Sharpening Filters

Sharpening Spatial Filters

- The principal objective of sharpening is to
  - highlight fine detail in an image or
  - to enhance detail that has been blurred, either in error or as an natural effect of a particular method of image acquisition.

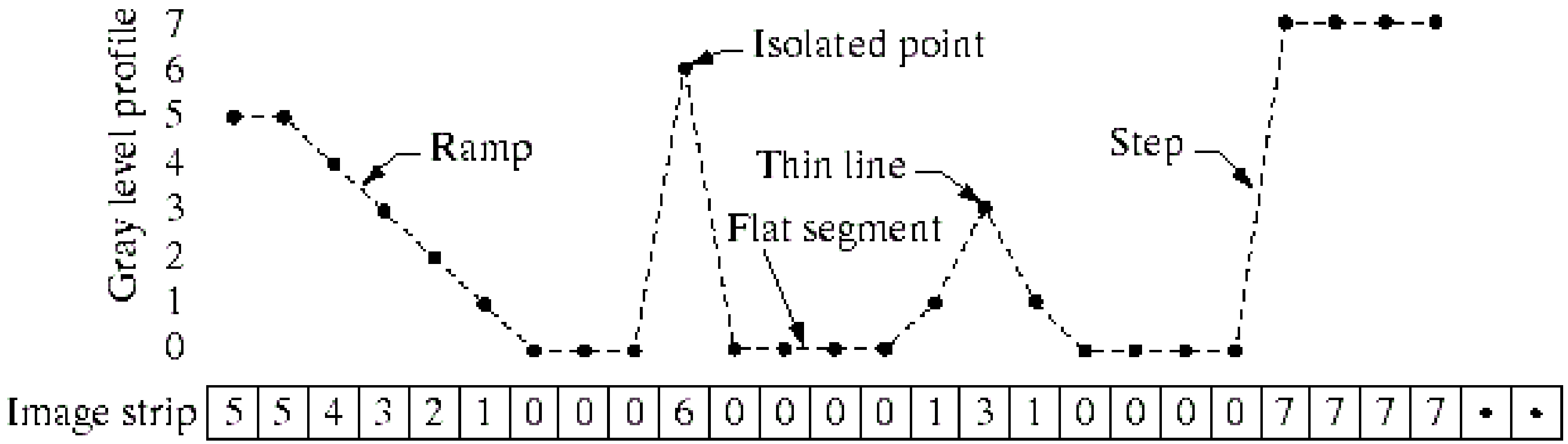
Introduction

- The image blurring is accomplished in the spatial domain by pixel averaging in a neighborhood.
- Since averaging is analogous to integration.
- Sharpening could be accomplished by spatial differentiation.

Foundation

- We are interested in the behavior of these derivatives in
  - areas of constant gray level(flat segments)
  - at the onset and end of discontinuities(step and ramp discontinuities)
  - along gray-level ramps.
- These types of discontinuities can be noise points, lines, and edges.

Sharpening Spatial Filters
An Example



Definition for a first derivative

- Must be zero in flat segments
- Must be nonzero at the onset of a gray-level step or ramp
- Must be nonzero along ramps.

Definition for a second derivative

- Must be zero in flat areas;
- Must be nonzero at the onset and end of a gray-level step or ramp;
- Must be zero along ramps of constant slope

