Last course: Intensity Level Resolution

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

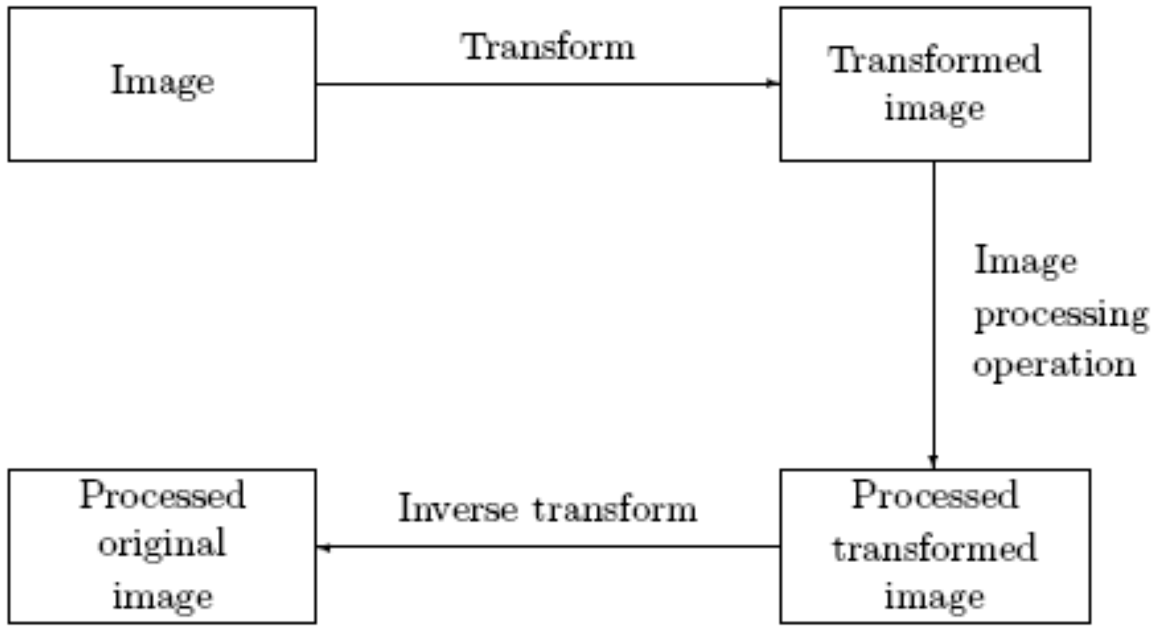
Low Detail

Medium Detail

High Detail

Last course POINT PROCESSING

Transform: represents the pixel values in some other, but equivalent form



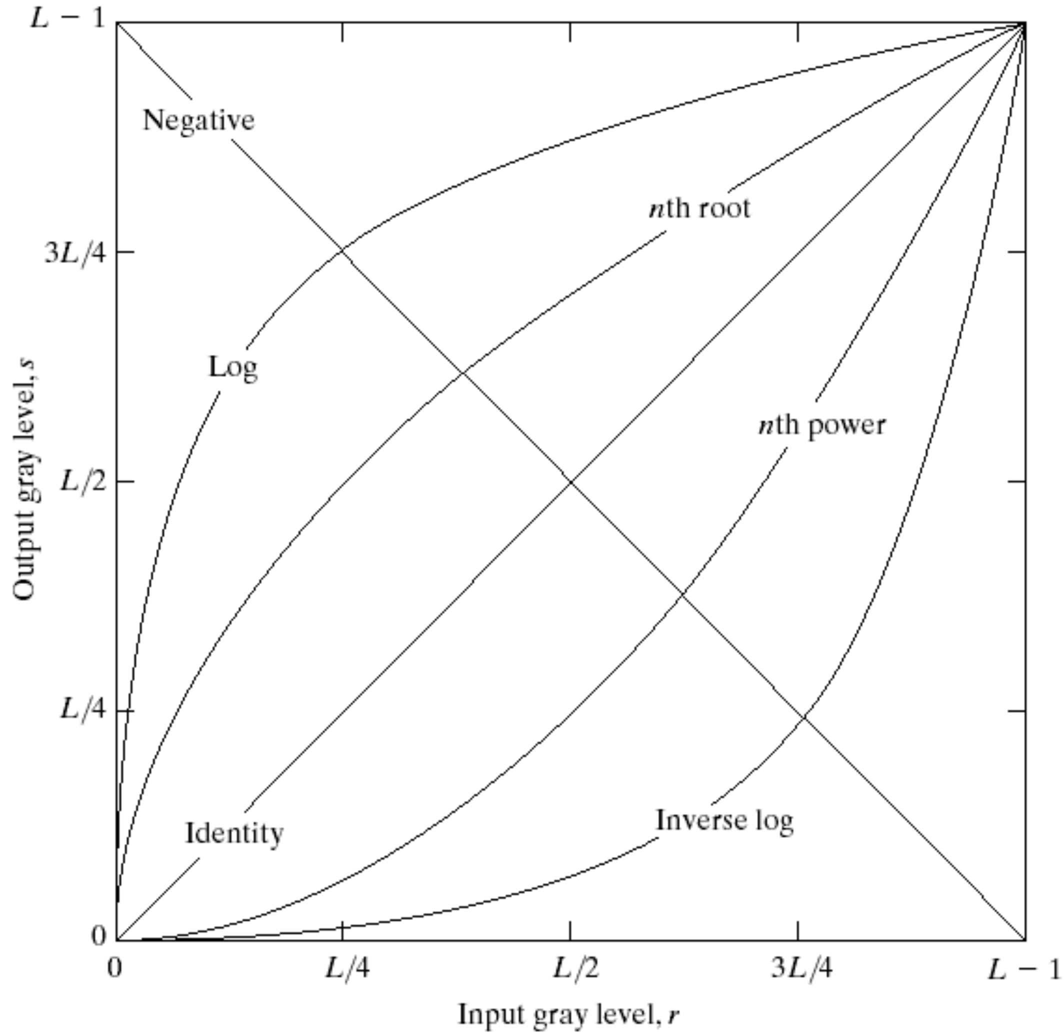
Schema for transform processing

Neighbourhood processing. To change the grey level of a given pixel we need only know the value of the grey levels in a small neighbourhood of pixels around the given pixel.

Point operations. A pixel's grey value is changed without any knowledge of its surrounds.

Last Course Basic Transformations

FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



Negative:

$$s = L - 1 - r$$

Log:

Inverse Log:

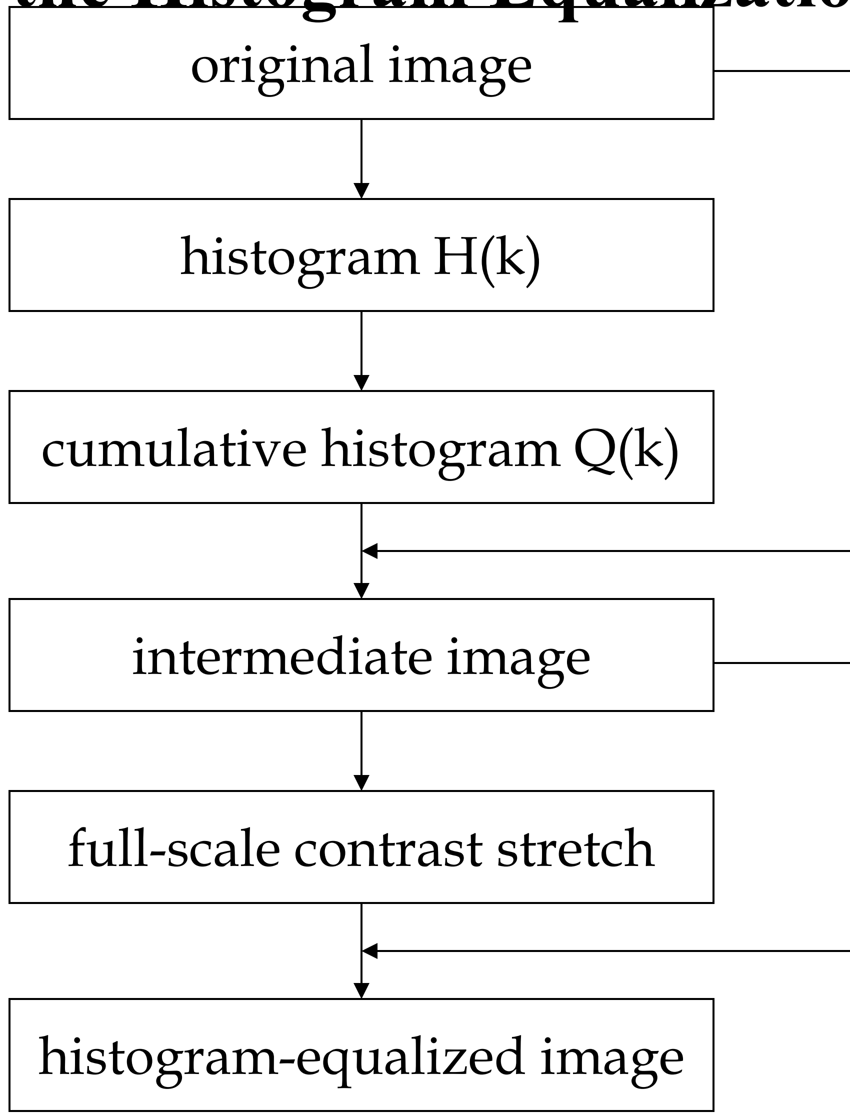
$$s = e^{cr} - 1$$

Power-law:

$$s = cr^{\gamma}$$

……

Last course
Summary of the Histogram Equalization Algorithm

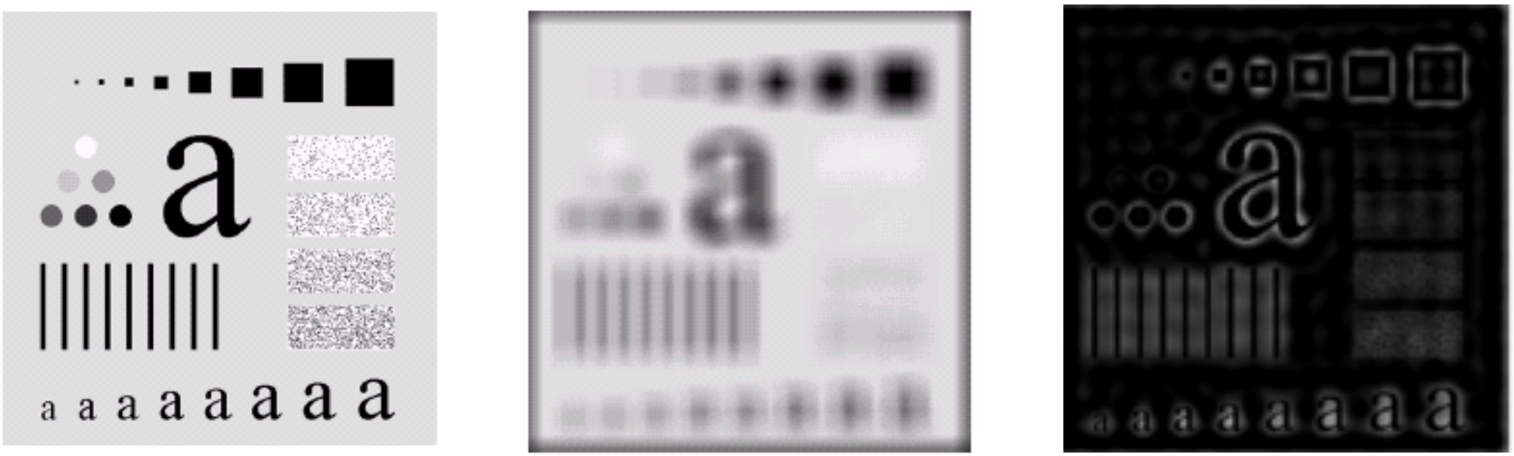


SPATIAL FILTERING

- Understand the mechanics of spatial filtering, and how spatial filters are formed
- Understand the principles of spatial convolution and correlation.
- Be familiar with the principal types of spatial filters, and how they are applied.
- Be aware of the relationships between spatial filters, and the fundamental role of lowpass filters.
- Understand how to use combinations of enhancement methods in cases where a single approach is insufficient.


