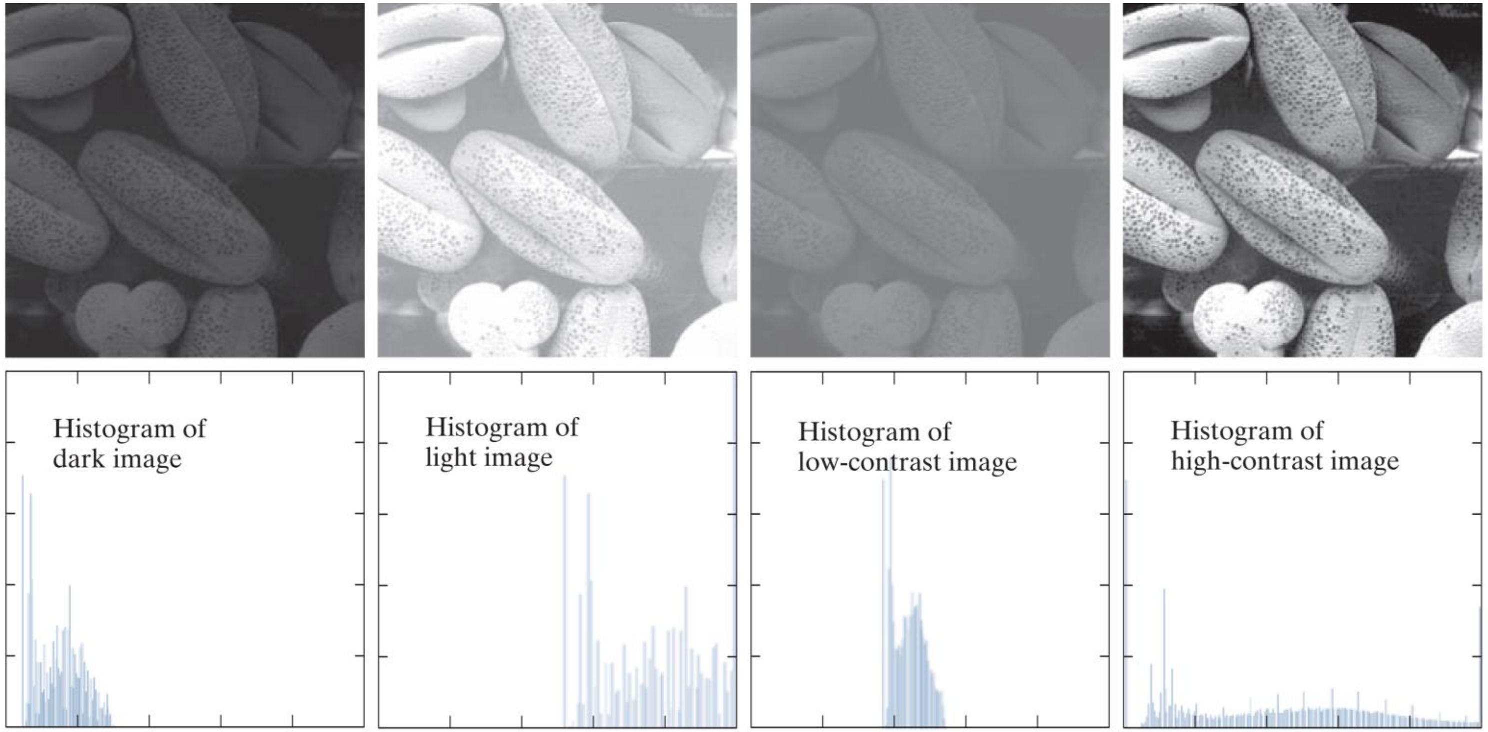
Digital Image Processing

Image Enhancement

Emad Fatemizadeh
Distance/online Course: Session 03
Date: 21 February 2021, 3rd Esfand 1399

