Digital Image Processing

Image Enhancement
(Spatial Filtering 2)

By

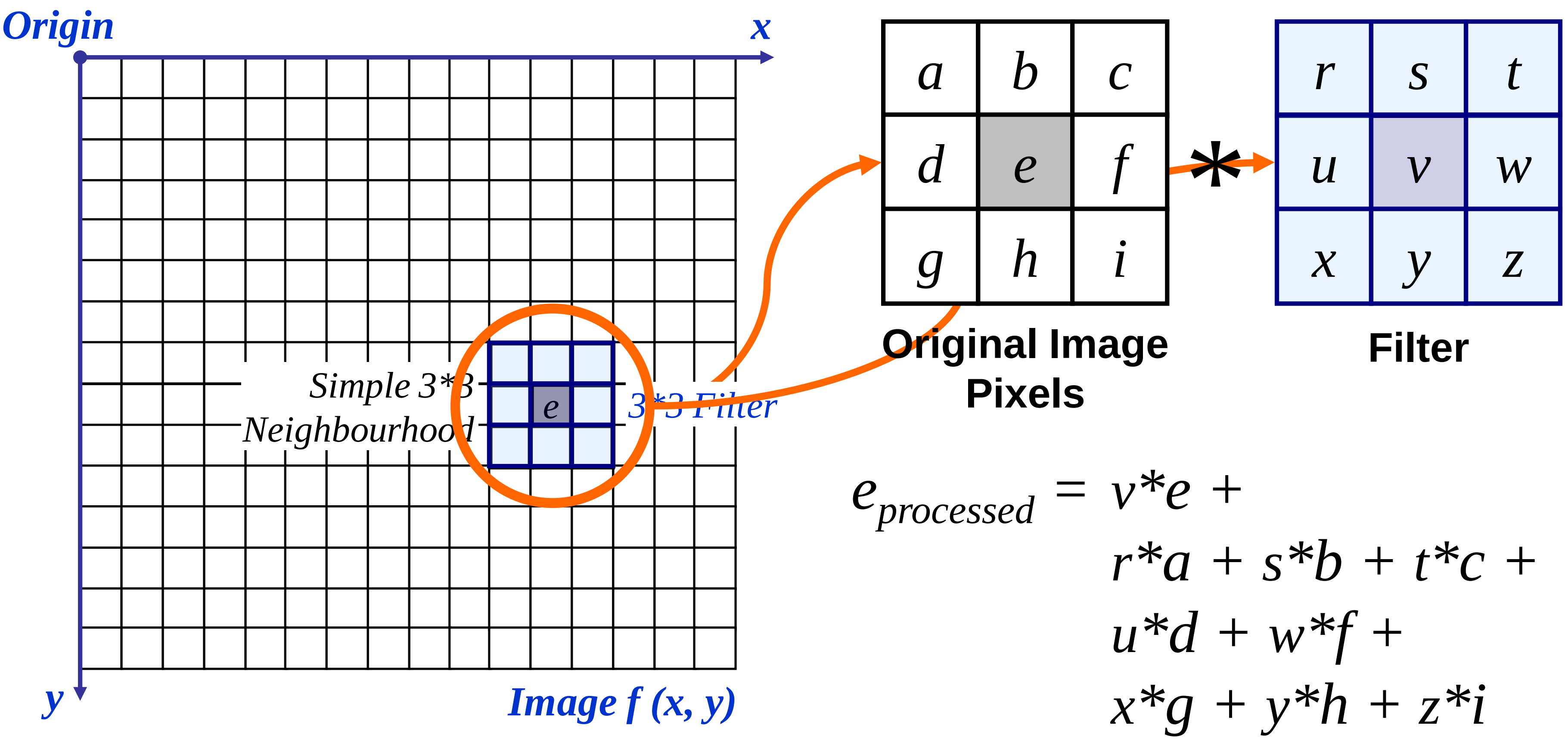
Dr. Hafeez Ur Rehman

Contents

In this lecture we will look at more spatial filtering techniques

- Spatial filtering refresher
- Sharpening filters
  - 1^{st} derivative filters
  - 2^{nd} derivative filters
- Combining filtering techniques

Spatial Filtering Refresher



$$
\begin{aligned}
e_{processed} = {} & v*e + \\
& r*a + s*b + t*c + \\
& u*d + w*f + \\
& x*g + y*h + z*i
\end{aligned}
$$

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*

Sharpening Spatial Filters

Job: to **highlight fine detail in an image or to enhance detail that has been blurred**, either in error or as a natural effect of a particular method of image acquisition.

- **Averaging** is like **Integration**
- **Sharpening** is like **Differentiation**

Sharpening Spatial Filters

This means that

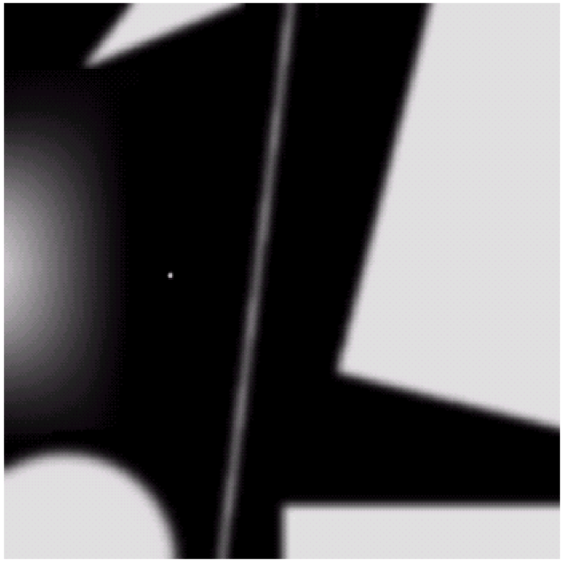
– any definition we use for a first derivative **(1)** must be **zero in flat segments** (areas of constant gray-level values); **(2)** must be **non-zero at the onset of a gray-level** step or ramp; and **(3)** must be **nonzero along ramps**.