HIGH PASS
LOW PASS
OUT
APPLY TO
7Q

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$



Linear image processing

- Image processing system $S(.)$ is linear, iff superposition principle holds:

$$S\left(\alpha \cdot f[x,y] + \beta \cdot g[x,y]\right) = \alpha \cdot S\left(f[x,y]\right) + \beta \cdot S\left(g[x,y]\right) \text{ for all } \alpha, \beta \in \mathbb{R}$$

- Any linear image processing system can be written as

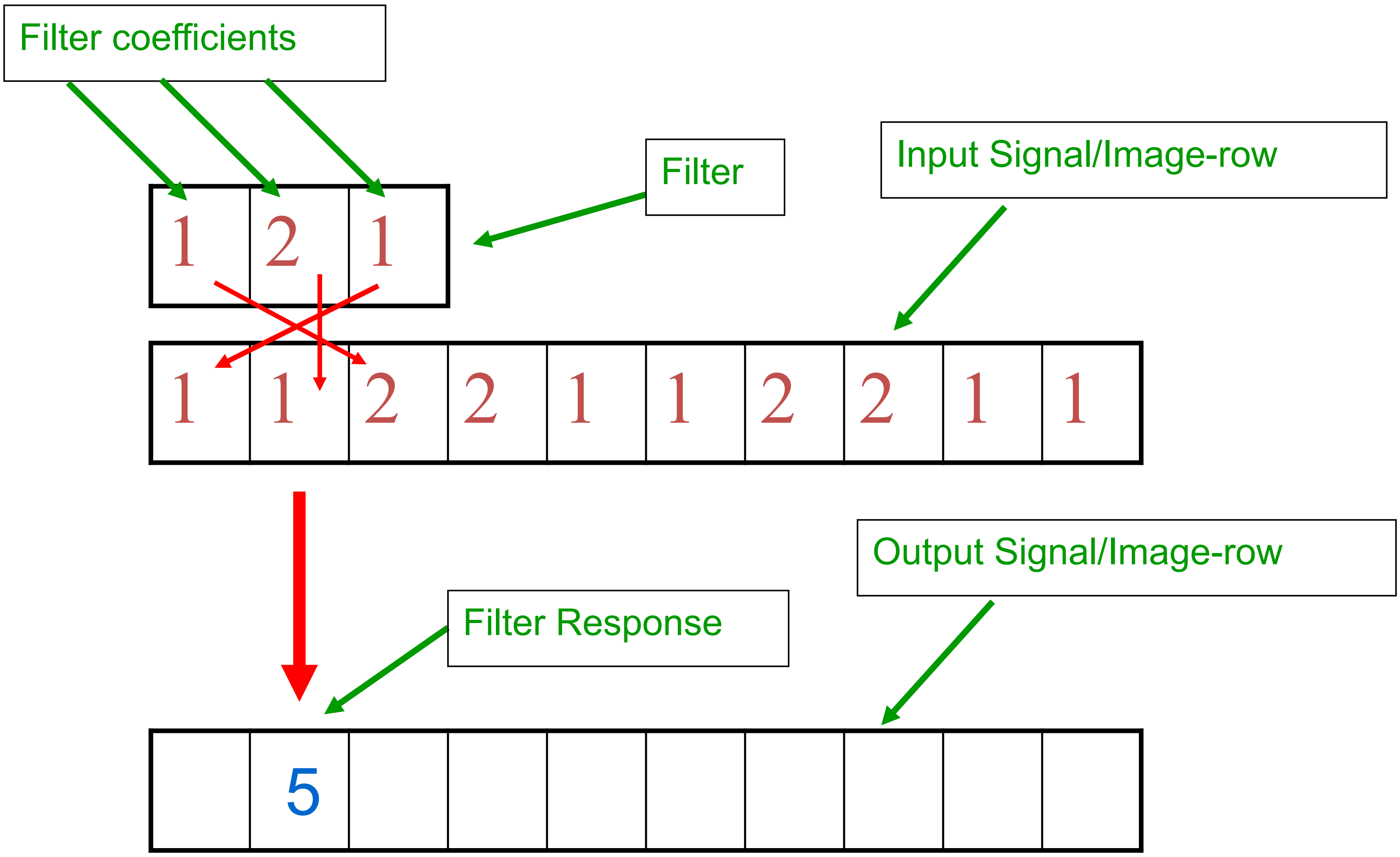
$$\vec{g} = H\vec{f}$$

Note: matrix H need not be square.

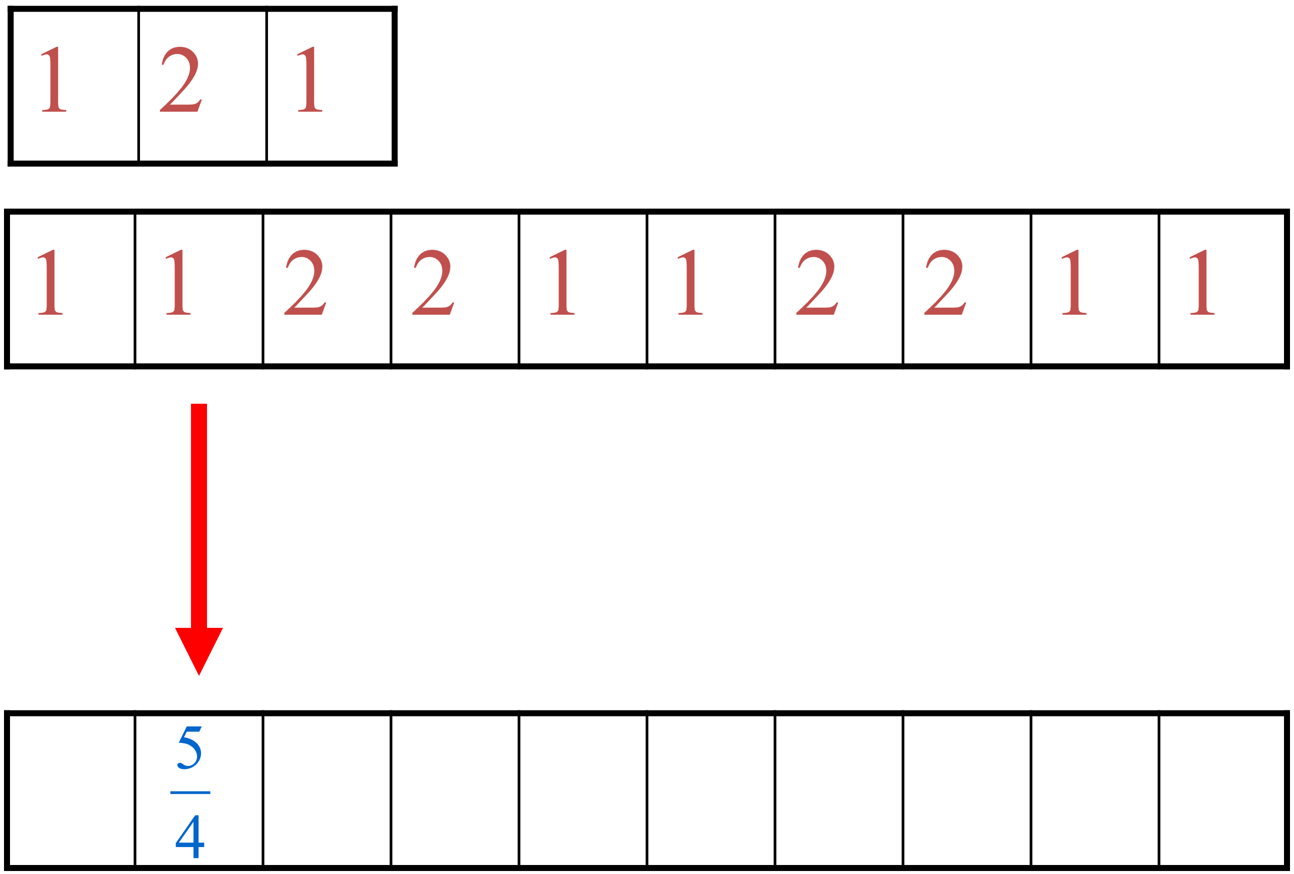
by sorting pixels into a column vector

$$\vec{f} = \begin{pmatrix} f[0,0] & f[1,0] & \cdots & f[N-1,0] & f[0,1] & \cdots & f[N-1,1] & \cdots & \cdots & f[0,L-1] & \cdots & f[N-1,L-1] \end{pmatrix}^T$$

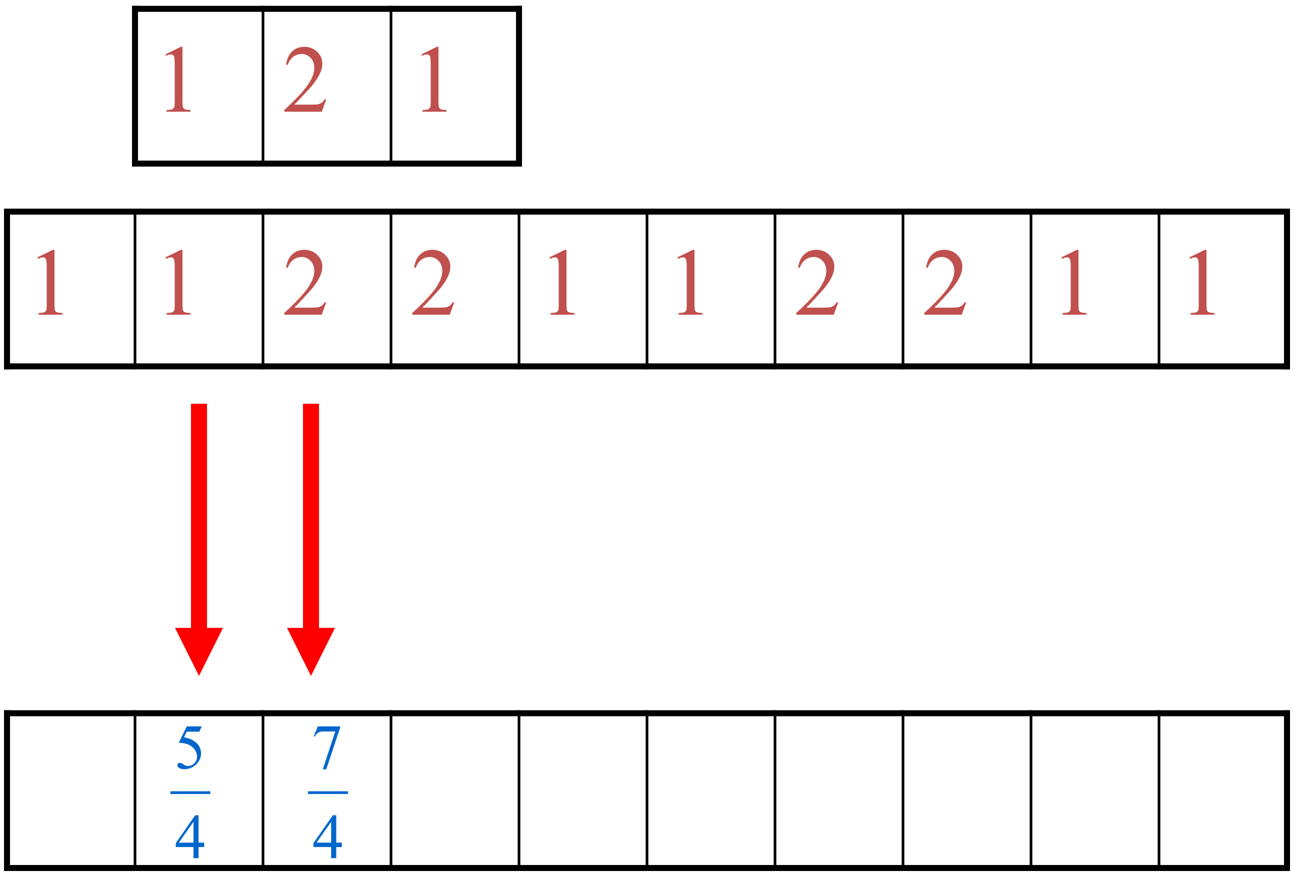
Convolution (1D)



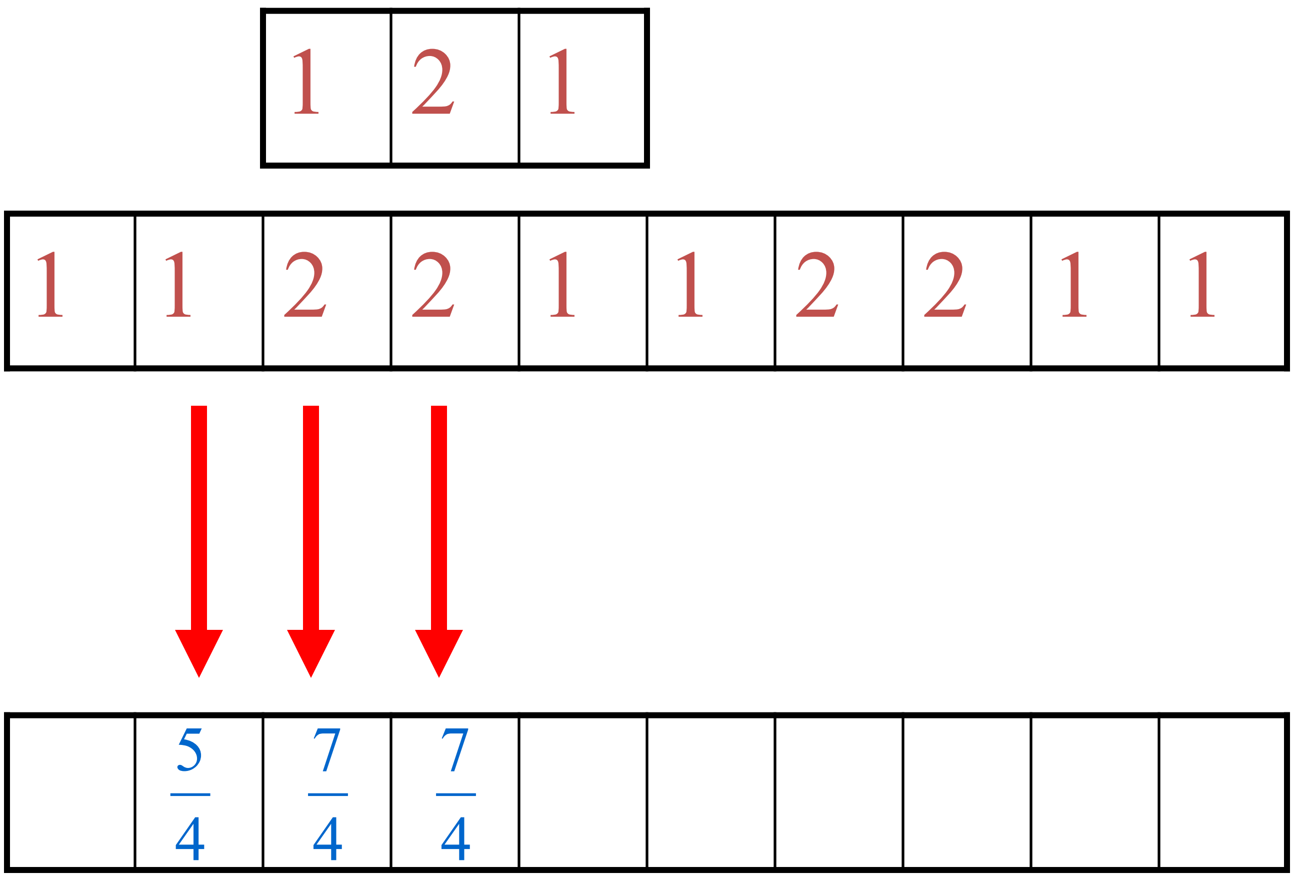
Convolution (1D)



