

The background of the slide features a large, light blue watermark of the University of Delaware seal. The seal is circular and contains an open book with Latin text on its pages: 'GRAMM', 'METAPH', 'PHIOL', 'LOGICA', 'RHETOR', 'MATHEM', 'ETHICA', and 'PHYSICA'. Below the book is a banner with the motto 'SOL MENESTIS'. The outer ring of the seal contains the text 'UNIVERSITY OF DELAWARE' and the year '1743'.

ELEG404/604: Imaging & Deep Learning

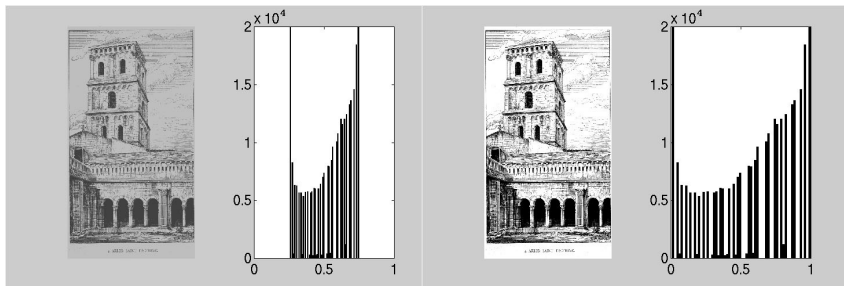
Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

Chapter II

Image Enhancement in the Spatial Domain

- ▶ Algorithms for improving the visual appearance of images
 - ▶ Gamma correction
 - ▶ Contrast improvements
 - ▶ Histogram equalization
 - ▶ Noise reduction
 - ▶ Image sharpening
- ▶ Optimality is in the eye of the observer: Ad hoc.



What is a histogram

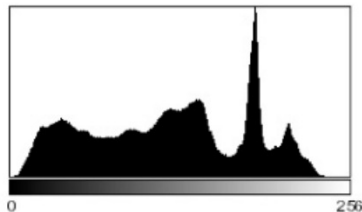
Each individual histogram entry is defined as

$$h(i) = \text{Number of pixels in } l \text{ with the intensity value } i$$

for all $0 \leq i < K$, that is

$$h(i) = \text{card} \{ (u, v) \mid I(u, v) = i \}$$

$h(0)$ is the number of pixels with the value 0, $h(1)$ the number of pixels with the value 1, and so forth.



Count: 1920000
Mean: 118.848
StdDev: 59.179
Min: 0
Max: 251
Mode: 184 (30513)

Histograms

Three very different images with identical histograms.

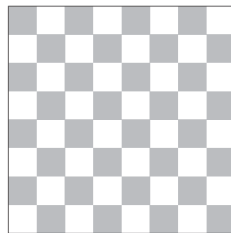
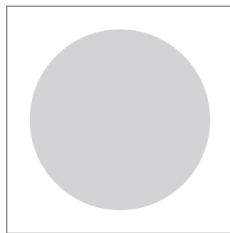
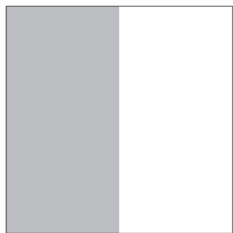
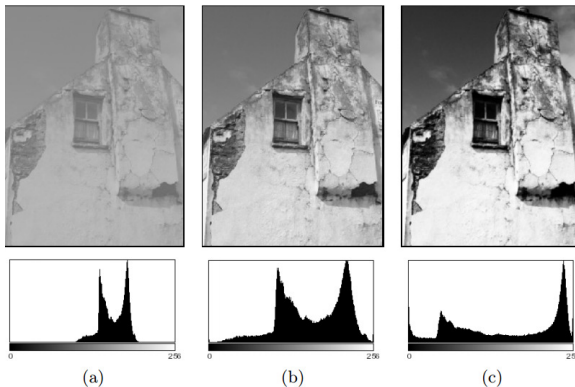


Image Acquisition- Contrast

- ▶ Range of intensity values effectively used.
- ▶ Difference between the image's maximum and minimum pixel values.
- ▶ A full-contrast image makes effective use of the entire range of available intensity values from $a = a_{min} \cdots a_{max} = 0 \cdots K - 1$ (black to white).



(a) Low contrast, (b) Normal contrast, (c) High contrast

Spatial Domain Processing

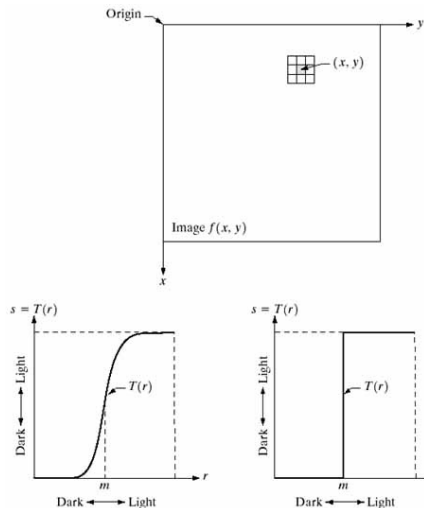
- ▶ Utilize neighborhood operations

$$g(x, y) = T[f(x, y)]$$

- ▶ Simple case: point operations

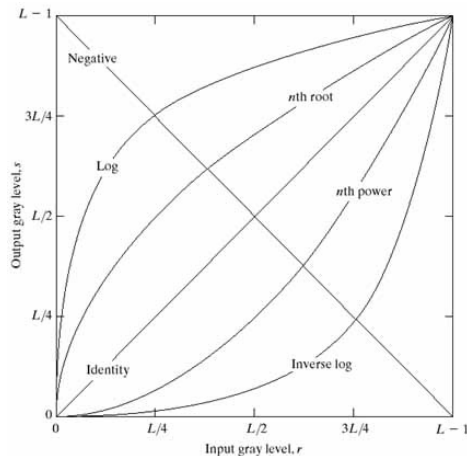
$$s = T(r)$$

- ▶ Contrast stretching
- ▶ Thresholding



Gray Level Transformation Curves

- ▶ More general transformations
 - ▶ Log
 - ▶ Inverse Log
 - ▶ n^{th} power
 - ▶ n^{th} root
- ▶ Used to map narrow dark (log/root) or bright (inverse log/power) range to a greater dynamic range