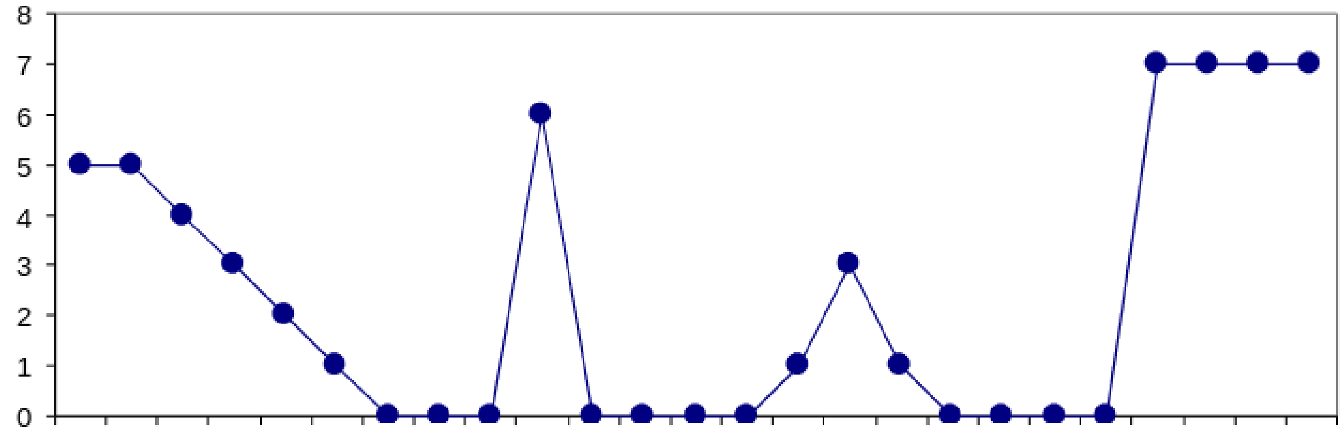
Definition of the 1st-order derivative

- A basic definition of the first-order derivative of a one-dimensional function f(x) is

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

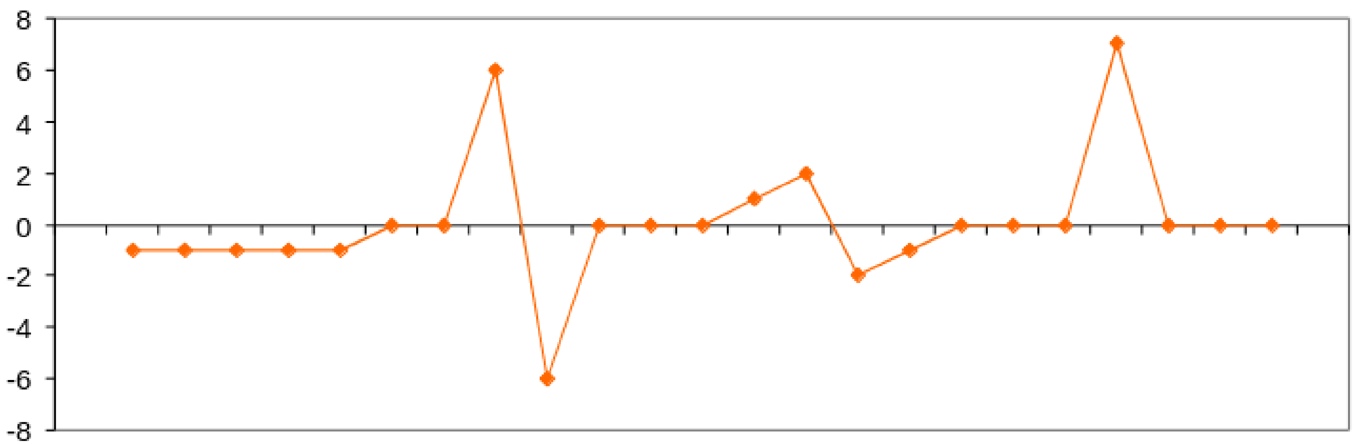
- It's just the difference between subsequent values and measures the rate of change of the function