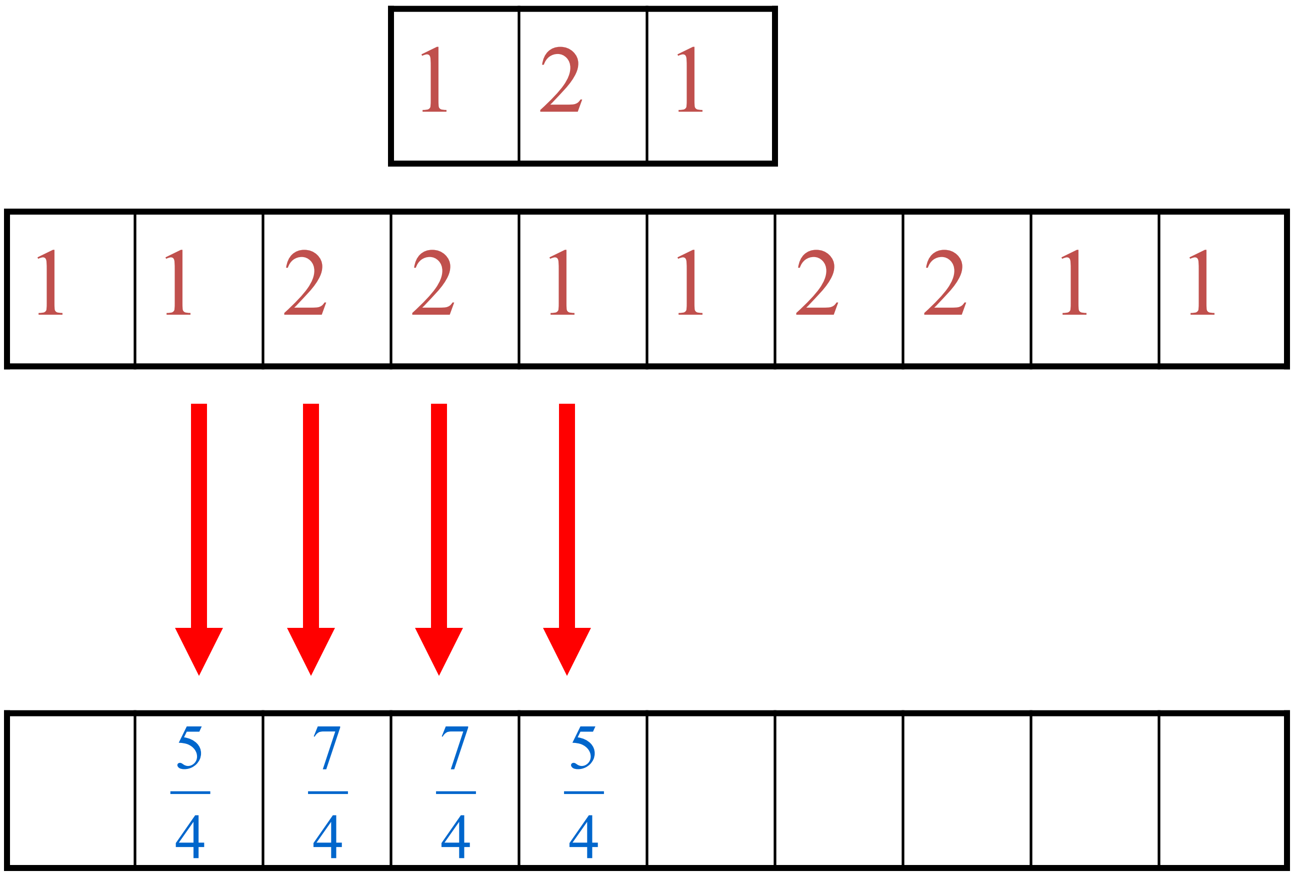
Convolution (1D)



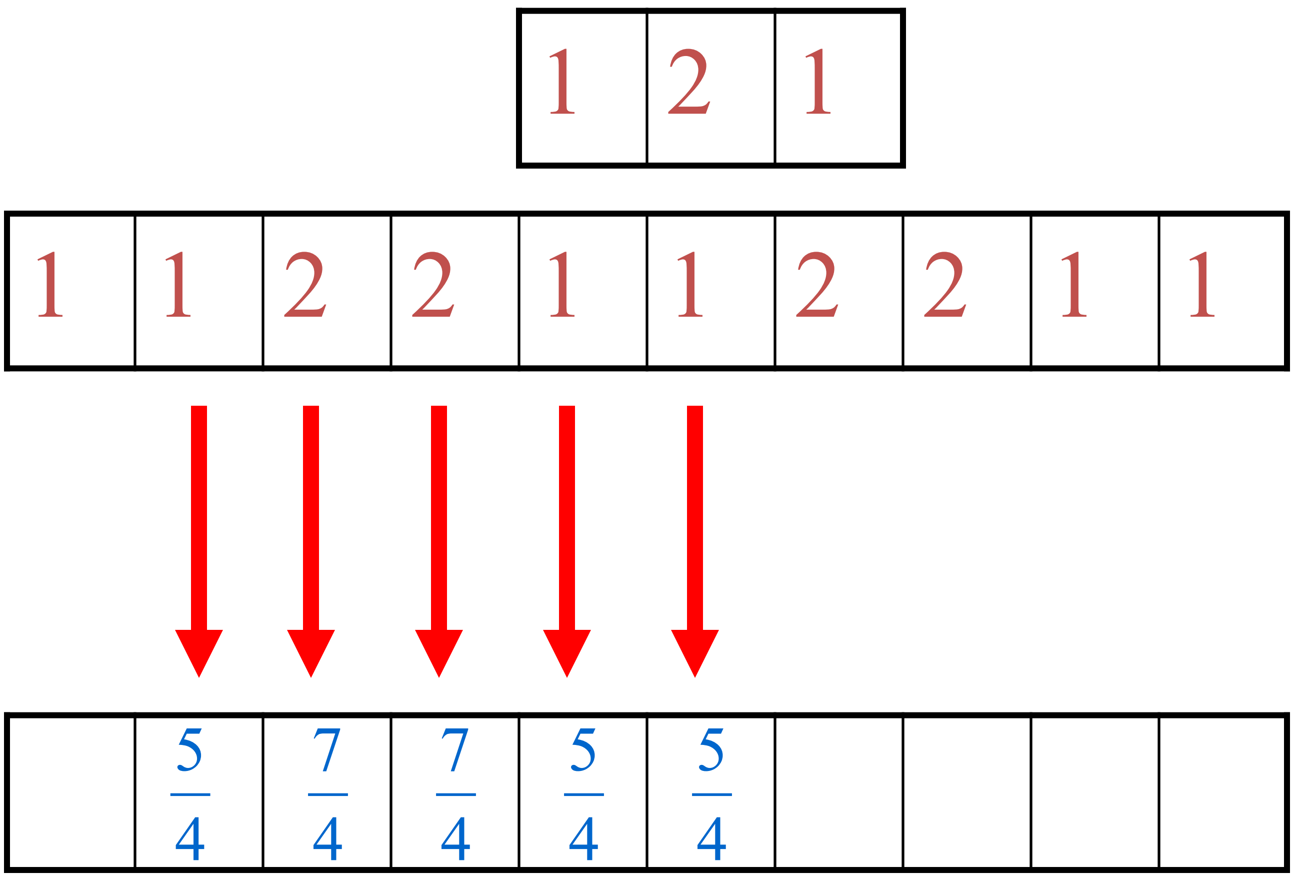
Convolution (1D)



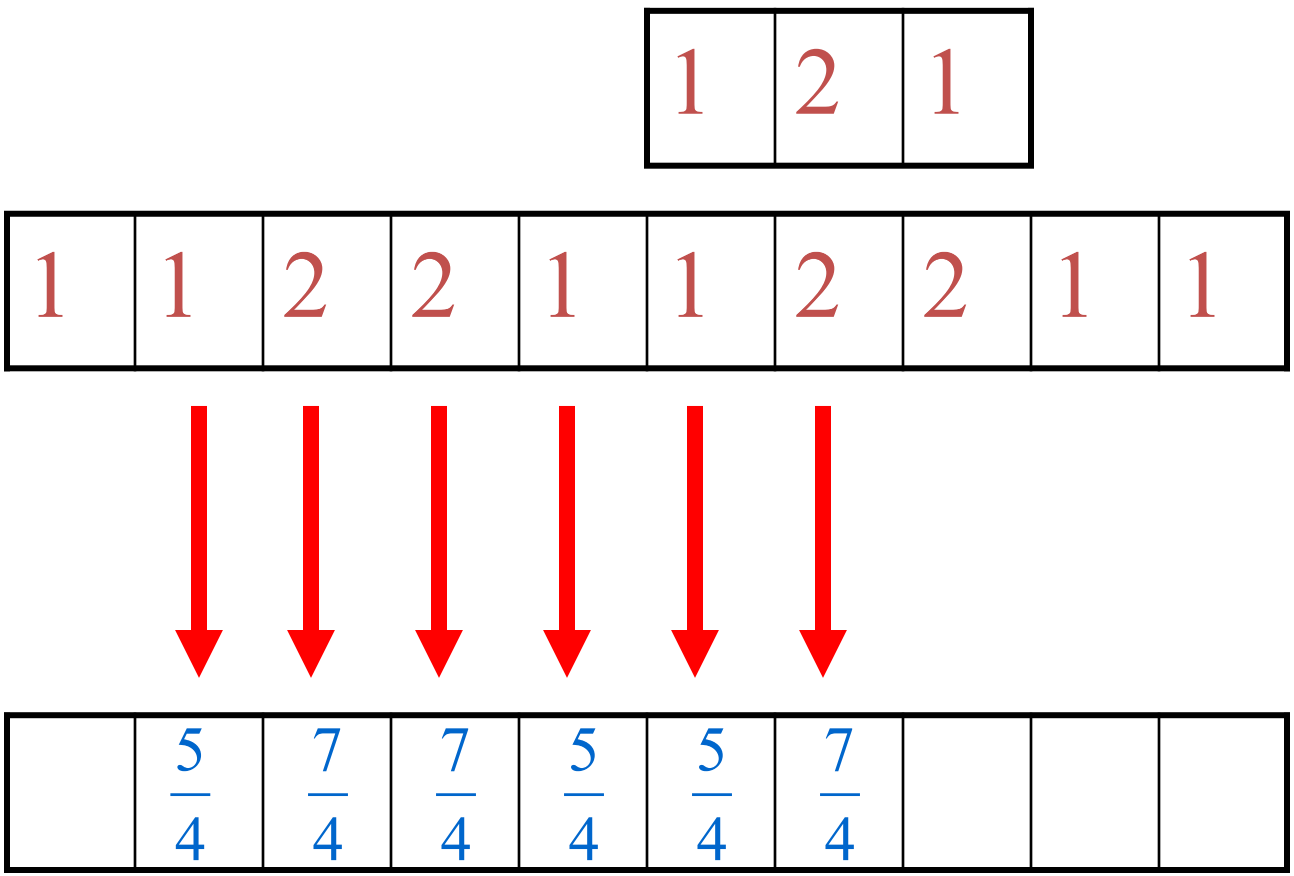
Convolution (1D)



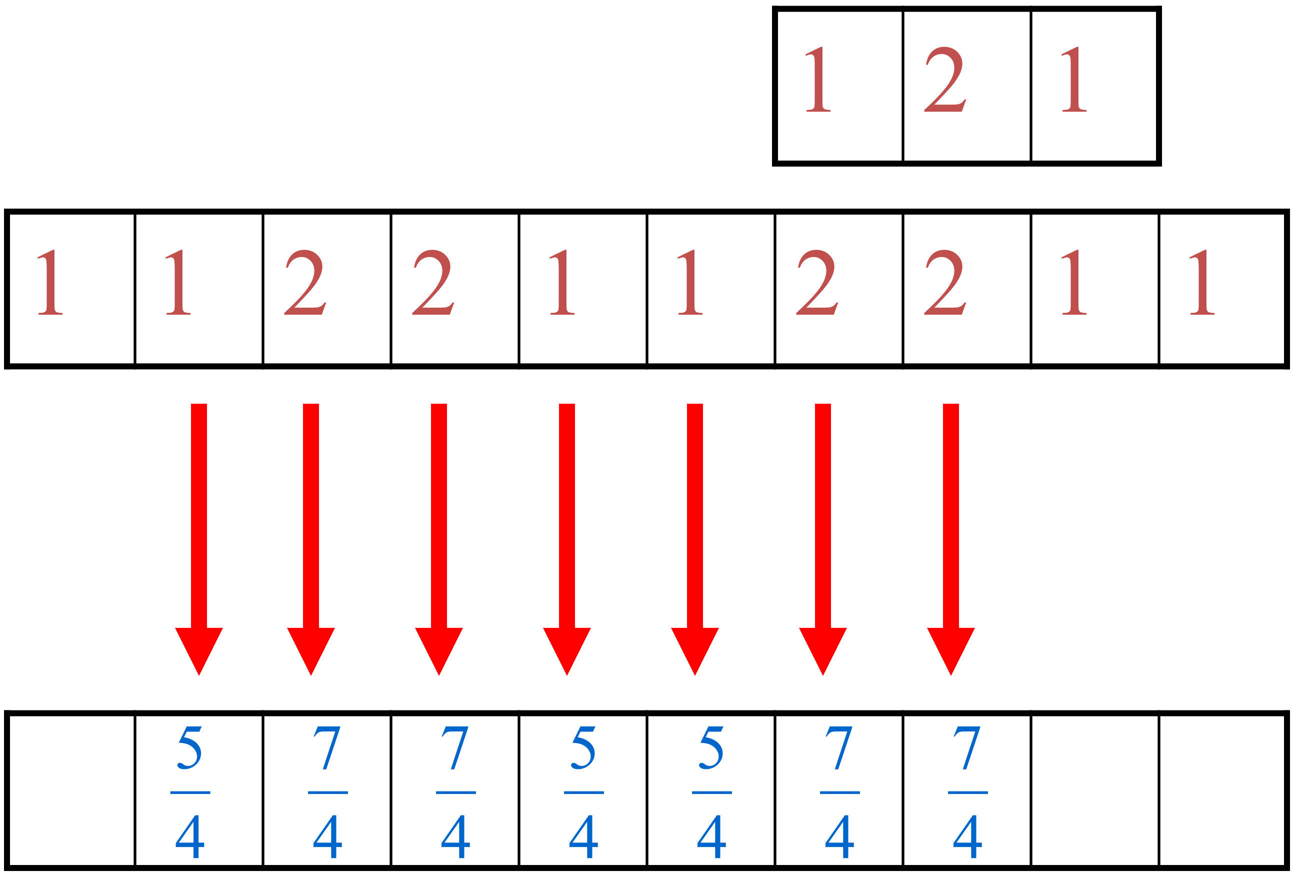
Convolution (1D)



Convolution (1D)



Convolution (1D)



Convolution (1D)

This process is called ***Convolution***!!

