ECE 468 / CS 519:
Digital Image Processing

Spatial Sharpening

Prof. Sinisa Todorovic

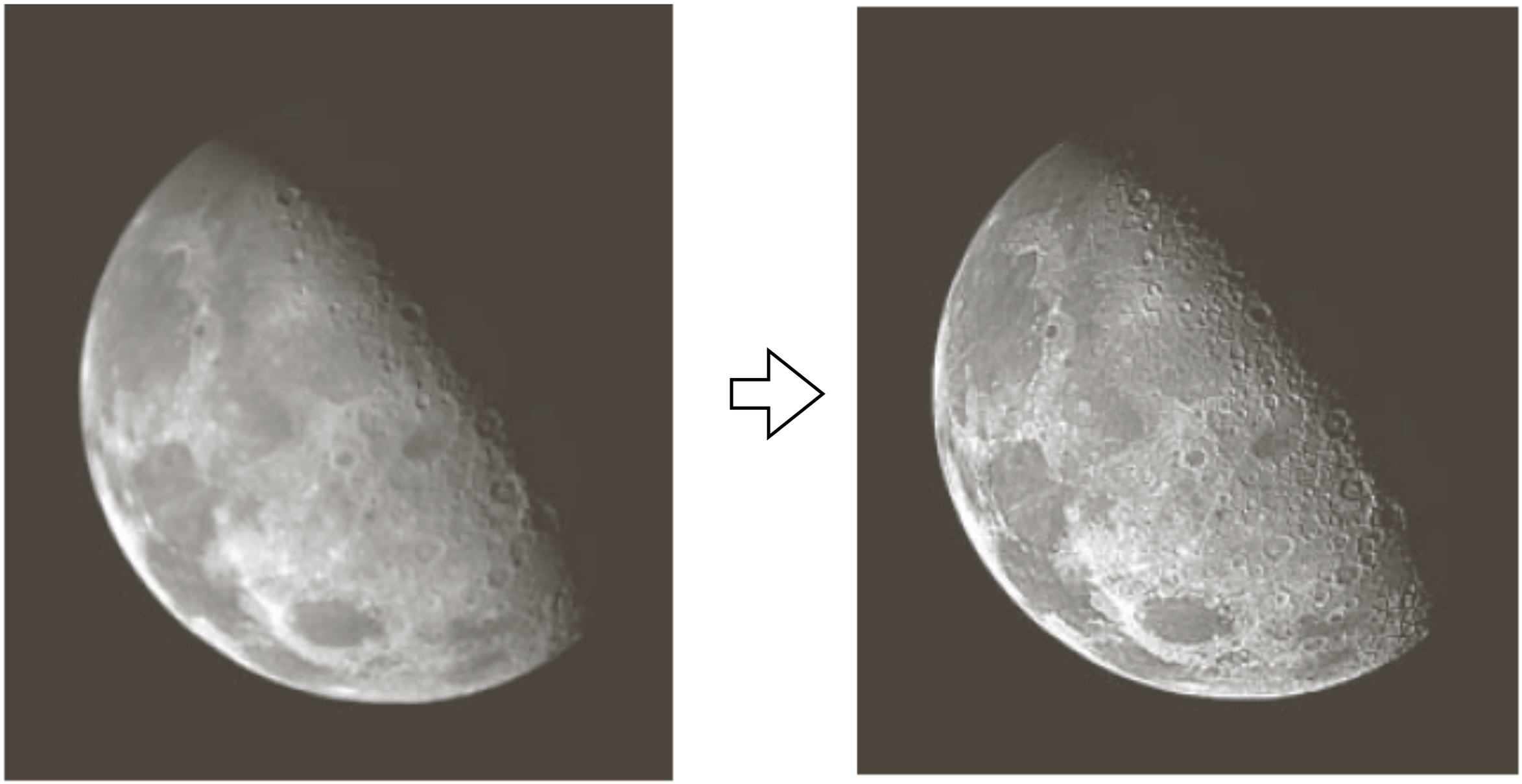
sinisa@eecs.oregonstate.edu

OSU Oregon State University

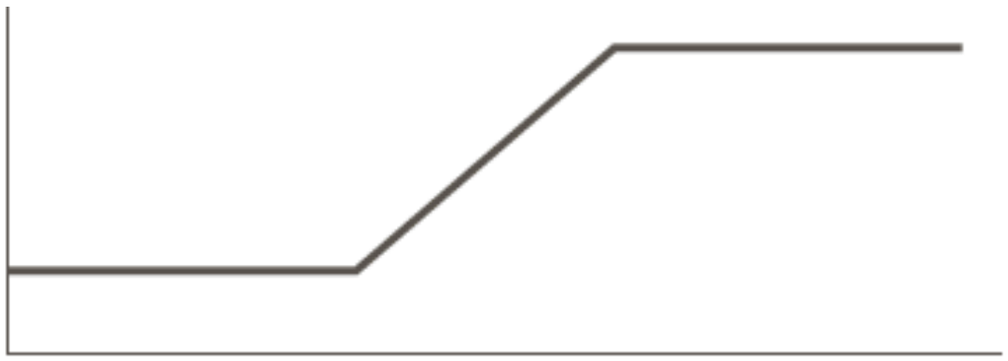
Outline

- MATLAB
- Sharpening spatial filters (Textbook: 3.5)