– Similarly, any definition of a second derivative **(1)** must be **zero in flat areas**; **(2)** must be **non zero at the onset and end of a gray-level step** or ramp; and (3) must be **zero along ramps** of constant slope.

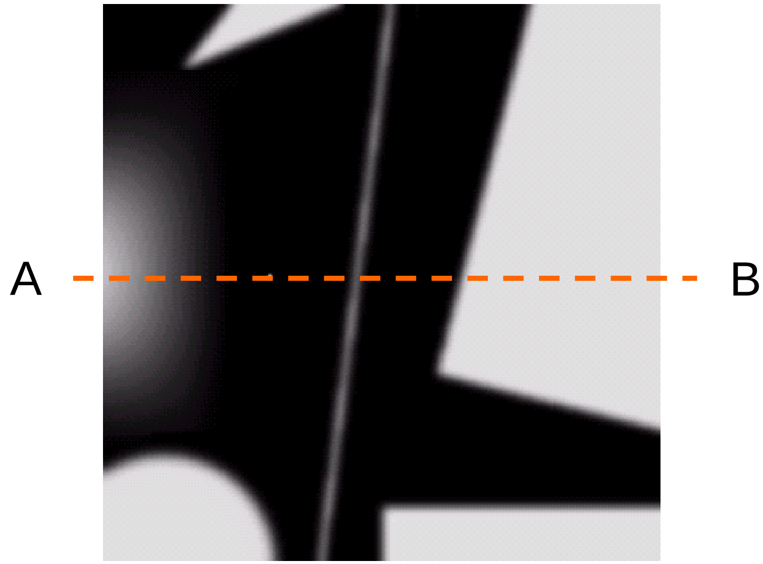
Spatial Differentiation

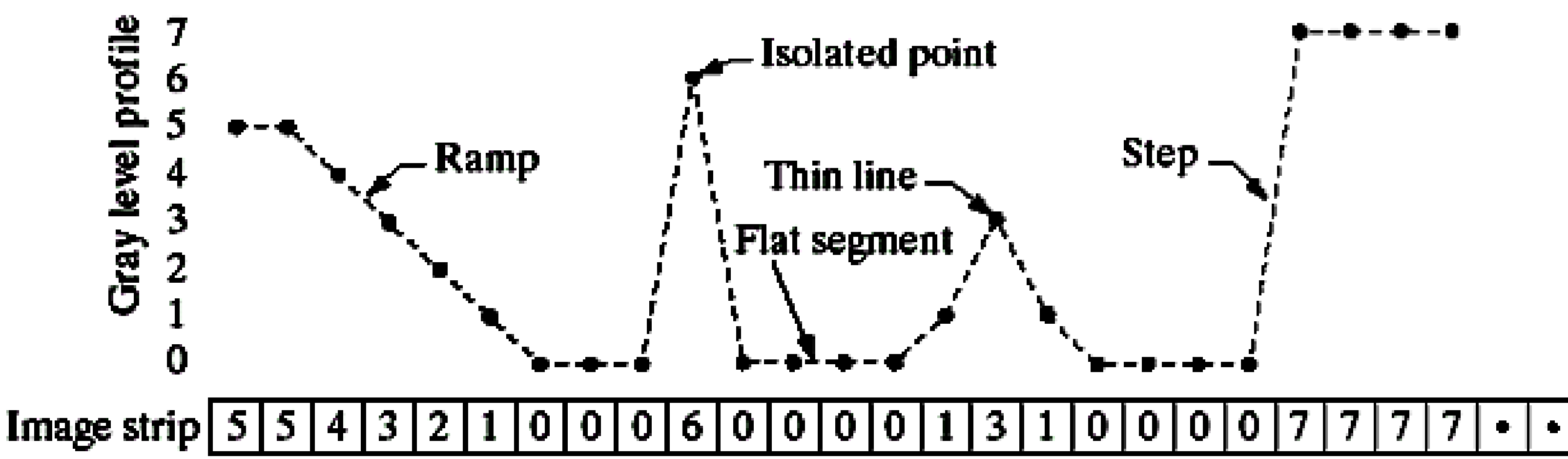
Differentiation measures the *rate of change* of a function