1	2	1

1	1	2	2	1	1	2	2	1	1

	$\frac{5}{4}$	$\frac{7}{4}$	$\frac{7}{4}$	$\frac{5}{4}$	$\frac{5}{4}$	$\frac{7}{4}$	$\frac{7}{4}$	$\frac{5}{4}$	

Convolution/correlation on images

- The filter is now 2D
- Kernel (mask), kernel coefficients
- Size: **3x3**, 5x5, 7x7, ….

Normalisation

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

Input

1	2	0	1	3	
2	1	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

	$\frac{12}{9}$				

Convolution/correlation on images

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

Input

1	2	0	1	3	
2	1	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

	$\frac{12}{9}$	$\frac{11}{9}$			

Convolution/correlation on images

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

Input

1	2	0	1	3	
2	1	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		

Convolution/correlation on images

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

Input

1	2	0	1	3	
2	1	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		
	$\frac{13}{9}$	$\frac{11}{9}$	$\frac{13}{9}$		
	$\frac{16}{9}$	$\frac{12}{9}$	$\frac{11}{9}$		

Math. of 2D Convolution/Correlation

Convolution

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

Correlation

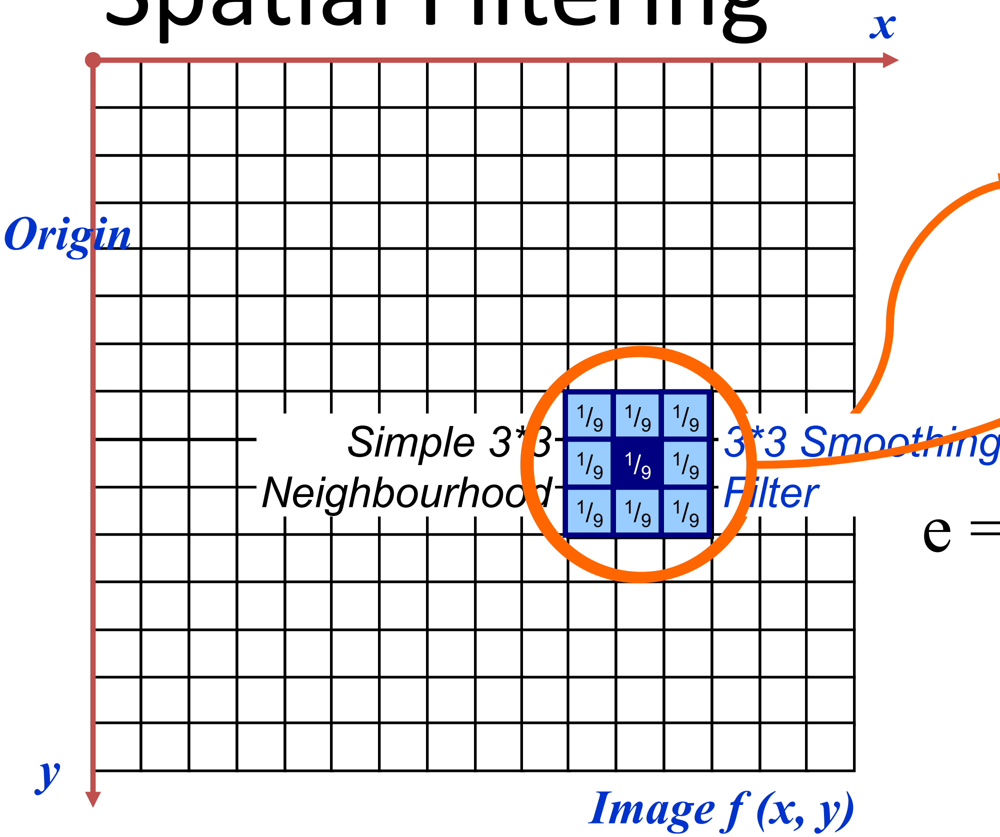
$$g(x,y) = h \circ f(x,y) = \sum_{j=-n}^{n} \sum_{i=-m}^{m} h(i,j) f(x+i, y+j)$$

Note: When the filter is symmetric: correlation = convolution!

1	1	1
1	1	1
1	1	1

2	3	2
-1	0	-1
2	3	2

Spatial Filtering



104	100	108
99	106	98
95	90	85

Original Image Pixels

$*$

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

Filter

$$
\begin{aligned}
e = {} & ^1/_9*106 + \\
& ^1/_9*104 + {}^1/_9*100 + {}^1/_9*108 + \\
& ^1/_9*99 + {}^1/_9*98 + \\
& ^1/_9*95 + {}^1/_9*90 + {}^1/_9*85 \\
= {} & 98.3333
\end{aligned}
$$

The above is repeated for every pixel in the original image.

$$
\begin{aligned}
R &= w_1 z_1 + w_2 z_2 + \ldots + w_{mn} z_{mn} \\
&= \sum_{k=1}^{mn} w_k z_k \\
&= \mathbf{w}^T \mathbf{z}
\end{aligned}
$$

Problems at the borders

- Why is the output image smaller than the input?
  We are lacking information
- The bigger the kernel the bigger the problem
- Does it matter? Yes, if we are going to combine the images afterwards

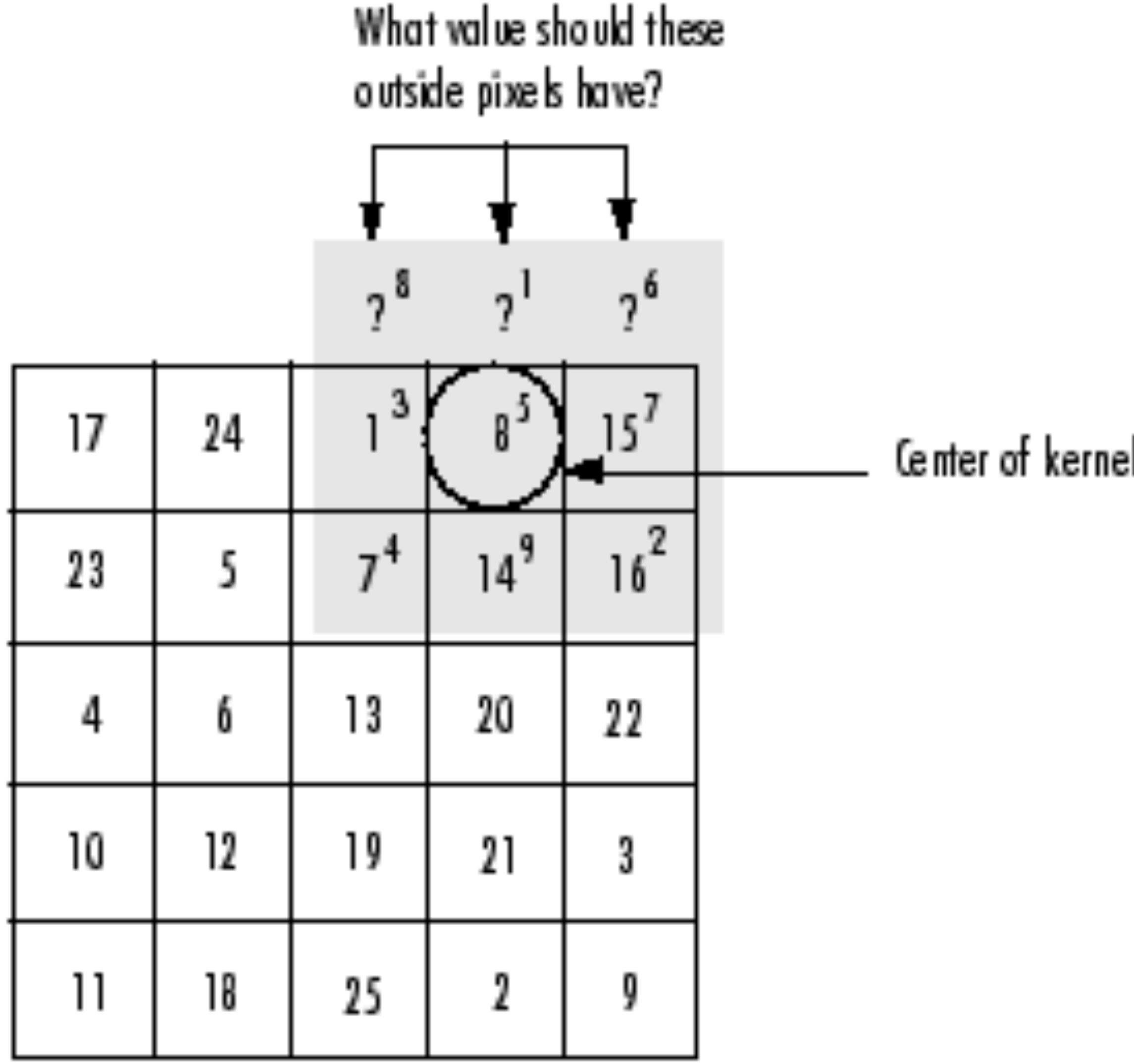
Input

1	2	0	1	3	
2	1	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

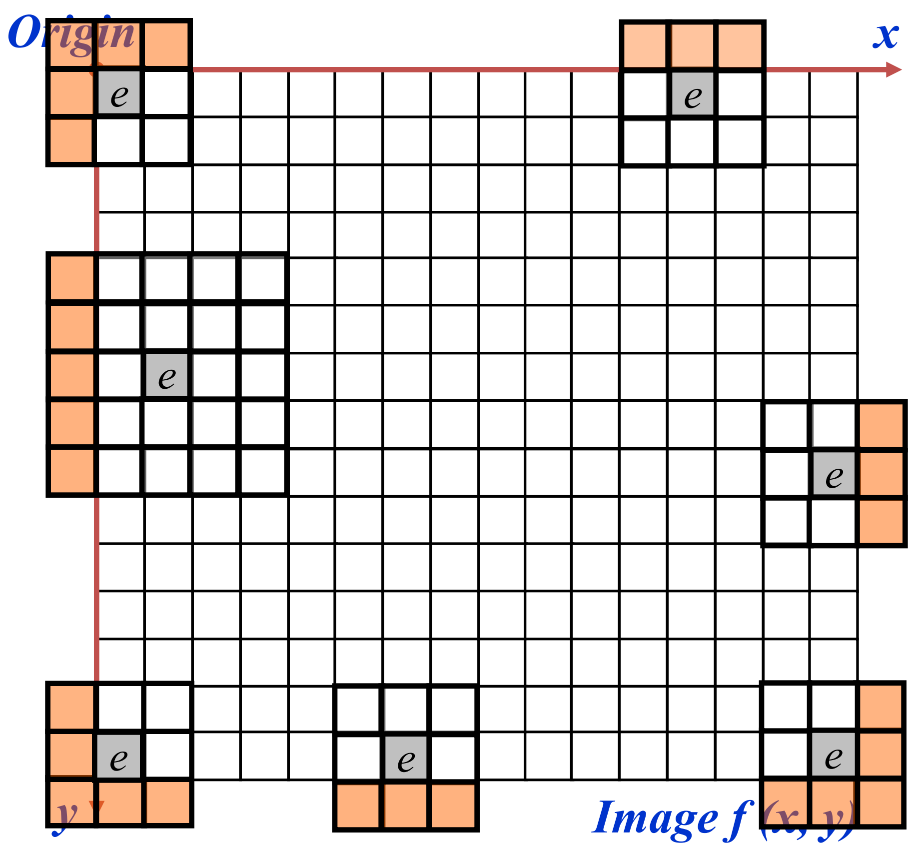
	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		
	$\frac{13}{9}$	$\frac{11}{9}$	$\frac{13}{9}$		
	$\frac{16}{9}$	$\frac{12}{9}$	$\frac{11}{9}$		

Problems at the borders



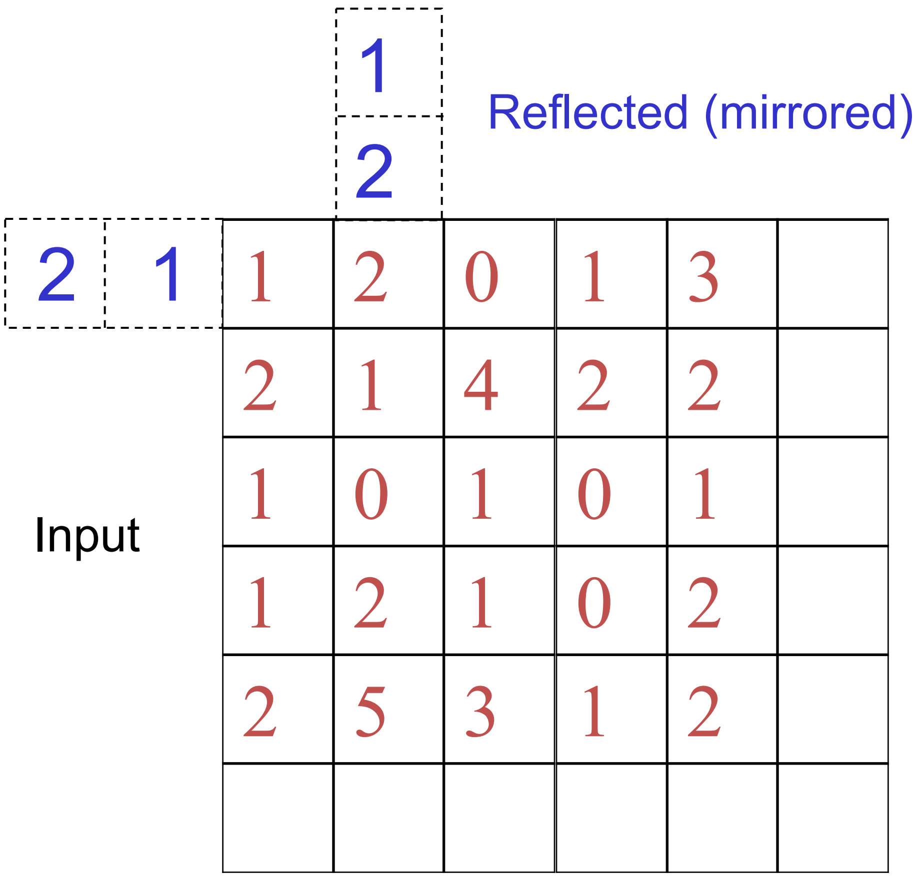
Strange Things Happen At The Edges!