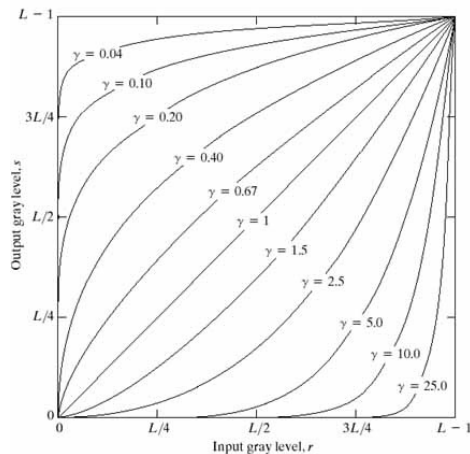


Power-Law (Gamma) Transformations

- ▶ Power-law transformation:

$$s = cr^\gamma$$

- ▶ Many devices require gamma correction
 - ▶ CRTs have power function intensity-to-voltage responses
 - ▶ Monitor specific
 - ▶ Applied in color planes



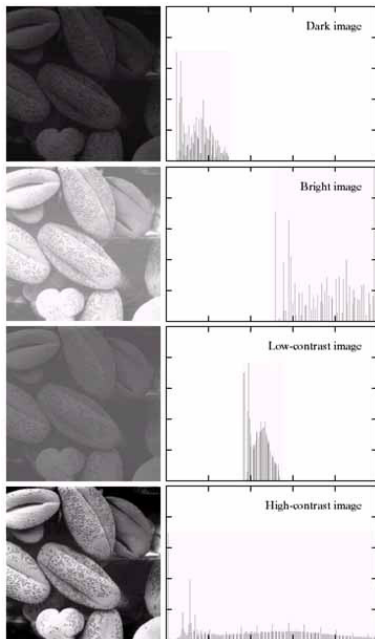
Gamma Correction Example - Aerial

 $\gamma=3$  $\gamma=4$  $\gamma=8$

Top Left: aerial image. Top right, Bottom left and bottom right: results of applying Gamma correction with $c = 1$ and $\gamma = 3, 4, 5$ respectively.

Histogram Processing

- ▶ Light, dark, and low contrast images have concentrated histograms
- ▶ Images with uniform histograms
 - ▶ Contain the full range of gray values
 - ▶ Have high contrast
 - ▶ Better general visual appearance



Histogram Equalization

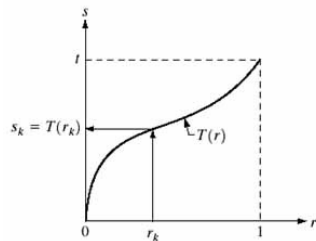
- ▶ We focus on the (normalized) scalar mapping

$$s = T(r) \quad 0 \leq r \leq 1$$

where the following are satisfied:

- ▶ $T(r)$ is single-valued and monotonically increasing in $[0,1]$
- ▶ $0 \leq T(r) \leq 1$ for $0 \leq r \leq 1$
- ▶ The single-valued condition allows the inverse transformation to be defined

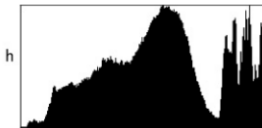
$$r = T^{-1}(s) \quad 0 \leq s \leq 1$$



Probability Density Function

- ▶ Let the PDF of r be the $p_r(r)$
- ▶ The CDF is $P_r(r) = \int_0^r p_r(w)dw$
 - ▶ Note CDFs are monotonically increasing and have range $[0,1]$
- ▶ Defined the RV $s = T(r)$
 - ▶ The PDF of a RV function is

$$p_s(s) = p_r(r) \left| \frac{dr}{ds} \right|$$



CDF Distribution

- ▶ Set $T(r) = P_r(r)$

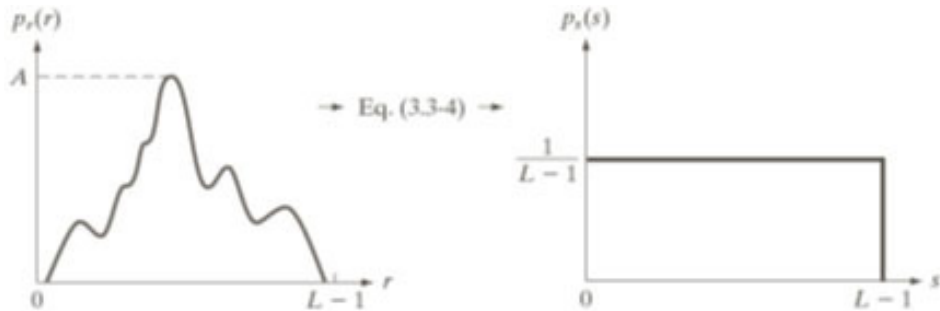
$$\begin{aligned}\frac{ds}{dr} &= \frac{dT(r)}{dr} \\ \frac{d[\int_0^r p_r(w)dw]}{dr} &= p_r(r)\end{aligned}$$

- ▶ Thus the PDF of s is

$$\begin{aligned}p_s(s) &= p_r(r) \left| \frac{dr}{ds} \right| \\ p_r(r) \left| \frac{1}{p_r(r)} \right| &= 1\end{aligned}$$

- ▶ The CDF is uniformly distributed.

Histogram Equalization



a b

FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels, r . The resulting intensities, s , have a uniform PDF, independently of the form of the PDF of the r 's.

Histogram Equalization

A 3-bit image ($L = 8$) of size 64×64 ($MN = 4096$) has the intensity distribution in table.

Intensity levels are integers in $[0, L - 1] = [0, 7]$.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

TABLE 3.1
Intensity distribution and histogram values for a 3-bit, 64×64 digital image.

Histogram Equalization

Values of the histogram equalization are obtained using:

$$s_0 = T(r_0) = 7 \sum_{j=0}^0 P_r(r_j) = 7p_r(r_0) = 1.33$$

Similarly,

$$s_1 = T(r_1) = 7 \sum_{j=0}^1 P_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08$$

and $s_2 = 4.55$, $s_3 = 5.67$, $s_4 = 6.23$, $s_5 = 6.65$, $s_6 = 6.86$, $s_7 = 7.00$