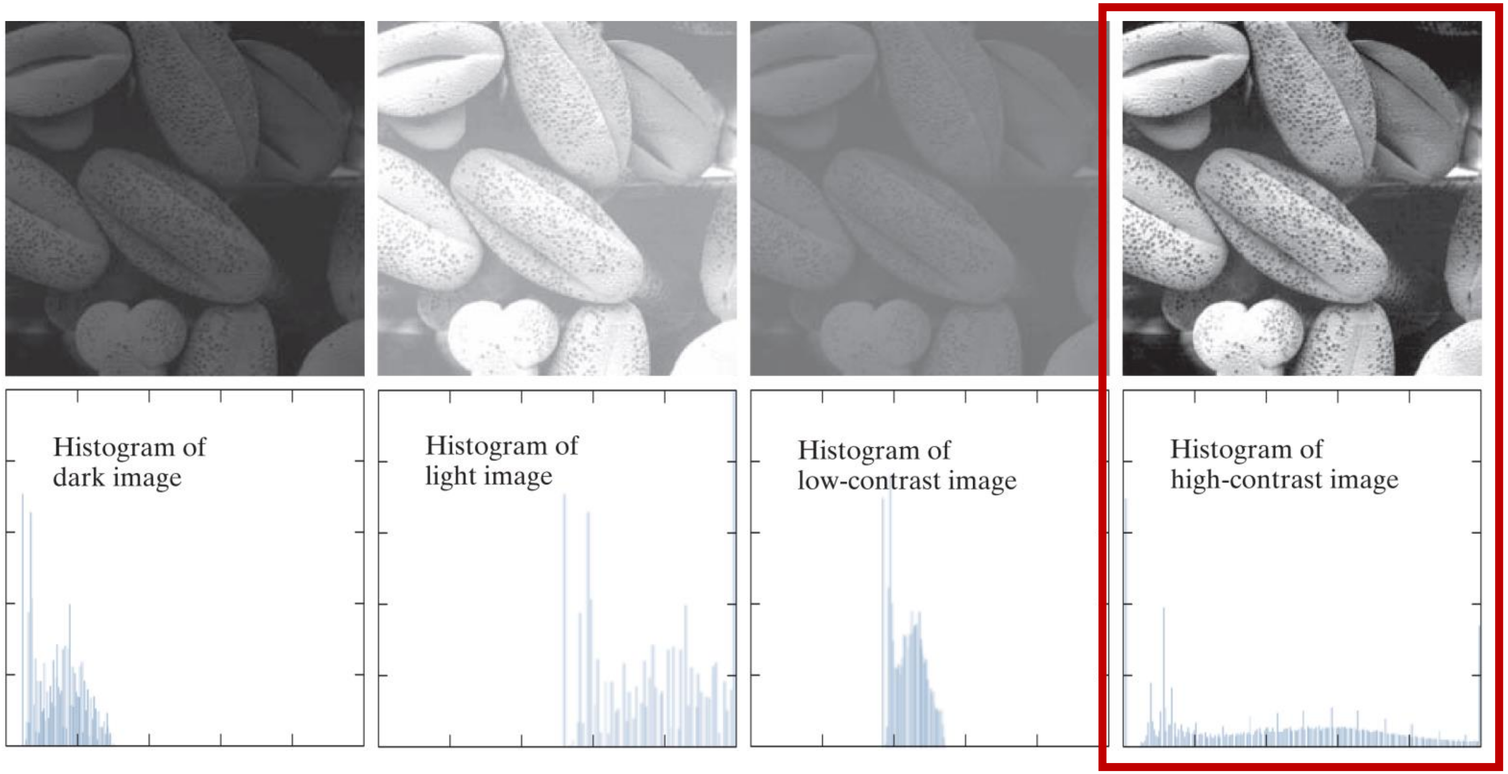
Histogram Processing

- Image Histogram and Normalized Histogram:
- $h(r_k) = n_k, \ \ k = 0,1,2,\cdots,L-1 \Rightarrow p(r_k) = \frac{n_k}{MN}$



Histogram Processing

› Effect of histogram on Image quality:

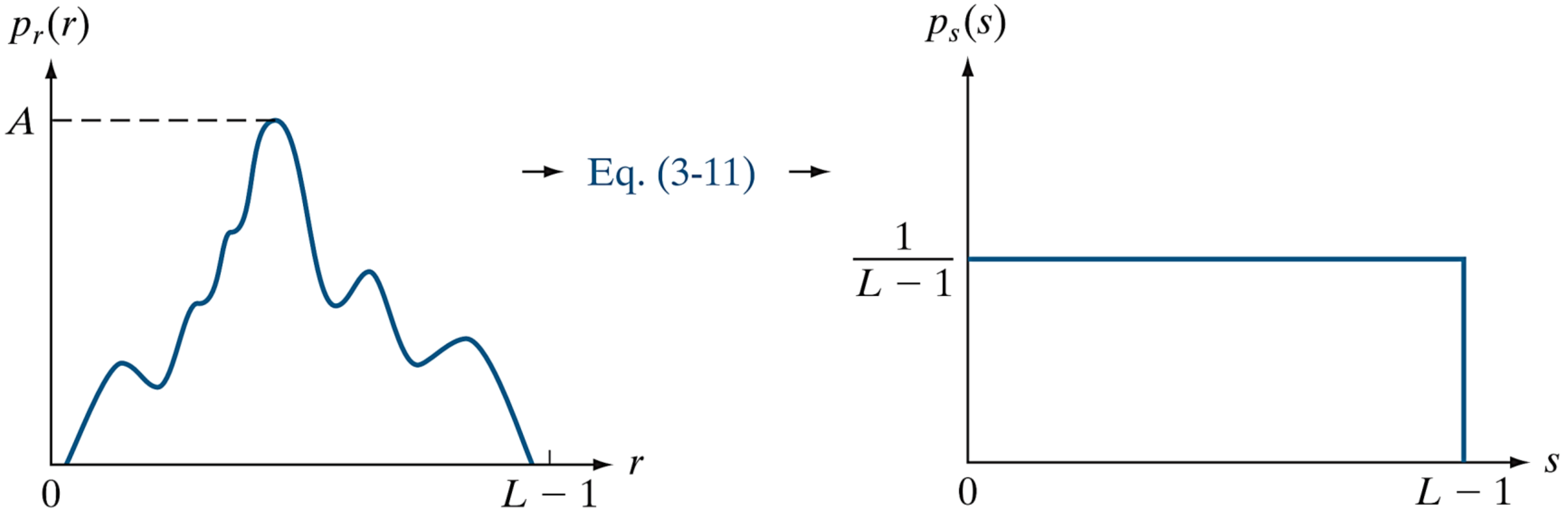


Histogram Equalization

› Continuous case:

$$p_r(r) \sim Input\ Image\ pdf$$

$$s = T(r) = (L-1)\int_0^r p_r(w)dw \propto Uni[0, L-1]$$

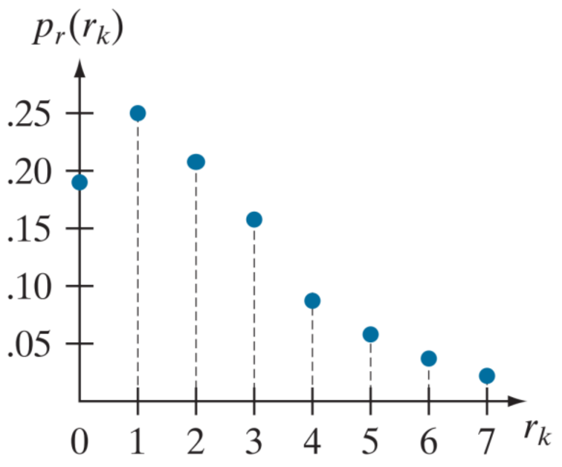


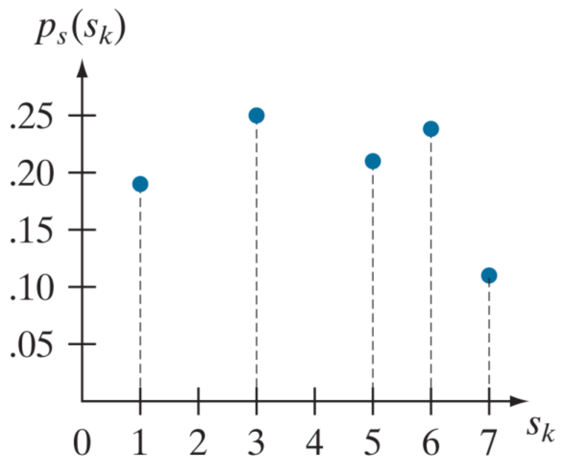
Histogram Equalization

› Discrete case:

$$p_r(r_k) = \frac{n_k}{MN}$$

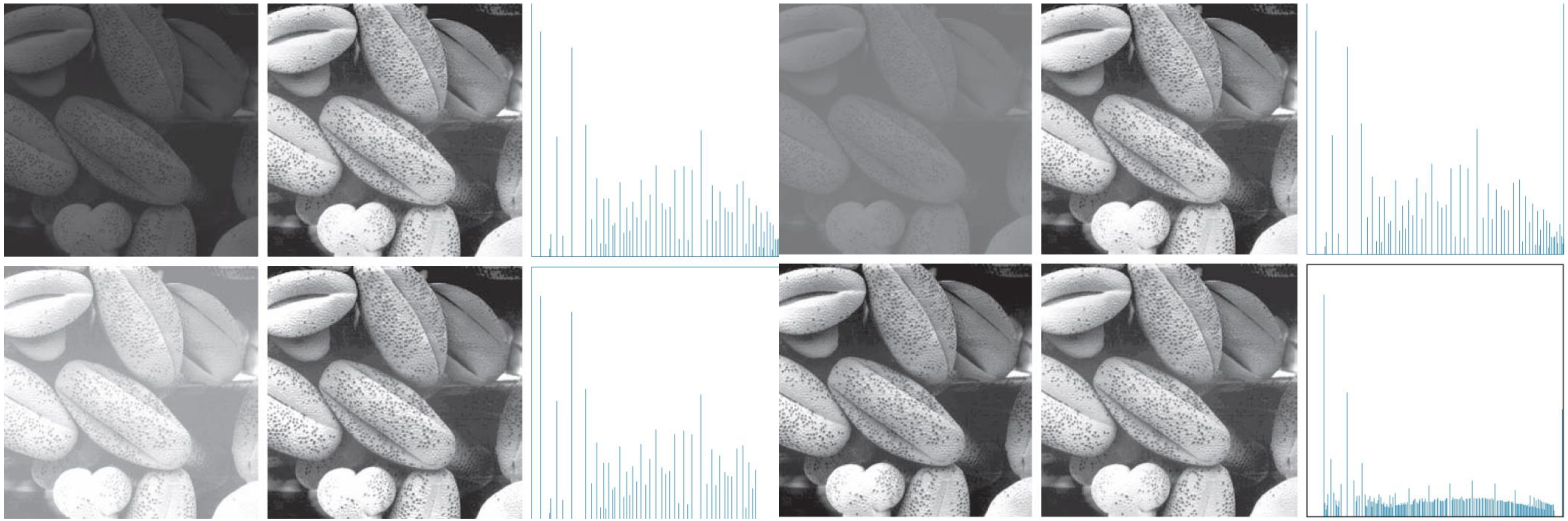
$$s = T(r) = (L-1)\sum_{j=0}^{k} p_r(r_j) \propto ???$$





Histogram Equalization

- Example:
  - Exact equalization in NOT achieved!