1st 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	



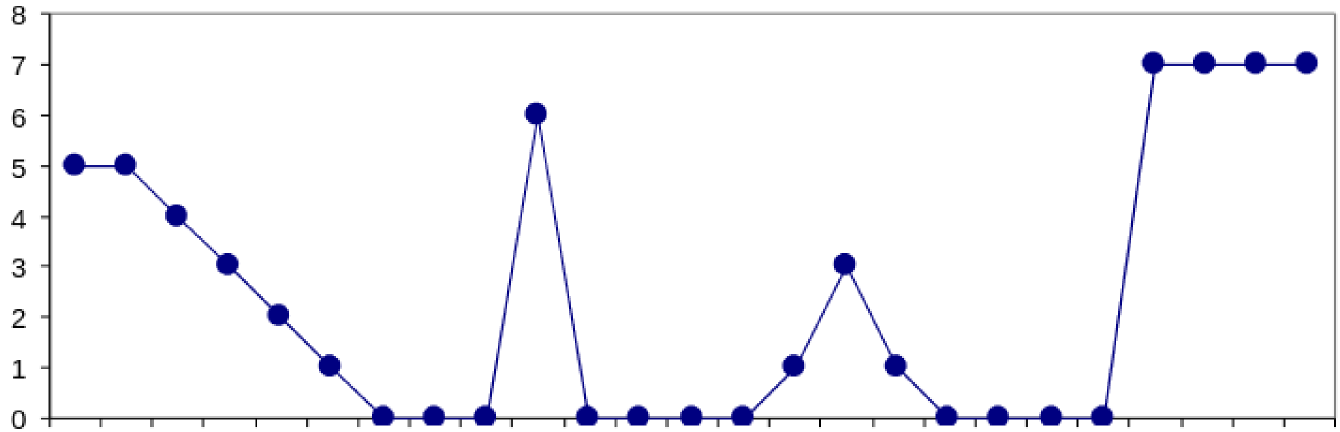
Definition of the 2nd-order derivative

- We define a second-order derivative as the difference

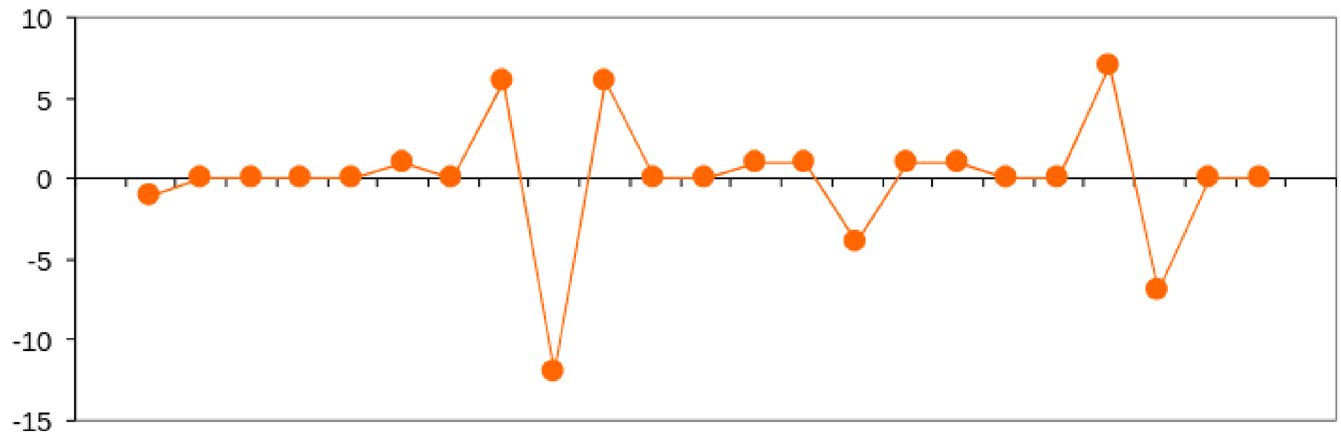
$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2\mathrm{f}(x).$$

