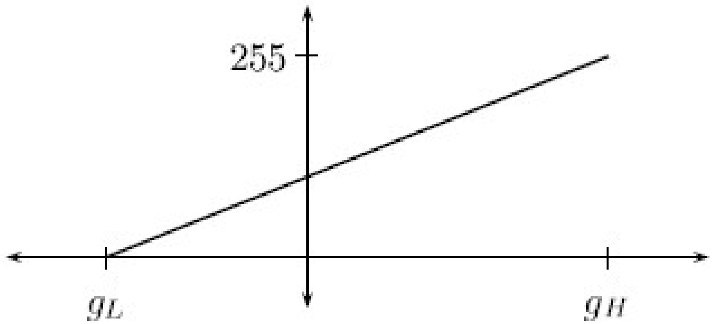
Laplacian of Gaussian

- Laplacian of Gaussian: LoG

Frequencies

- values outside the range 0-255
  - make negative values positive (these values are themselves close to zero)
  - clip values
  - scaling transformation

$$y = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } 0 \leq x \leq 255 \\ 255 & \text{if } x > 255 \end{cases}$$

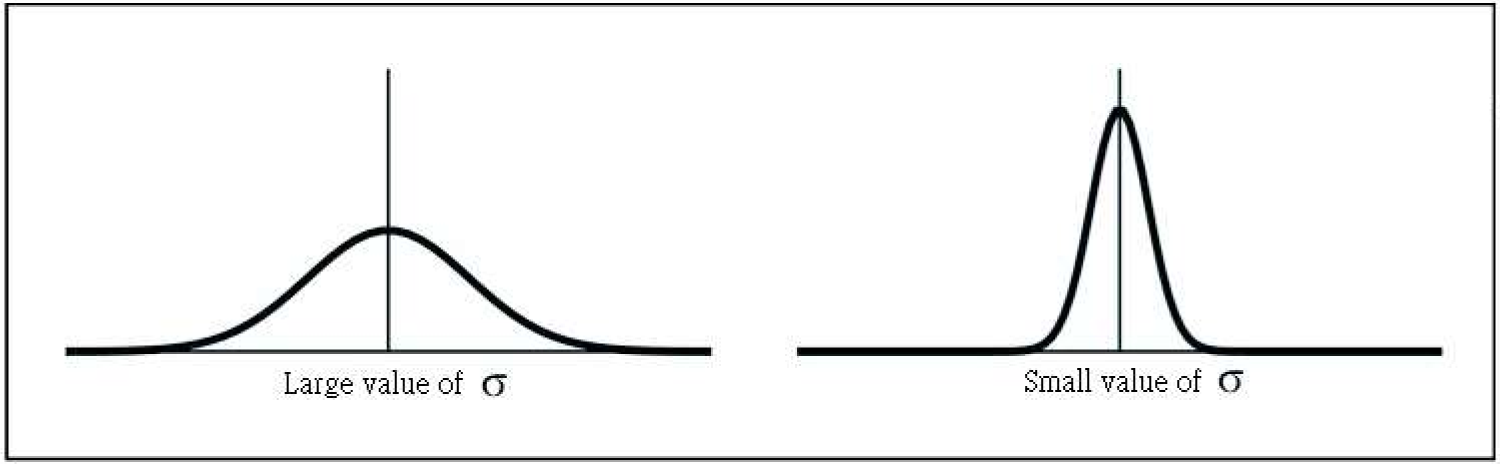


$$y = 255\frac{x - g_L}{g_H - g_L}$$

Gaussian Filters

- A one-dimensional Gaussian probability distribution function

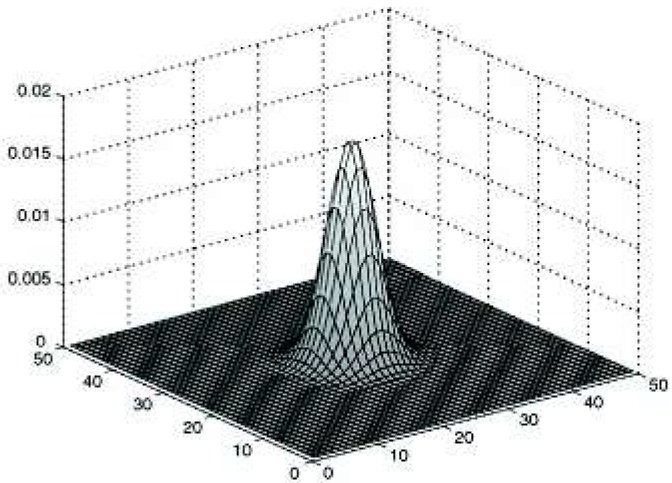