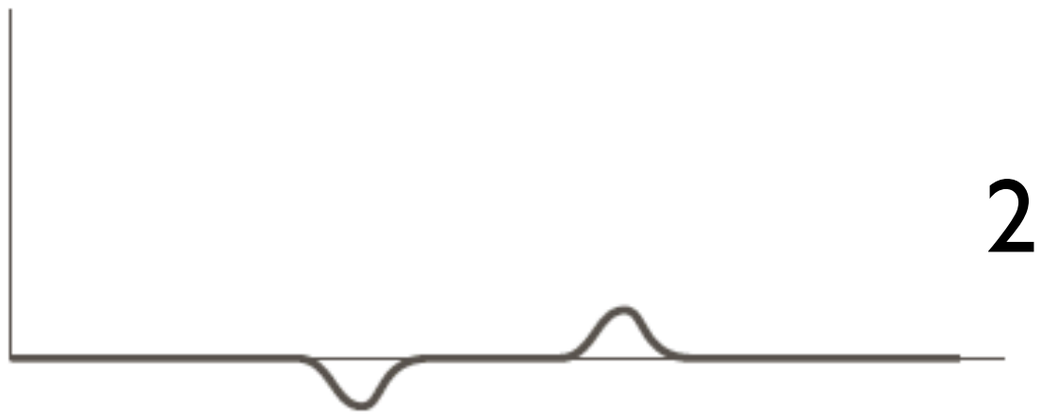
Image Sharpening



Second Derivative

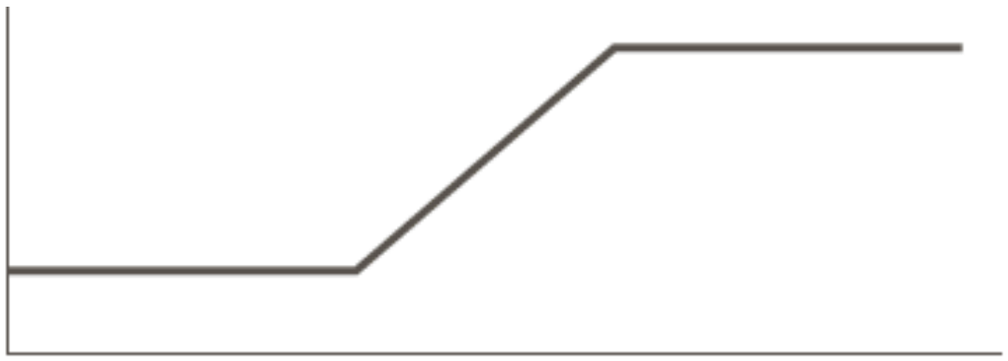


Original signal



2nd derivative

Second Derivative



Original signal

Original signal - 2nd derivative

Image Derivatives

$$\frac{\partial f(x,y)}{\partial x} \approx f(x+1,y) - f(x,y)$$

$$\frac{\partial f(x,y)}{\partial y} \approx f(x,y+1) - f(x,y)$$

Image Derivatives

Image Derivatives

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

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

Laplacian

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

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

0	1	0
1	−4	1
0	1	0

Sharpening Spatial Filters

