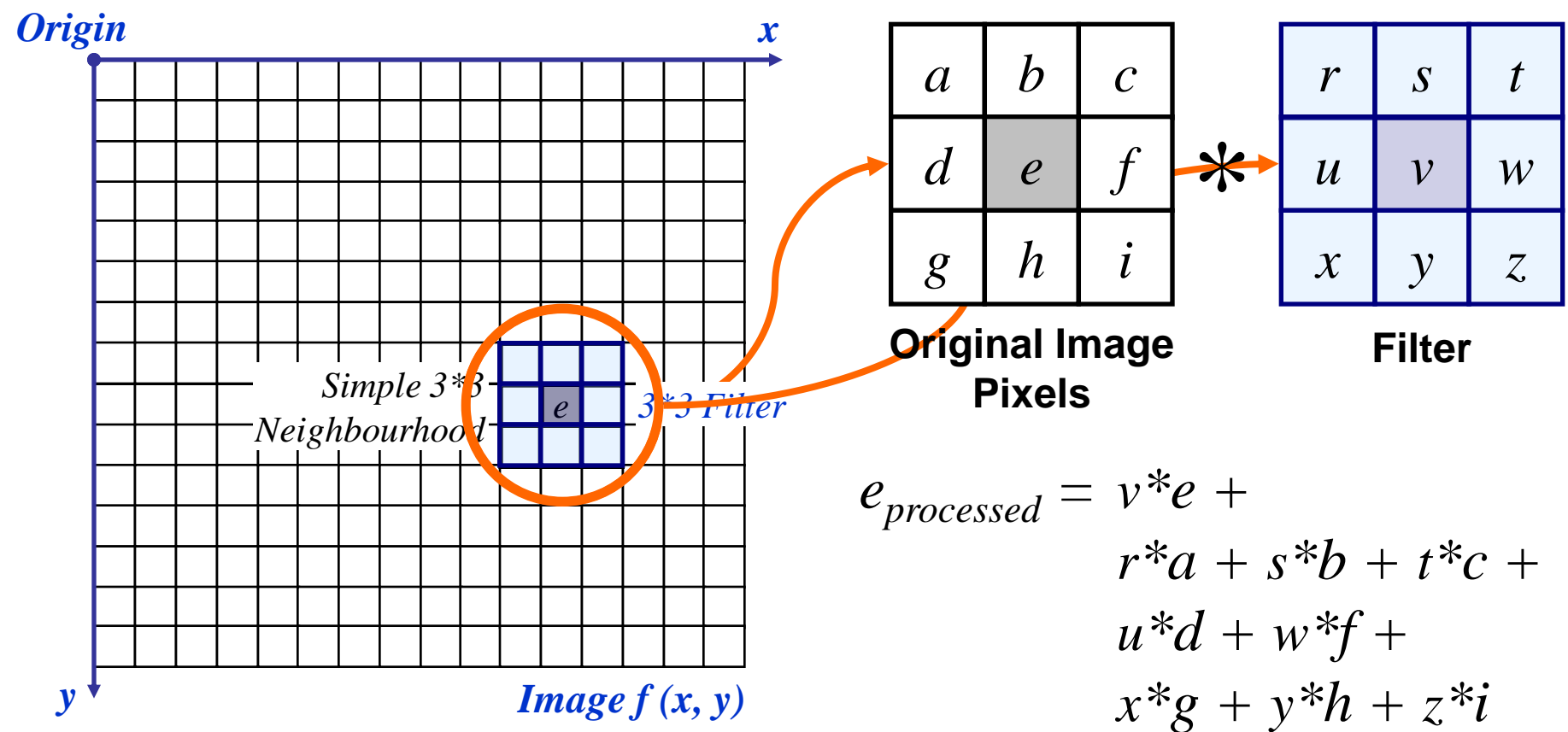


# Image Enhancement (Spatial Filtering 2)

In this lecture we will look at more spatial filtering techniques

- Spatial filtering refresher
- Sharpening filters
  - 1<sup>st</sup> derivative filters
  - 2<sup>nd</sup> derivative filters
- Combining filtering techniques

# Spatial Filtering Refresher



The above is repeated for every pixel in the original image to generate the smoothed image

