Chapter 3: Intensity Transformations and Spatial Filtering

Enhancement with Averaging Operations

When images are displayed (or printed), they often have suffered from noise and interferences from several sources including:

- electrical sensor noise,
- photographic grain noise, and
- channel errors.

Noisy image

These noise effects can be removed by simple ad hoc "noise-cleaning" techniques or more advanced methods, including deep (machine) learning) applied to local neighborhoods of input pixels.

Enhancement with Averaging Operations

Consider a (simplified) noisy image:

$$g(x,y) = f(x,y) + \eta(x,y)$$

where the second term is **noise**,

We assume that the noise is *uncorrelated* with the input and has *zero mean*. Then, averaging K different noisy images:

$$\bar{g}(x,y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x,y)$$

produces an output image with

$$E[\bar{g}(x,y)] = f(x,y) \quad and \quad \sigma_{\bar{g}}^2 = \frac{1}{K}\sigma_\eta^2$$

Chapter 3: Intensity Transformations and Spatial Filtering

Enhancement with Averaging Operations

original

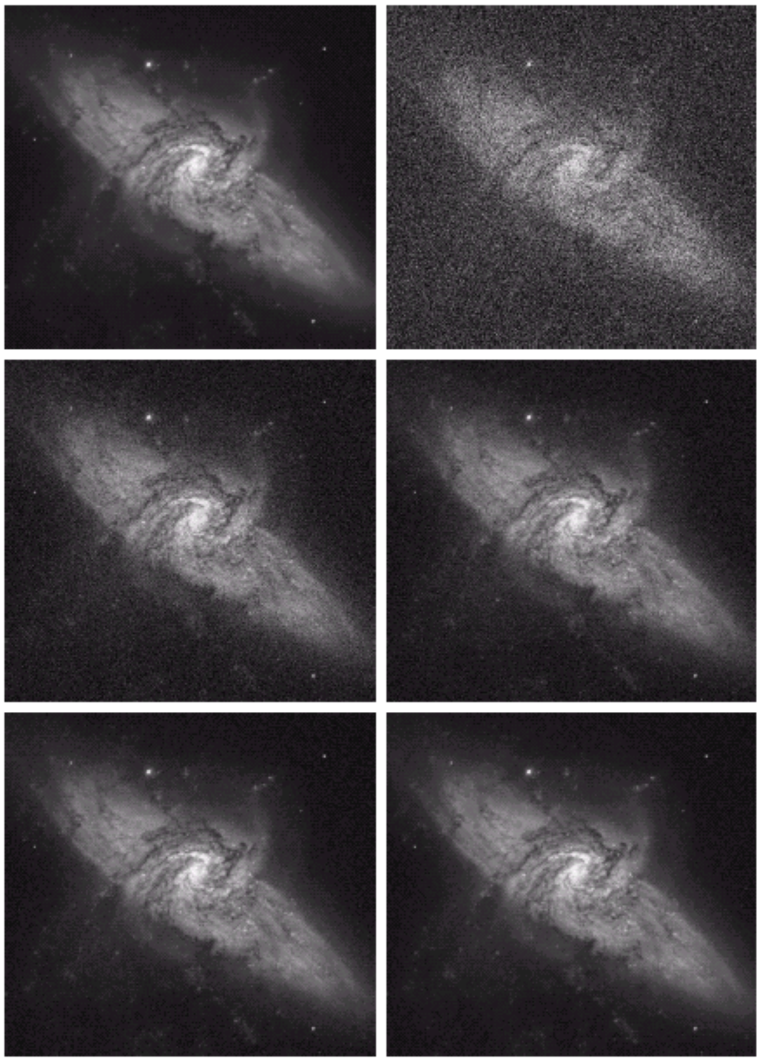
noisy image, N(0,64^2)

result of averaging 8 noisy images

16 noisy images

64 noisy images

128 noisy images



a b
c d
e f

FIGURE 3.30 (a) Image of Galaxy Pair NGC 3314. (b) Image corrupted by additive Gaussian noise with zero mean and a standard deviation of 64 gray levels. (c)–(f) Results of averaging $K = 8, 16, 64$, and 128 noisy images. (Original image courtesy of NASA.)