Let’s consider a simple 1 dimensional example



Spatial Differentiation





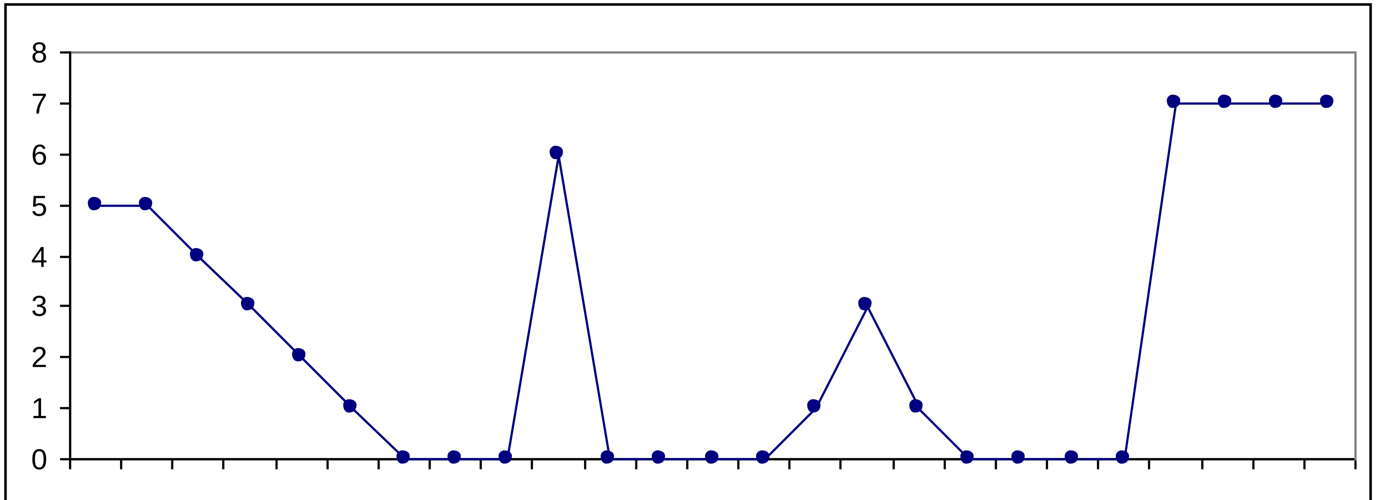
1^{st} Derivative

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

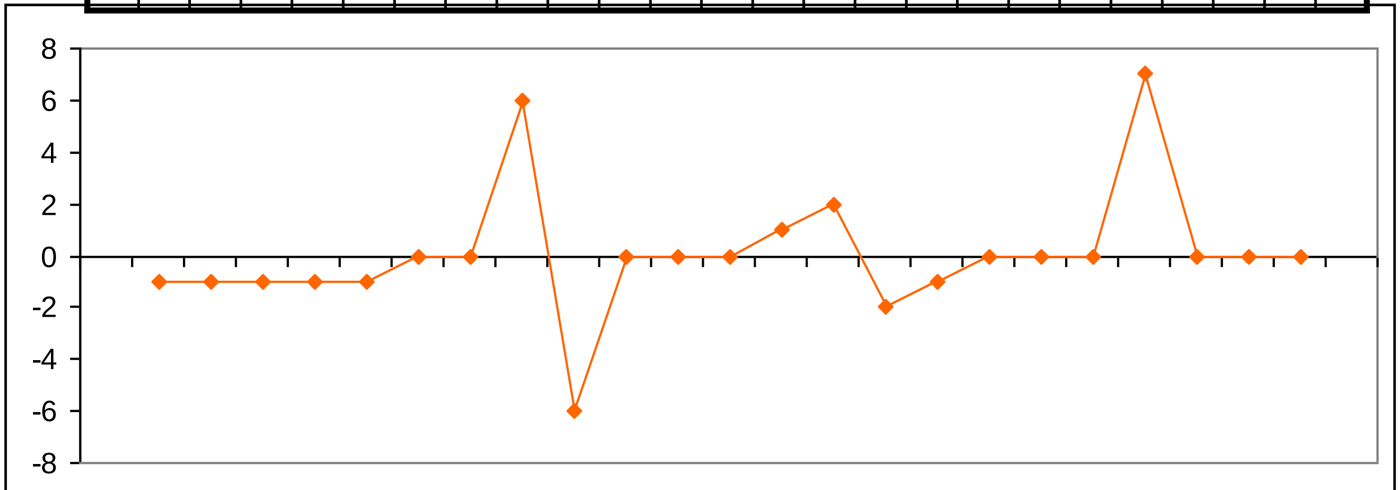
$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

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

1^{st} 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	



2^{nd} Derivative

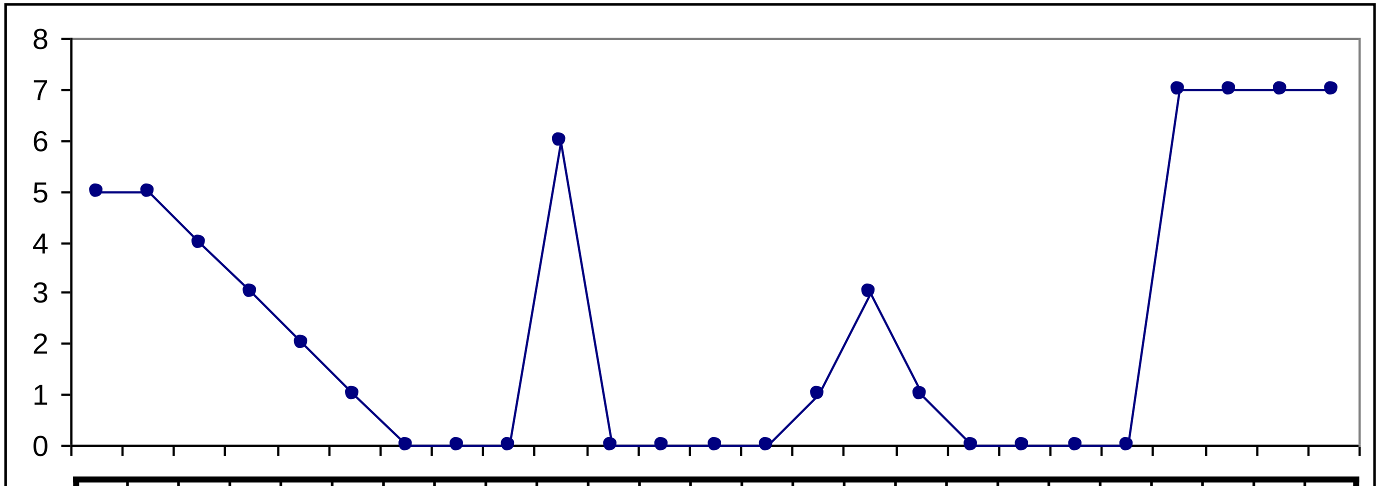
The formula for the 2^{nd} derivative of a function is as follows:

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

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

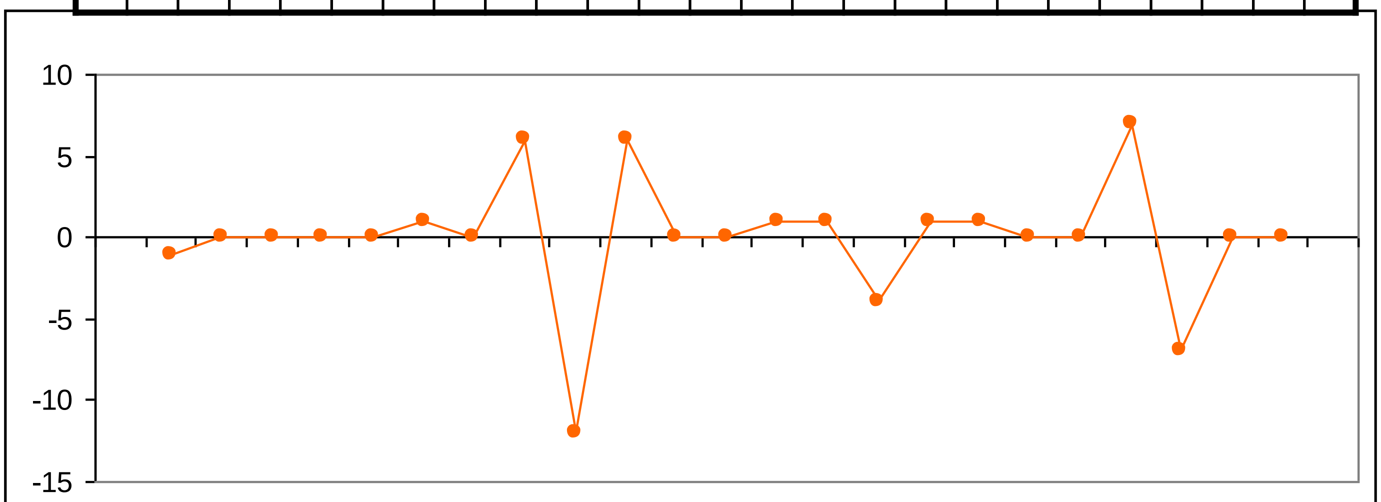
The derivative is centered at the middle of the filter.

2^{nd} Derivative (cont…)

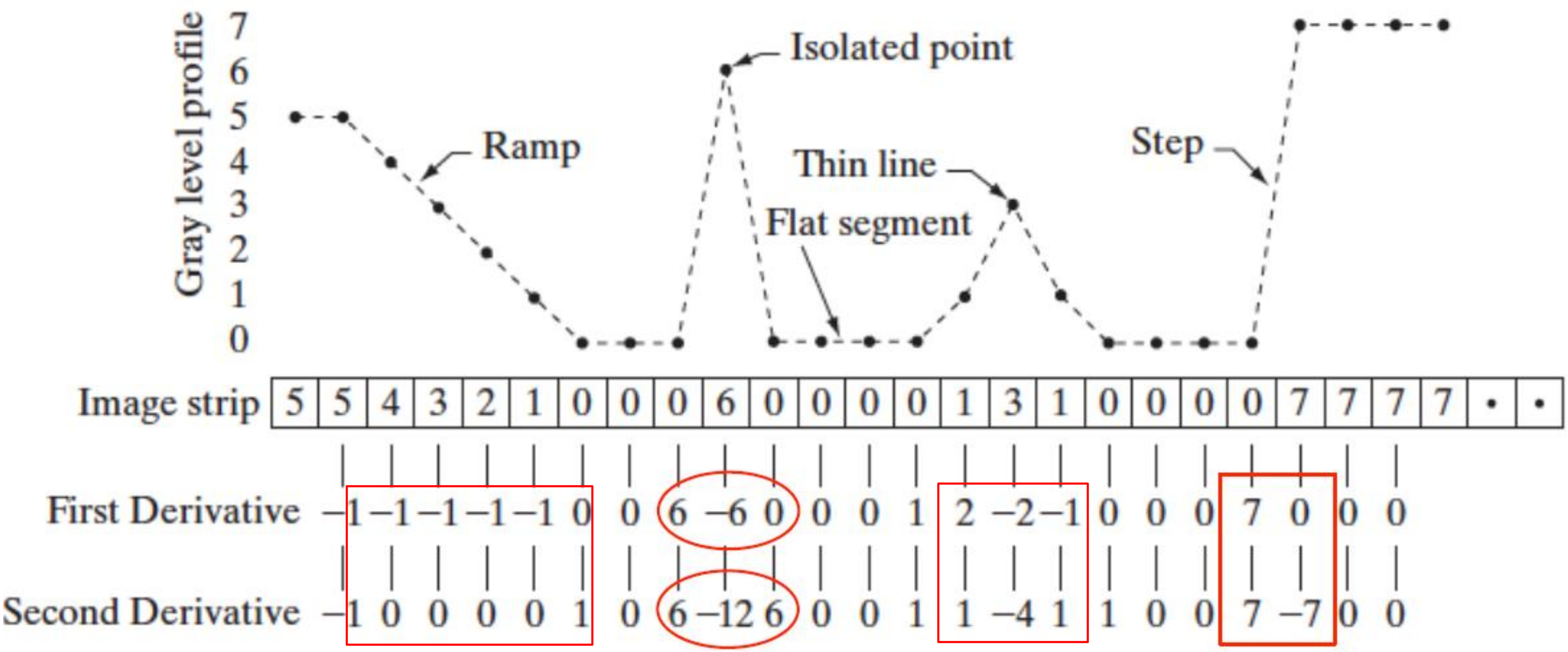


5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7

	-1	0	0	0	0	1	0	6	-12	6	0	0	1	1	-4	1	1	0	0	7	-7	0	0	



Combined 1st and 2nd derivative



First vs Second order derivatives

1. **Ramp:** the first-order derivative is nonzero along the entire ramp, while the second-order derivative is nonzero only at the onset and end of the ramp.
   – Because edges in an image resemble this type of transition, we conclude that **first-order derivatives produce “ thick” edges and second-order derivatives, much finer ones**.
2. **Isolated noise point:** Here, the response at and around the point is much **stronger for the second- than for the first-order derivative**.
   – A second-order derivative is much more aggressive than a first-order derivative in enhancing sharp changes.

First vs Second order derivatives

3. **Grey-level step:** the response of the two derivatives is the same at the gray-level step
   - in most cases when the transition into a step is not from zero, the second derivative will be weaker
   - the second derivative has a transition from positive back to negative. In an image, this shows as a thin double line.

Summary

1. First-order derivatives generally produce thicker edges in an image.
2. Second-order derivatives have a stronger response to fine details, such as thin lines and isolated points.
3. First order derivatives generally have a stronger response to a gray-level step.
4. Second-order derivatives produce a double response at step changes in gray level.