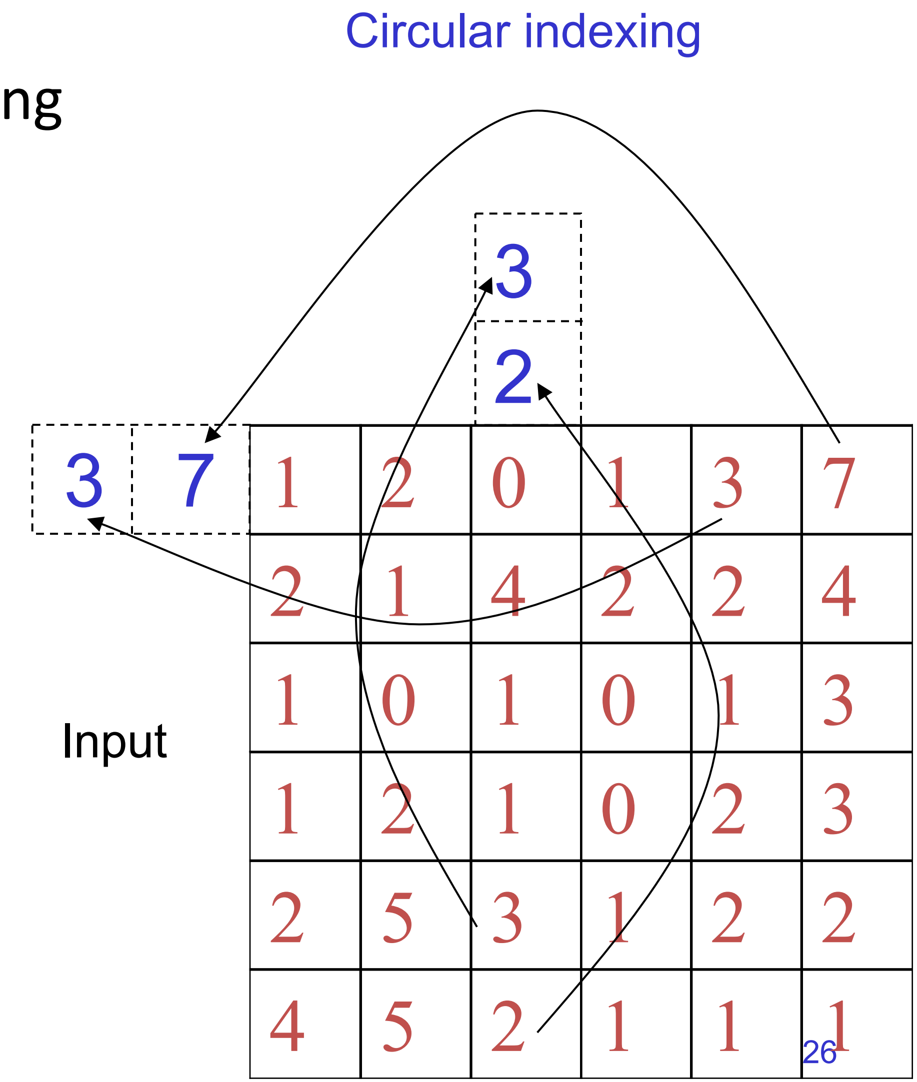
At the edges of an image we are missing pixels to form a neighbourhood



Problems at the borders

- Solutions:
  - Complex and perhaps wrong





Smoothing Spatial Filters

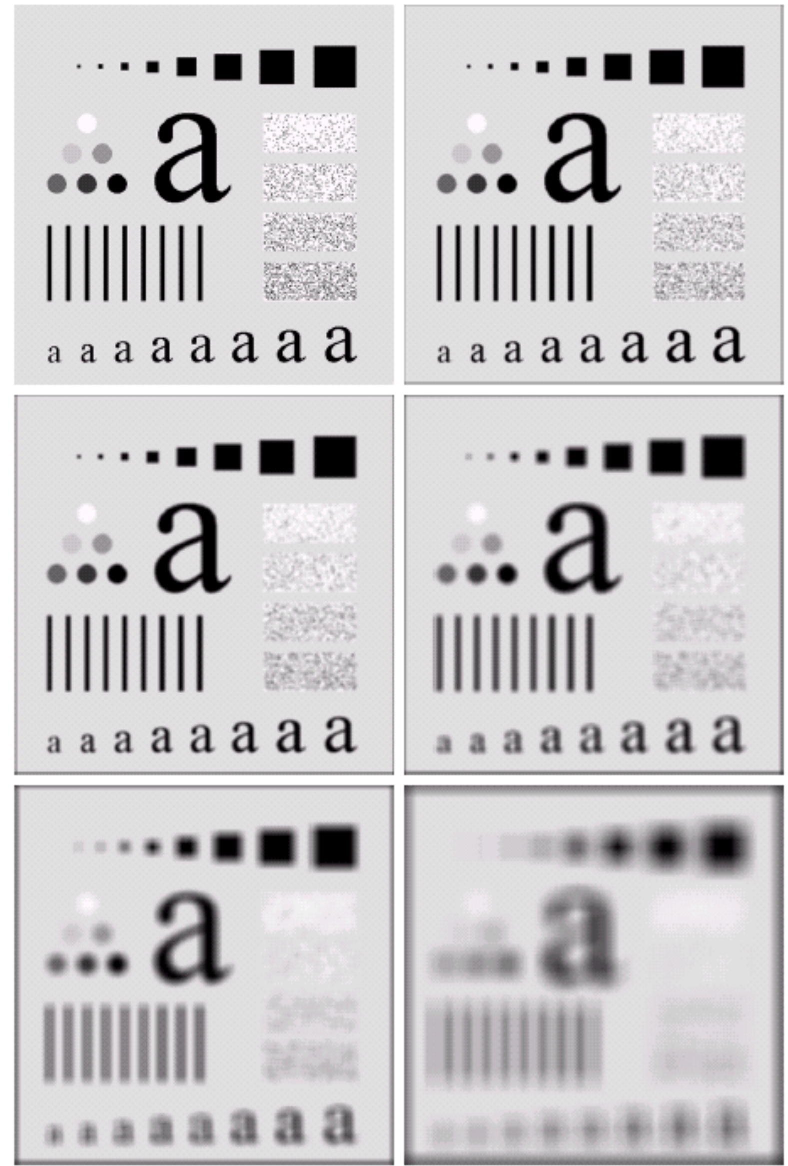
$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

a b

FIGURE 3.32 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

Image Smoothing Example

- The image at the top left is an original image of size 500*500 pixels
- The subsequent images show the image after filtering with an averaging filter of increasing sizes
  - 3, 5, 9, 15 and 35
- Notice how detail begins to disappear



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Smoothing Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

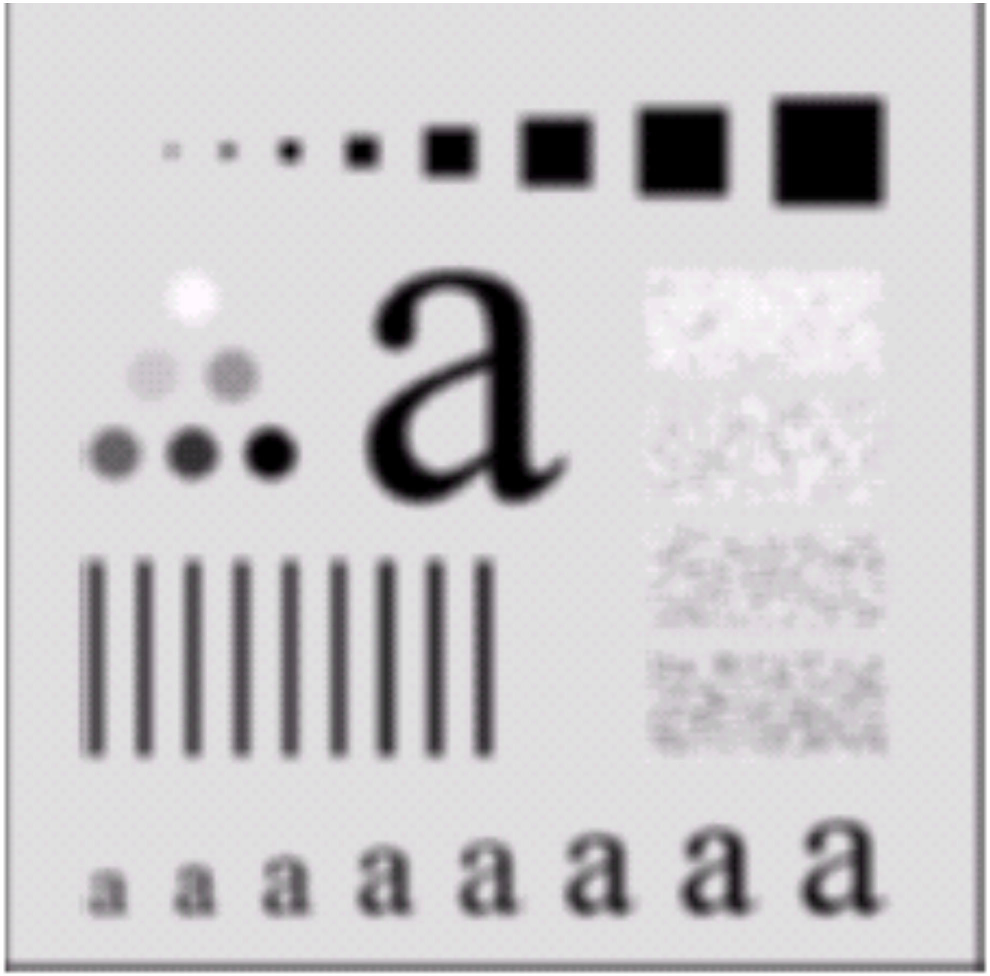
Image Smoothing Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Smoothing Example

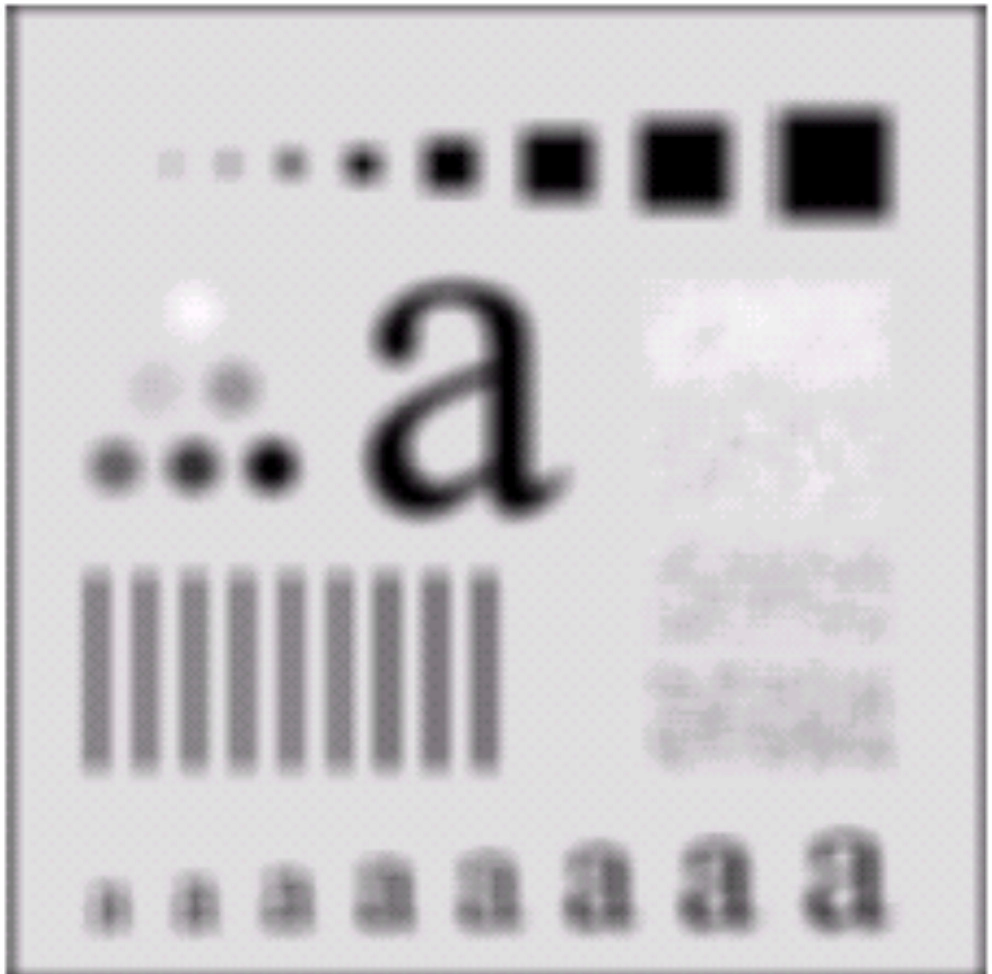
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Smoothing Example