$$f(x) = e^{-\frac{x^2}{2\sigma^2}}$$



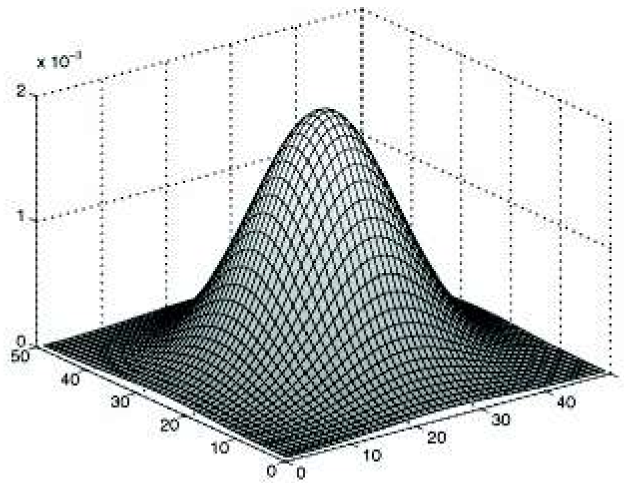
Gaussian Filters

- A two-dimensional Gaussian probability distribution function

$$f(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$



$\sigma = 3$



$\sigma = 9$

Effect of different Gaussian filter

$5 \times 5, \sigma = 0.5$ (a)

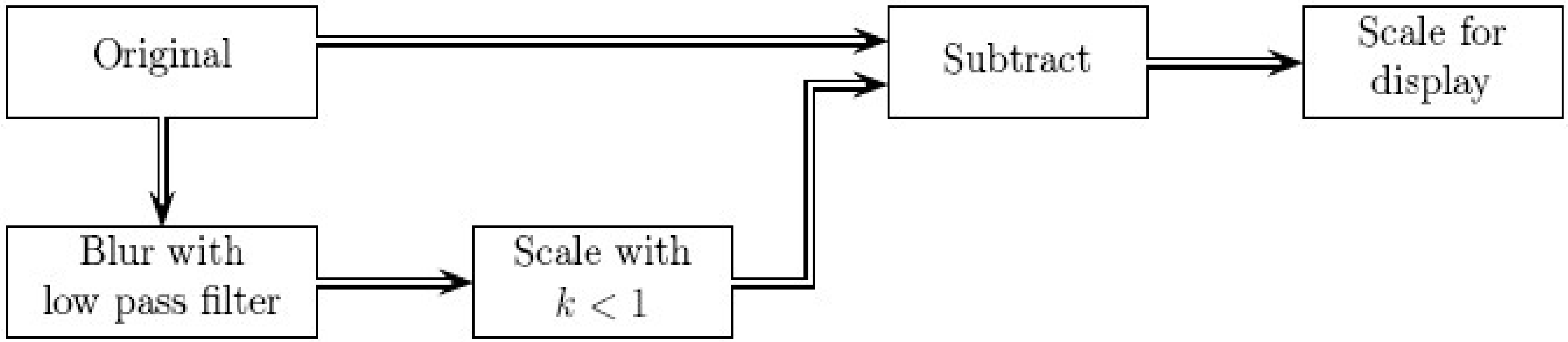
$5 \times 5, \sigma = 2$

$11 \times 11, \sigma = 1$ (c)

$11 \times 11, \sigma = 5$

Edge sharpening

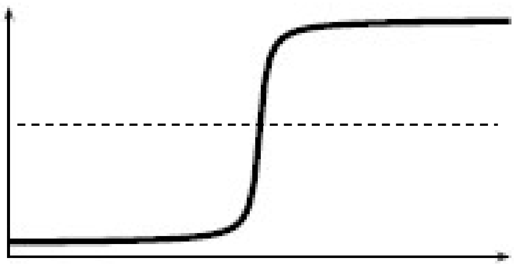
- The operation is variously called edge enhancement, edge crispening, or unsharp masking.