The s values still have fractions because they were generated by summing probability values, so we round them to the nearest integer:

$$s_0 = 1.33 \rightarrow 1$$

$$s_1 = 3.08 \rightarrow 3$$

$$s_2 = 4.55 \rightarrow 5$$

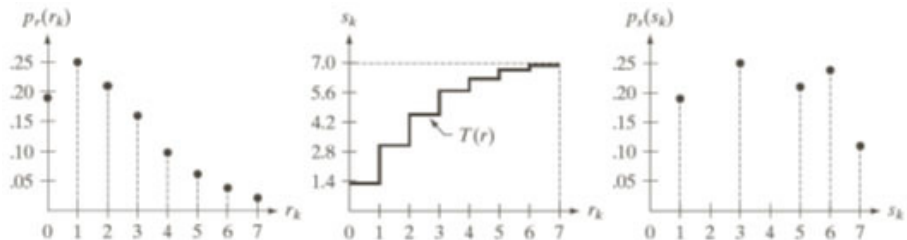
$$s_3 = 5.67 \rightarrow 6$$

$$s_4 = 6.23 \rightarrow 6$$

$$s_5 = 6.65 \rightarrow 7$$

$$s_6 = 6.86 \rightarrow 7$$

$$s_7 = 7.00 \rightarrow 7$$

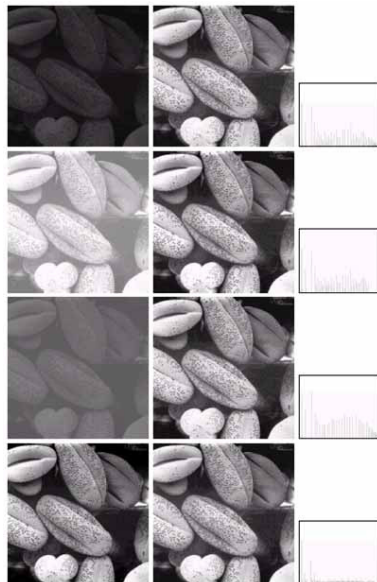


a b c

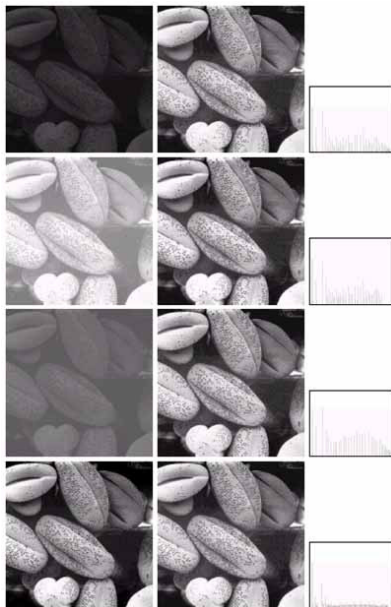
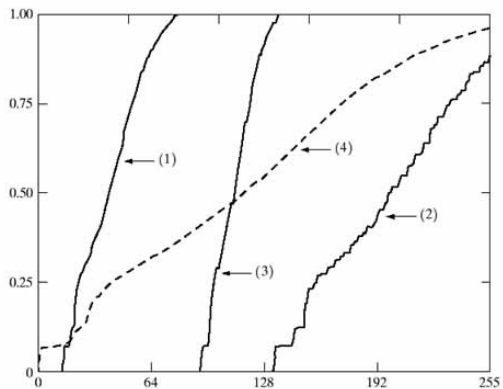
FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Histogram Equalization

- ▶ CDF mapping of gray values
 - ▶ Yields uniformed histogram
 - ▶ Simple, parameter-free
- ▶ Discrete case
 - ▶ Results not strictly uniform
 - ▶ Implementation issues



Histogram EQ Mappings



Histogram Equalization Result

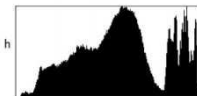
Image



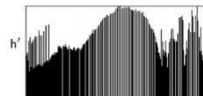
(a)

(b)

Histograms



(c)



(d)

Cumulative
Histograms



(e)



(f)

Original image and corresponding histograms. Notice that new peaks are created in the resulting histogram h (d) by merging original histogram cells.

Filters

What is a filter?

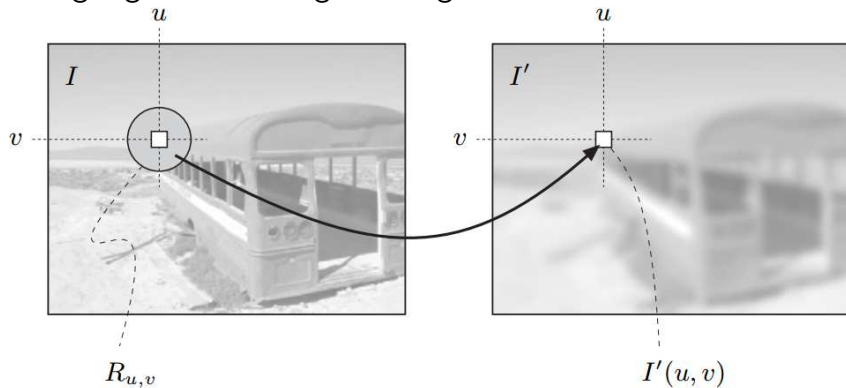
Main difference between filters and point operations is that filters generally use more than one pixel to compute each new pixel input.



No point operation can blur or sharpen an image.

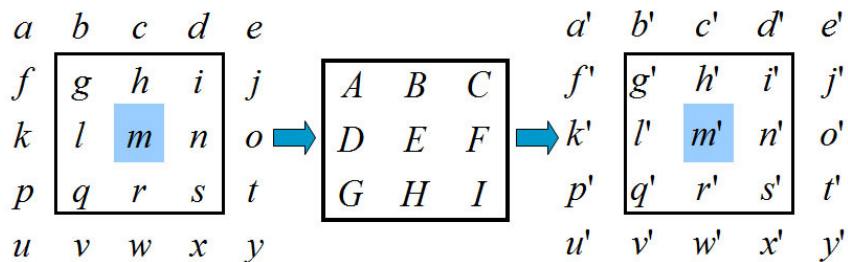
Filters

1 Each new pixel value $I'(x,y)$ is computed as a function of the pixels in a corresponding region in the original image I .