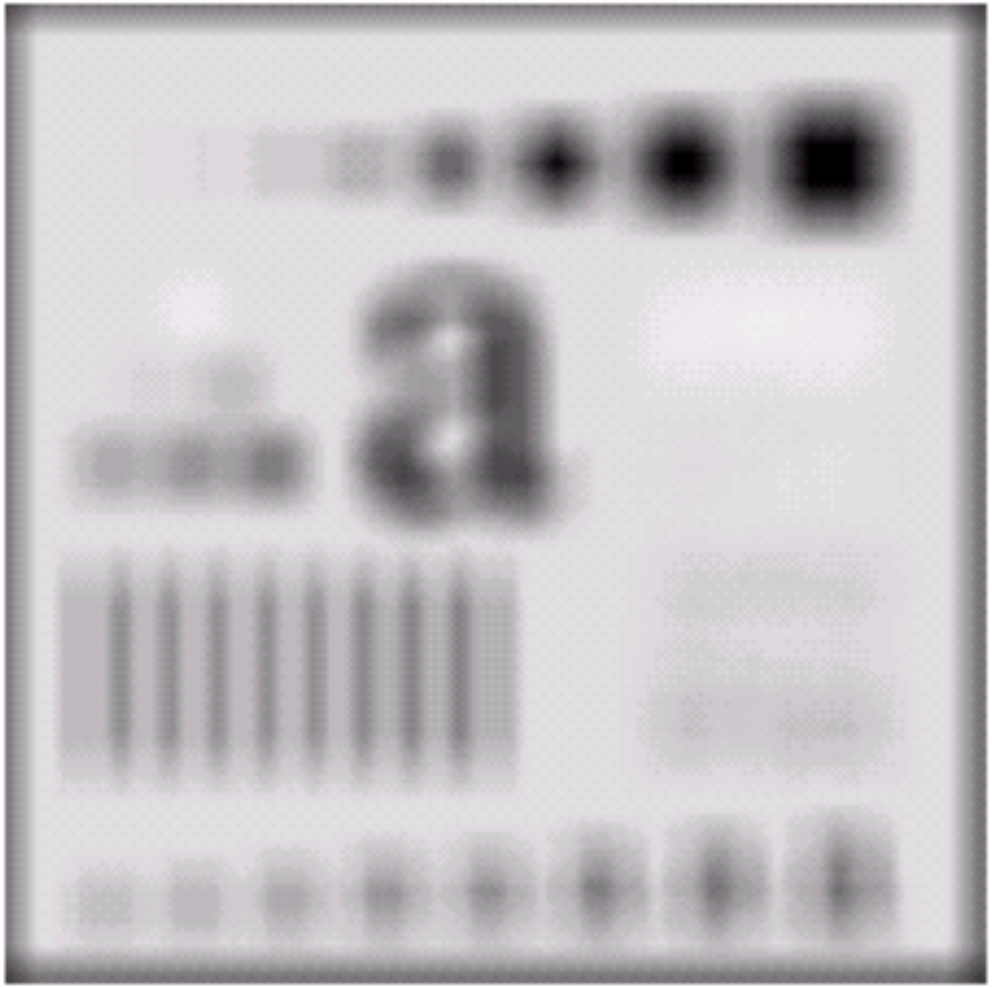
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Smoothing Example



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

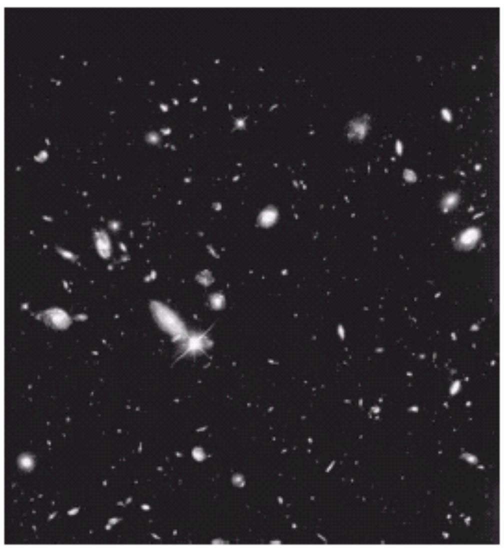
Image Smoothing Example



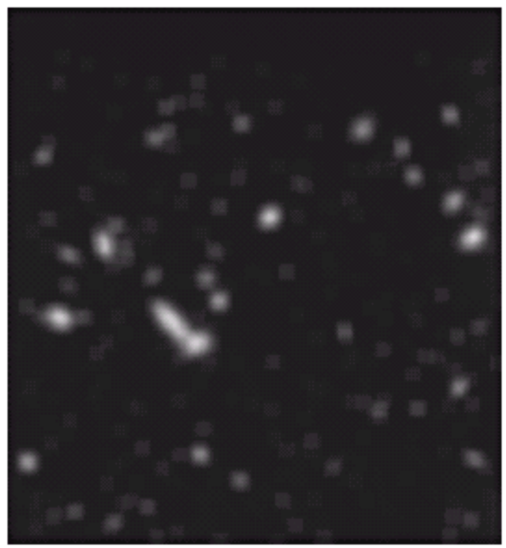
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Another Smoothing Example

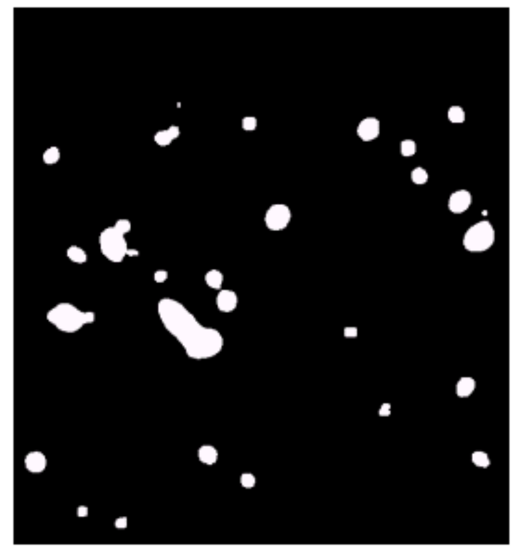
- By smoothing the original image we get rid of lots of the finer detail which leaves only the gross features for thresholding



Original Image



Smoothed Image



Thresholded Image

Convolution examples

Original
Bike

Bike blurred by convolution
Impulse response „box filter“

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

Convolution examples

Original
Bike

Bike blurred horizontally
Filter impulse response

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

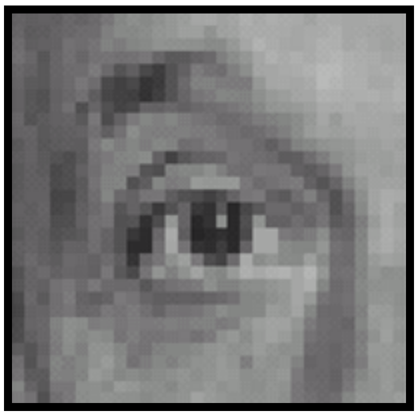
Convolution examples

Original
Bike

Bike blurred vertically
Filter impulse response

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

Practice with linear filters



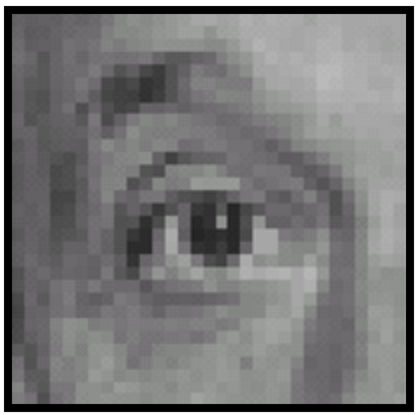
Original

0	0	0
0	1	0
0	0	0

?

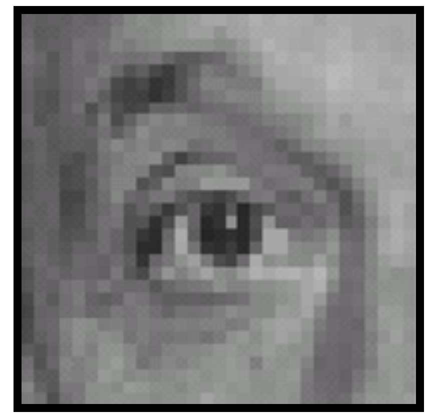
Source: D. Lowe

Practice with linear filters



Original

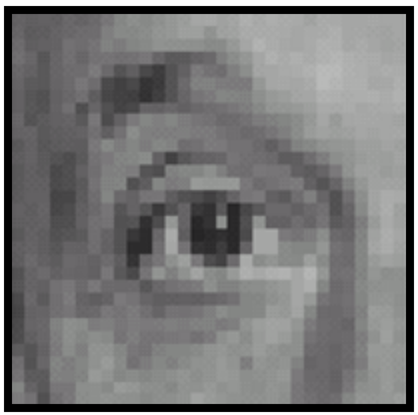
0	0	0
0	1	0
0	0	0



Filtered
(no change)

Source: D. Lowe

Practice with linear filters



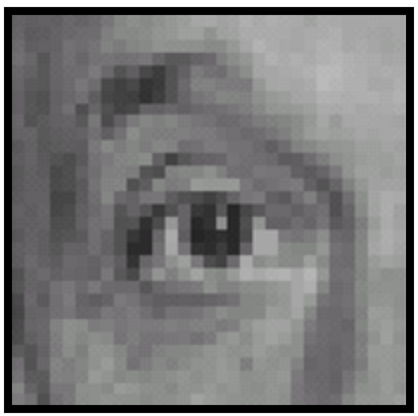
Original

0	0	0
0	0	1
0	0	0

?

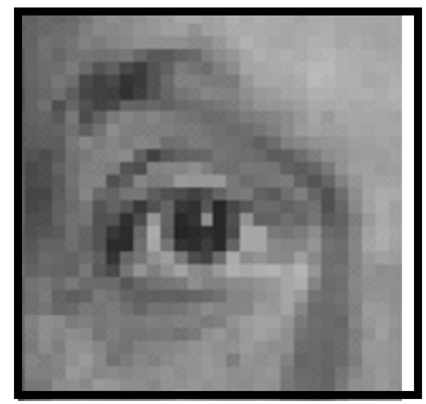
Source: D. Lowe

Practice with linear filters



Original

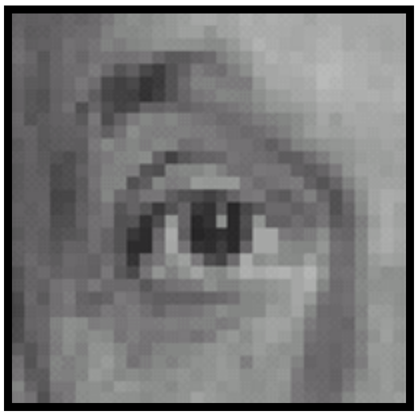
0	0	0
0	0	1
0	0	0



Shifted left
By 1 pixel

Source: D. Lowe

Practice with linear filters



Original

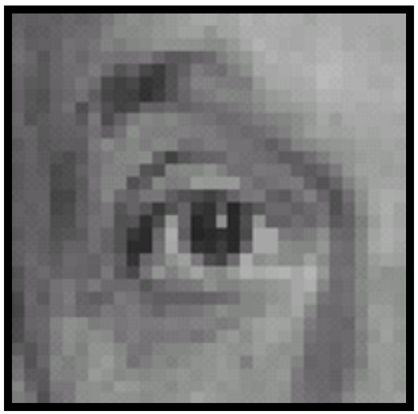
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

?

(Note that filter sums to 1)

Source: D. Lowe

Practice with linear filters



Original

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Sharpening filter

- Accentuates differences with local average

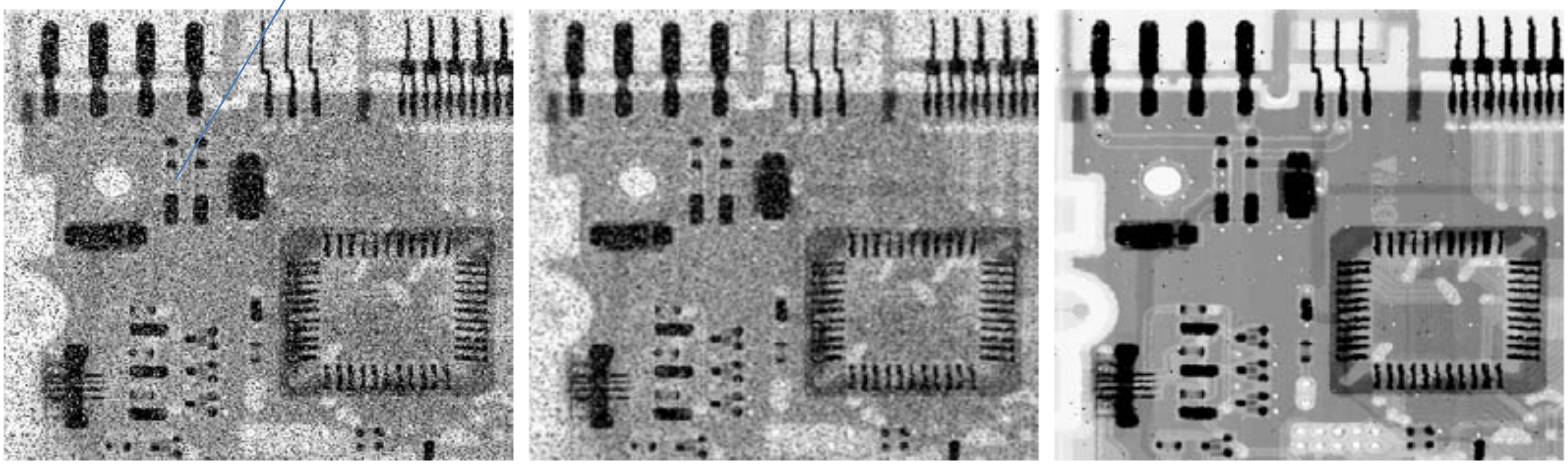
Source: D. Lowe

Order-Statistic (Nonlinear) Filters

5 8 10 15 20 22 25 32 100

10	22	20
5	8	15
32	25	100

10	22	20
5	20	15
32	25	100



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening Spatial Filters

Previously we have looked at smoothing filters which remove fine detail

Sharpening spatial filters seek to highlight fine detail

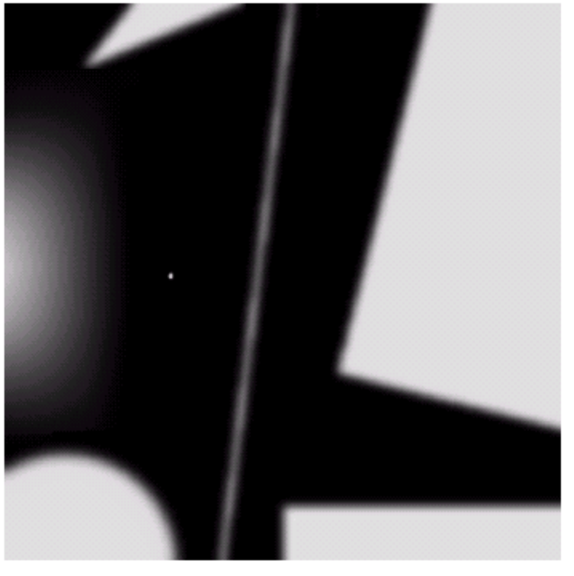
- Remove blurring from images
- Highlight edges