Chapter 3: Intensity Transformations and Spatial Filtering

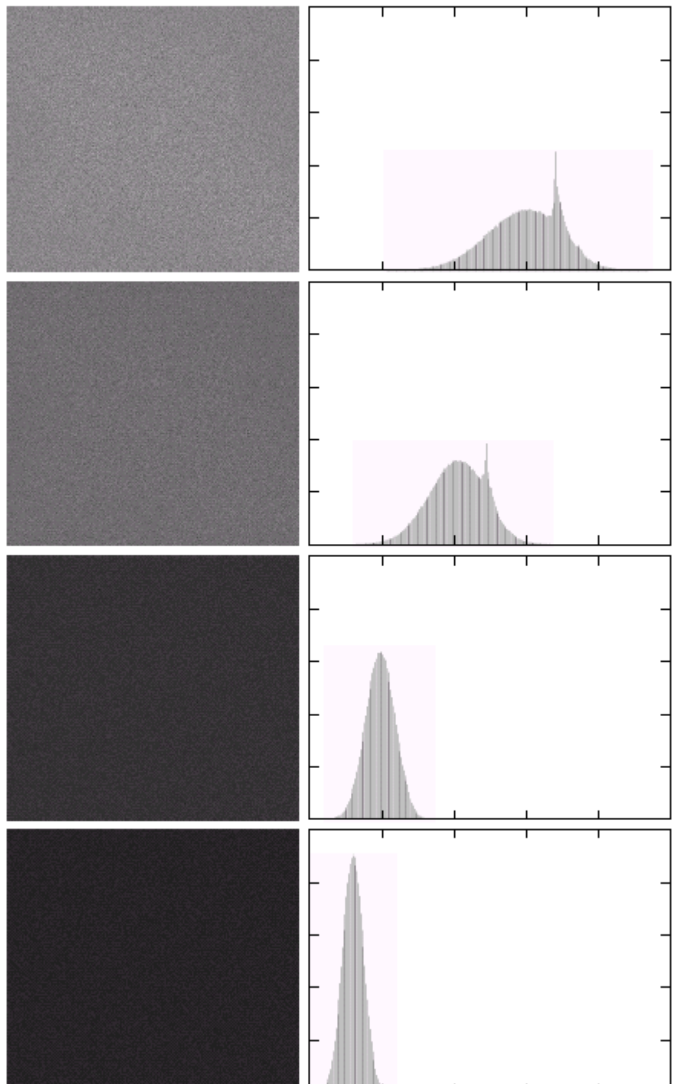
Enhancement with Averaging Operations

K=8

K=16

K=64

K=128



a b

FIGURE 3.31
(a) From top to bottom: Difference images between Fig. 3.30(a) and the four images in Figs. 3.30(c) through (f), respectively. (b) Corresponding histograms.

Remarks:

1. Notice how the noise is
- smoothed, and its
- variance is decreasing

with increasing K.

2. The operation is costly in computation and memory.

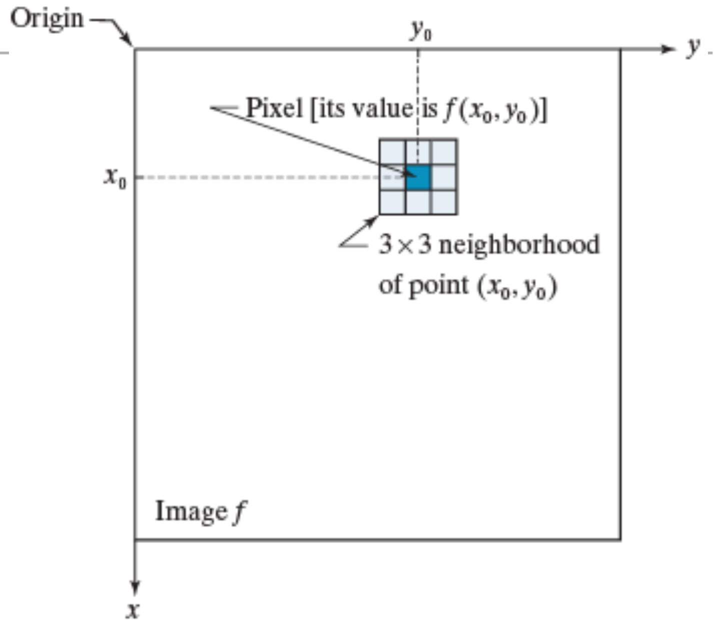
Chapter 3: Intensity Transformations and Spatial Filtering

Alternative solution:
Spatial Domain Operation:

$$g(x,y) = T[f_{N(x,y)}(x,y)]$$

$f_{N(x,y)}(x,y)$ is the original image $f(x,y)$ defined over the neighborhood $N(x,y)$ of (x,y); $g(x,y)$ is the output image and $T[.]$ is an operator on $f(x,y)$ defined over a neighborhood $N(x,y)$ of (x,y).