- We also note of second-order derivatives that, for similar changes in gray-level values in an image, their response is stronger to a line than to a step, and to a point than to a line.
- First and second order derivatives can be used in combination

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 transformation

- Premise: we are interested in ***isotropic* filters**, whose response is independent of the direction of the discontinuities in the image to which the filter is applied.
- In other words, **isotropic filters are *rotation invariant***, in the sense that rotating the image and then applying the filter gives the same result as applying the filter to the image first and then rotating the result.

The Laplacian transformation

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:

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

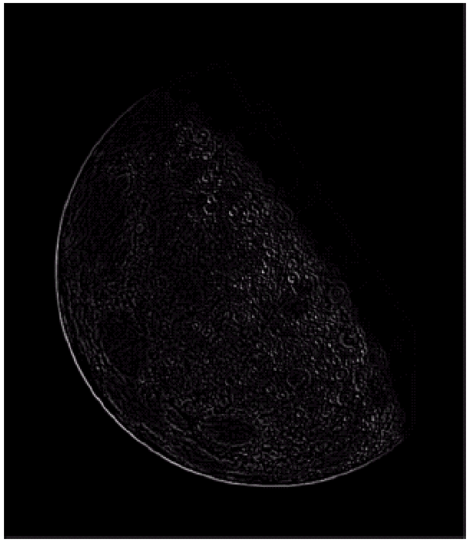
We can easily **build a filter** based on this

0	1	0
1	-4	1
0	1	0

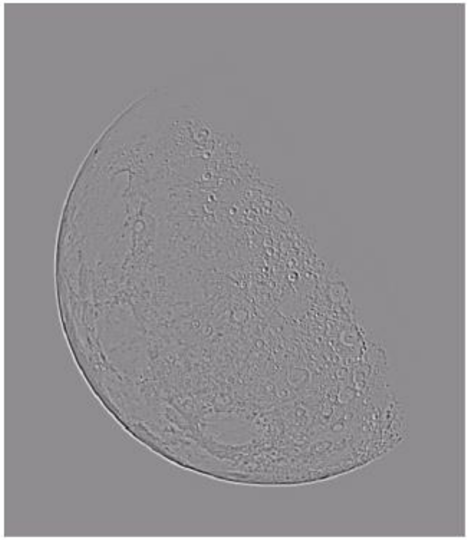
The Laplacian (cont…)

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

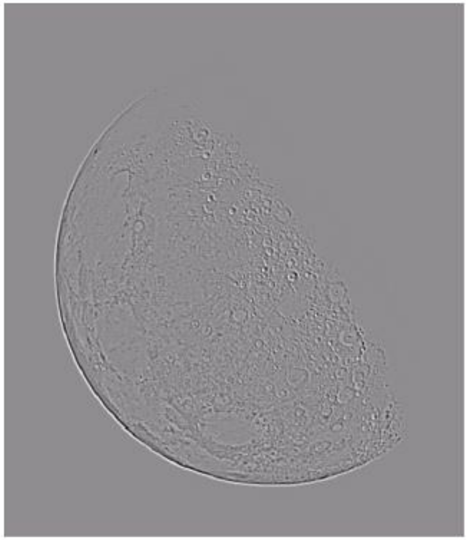
But That Is Not Very Enhanced!

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

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



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

Laplacian Image Enhancement

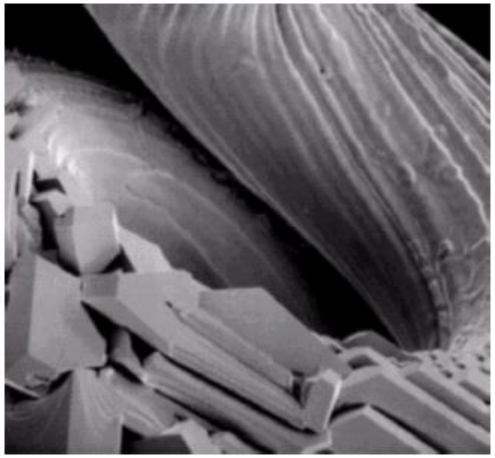
Simplified Image Enhancement

The entire enhancement can be combined into a single filtering operation

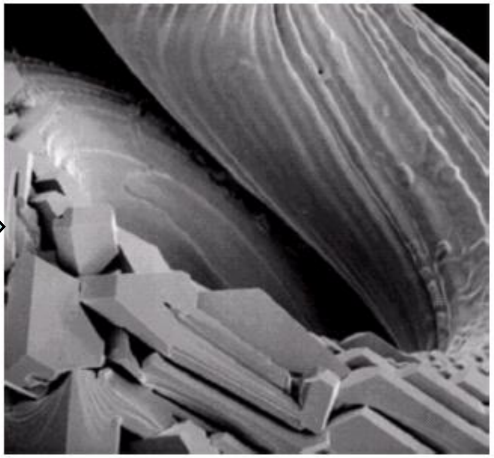
$$\begin{aligned}
g(x, y) &= f(x, y) - \nabla^2 f \\
&= f(x, y) - [f(x+1, y) + f(x-1, y) \\
&\qquad + f(x, y+1) + f(x, y-1) \\
&\qquad - 4f(x, y)] \\
&= 5f(x, y) - f(x+1, y) - f(x-1, y) \\
&\qquad - f(x, y+1) - f(x, y-1)
\end{aligned}$$

Simplified Image Enhancement (cont…)