Filtering

To sharpen or filter an image that was taken out-of-focus and is blurred, each pixel is replaced with a linear combination of its neighbors.



$$m' = A \cdot g + B \cdot h + C \cdot i + D \cdot l + \dots$$
$$\dots + E \cdot m + F \cdot n + G \cdot q + H \cdot r + I \cdot s$$

Filtering

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>f</i>	<i>g</i>	<i>h</i>	<i>i</i>	<i>j</i>
<i>k</i>	<i>l</i>	<i>m</i>	<i>n</i>	<i>o</i>
<i>p</i>	<i>q</i>	<i>r</i>	<i>s</i>	<i>t</i>
<i>u</i>	<i>v</i>	<i>w</i>	<i>x</i>	<i>y</i>



<i>a'</i>	<i>b'</i>	<i>c'</i>	<i>d'</i>	<i>e'</i>
<i>f'</i>	<i>g'</i>	<i>h'</i>	<i>i'</i>	<i>j'</i>
<i>k'</i>	<i>l'</i>	<i>m'</i>	<i>n'</i>	<i>o'</i>
<i>p'</i>	<i>q'</i>	<i>r'</i>	<i>s'</i>	<i>t'</i>
<i>u'</i>	<i>v'</i>	<i>w'</i>	<i>x'</i>	<i>y'</i>

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>f</i>	<i>g</i>	<i>h</i>	<i>i</i>	<i>j</i>
<i>k</i>	<i>l</i>	<i>m</i>	<i>n</i>	<i>o</i>
<i>p</i>	<i>q</i>	<i>r</i>	<i>s</i>	<i>t</i>
<i>u</i>	<i>v</i>	<i>w</i>	<i>x</i>	<i>y</i>



<i>a'</i>	<i>b'</i>	<i>c'</i>	<i>d'</i>	<i>e'</i>
<i>f'</i>	<i>g'</i>	<i>h'</i>	<i>i'</i>	<i>j'</i>
<i>k'</i>	<i>l'</i>	<i>m'</i>	<i>n'</i>	<i>o'</i>
<i>p'</i>	<i>q'</i>	<i>r'</i>	<i>s'</i>	<i>t'</i>
<i>u'</i>	<i>v'</i>	<i>w'</i>	<i>x'</i>	<i>y'</i>

Simple Smoothing Masks

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

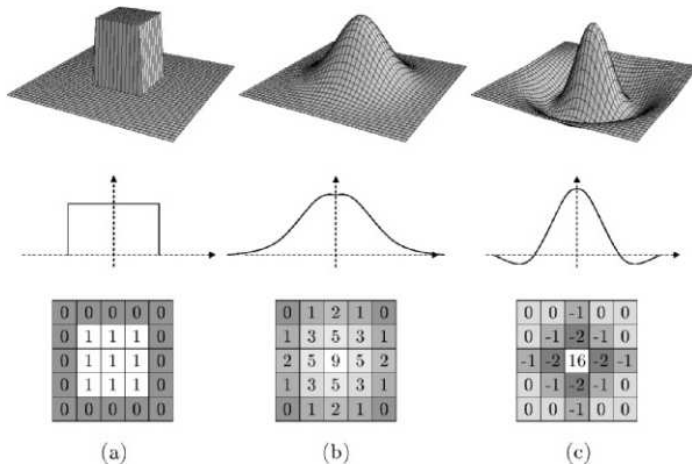
- ▶ Simplest linear filter: spatial average: reduces noise, but introduces blurring
- ▶ Distance weight samples
 - ▶ Centrally located samples are more important
 - ▶ Reduces blurring (somewhat)
 - ▶ Example above: simple integer arithmetic
- ▶ Alternative approach: utilize spectral characteristics

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



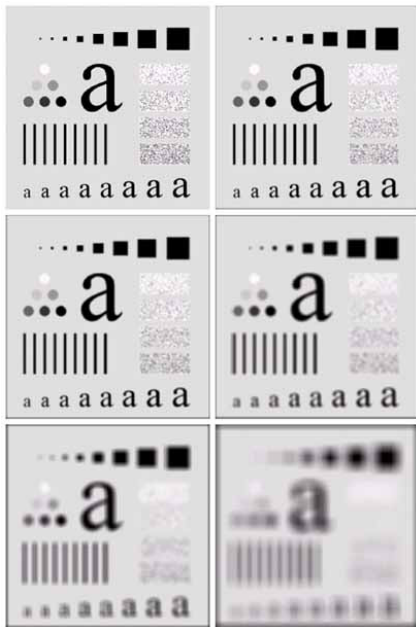
Linear Filters

Examples of linear filters. The “box” filter (a) and the Gauss filter (b) are both smoothing filters with all-positive coefficients. The “Laplace” or “Mexican hat” filter (c) is a difference filter.



Smoothing Example

- ▶ Square averaging filter results
- ▶ Window sizes: 3, 5, 9, 15, and 35



Order-Statistic Filters

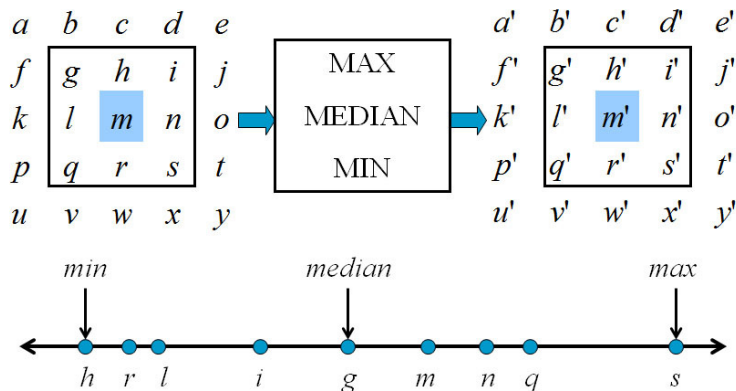
- ▶ Linear (weighted sum) filters
 - ▶ Blur edges and details
 - ▶ Are susceptible to outliers
- ▶ Order-statistic filters (nonlinear)
 - ▶ Preserve edges and details
 - ▶ Are less susceptible to outliers
- ▶ Spatially ordered samples: z_1, z_2, \dots, z_N
- ▶ Rank ordered samples: $z_{(1)}, z_{(2)}, \dots, z_{(N)}$

$$z_{(1)} \leq z_{(2)} \leq \dots \leq z_{(N)}$$
$$MED[z_1, z_2, \dots, z_N] = z_{((N+1)/2)}$$

Selection-type filter

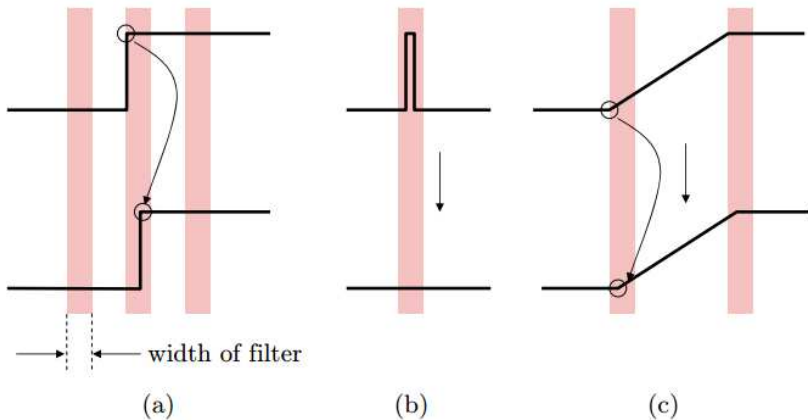
Filtering

Sometimes it is very useful to apply a non-linear filter such as the median.