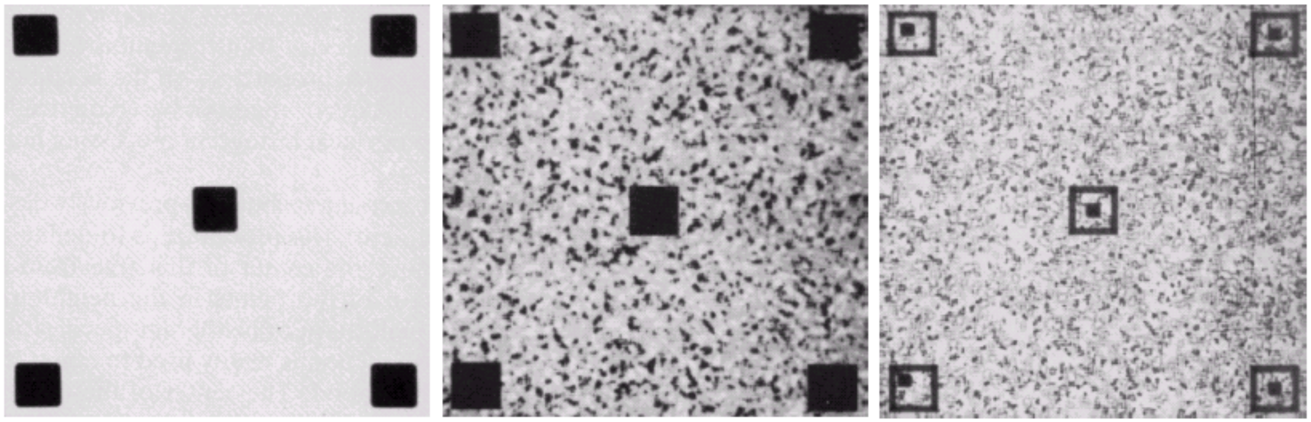
FIGURE 3.1 A 3×3 neighborhood about a point (x_0, y_0) in an image. The neighborhood is moved from pixel to pixel in the image to generate an output image. Recall from Chapter 2 that the value of a pixel at location (x_0, y_0) is $f(x_0, y_0)$, the value of the image at that location.



Special case: when the neighborhood N(x,y) contains 1 pixel, then $T[.]$ is called intensity transformation function.

Now we consider the information from larger neighborhoods

Localized Histogram Equalization

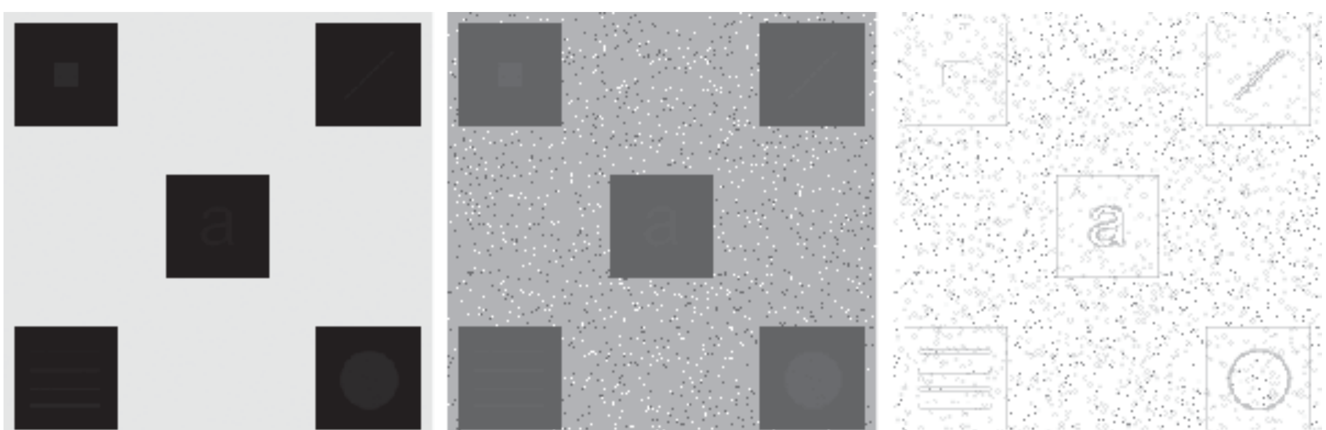


a b c

~~FIGURE 3.23~~ (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

a b c

FIGURE 3.32
(a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization.