Schema for unsharp masking

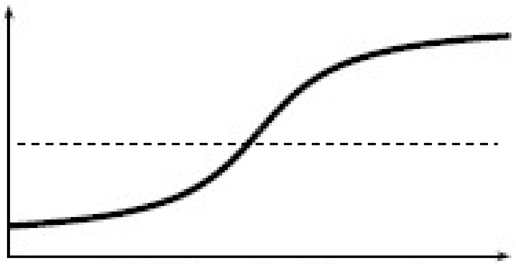
Edge sharpening

(a)

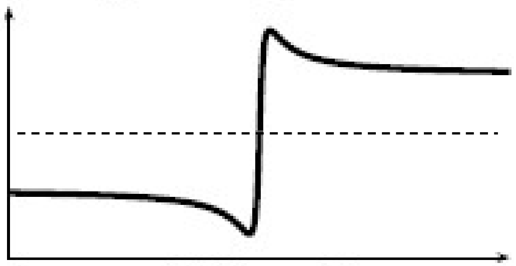


(a) Pixel values over an edge

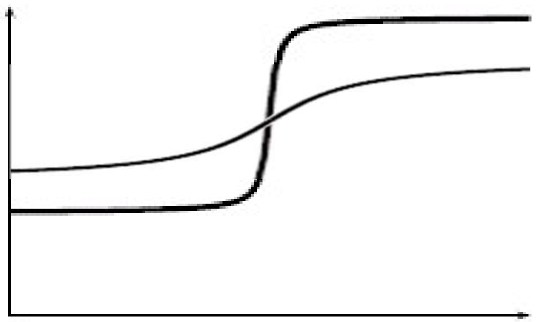
(b)



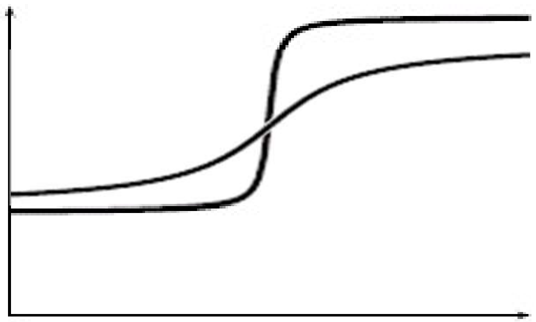
(b) The edge blurred



(c) (a) − k(b)



?



?

Unsharp masking

Edge Sharpening

$$f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{k} \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$f = k \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

$$\frac{1}{\alpha + 1} \begin{bmatrix} -\alpha & \alpha - 1 & -\alpha \\ \alpha - 1 & \alpha + 5 & \alpha - 1 \\ -\alpha & \alpha - 1 & -\alpha \end{bmatrix}$$

$$\frac{1}{3} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 11 & -1 \\ -1 & -1 & -1 \end{bmatrix} = 4 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - 3 \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

1 1 1

High boost filtering

- ordinary high-pass filter

  = original-(low pass)

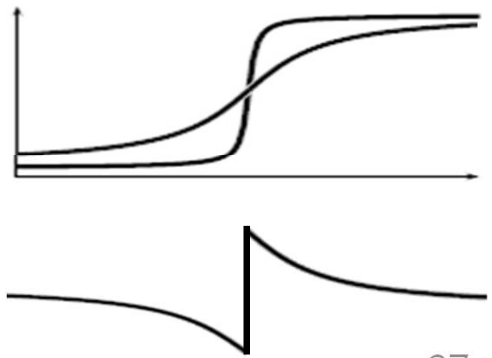
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

- high boost filter

  = A*(original)-(low pass)

  = A*((original)-((original)-(high pass))

  = (A-1)*(original)+(high pass)



High boost filtering

high boost filter

=w*A*(original)-w*(low pass)

Let w*A-w=1 or w=1/(A-1)

=(A/(A-1))*(original)-(1/(A-1))*(low pass)

A=3/5~5/6

$$\underset{1}{\begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline ? & ? & ? \\ \hline ? & ? & ? \\ \hline \end{array}} = A* \underset{1}{\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}} - \underset{1}{\begin{array}{|c|c|c|}\hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline \end{array}}$$

Example

(a) High boost filtering with `hb1`

(b) High boost filtering with `hb2`

Non-linear filters

- **rank-order filter**
  - maximum filter
  - minimum filter
  - median filter
- geometric mean filter

$$\left(\prod_{(i,j)\in M} x(i,j)\right)^{(1/|M|)}$$

- **alpha-trimmed mean filter**

$$x_1 \le x_2 \le x_3 \le \cdots \le x_9$$

$$(x_3 + x_4 + x_5 + x_6 + x_7)/5.$$

Mask

$$(a)\ \begin{matrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{matrix} \qquad (b)\ \begin{matrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{matrix} \qquad (c)\ \begin{matrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{matrix} \qquad (d)\ \begin{matrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{matrix}$$

$$(e)\ \begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix} \qquad (f)\ \begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix} \qquad (g)\ \begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix} \qquad (h)\ \begin{matrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{matrix}$$