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

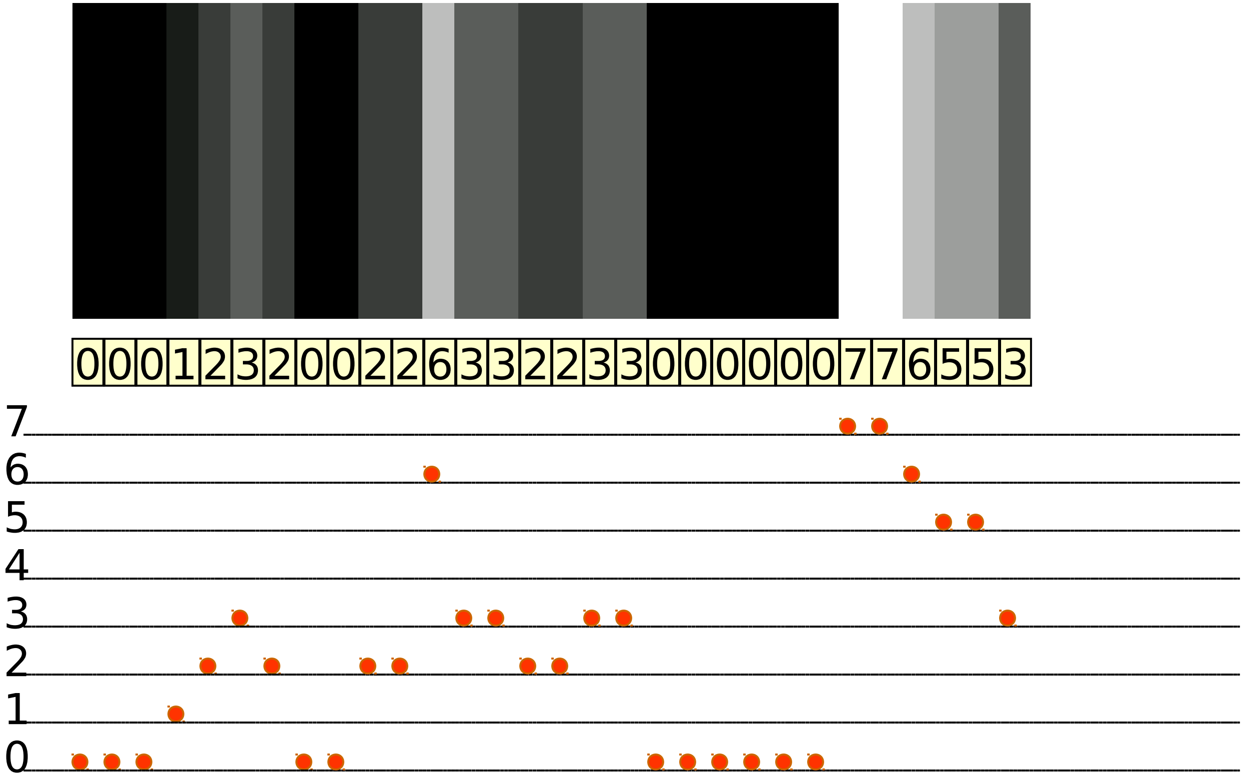
2^{nd} Derivative (cont…)



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	



Gray-level profile



Derivative of image profile

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

first **0 0 1 1 1 -1 -2 0 2 0 4 -3 0 -1 0 1 0 -3 0 0 0 0 0 -7 0 -1 -1 0 -2**

second **0 -1 0 0 -2 -1 2 2 -2 4 -7 3 -1 1 1 -1 -3 3 0 0 0 0 -7 7 -1 0 1 -2**

Analyze

- Edges in digital images are ramp like transitions in intensity .
- The 1st-order derivative is nonzero along the entire ramp result in thick edges
- The 2nd-order derivative is nonzero only at the onset and end of the ramp and produce a double edge
- The response at and around the point is much stronger for the 2nd order derivative and enhances fine detail

1st make thick edge and 2nd make thin edge

The Laplacian (2nd order derivative)

- Shown by Rosenfeld and Kak[1982] that the simplest isotropic derivative operator is the Laplacian is defined as

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

- Isotropic filters are rotation invariant
- Laplacian is a linear operator

Discrete form of derivative

In the x-direction and in the y-direction

f(x-1,y)	*f(x,y)*	*f(x+1,y)*

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

f(x,y-1)
f(x,y)
f(x,y+1)

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

2-Dimentional Laplacian

- The digital implementation of the 2-Dimensional Laplacian is obtained by summing 2 components