Chapter 3: Intensity Transformations and Spatial Filtering

Local Histogram Statistics

How to obtain this?

~~FIGURE 3.24~~ SEM image of a tungsten filament and support, magnified approximately 130×. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene).

There is another filament behind the front one, but it’s too dark to see. Can we use local information to highlight regions of low-intensity while preserving the visible ones as they are?

Local Histogram Statistics

Normalized histogram values can be treated as probabilities.

We can then compute global statistics over the whole image:

Mean: $$m = \sum_{i=0}^{L-1} r_i p(r_i)$$ Variance: $$\sigma^2 = \sum_{i=0}^{L-1} [r_i - m]^2 p(r_i)$$

Local statistics can be computed in neighborhoods of the image:

Mean: $$m_{S_{xy}} = \sum_{(s,t)\in S_{xy}} r_{s,t} p(r_{s,t})$$

Variance: $$\sigma^2{}_{S_{xy}} = \sum_{(s,t)\in S_{xy}} [r_{s,t} - m_{S_{xy}}]^2 p(r_{s,t})$$

(index *i* omitted here for clarity)

Chapter 3: Intensity Transformations and Spatial Filtering
Local Histogram Statistics

We can use local statistics to selectively apply intensity transforms.

Local mean compared to global mean indicates whether the area is dark or light.

Difference between local and global variance indicates contrast.

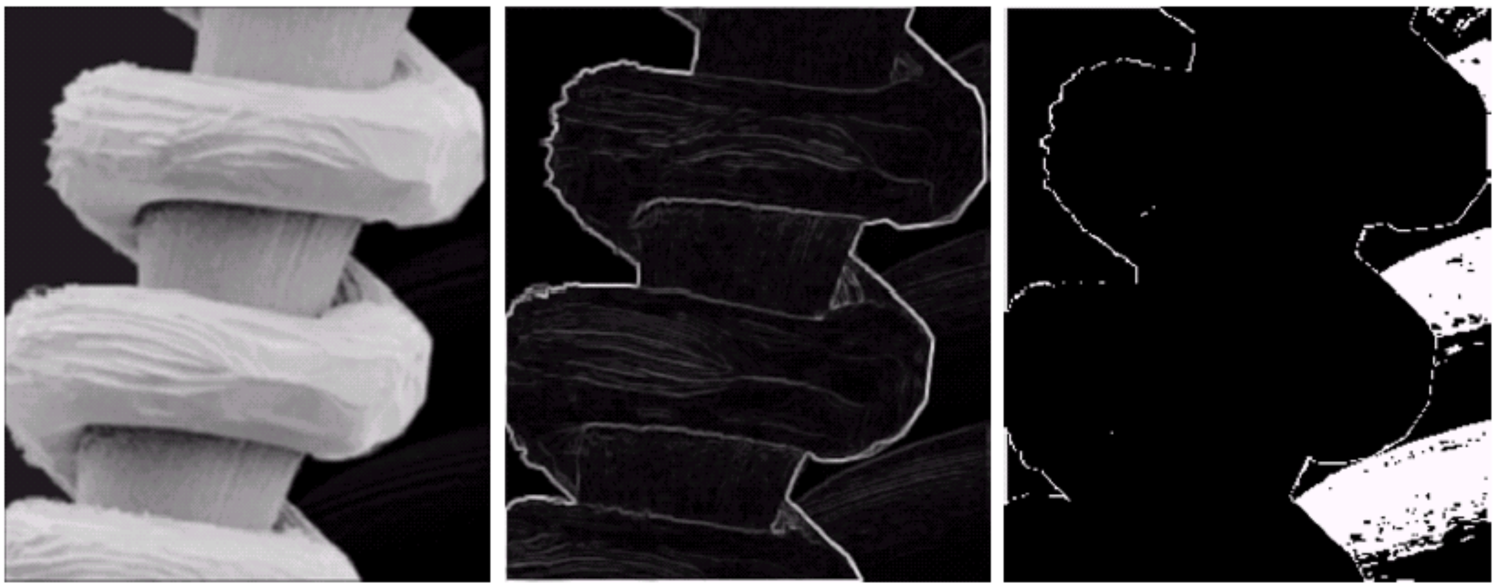
By setting thresholds we obtain a selective transformation:

$$g(x,y) = \begin{cases} C \cdot f(x,y) & \text{if } k_0 m_G < m_{S(x,y)} < k_1 m_G \text{ AND } k_2 \sigma_G < \sigma_{S(x,y)} < k_3 \sigma_G \\ f(x,y) & \text{otherwise} \end{cases}$$