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

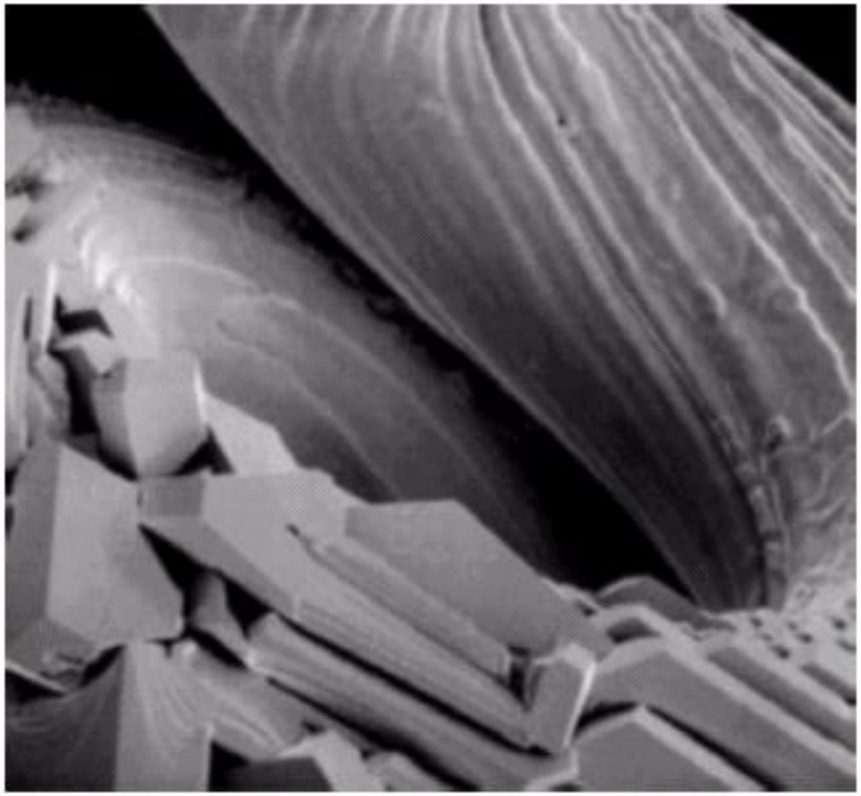


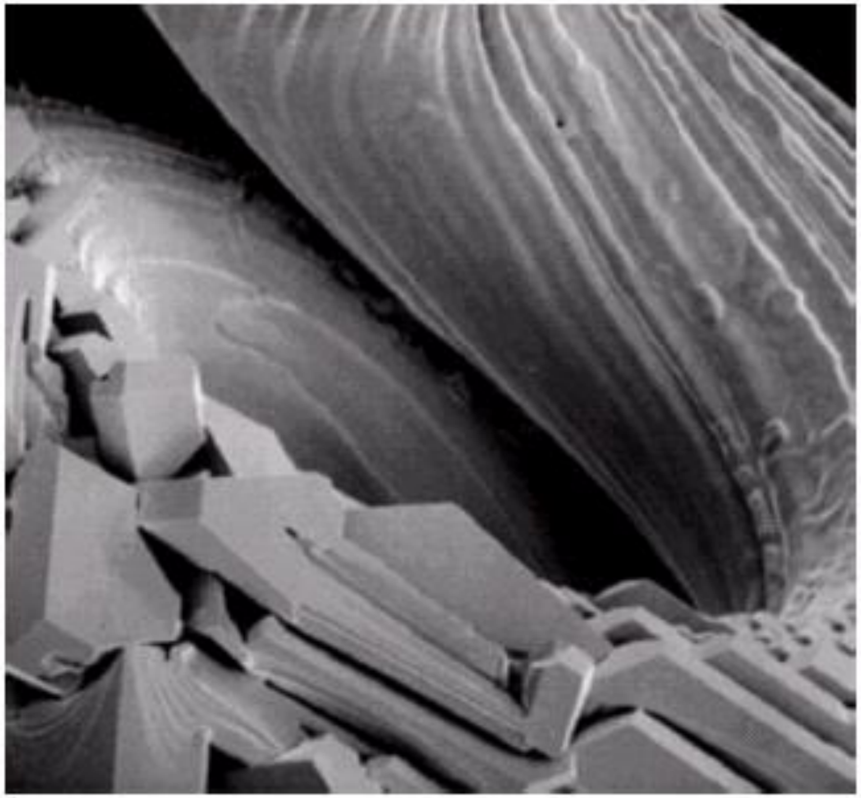
0	-1	0
-1	5	-1
0	-1	0



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

Simplified Image Enhancement (cont…)





Variants On The Simple Laplacian

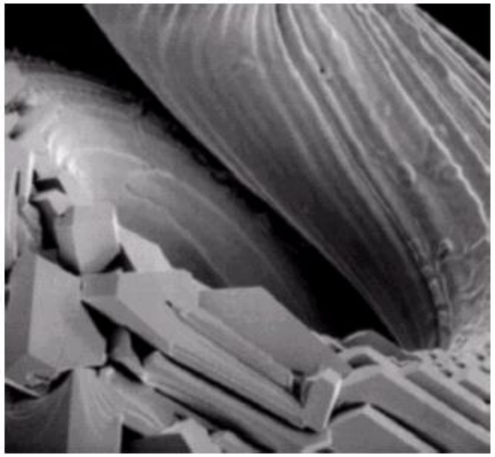
There are lots of slightly different versions of the Laplacian that can be used:

0	1	0
1	-4	1
0	1	0

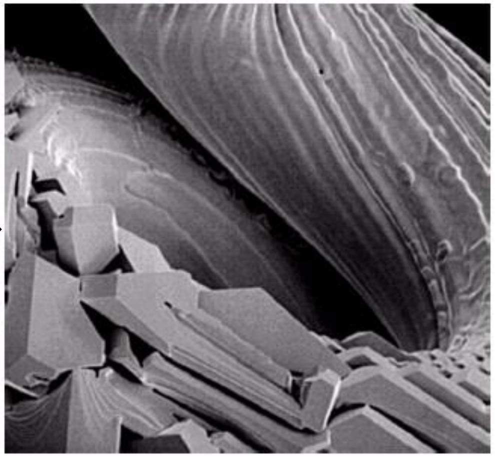
Simple Laplacian

1	1	1
1	-8	1
1	1	1

Variant of Laplacian



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



Variants On The Simple Laplacian

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) & \text{if the center coefficient of the} \\ & \text{Laplacian mask is negative} \\ f(x, y) + \nabla^2 f(x, y) & \text{if the center coefficient of the} \\ & \text{Laplacian mask is positive.} \end{cases}$$

1st Derivative Filtering

Implementing 1st 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:

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

1^{st} Derivative Filtering (cont…)

The magnitude of this vector is given by:

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

For practical reasons this can be simplified as:

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

Definition of 1st Derivatives

Recall that:

– any definition we use for a first derivative **(1)** must be **zero in flat segments** (areas of constant gray-level values); **(2)** must be **non-zero at the onset of a gray-level** step or ramp; and **(3)** must be **nonzero along ramps**.