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

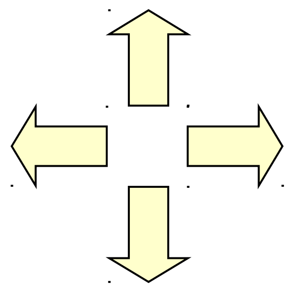
$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4\mathrm{f}(x, y)$$

	1	
1	-4	1
	1	

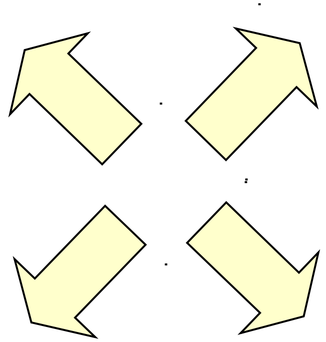
Laplacian

Variant of Laplacian

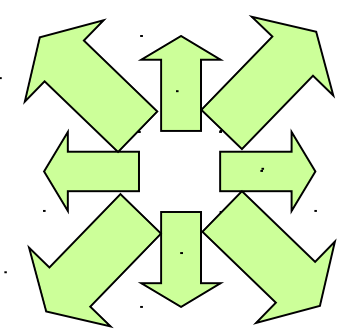
0	1	0
1	-4	1
0	1	0



1	0	1
0	-4	0
1	0	1



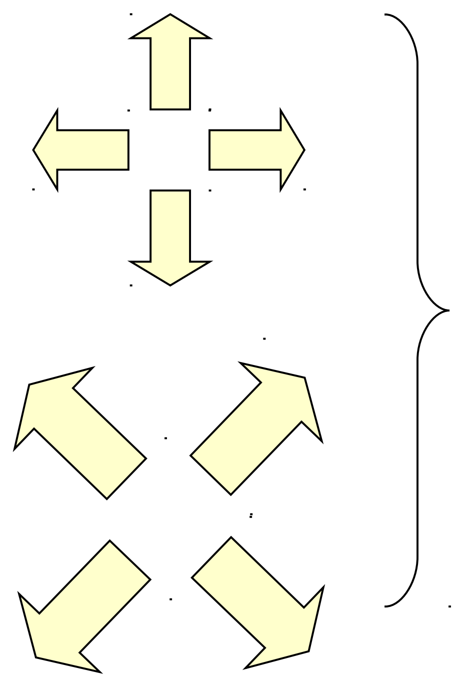
1	1	1
1	-8	1
1	1	1



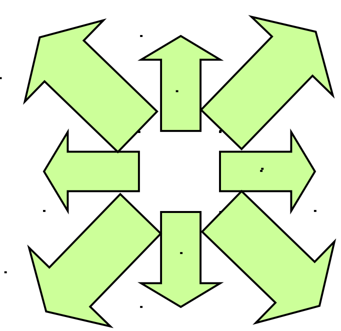
Laplacian

0	-1	0
-1	4	-1
0	-1	0

-1	0	-1
0	4	0
-1	0	-1



-1	-1	-1
-1	8	-1
-1	-1	-1



Implementation

$$g(x,y)=\begin{cases} f(x,y)-\nabla^2 f(x,y) & \textit{If the center coefficient is negative} \\ f(x,y)+\nabla^2 f(x,y) & \textit{If the center coefficient is positive} \end{cases}$$

Where f(x,y) is the original image

$\nabla^2 f(x,y)$ is Laplacian filtered image

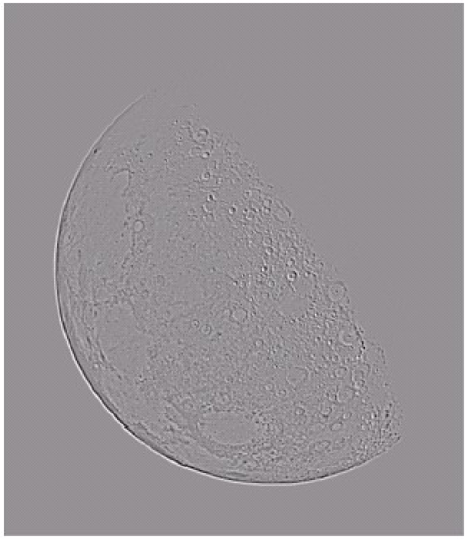
g(x,y) is the sharpen image

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
- Subtract the Laplacian result from the original image to generate our final sharpened enhanced image
- Subtracting the orginal image to the laplican restored overall intensity variations and increasing the contrast at the locations of intensity discontinuities