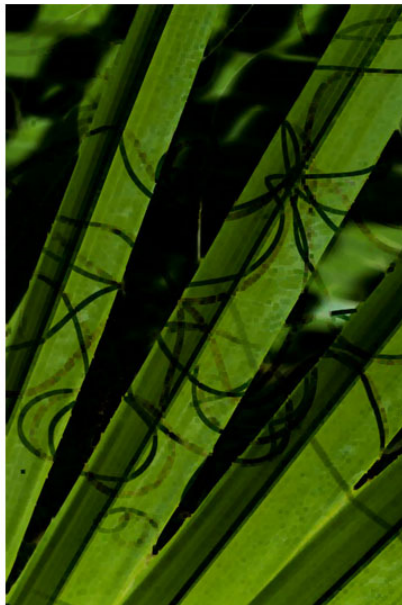
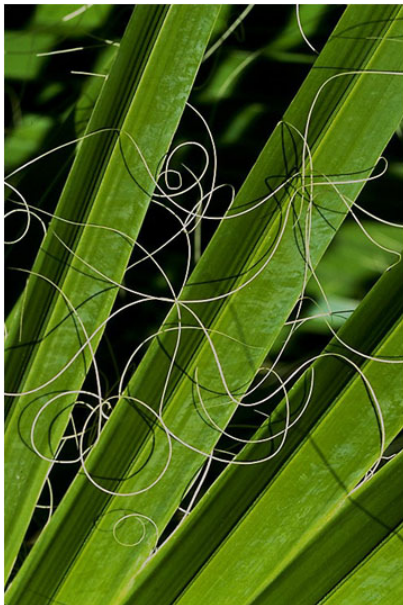


Effects of a minimum filter on various local signal structures

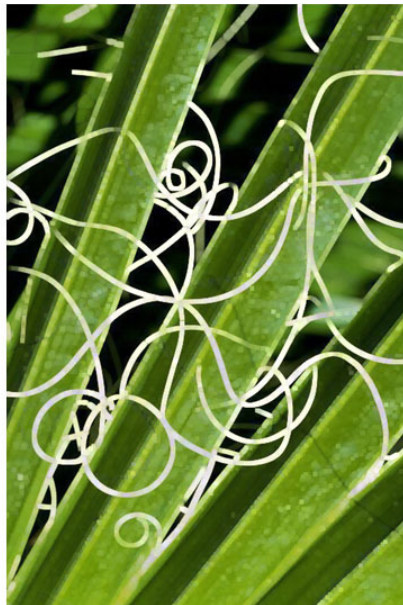
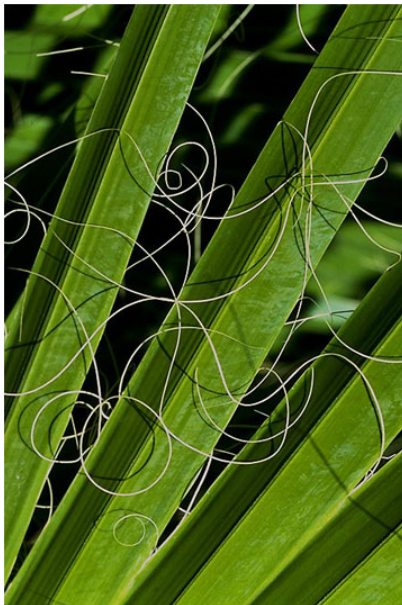
Original signal (top) and result after filtering (bottom), where color bars indicate the extent of the filter. Step edge (a) and linear ramp (c) are shifted to the right. Narrow pulse (b) is removed.



Minimum Filter

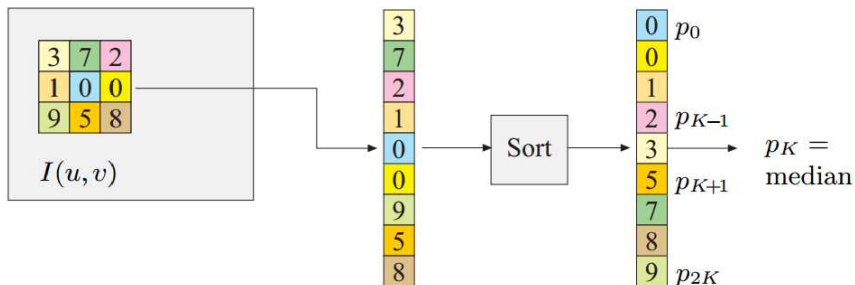


Maximum Filter

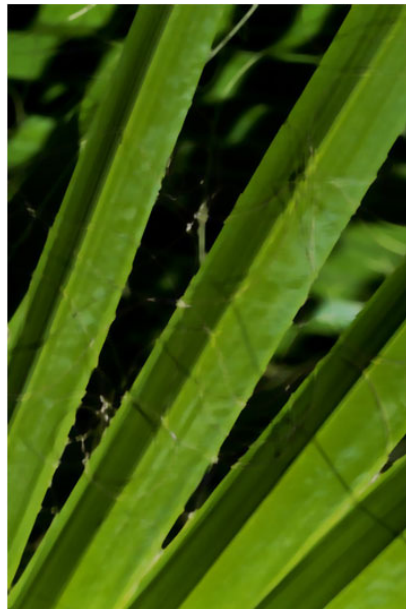
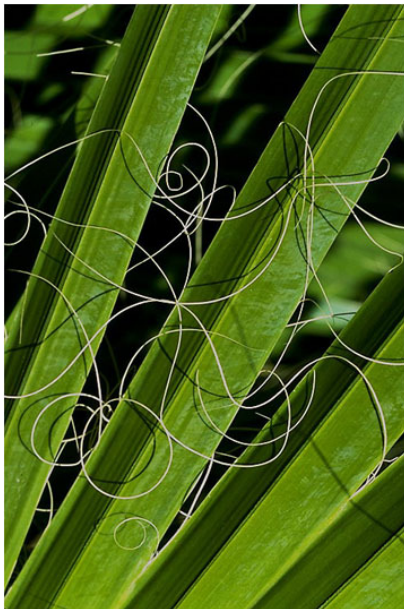


Median Filter

Computation of a 3×3 median filter. The nine pixel values extracted from the window are sorted and the center value is the median



Median Filter



Linear smoothing filter vs median filter



(a) Original image corrupted with "salt-and-pepper" noise. (b) Linear 3×3 pixel box filter (b) reduces noise but the entire image is blurred. (c) The median filter eliminates the noise dots and keeps the structures largely intact.