# Sharpening Spatial Filters

Previously we have looked at smoothing filters which remove fine detail

*Sharpening spatial filters* seek to highlight fine detail

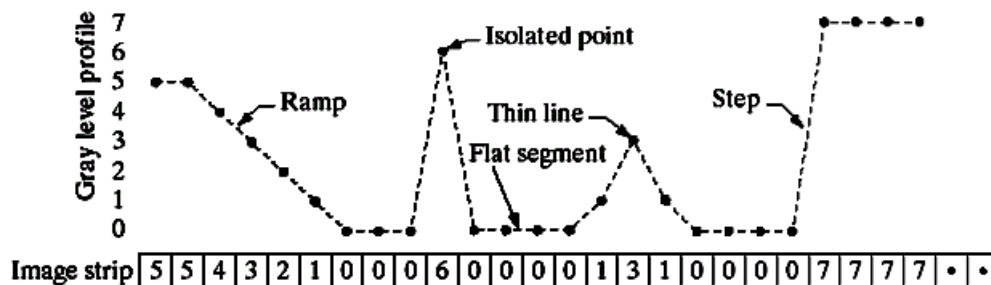
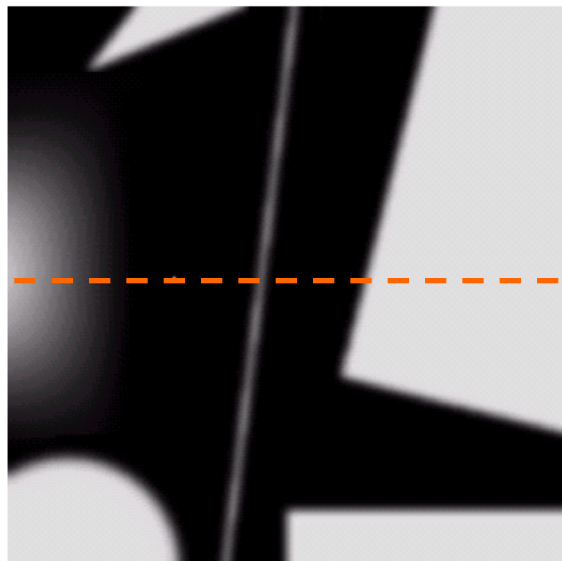
- Remove blurring from images
- Highlight edges

Sharpening filters are based on *spatial differentiation*

# Spatial Differentiation

Differentiation measures the *rate of change* of a function

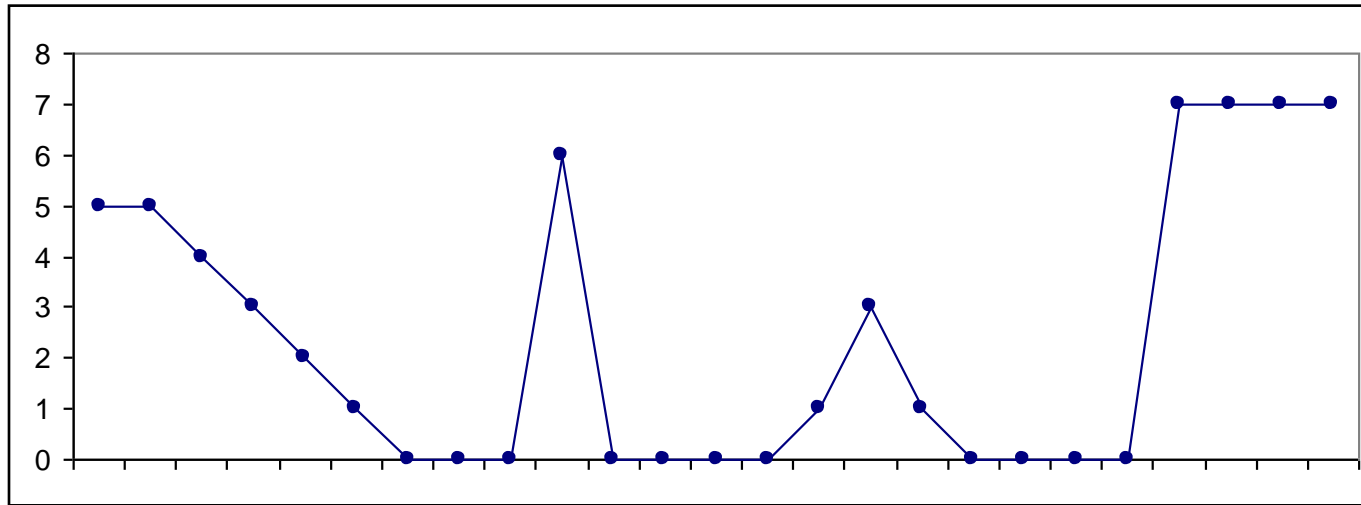
Let's consider a simple 1 dimensional example



The formula for the 1<sup>st</sup> derivative of a function is as follows:

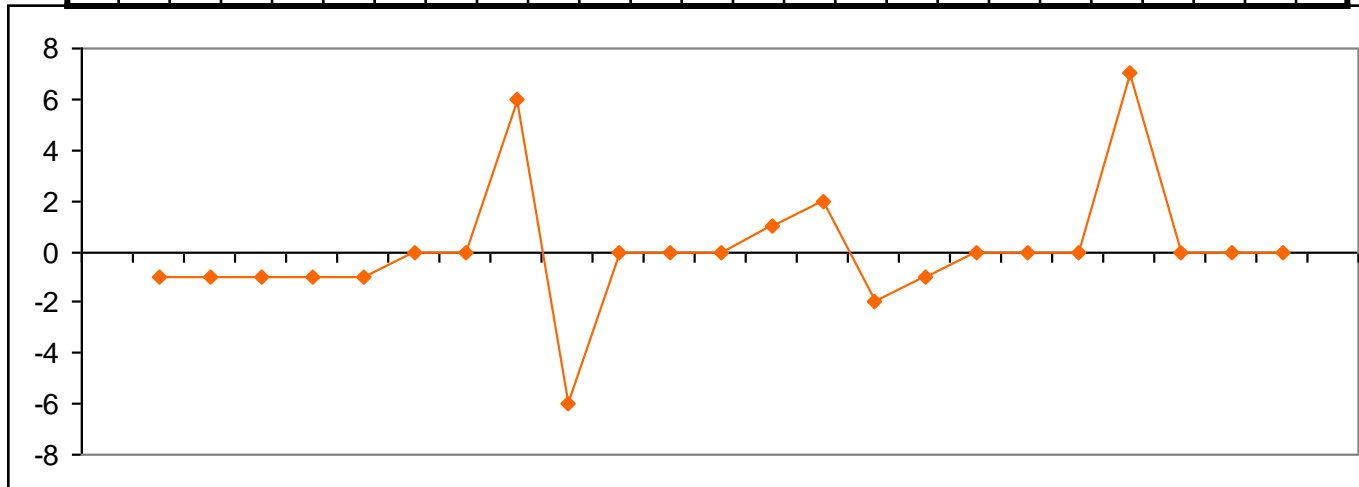
$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

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

1<sup>st</sup> Derivative (cont...)

5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

	0	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	0	0	
--	---	----	----	----	----	---	---	---	----	---	---	---	---	---	----	----	---	---	---	---	---	---	---	--



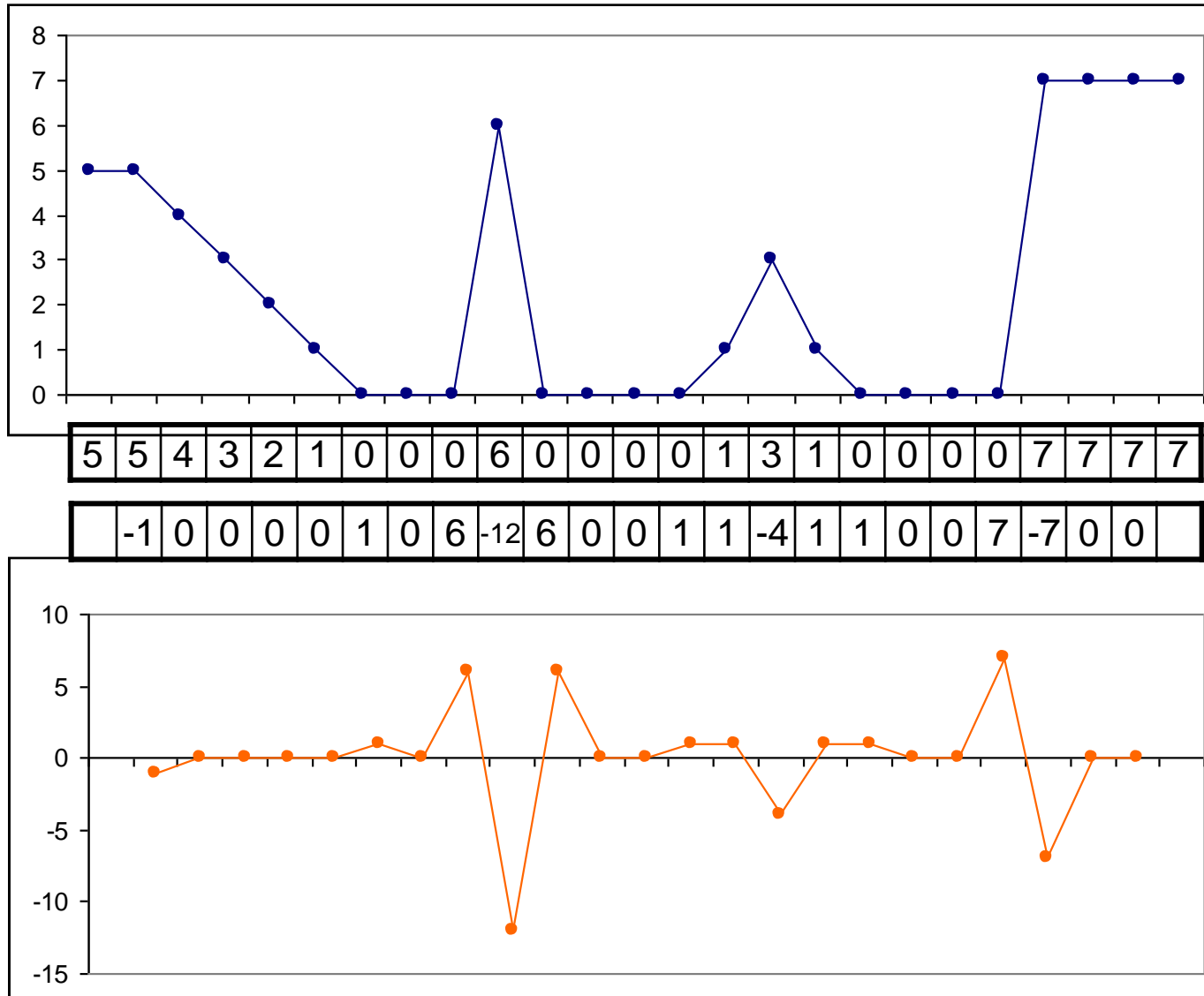
The formula for the 2<sup>nd</sup> derivative of a function is as follows:

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

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



# 2<sup>nd</sup> Derivative (cont...)



Comparing the 1<sup>st</sup> and 2<sup>nd</sup> derivatives we can conclude the following:

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

# Using Second Derivatives For Image Enhancement

The 2<sup>nd</sup> derivative is more useful for image enhancement than the 1<sup>st</sup> derivative

- Stronger response to fine detail
- Simpler implementation
- We will come back to the 1<sup>st</sup> 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 is defined as follows:

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

where the partial 1<sup>st</sup> 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:

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

We can easily build a filter based on this

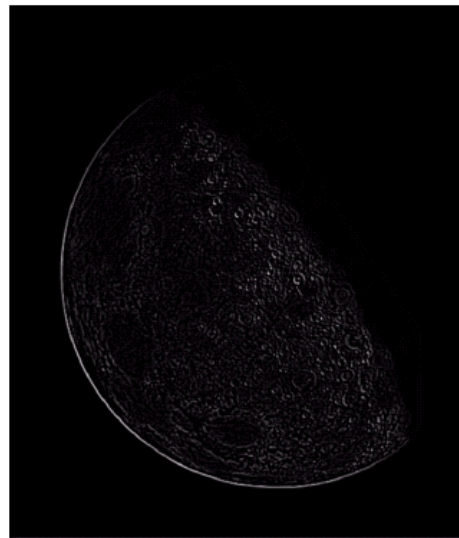
0	1	0
1	-4	1
0	1	0

# The Laplacian (cont...)

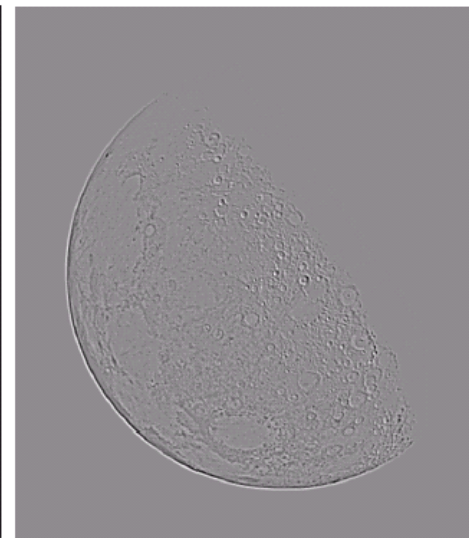
Applying the Laplacian to an image we get a new image that highlights edges and other discontinuities



Original  
Image



Laplacian  
Filtered Image



Laplacian  
Filtered Image  
Scaled for Display

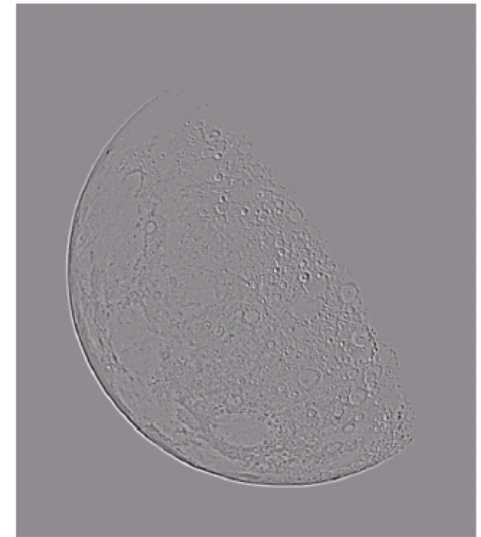
# But That Is Not Very Enhanced!