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

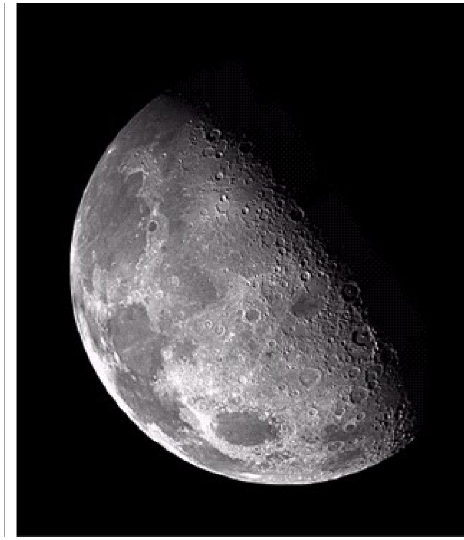
Laplacian Image Enhancement

Original Image

-

Laplacian Filtered Image

=



Sharpened Image

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

Unsharp Masking

- **Unsharp masking:** Process used by the printing and publishing industry to sharpen images consists of subtracting an unsharp version of an image from the orginal image
- Process consists of following steps:
  - Blur the orginal image
  - Subtract the blurred image from the orginal (mask)
  - Add mask to the orginal

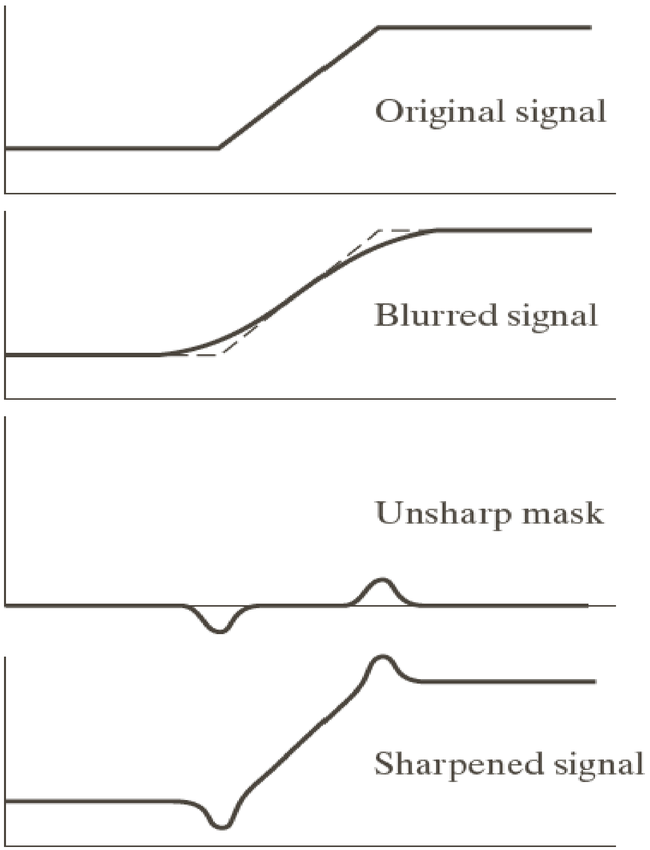
Unsharp Masking

- Let f'(x,y) denote the blurred image, unsharp masking is expressed in
- We add a weighted portion of the mask to the original image

$$g_{mask}(x,y) = f(x,y) - f'(x,y)$$
$$g(x,y) = f(x,y) + k*g_{mask}(x,y)$$

Where weight k k>=0, where k=1 -> unsharp masking
k>1-> highboost filtering
K<1-> deemphasizes the contribution of unsharp mask