Weighted Median Filters (WMF)

The median filter can be generalized to weighted order statistic filters.

Given $x(n) = [x_1, \dots, x_N]$, and a set of weights w_1, \dots, w_N , the WMF output is given by

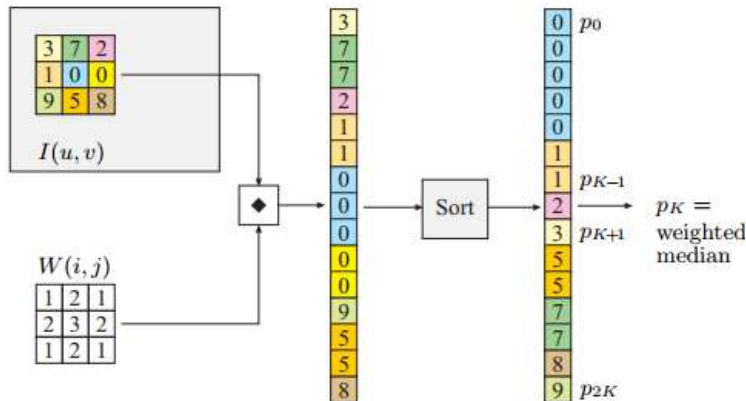
$$y(n) = \text{MEDIAN} [x_1 \diamond w_1, x_2 \diamond w_2, \dots, x_N \diamond w_N] \quad (1)$$

where $x_i \diamond w_i$ denotes replication:

$$x_i \diamond w_i = \overbrace{x_i, x_i, \dots, x_i}^{w_i \text{ times}} \quad (2)$$

and $w_i \in \mathbb{Z}^+$

Weighted median example



Each pixel value is replicated multiple times, as specified by the weight matrix W . For example, the value 0 from the center pixel is inserted three times. The pixel vector is sorted and the center value (2) is the weighted median.

Center Weighted Median Filter (CWMF)

A special case is the center weighted median filter (CWMF)

$$y(n) = \text{MEDIAN}[x_1, x_2, \dots, x_{c-1}, x_c \diamond w_c, x_{c+1}, \dots, x_N] \quad (3)$$

where $c = (N + 1)/2 = N_1 + 1$ is the index of the center sample.

For $w_c = 1$, the CWMF reduces to a median filter.

For $w_c \geq N$ the CWMF is an identity filter.

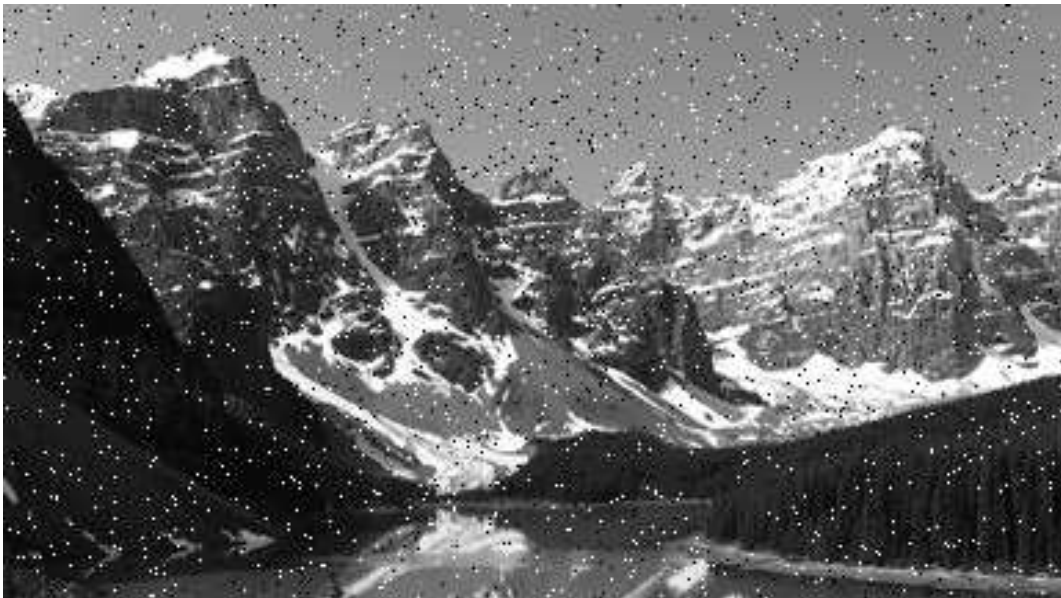
By varying the parameter

$$\begin{array}{ccc} \text{more} & \text{smoothing} & \text{less} \\ 1 & \longleftarrow \xrightarrow{w_c} & N \end{array}$$

Original



Salt and pepper noise



5×5 window - $w_c = 1$




5×5 window - $w_c = 3$



5×5 window - $w_c = 5$



Denoising – Most General Approach

$$i_{denoised}(x) = \frac{1}{\sum_{\text{all pixels } x'} w(x, x')} \sum_{\text{all pixels } x'} i_{noisy}(x') \cdot w(x, x')$$


- many (not all) denoising techniques work like this
- idea: average a number of similar pixels to reduce noise
- question/difference in approach: how similar are two noisy pixels?

Denoising – 3. Bilateral Filtering

$$i_{denoised}(x) = \frac{1}{\sum_{\text{all pixels } x'} w(x, x')} \sum_{\text{all pixels } x'} i_{noisy}(x') \cdot w(x, x')$$

spatial distance

distance of intensities

$$w(x, x') = \exp\left(-\frac{\|x' - x\|^2}{2\sigma^2}\right) \cdot \exp\left(-\frac{\|i_{noisy}(x') - i_{noisy}(x)\|^2}{2\sigma_i^2}\right)$$

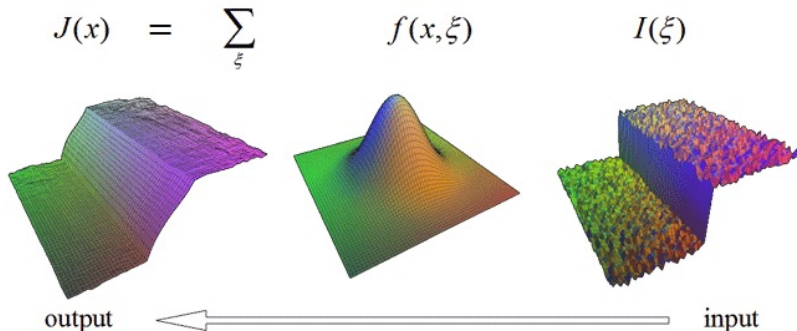
- more clever: average in local neighborhood, but only average similar intensities!