Variants of the Laplacian filter

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Image Sharpening

$$g(x,y) = f(x,y) \pm c \cdot \nabla^2 f(x,y)$$

- Blur the original image
- **Subtract** the Laplacian

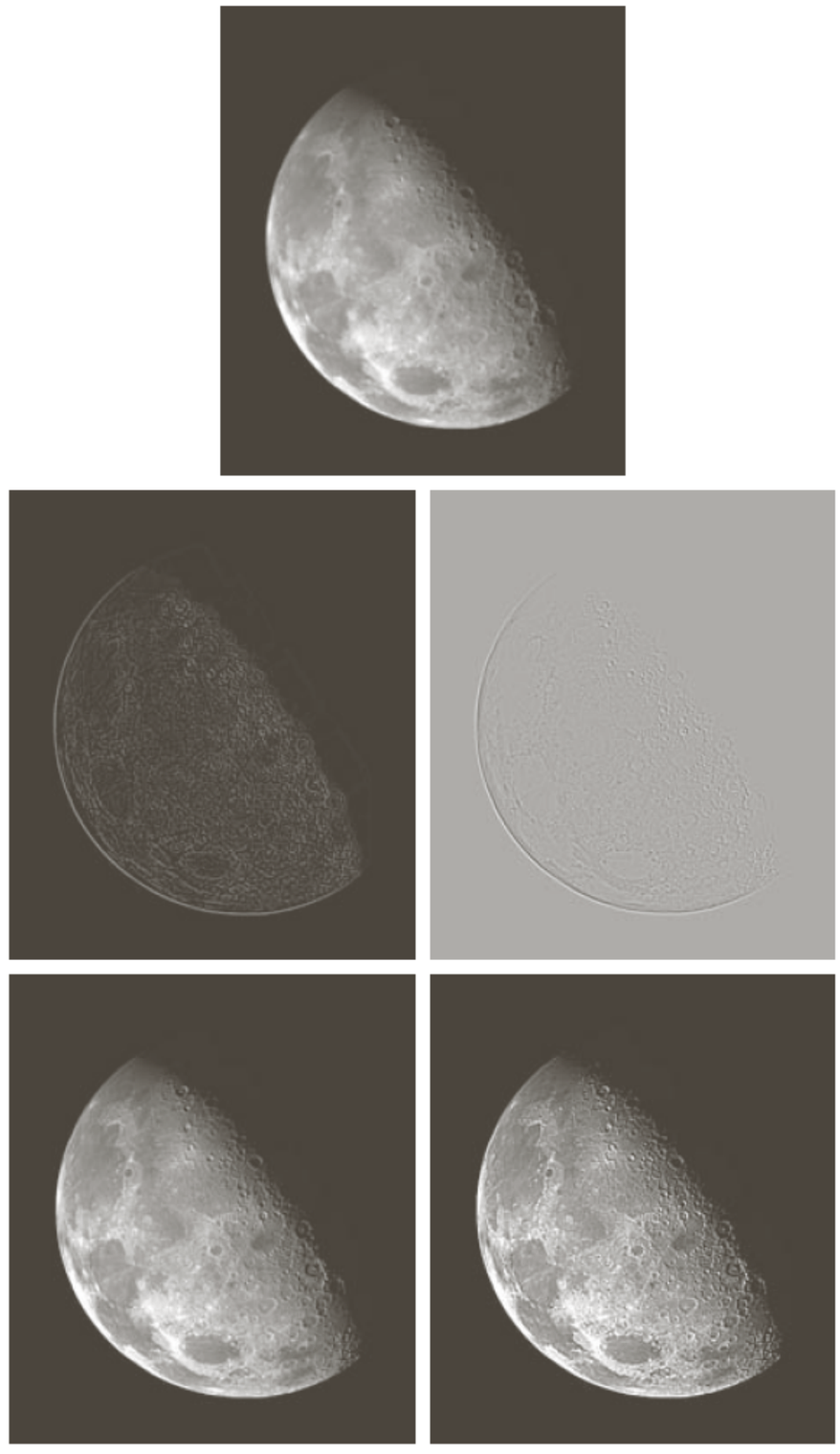


Image Sharpening

$$g(x,y) = f(x,y) \pm c \cdot \nabla^2 f(x,y)$$

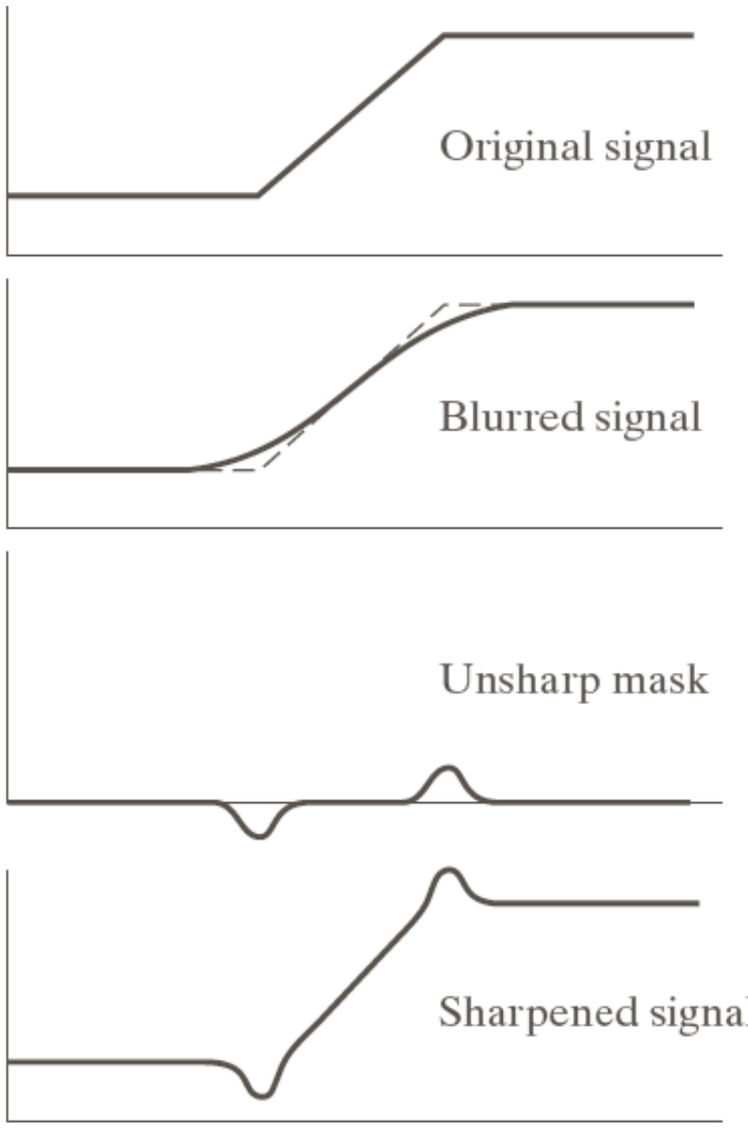
use $c = -1$ for

0	1	0	1	1	1
1	−4	1	1	−8	1
0	1	0	1	1	1

use $c = +1$ for

0	−1	0	−1	−1	−1
−1	4	−1	−1	8	−1
0	−1	0	−1	−1	−1

Unsharp Masking



- Blur the original image
- Subtract the blurred image from the original
- Add the resulting mask to the original

$$g(x,y) = f(x,y) + k(f(x,y) - \overline{f(x,y)})$$

$k = 1$ unsharp masking

$k > 1$ highboost filtering

Unsharp Masking

input

blurred

input - blurred

input + (input - blurred)

input + 4.5 (input - blurred)

MATLAB

MATLAB

- `g = imfilter(f, w, 'corr', 'replicate')`

MATLAB

- `g = imfilter(f, w, 'corr', 'replicate')`

- `f = padarray(f, [delta_row delta_col], 'replicate','post')`

MATLAB

- `g = imfilter(f, w, 'corr', 'replicate')`

- `f = padarray(f, [delta_row delta_col], 'replicate','post')`

- `w = fspecial('type', parameters)`

MATLAB

- `g = imfilter(f, w, 'corr', 'replicate')`

- `f = padarray(f, [delta_row delta_col], 'replicate','post')`

- `w = fspecial('type', parameters)`

  - `fspecial('gaussian', [r c], sigma)`