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

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



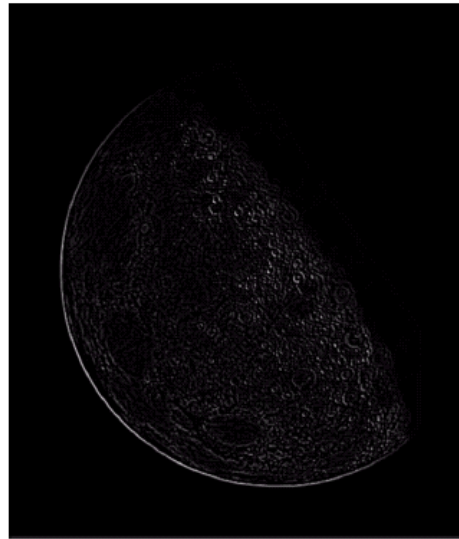
Laplacian  
Filtered Image  
Scaled for Display

# Laplacian Image Enhancement



Original  
Image

-



Laplacian  
Filtered Image

=



Sharpened  
Image

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



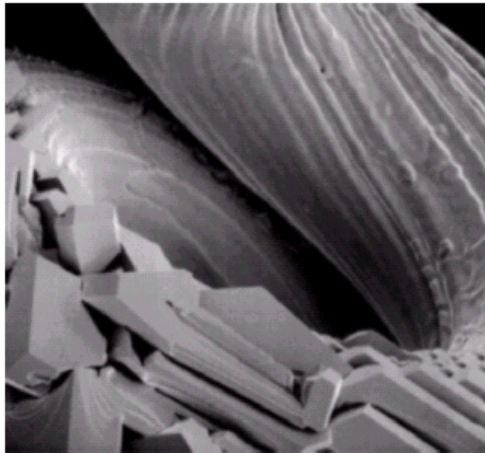
# Simplified Image Enhancement

The entire enhancement can be combined into a single filtering operation

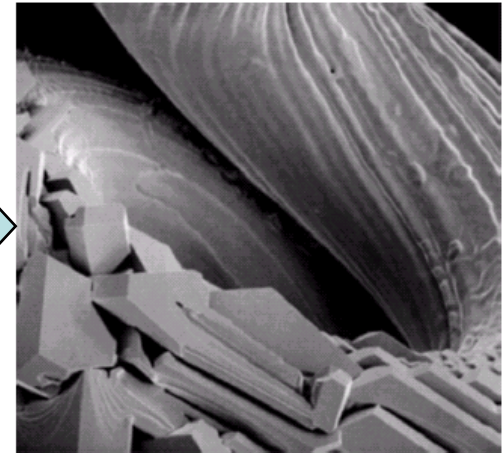
$$\begin{aligned} g(x, y) &= f(x, y) - \nabla^2 f \\ &= f(x, y) - [f(x+1, y) + f(x-1, y) \\ &\quad + f(x, y+1) + f(x, y-1) \\ &\quad - 4f(x, y)] \\ &= 5f(x, y) - f(x+1, y) - f(x-1, y) \\ &\quad - f(x, y+1) - f(x, y-1) \end{aligned}$$