– Similarly, any definition of a second derivative **(1)** must be **zero in flat areas**; **(2)** must be **non zero at the onset and end of a gray-level step** or ramp; and (3) must be **zero along ramps** of constant slope.

1^{st} Derivative Filtering (cont…)

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

i.e., Normal->Roberts->Prewitts->**Sobel**

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

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

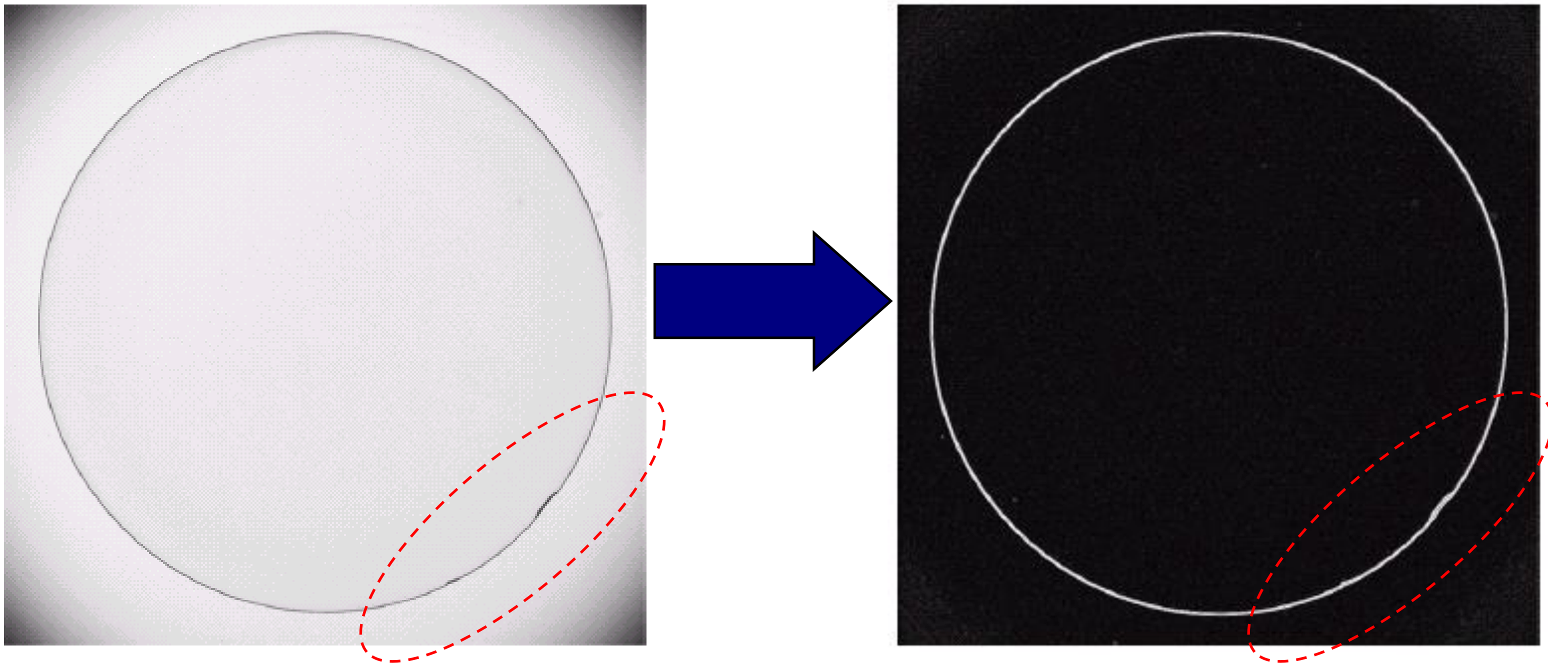
Based on the previous equations we can derive the *Sobel Operators*

-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

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

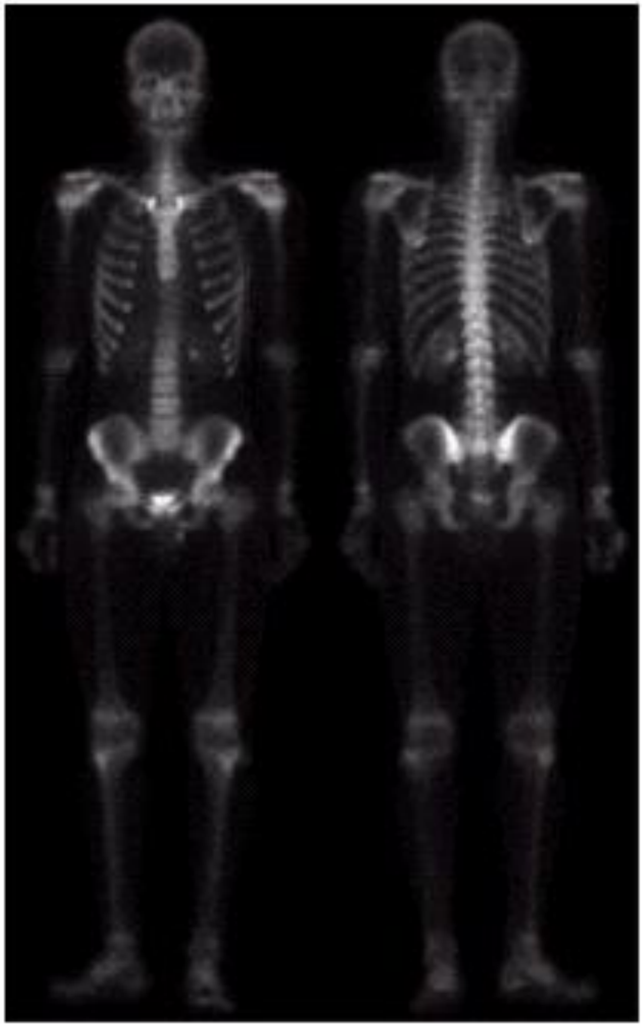
Comparing the 1^{st} and 2^{nd} derivatives we can conclude the following:

- 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

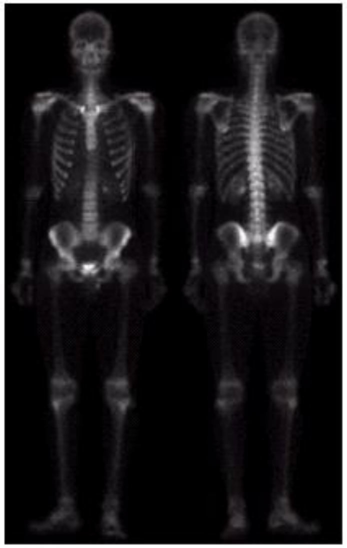
Combining Spatial Enhancement Methods

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

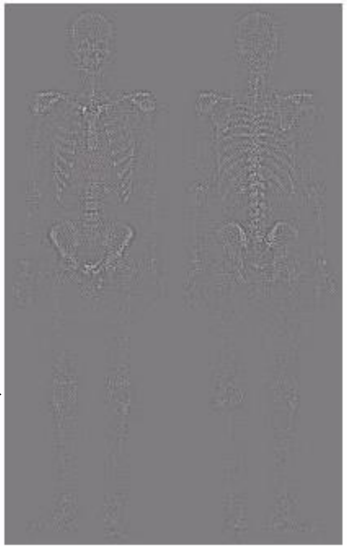
- Successful image enhancement is typically not achieved using a single operation
- Rather we combine a range of techniques in order to achieve a final result
- This example will focus on enhancing the bone scan to the right



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…)

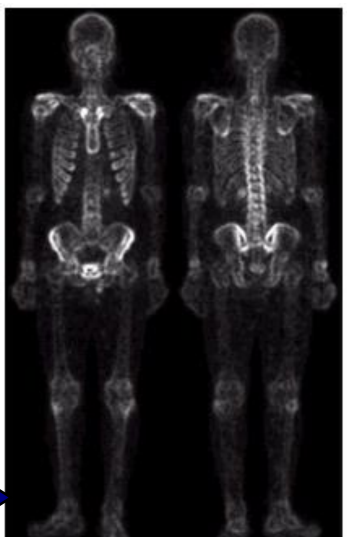
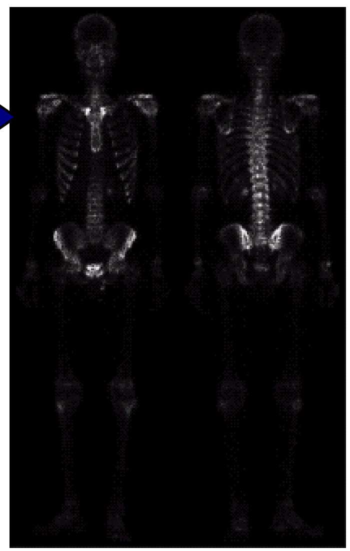


Image (d) smoothed with a 5*5 averaging filter

(e)

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



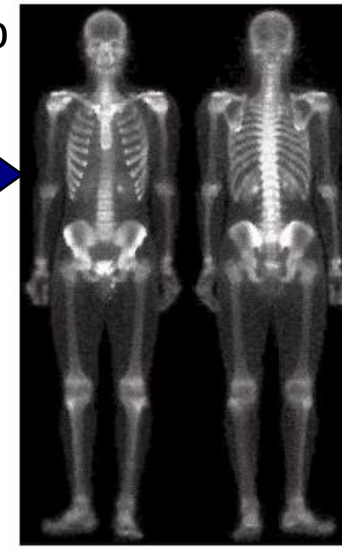
(f)

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

(g)

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

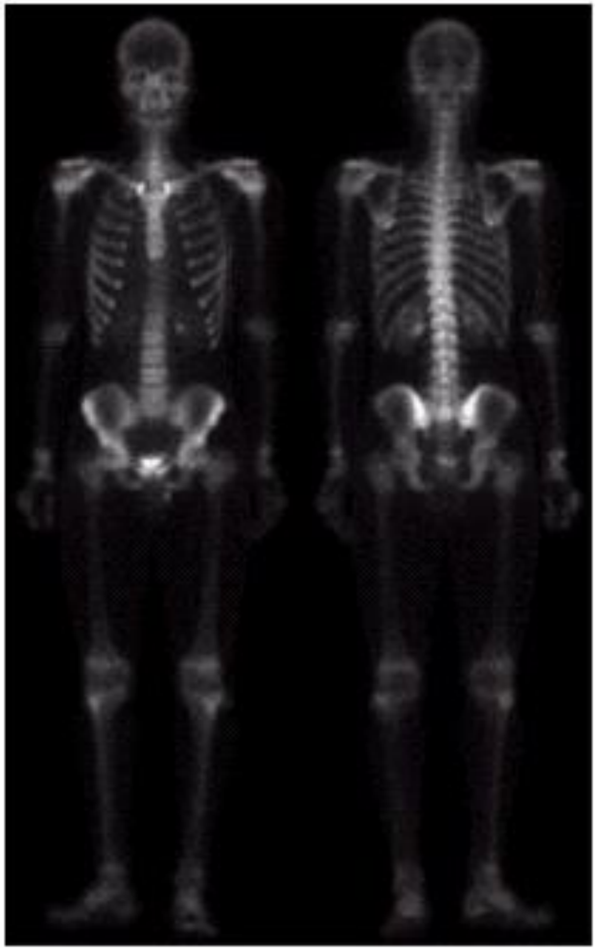
(h)



Combining Spatial Enhancement Methods (cont…)

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

Compare the original and final images



Summary

In this lecture we looked at:

- Sharpening filters
  - 1^{st} derivative filters
  - 2^{nd} derivative filters
- Combining filtering techniques