Histogram Matching

› Image with highest quality in field does NOT obey uniform distribution, necessarily!

› Assume we know target optimal distribution, $p_z(r)$

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

$$G(z) = (L-1)\int_0^z p_z(v)dv = s$$

$$z = G^{-1}\big(T(r)\big) \propto p_z(z)$$

Local Processing

- Local processing is very effective approach in image processing (Natural images are Non-stationary)
  - Local histogram equalization
  - Local and adaptive intensity transform
  - Local statistics (mean and variance)

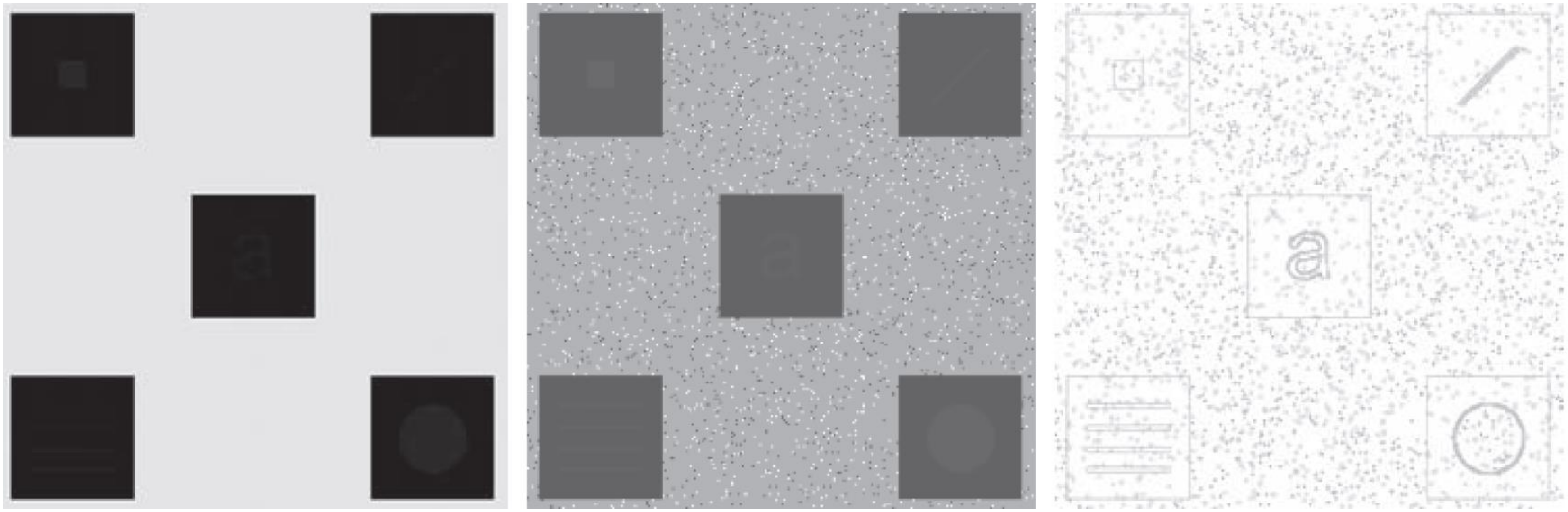
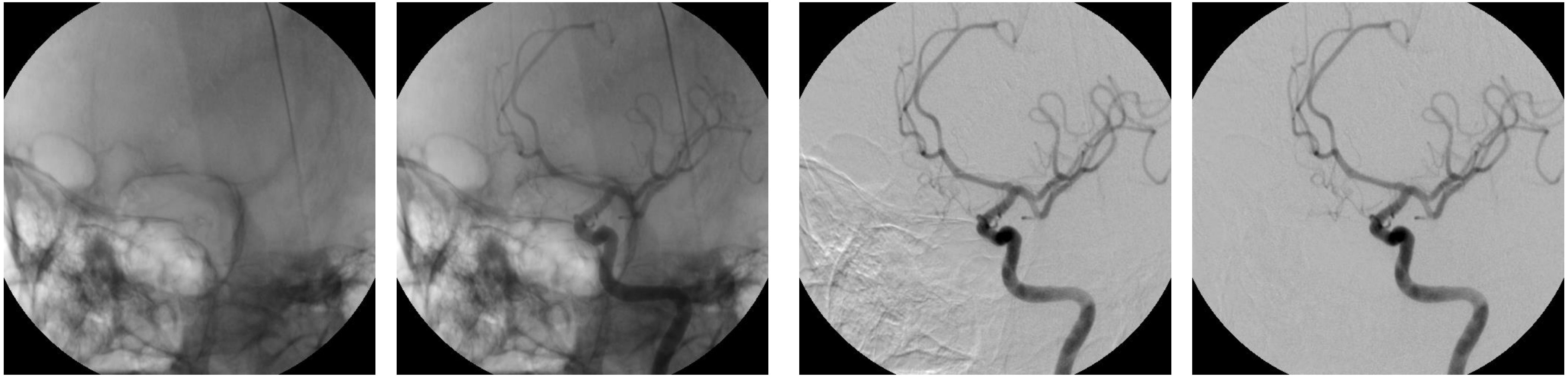