Chapter 3: Intensity Transformations and Spatial Filtering

Local Histogram Statistics

$m_{S_{xy}}$ $\quad$ ${\sigma^2}_{S_{xy}}$ $\quad$ $(1, C)$



a b c

~~FIGURE 3.25~~ (a) Image formed from all local means obtained from Fig. 3.24 using Eq. (3.3-21). (b) Image formed from all local standard deviations obtained from Fig. 3.24 using Eq. (3.3-22). (c) Image formed from all multiplication constants used to produce the enhanced image shown in Fig. 3.26.

Here all the neighborhoods of size 3x3 are used.

Chapter 3: Intensity Transformations and Spatial Filtering

Local Histogram Statistics

~~FIGURE 3.26~~
Enhanced SEM image. Compare with Fig. 3.24. Note in particular the enhanced area on the right side of the image.

The following values are used:

$$C = 4.0,$$

$$k_0 = 0, \qquad k_1 = 0.4,$$

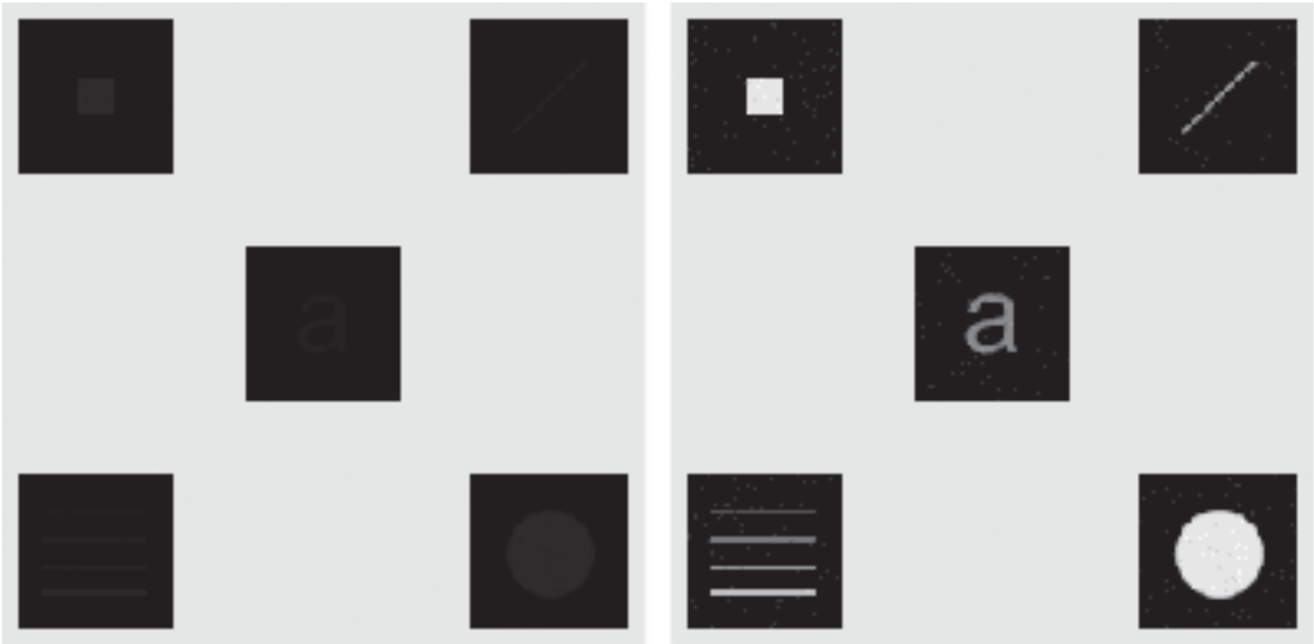
$$k_2 = 0.02, \quad k_3 = 0.4$$

Chapter 3: Intensity Transformations and Spatial Filtering

Local Histogram Statistics

Same technique used to obtain this result

$$C = 22.8,$$
$$k_0 = 0, \qquad k_1 = 0.1,$$
$$k_2 = 0, \qquad k_3 = 0.1$$



a b

FIGURE 3.33 (a) Original image. (b) Result of local enhancement based on local histogram statistics. Compare (b) with Fig. 3.32(c).