# 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



# Variants On The Simple Laplacian

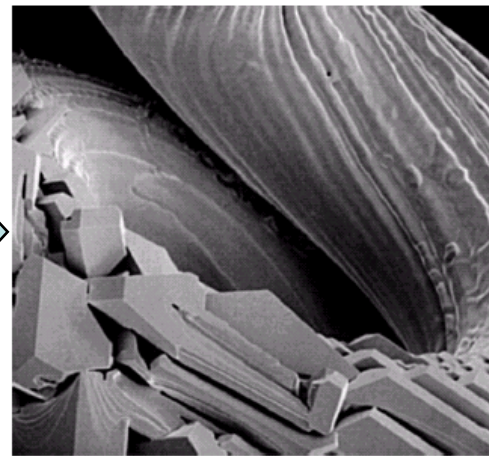
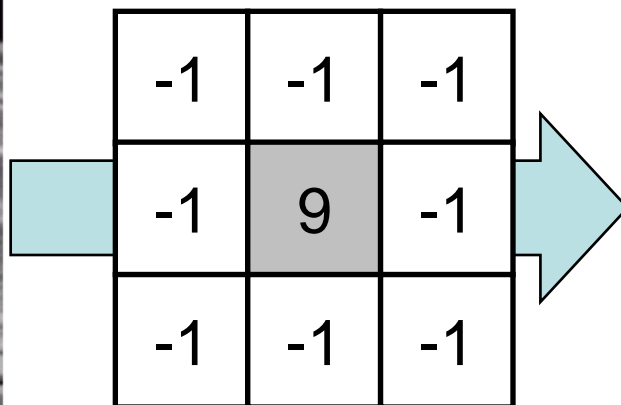
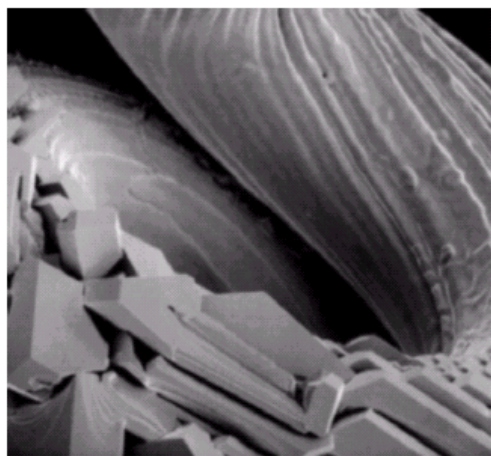
There are lots of slightly different versions of the Laplacian that can be used:

0	1	0
1	-4	1
0	1	0

Simple  
Laplacian

1	1	1
1	-8	1
1	1	1

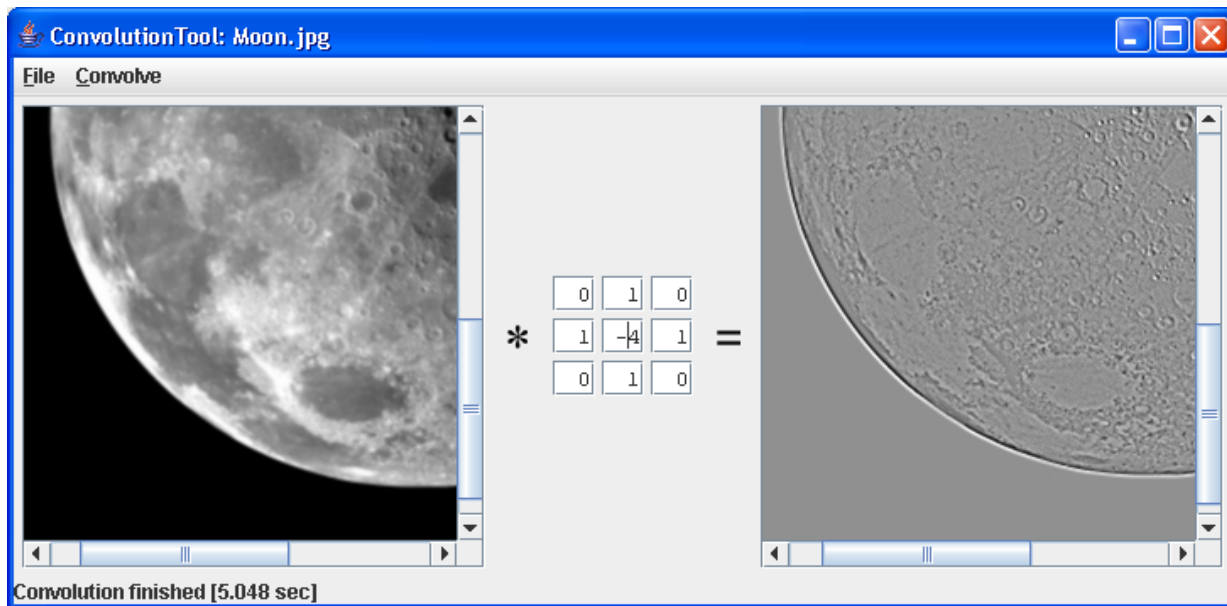
Variant of  
Laplacian



# Simple Convolution Tool In Java

A great tool for testing out different filters

- From the book “Image Processing tools in Java”
- Available from my [website](#)
- To launch: `java ConvolutionTool Moon.jpg`



Implementing 1<sup>st</sup> derivative filters is difficult in practice

For a function  $f(x, y)$  the gradient of  $f$  at coordinates  $(x, y)$  is given as the column vector:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