Image Subtraction

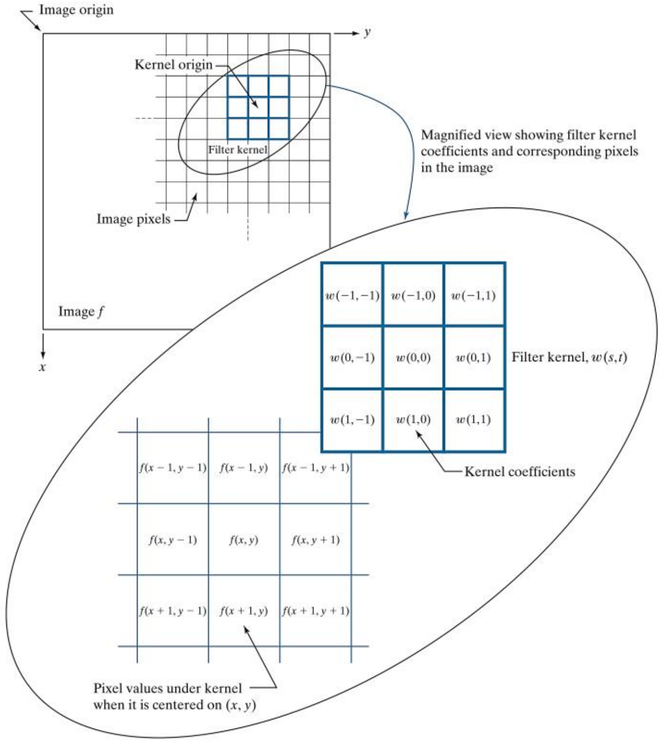
- Digital Subtraction Angiography: $f_{Enh} = K(f_{Post} - f_{Pre}) + f_{Pre}$



Pre Contrast Agent Injection	Post Contrast Agent Injection	Subtraction without Registration	Subtraction with Registration

Spatial Domain Process

- For $n \times n$ window,
  - Filtering/Mask/Kernel/Window/Template Processing



Spatial Domain Process

- Smoothing Linear Filtering (Correlation and/or Convolution)

- $g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x - s, y - t)$
- $g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)$

Spatial Domain Process

› Correlation/Convolution, valid (center) and same (right)

Initial position for w

$$\begin{matrix} 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 4 & 5 & 6 & 0 & 0 & 0 & 0 \\ 7 & 8 & 9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Correlation result

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 9 & 8 & 7 & 0 \\ 0 & 6 & 5 & 4 & 0 \\ 0 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Full correlation result

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 9 & 8 & 7 & 0 & 0 \\ 0 & 0 & 6 & 5 & 4 & 0 & 0 \\ 0 & 0 & 3 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Rotated w

$$\begin{matrix} 9 & 8 & 7 & 0 & 0 & 0 & 0 \\ 6 & 5 & 4 & 0 & 0 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Convolution result

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 \\ 0 & 4 & 5 & 6 & 0 \\ 0 & 7 & 8 & 9 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