Unsharp Masking



The points at which a change of slope in intensity occurs in the signal are now emphazied

Using First-order Derivatives for image Sharpening-gradient

- First derviatives in image processing implemented using magnitude of gradient
- Let f(x,y) the gradient of f at coordinates (x,y) is defined 2d column vector

$$\nabla f = grad(f) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix},$$

This vector points to the direction of greatest rate of change of f at location (x,y)

Using First-order Derivatives for image Sharpening-gradient

- Magnitude of the vector ∇f denoted by M(x,y) where

$$magnitude(grad(f)) = \sqrt{\frac{\partial f}{\partial x}^2 + \frac{\partial f}{\partial y}^2}$$

$$direction(grad(f)) = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

The square and square root operations by absolute values $M(x,y)=|g_x|+|g_y|$----*

Using First-order Derivatives for image Sharpening-gradient

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

- Z_5 denotes f(x,y) at an arbirary location (x,y).
- z_1 denotes f(x-1,y-1)
- Roberts use cross differences

$g_x=(z_9-z_5)$ and $g_y=(z_8-z_6)$-----**

- The gradiant of the image is defined as

$M(x,y)=[(z_9-z_5)^2+(z_8-z_6)^2]^{1/2}$

From * and ** we get

- $M(x,y)=|z_9-z_5|+|z_8-z_6|$ (Roberts cross-gradient operators)

$g_x=(z_7+2z_8+z_9)-(z_1+2z_2+z_3)$

$g_y=(z_3+2z_6+z_9)-(z_1+2z_4+z_7)$ --> sobel operators

The Laplacian - Masks

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

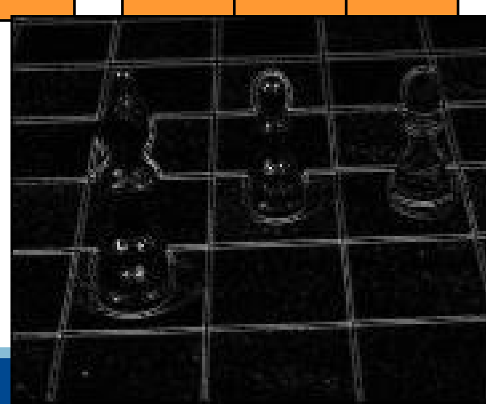
To recover the image:

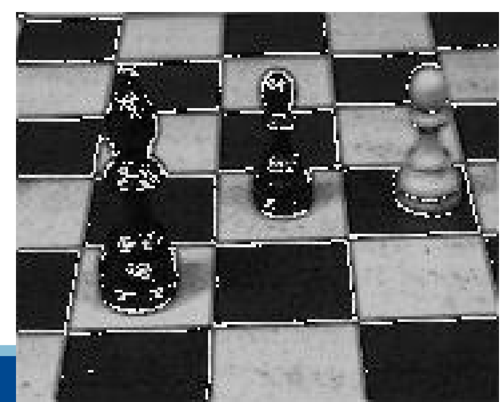
$$g(x,y) = f(x,y) + \nabla^2 f(x,y)$$

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	1
-1	-1	-1

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





Roberts
Prewitt
Sobel

Spatial filters

Purpose:

Blur or noise reduction

Lowpass/Smoothing spatial filtering

Sum of the mask coefficients is 1

Visual effect: reduced noise but blurred edge as well

Smoothing linear filters

Averaging filter

Weighted average (e.g. Gaussian)

Smoothing nonlinear filters

Order statistics filters (e.g. median filter)

Purpose

Highlight fine detail or enhance detail that has been blurred

Highpass/Sharpening spatial filter

Sum of the mask coefficients is 0

Visual effect: enhanced edges on a dark background

High-boost filtering and unsharp masking

Derivative filters

1st

2nd