Denoising – Gaussian Filter

J: filtered output (is blurred)

f: Gaussian convolution kernel

I: step function & noise



Denoising – Bilateral Filter

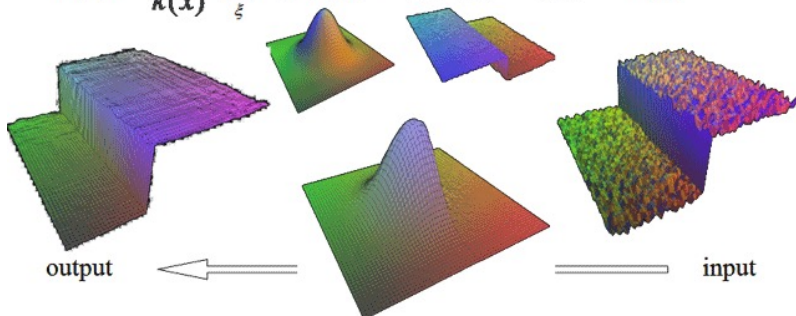
J: filtered output (is not blurred)

f: Gaussian convolution kernel

I: noisy image (step function & noise)

difference in intensity as scale!

$$J(x) = \frac{1}{k(x)} \sum_{\xi} f(x, \xi) g(I(\xi) - I(x)) I(\xi)$$



Denoising – Bilateral Filter



original image



bilateral filter = “edge-aware smoothing”

Denoising – Bilateral Filter



noisy image



bilateral filter = “edge-aware smoothing”

Denoising – 4. Non-local Means

$$i_{denoised}(x) = \frac{1}{\sum_{\text{all pixels } x'} w(x, x')} \sum_{\text{all pixels } x'} i_{noisy}(x') \cdot w(x, x')$$

$$w(x, x') = \exp\left(-\frac{\|W(i_{noisy}, x') - W(i_{noisy}, x)\|^2}{2\sigma^2}\right)$$

[Buades 2005]

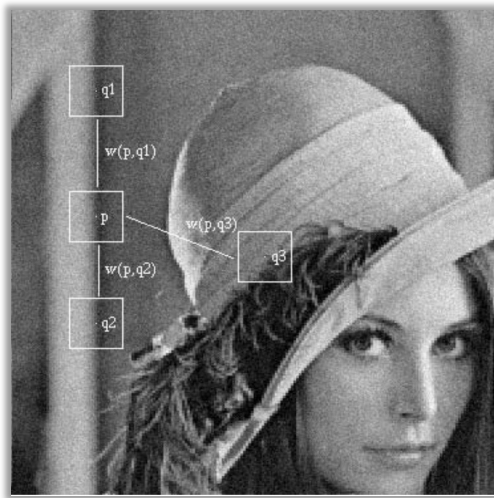
- very powerful approach: exploit self-similarity in image; average pixels with a similar neighborhood, but don't need to be close → non-local

Denoising – 4. Non-local Means

- define distance between global image patches
- average distant pixels with similar neighborhood!

$$i_{denoised}(x) = \sum_{\text{all pixels } x'} i_{noisy}(x') \cdot w(x, x')$$

[Buades 2005]



Denoising – 4. Non-local Means

noisy

Gaussian filtering

anisotropic filtering



TV

bilateral filtering

NL-means

[Buades 2005]

Filters Comparison

Original



Bilateral



FFDNet



BM3D



Filters Comparison

Original



Bilateral



FFDNet



BM3D



Filters Comparison

Original



Bilateral



FFDNet



BM3D



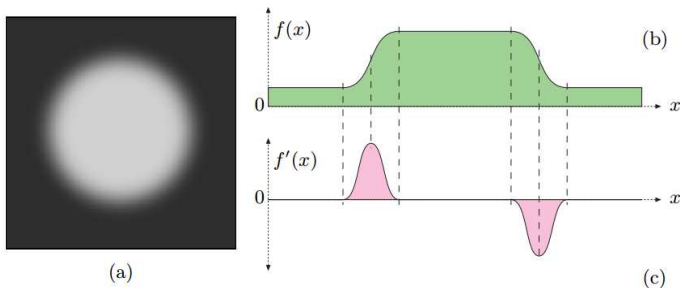
Sharpening Filters

- ▶ Objective: Highlight and enhance fine detail
- ▶ Utilize first- and second-order derivative
 - ▶ Derivatives identify signal changes (details/features)
- ▶ First-derivative:
 - ▶ Zero in flat regions
 - ▶ Nonzero along ramps
- ▶ Second-derivative requirements:
 - ▶ Zero in flat regions
 - ▶ Zero along ramps of constant slope

First derivative in one dimension

Original image (a), horizontal intensity profile $f(x)$ along the center image line (b), and first derivative $f'(x)$ (c).

$$\frac{df}{dx}(x) \approx \frac{f(x+1) - f(x-1)}{2} = 0.5 \cdot (f(x+1) - f(x-1)).$$



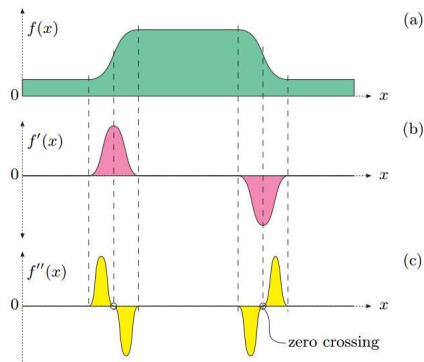
Second Derivative

- ▶ Utilize difference equations: First derivative:

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

Second derivative

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



Partial Derivatives and the Gradient

Derivative of a multidimensional function taken along one of its coordinates
partial derivative;

$$\frac{\partial I}{\partial x}(x, y) \text{ and } \frac{\partial I}{\partial y}(x, y)$$

The gradient function is

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

The *magnitude* of the gradient,

$$|\nabla I|(x, y) = \sqrt{\left(\frac{\partial I}{\partial x}(x, y)\right)^2 + \left(\frac{\partial I}{\partial y}(x, y)\right)^2}$$