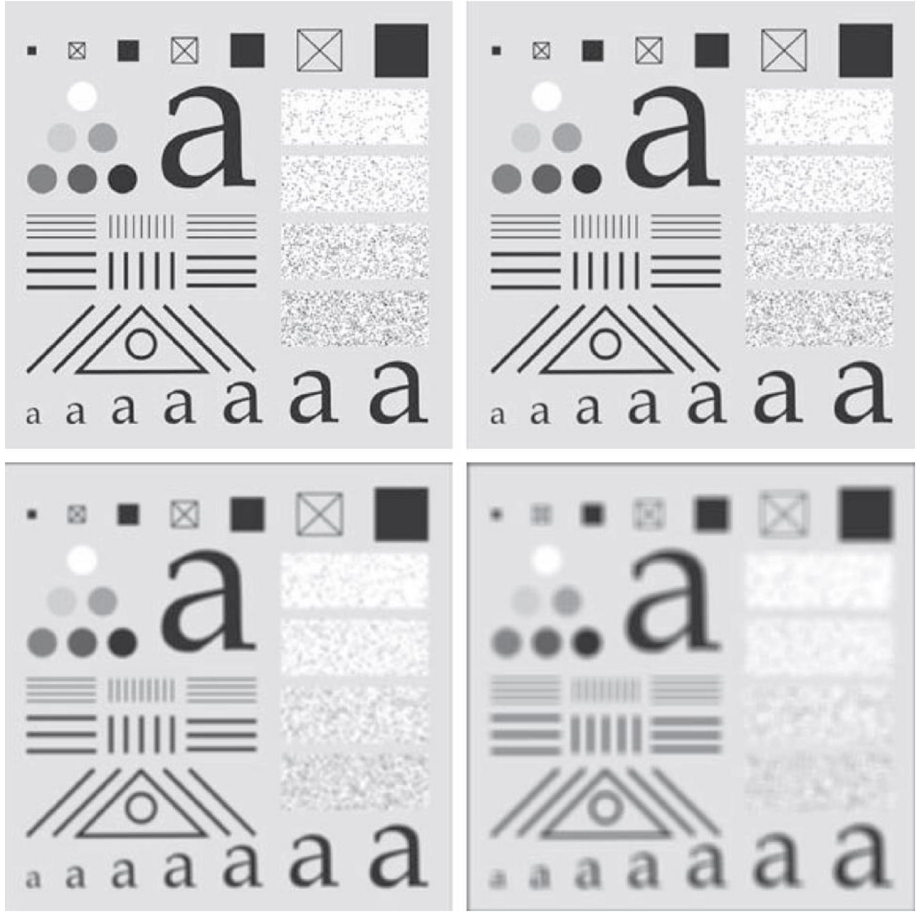
Full convolution result

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 4 & 5 & 6 & 0 & 0 \\ 0 & 0 & 7 & 8 & 9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Spatial Domain Process

- Blurring Effect:
- Boxcar windows:

1×1	3×3
11×11	21×21



$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Spatial Domain Process

› Most Common Spatial Filter:

–Gaussian Kernel:

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \xrightarrow{kernel\ limited\ size} G_\sigma(x,y) = Ke^{-\frac{x^2+y^2}{2\sigma^2}}$$

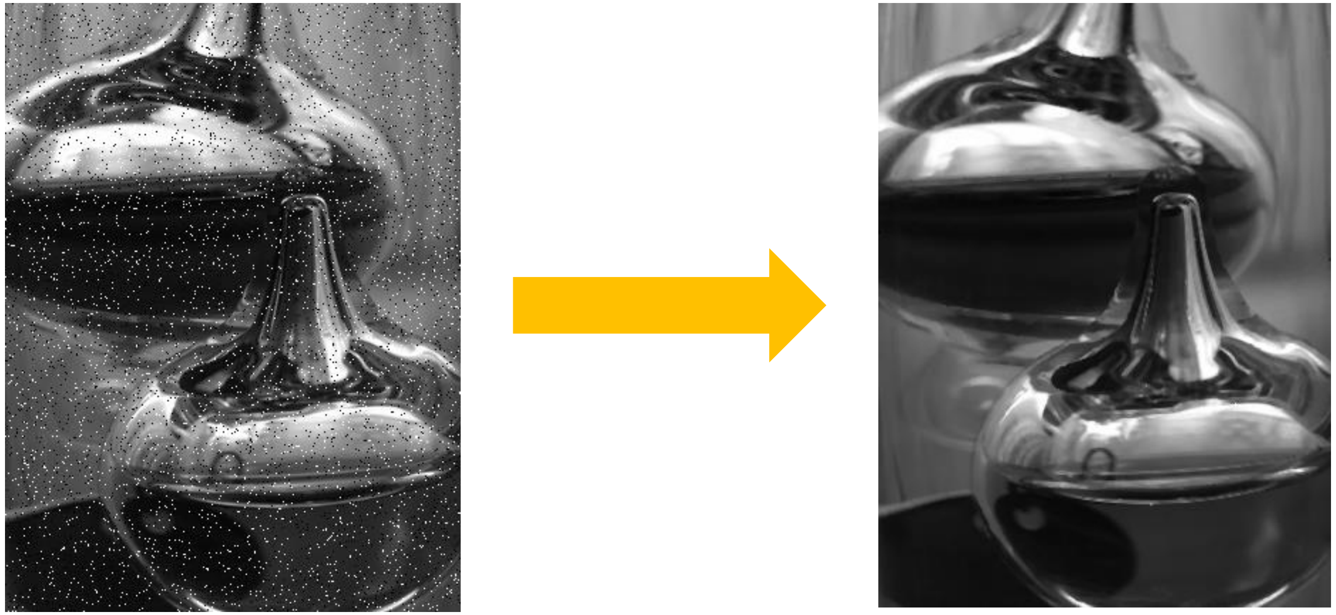
-Kernel/Windows size:≈ $([6\sigma] \times [6\sigma])$

–K: Normalization factor $\left(\sum_x \sum_y G_\sigma(x,y) = 1\right)$

–Less blurring

Spatial Domain Process

- Order statistics filter:
  - Median (Best simple choice for salt & pepper noise)
  - $g(x,y) = \sum_{(s,t)\in S(x,y)} median\{f(s,t)\}$



Spatial Domain Process

› Image Sharpening:

– Highlight edges using *first* or *second* derivative:

› Laplacian of image

$$\pm\left(\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial Y^2}\right)=\pm\nabla^2 f$$

› Discrete Implementation with + sign (left) and – sign (right):

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

Spatial Domain Process

- Image enhancement:

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

- Kernel formulation:

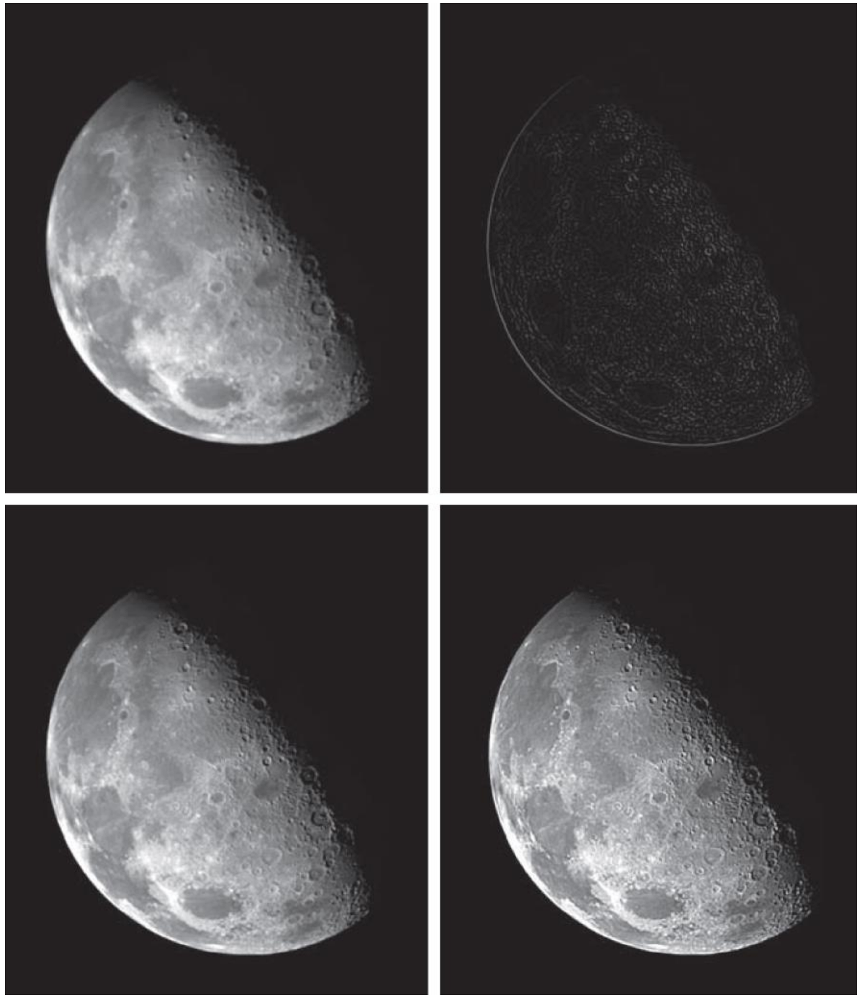
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 00 \end{bmatrix} + c\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Spatial Domain Process

- Example:

a	b
c	d

(a) Blurred image of the North Pole of the moon.
(b) Laplacian image obtained using the kernel in Fig. 3.45(a).
(c) Image sharpened using Eq. (3-54) with $c = -1$.
(d) Image sharpened using the same procedure, but with the kernel in Fig. 3.45(b).



Spatial Domain Process

› Noise suppression in Laplacian processing:

$$-\left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial Y^2}\right) = -\nabla^2 f$$

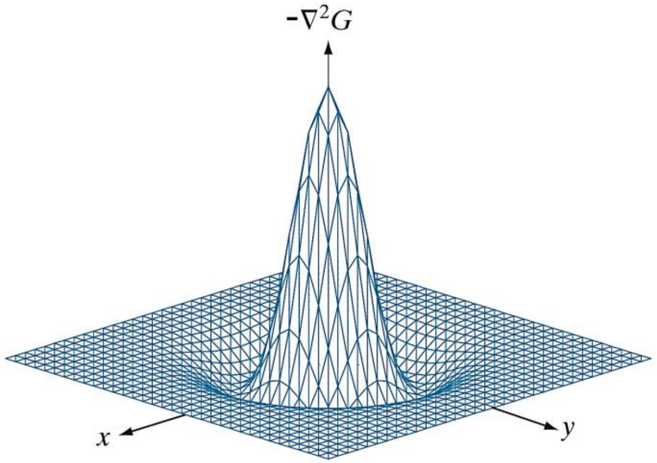
› Laplacian of Gaussian (LoG) of image:

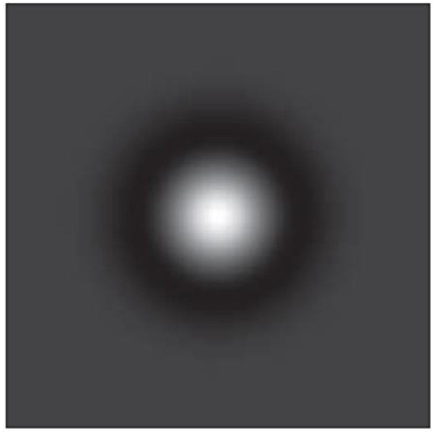
$$LoG(f) = -\nabla^2 (G_\sigma * f) = (-\nabla^2 G_\sigma) * f$$