# 1<sup>st</sup> Derivative Filtering (cont...)

The magnitude of this vector is given by:

$$\begin{aligned}\nabla f &= \text{mag}(\nabla f) \\ &= [G_x^2 + G_y^2]^{1/2} \\ &= \left[ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right]^{1/2}\end{aligned}$$

For practical reasons this can be simplified as:

$$\nabla f \approx |G_x| + |G_y|$$

# 1<sup>st</sup> Derivative Filtering (cont...)

There is some debate as to how best to calculate these gradients but we will use:

$$\nabla f \approx \left| (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \right| \\ + \left| (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \right|$$

which is based on these coordinates

$z_1$	$z_2$	$z_3$
$z_4$	$z_5$	$z_6$
$z_7$	$z_8$	$z_9$

# Sobel Operators

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

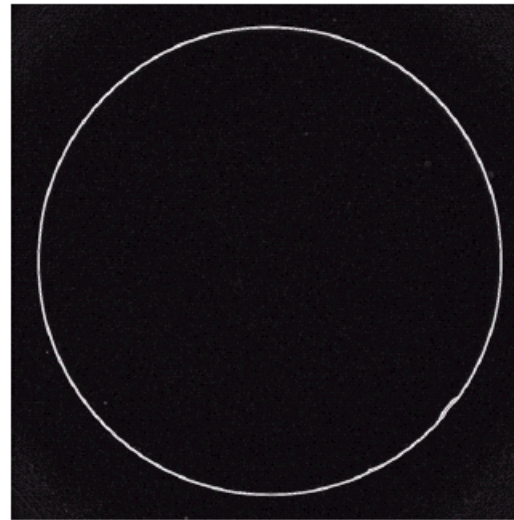
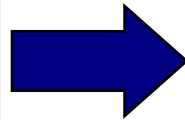
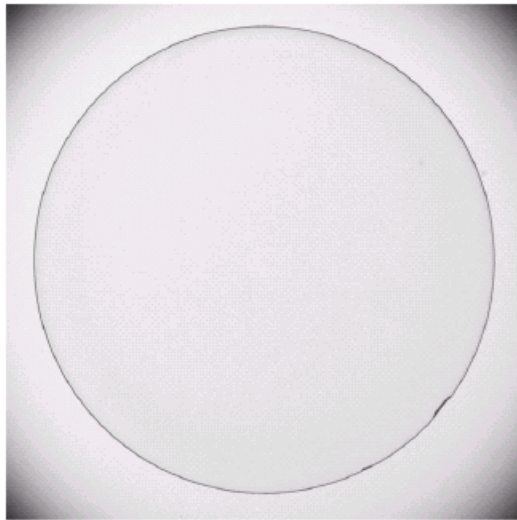
-1	-2	-1
0	0	0
1	2	1

-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

# Combining Spatial Enhancement Methods

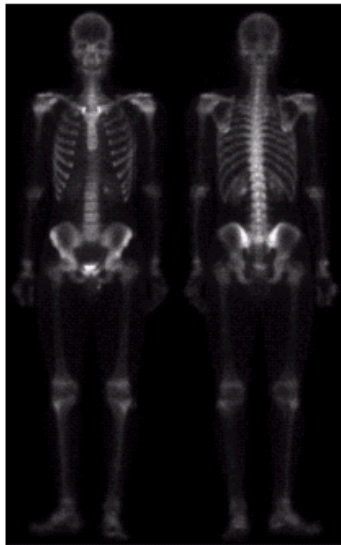
Successful image enhancement is typically not achieved using a single operation

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...)



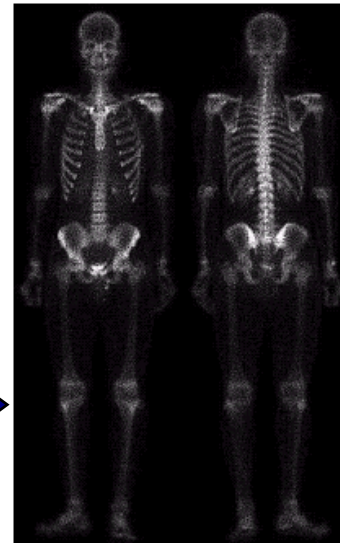
(a)

Laplacian filter of  
bone scan (a)