Sharpening filters are based on *spatial differentiation*

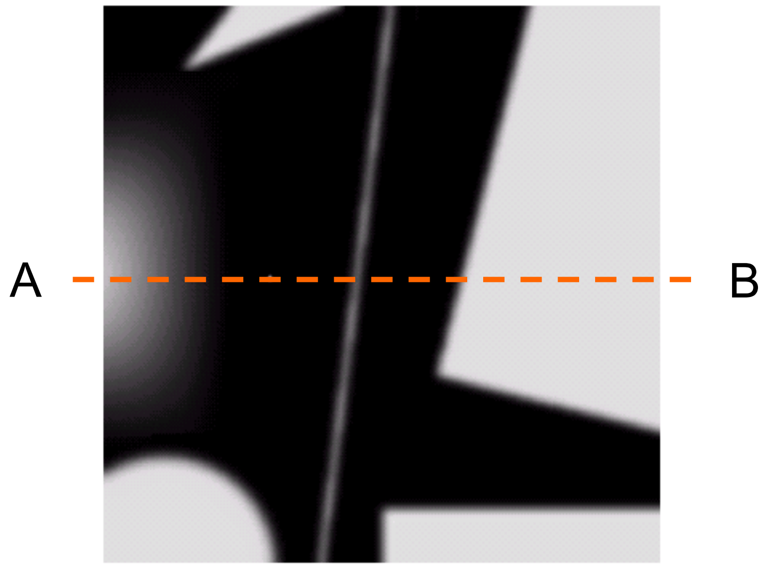
Spatial Differentiation

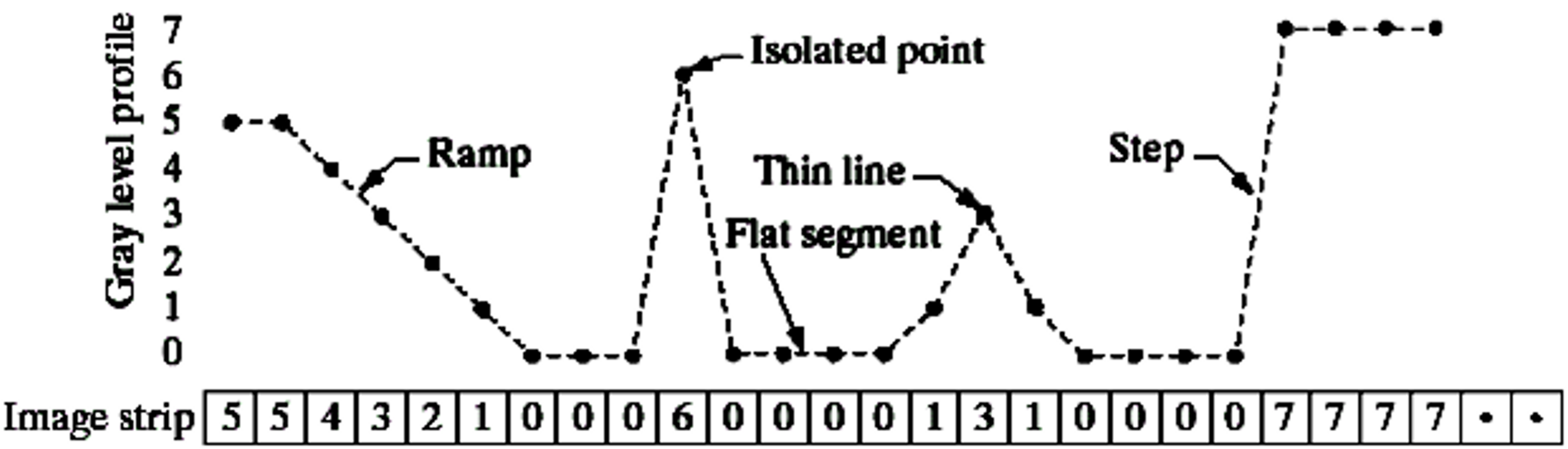
Differentiation measures the *rate of change* of a function

Let's consider a simple 1 dimensional example



Spatial Differentiation





Images taken from Gonzalez & Woods, Digital Image Processing (2002)

1st Derivative

The formula for the 1st derivative of a function is as follows:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

It's just the difference between subsequent values and measures the rate of change of the function

2nd Derivative

The formula for the 2nd derivative of a function is as follows:

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

Simply takes into account the values both before and after the current value

Using Second Derivatives For Image Enhancement

- The 2^{nd} derivative is more useful for image enhancement than the 1^{st} derivative
  - Stronger response to fine detail
  - Simpler implementation
  - We will come back to the 1^{st} order derivative later on
- The first sharpening filter we will look at is the *Laplacian*
  - Isotropic
  - One of the simplest sharpening filters
  - We will look at a digital implementation

The Laplacian

The Laplacian is defined as follows:

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

where the partial 1st order derivative in the x direction is defined as follows:

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

and in the y direction as follows:

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

The Laplacian (cont…)

So, the Laplacian can be given as follows:

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

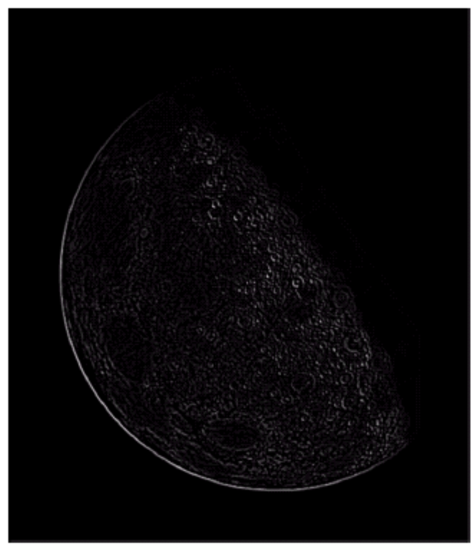
We can easily build a filter based on this

0	1	0
1	-4	1
0	1	0

The Laplacian (cont…)

Applying the Laplacian to an image we get a new image that highlights edges and other discontinuities

Original Image



Laplacian Filtered Image

Laplacian Filtered Image Scaled for Display

But That Is Not Very Enhanced!

The result of a Laplacian filtering is not an enhanced image

We have to do more work in order to get our final image

Subtract the Laplacian result from the original image to generate our final sharpened enhanced image

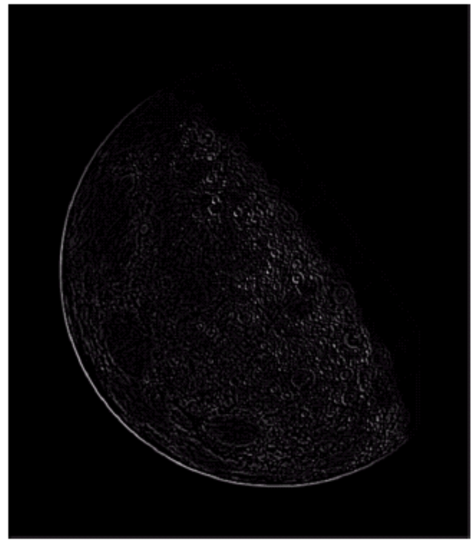
Laplacian Filtered Image Scaled for Display

$$g(x, y) = f(x, y) - \nabla^2 f$$

Laplacian Image Enhancement

Original Image

-



Laplacian Filtered Image

=

Sharpened Image

In the final sharpened image edges and fine detail are much more obvious

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Laplacian Image Enhancement

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

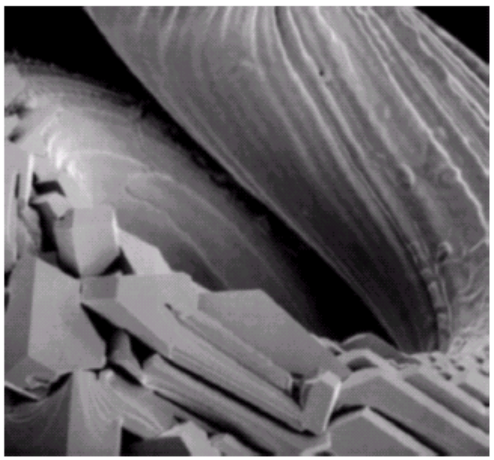
Simplified Image Enhancement

The entire enhancement can be combined into a single filtering operation

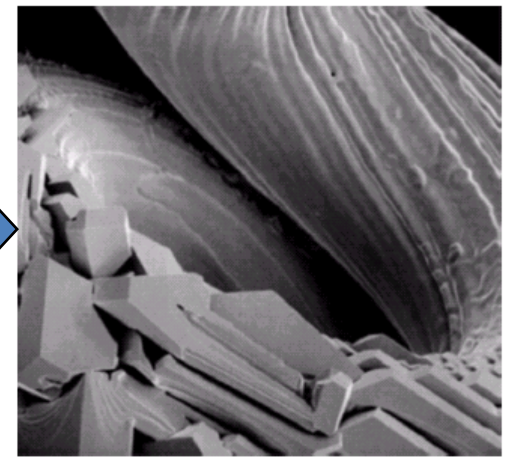
$$
\begin{aligned}
g(x,y) = f(x,y) - \nabla^2 f \\
= f(x,y) - [f(x+1,y) + f(x-1,y) \\
+ f(x,y+1) + f(x,y-1) \\
- 4f(x,y)] \\
= 5f(x,y) - f(x+1,y) - f(x-1,y) \\
- f(x,y+1) - f(x,y-1)
\end{aligned}
$$

Simplified Image Enhancement (cont…)

This gives us a new filter which does the whole job for us in one step

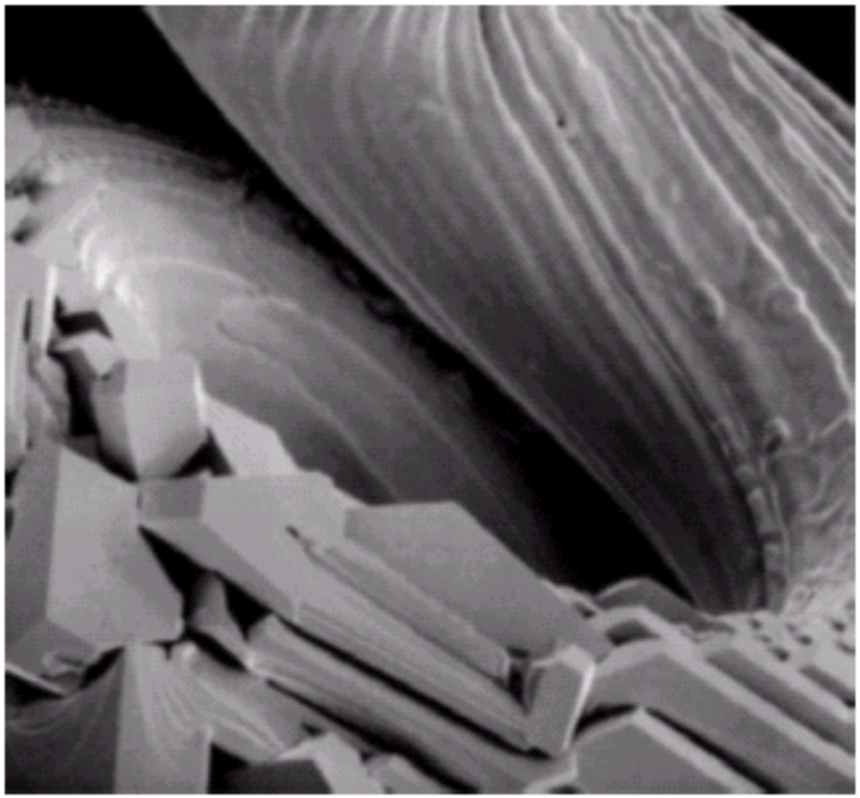


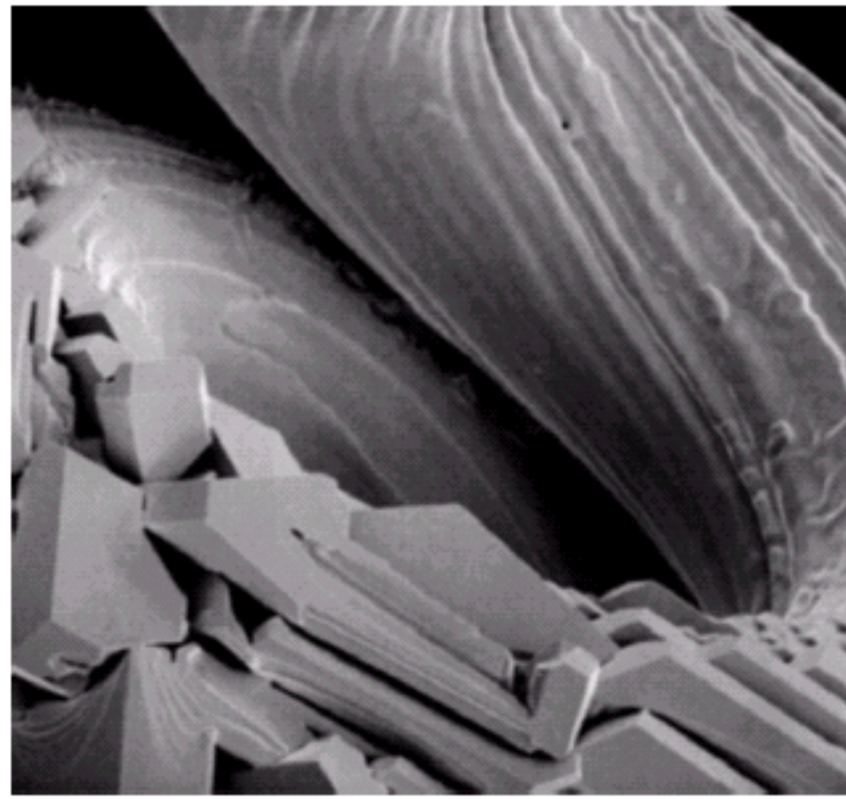
0	-1	0
-1	5	-1
0	-1	0



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Simplified Image Enhancement (cont…)





Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Variants On The Simple Laplacian

$$g(x, y) = f(x, y) + c\left[\nabla^2 f(x, y)\right]$$

There are lots of slightly different versions of the Laplacian that can be used:

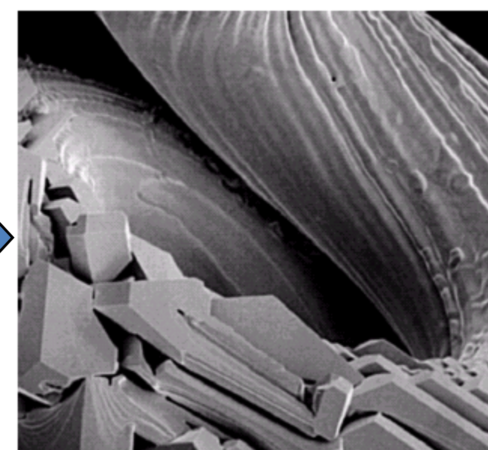
0	1	0
1	-4	1
0	1	0

Simple Laplacian

1	1	1
1	-8	1
1	1	1

Variant of Laplacian

-1	-1	-1
-1	9	-1
-1	-1	-1



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Unsharp Masking and Highboost Filtering

$$g_{\text{mask}}(x,y) = f(x,y) - \bar{f}(x,y)$$

$$g(x,y) = f(x,y) + k\,g_{\text{mask}}(x,y)$$

Original image.

Result of blurring with a Gaussian filter.

Unsharp mask.

Result of using unsharp masking.

Result of using highboost filtering.

Sobel Operators

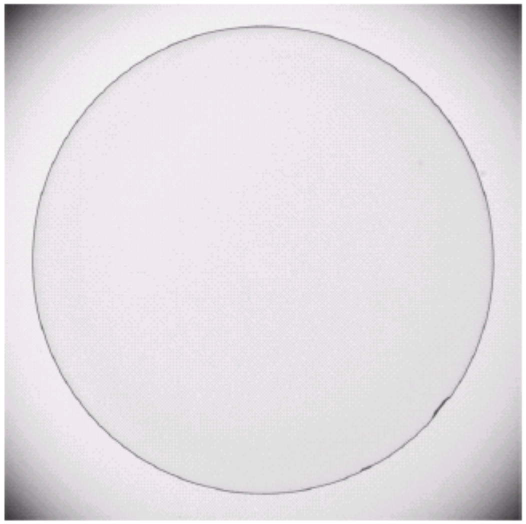
Based on the previous equations we can derive the *Sobel Operators*

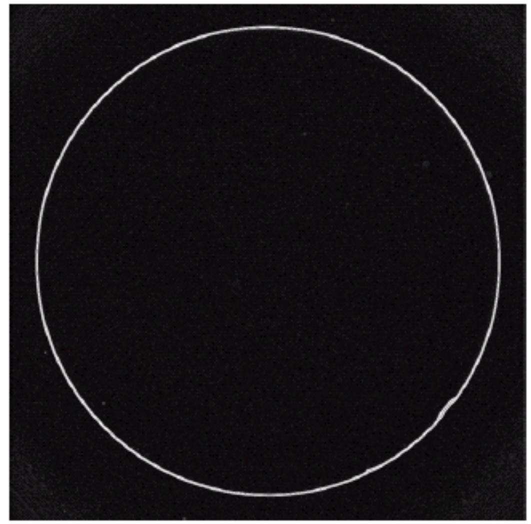
-1	-2	-1
0	0	0
1	2	1

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

To filter an image it is filtered using both operators the results of which are added together

Sobel Example





An image of a contact lens which is enhanced in order to make defects (at four and five o'clock in the image) more obvious

Sobel filters are typically used for edge detection

1^{st} & 2^{nd} Derivatives

Comparing the 1^{st} and 2^{nd} derivatives we can conclude the following:

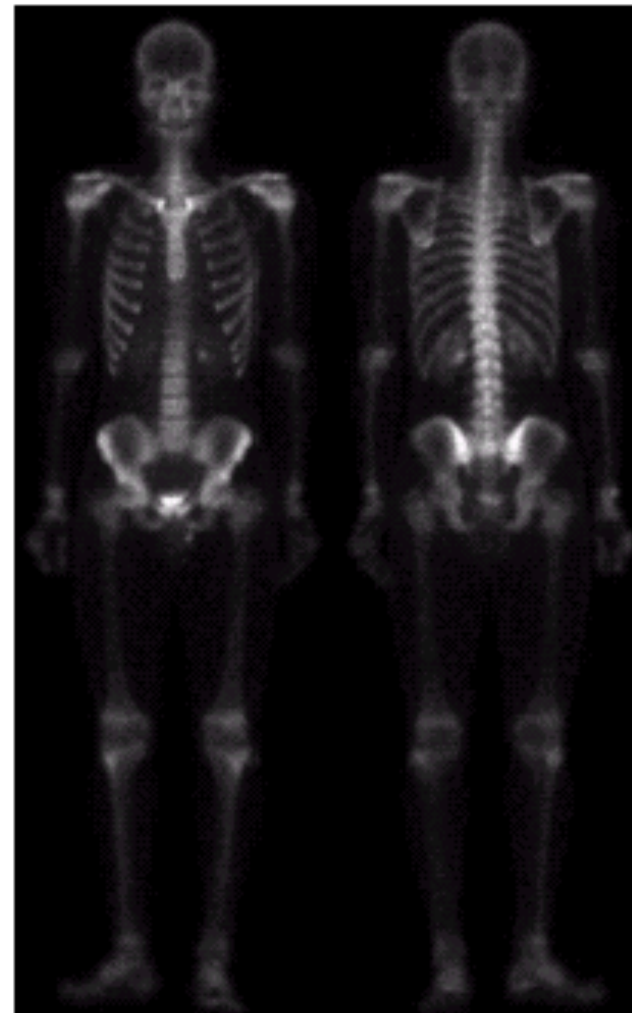
- 1^{st} order derivatives generally produce thicker edges
- 2^{nd} order derivatives have a stronger response to fine detail e.g. thin lines
- 1^{st} order derivatives have stronger response to grey level step
- 2^{nd} order derivatives produce a double response at step changes in grey level

Combining Spatial Enhancement Methods

Successful image enhancement is typically not achieved using a single operation

Rather we combine a range of techniques in order to achieve a final result

This example will focus on enhancing the bone scan to the right

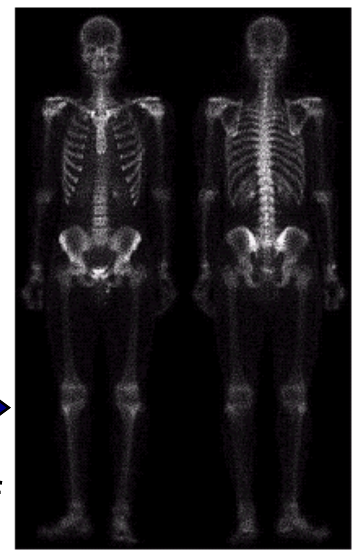


Combining Spatial Enhancement Methods (cont…)

(a)

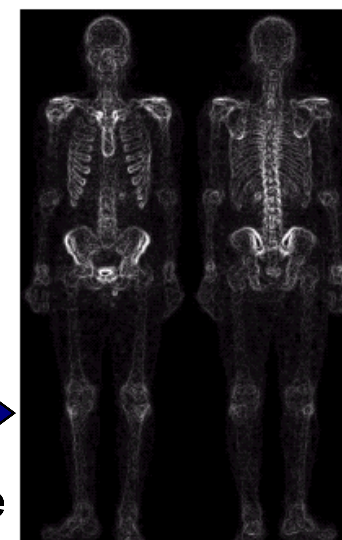
Laplacian filter of bone scan (a)

(b)



Sharpened version of bone scan achieved by subtracting (a) and (b)

(c)



Sobel filter of bone scan (a)

(d)

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Combining Spatial Enhancement Methods (cont…)

(e)

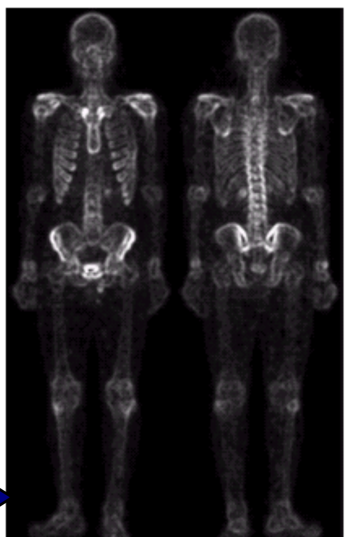
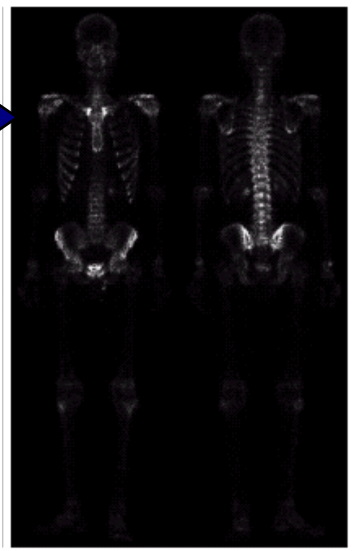


Image (d) smoothed with a 5*5 averaging filter

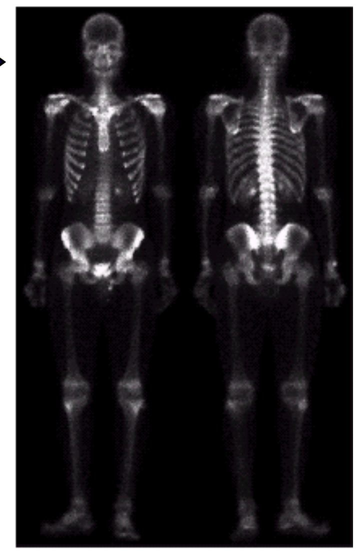
The product of (c) and (e) which will be used as a mask

(f)



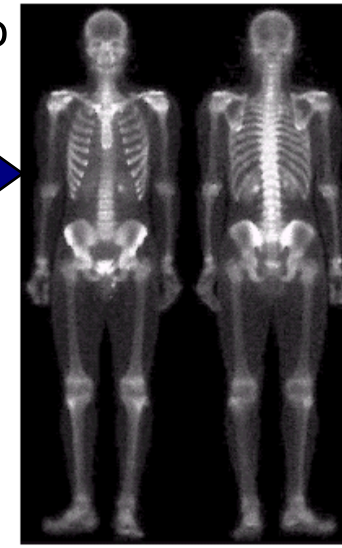
Sharpened image which is sum of (a) and (f)

(g)



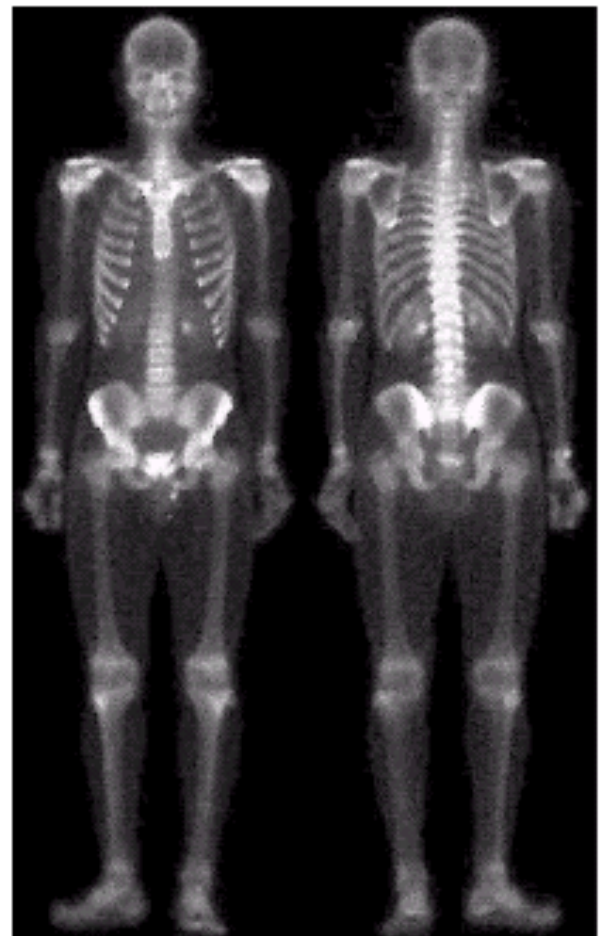
Result of applying a power-law trans. to (g)

(h)



Combining Spatial Enhancement Methods (cont…)

Compare the original and final images



Next Course: Frequency Spectra