Derivative Filters

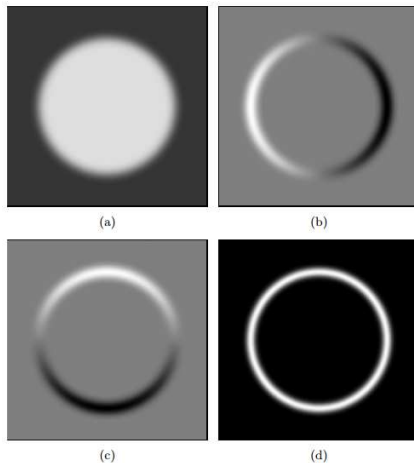
The gradient approximation of the horizontal derivatives is the coefficient matrix:

$$H_x^D = [-0.5 \quad \mathbf{0} \quad 0.5] = 0.5 \cdot [-1 \quad \mathbf{0} \quad 1]$$

where the coefficients -0.5 and $+0.5$ correspond to the image elements $I(x-1, y)$ and $I(x+1, y)$. Vertical component of the gradient

$$H_y^D = \begin{bmatrix} -0.5 \\ \mathbf{0} \\ 0.5 \end{bmatrix} = .5 \cdot \begin{bmatrix} -1 \\ \mathbf{0} \\ 1 \end{bmatrix}$$

Partial Derivatives of a Two Dimensional Function



(a) Synthetic image, (b) first derivatives in the horizontal direction $\partial I / \partial x$. (c) vertical direction $\partial I / \partial y$; (d) magnitude of the gradient $|\nabla I|$ (d).

Laplacian Edge Operator

Problem with edge operators based on first derivatives; edges are as wide as the underlying intensity transition. Alternative makes use of the second derivatives.

Laplacian is the simplest isotropic derivative:

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



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

- ▶ Isotropic to 90° rotations
- ▶ Add diagonal derivatives to make it 45° isotropic

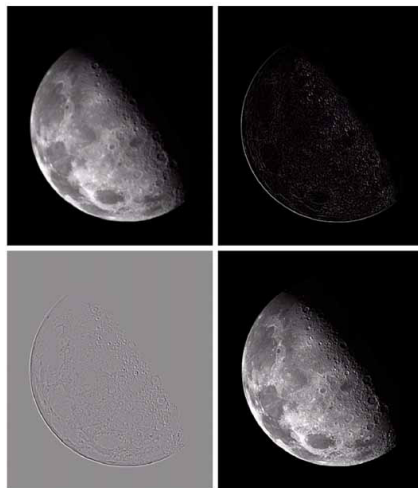
Mask Implementations and Enhancement

- ▶ Similar definition produces a sign change
- ▶ Enhancement adds (subtracts) derivative and observed image

0	1	0	1	1	1
1	-4	1	1	-8	1
0	1	0	1	1	1
0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

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

Example



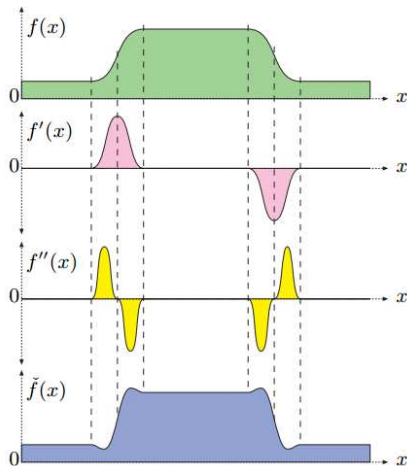
Top Left: North pole of the moon. Top Right: Laplacian-filtered image. Bottom Left: Laplacian image scaled for display purposes. Bottom Right: Image enhanced

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & +5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



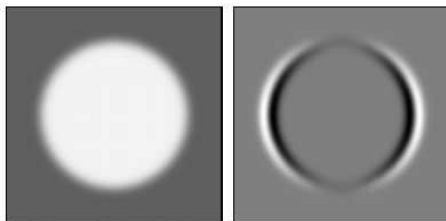
Edge Sharpening with second derivative

Original intensity function $f(x)$, first derivative $f'(x)$, second derivative $f''(x)$, and sharpened intensity function $\check{f}(x) = f(x) - w \cdot f''(x)$



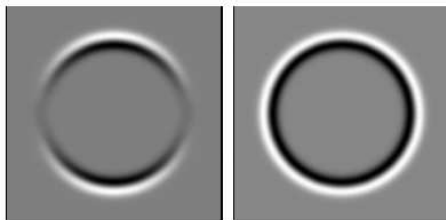
Results of Laplace filter H^L

(a) Image $I(a)$, (b) second partial derivative $\partial^2 I / \partial^2 u$ in the horizontal direction, (c) second partial derivative $\partial^2 I / \partial^2 v$ in the vertical direction, and (d) Laplace filter $\nabla^2 I(u, v)$.



(a)

(b)

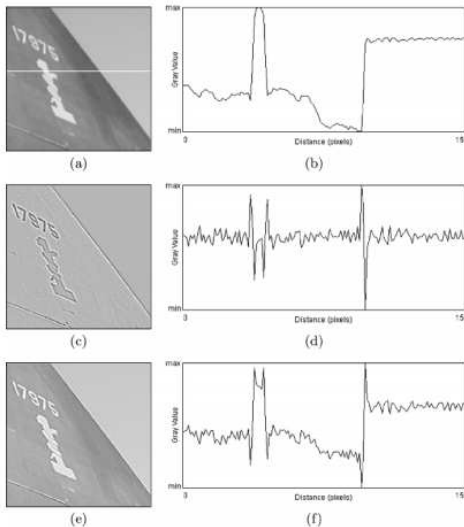


(c)

(d)

Edge Sharpening with the Laplace filter

Original image and marked line (a, b), result of Laplace filter H^L (c, d), and sharpened image (e, f).



Edge Detection

$$f(x)$$



$$|df(x)|$$



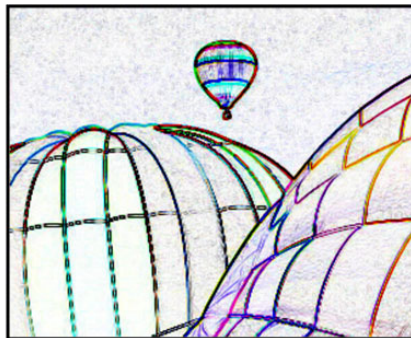
$$|df(x)|$$



Edge Detection

The derivative of a digital image can be found by applying the linear filter.

$$\begin{bmatrix} -1 & +2 & -1 \\ +2 & -4 & +2 \\ -1 & +2 & -1 \end{bmatrix}$$



Edge Detection

The edges are then thinned by applying a threshold.