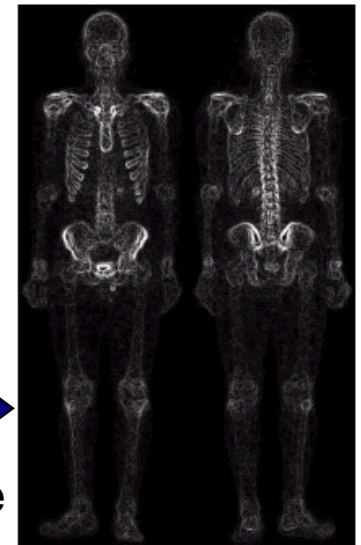
(b)

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



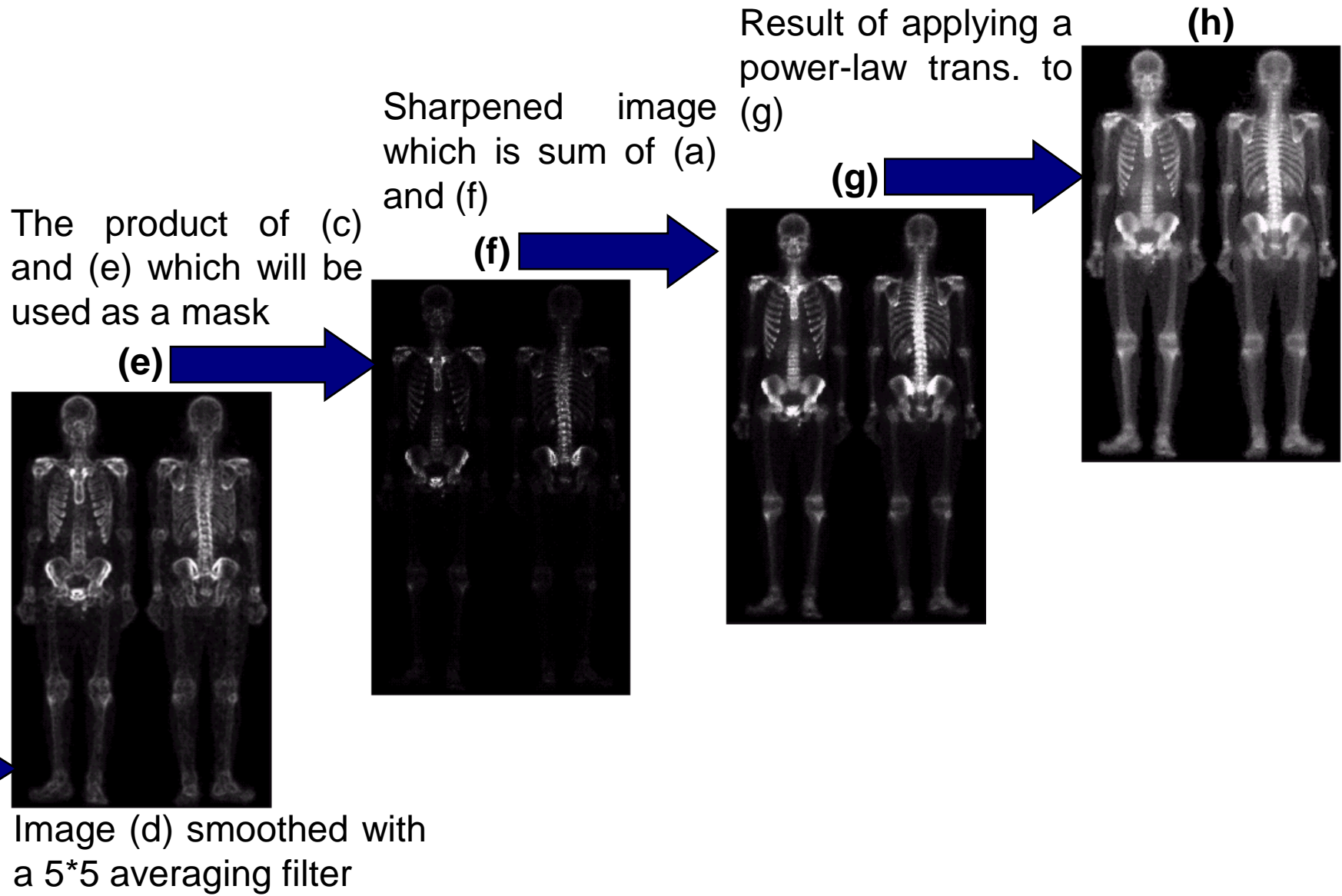
(c)

Sobel filter of bone  
scan (a)



(d)

# Combining Spatial Enhancement Methods (cont...)



# Combining Spatial Enhancement Methods (cont...)

Compare the original and final images



In this lecture we looked at:

- Sharpening filters
  - 1<sup>st</sup> derivative filters
  - 2<sup>nd</sup> derivative filters
- Combining filtering techniques