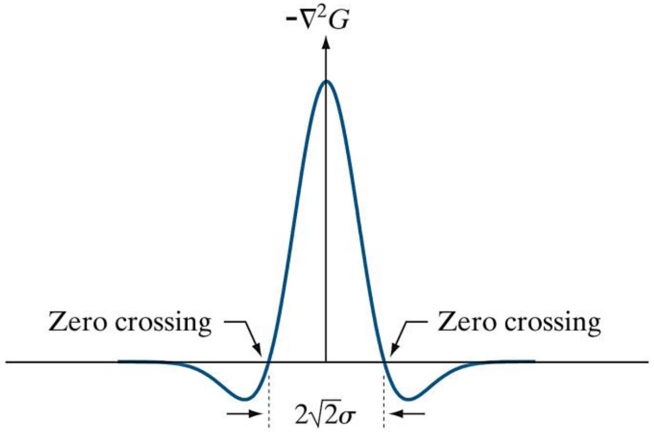
$$\nabla^2 G_\sigma = \left(\frac{2\sigma^2 - x^2 - y^2}{\sigma^4}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Spatial Domain Process

› *LoG Kernel*







0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

Spatial Domain Process

› LoG approximation via DoG (Difference of Gaussian)

$$G_D(x,y) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2+y^2}{2\sigma_1^2}} - \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}}$$

Spatial Domain Process

› Image Sharpening using image gradient:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x}, & \frac{\partial f}{\partial y} \end{bmatrix}^T \Rightarrow M(x,y) = \|\nabla f\|, \widehat{M}(x,y,) = \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|$$

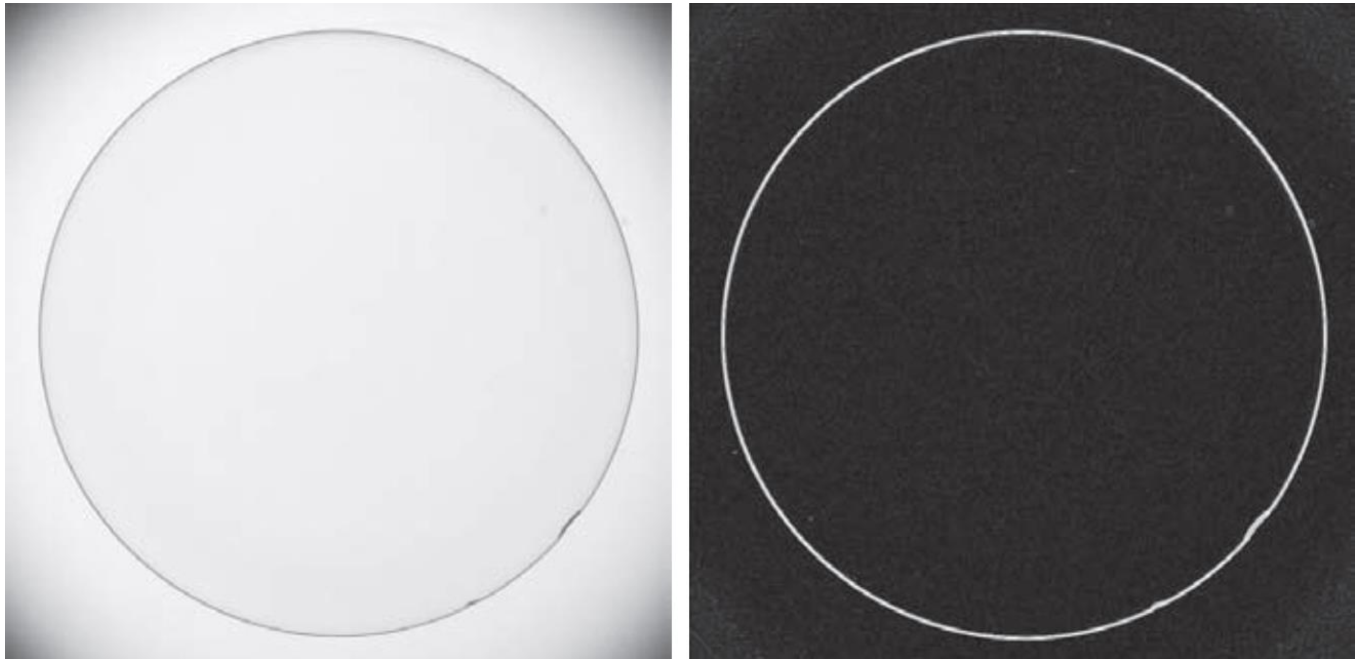
› Discrete implementation of g_x and g_y (*Sobel* Mask):

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Spatial Domain Process

› Image Sharpening using image gradient, $M(x, y)$:



Matlab Command

- Image Statistics:
  - means2, std2, corr2, imhist, regionprops
- Image Intensity Adjustment:
  - imadjust, histeq, adapthisteq, imnoise
- Linear Filter:
  - imfilter, fspecial, conv2, corr2,
- Nonlinear filter:
  - medfilt2, ordfilt2

The End

› AnY QuEsTiOn?