Chapter 3: Intensity Transformations and Spatial Filtering

Spatial Filtering Introduction

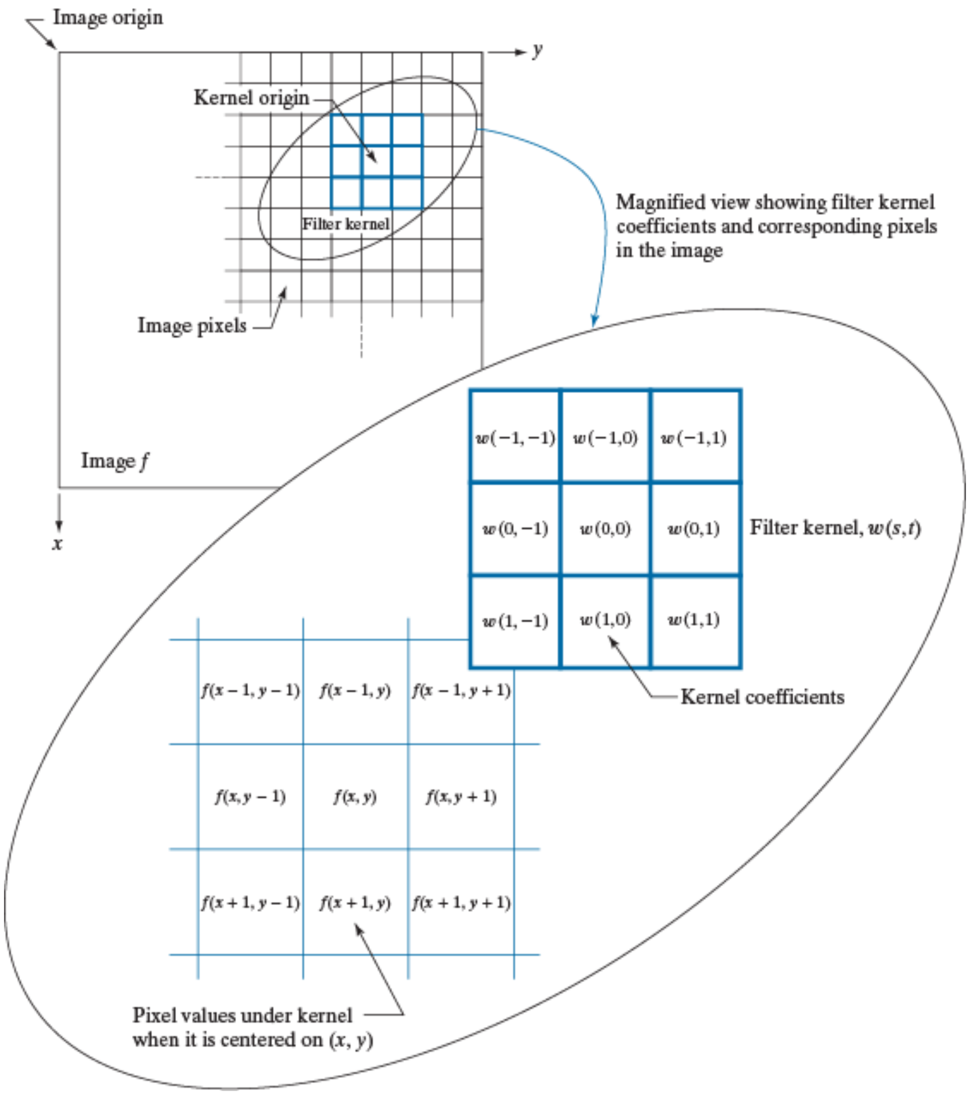


FIGURE 3.34 The mechanics of linear spatial filtering using a 3×3 kernel. The pixels are shown as squares to simplify the graphics. Note that the origin of the image is at the top left, but the origin of the kernel is at its center. Placing the origin at the center of spatially symmetric kernels simplifies writing expressions for linear filtering.

Each pixel is replaced by a weighted sum of its neighbourhood pixels in a sliding window

Chapter 3: Intensity Transformations and Spatial Filtering
Correlation and Convolution

Correlation and convolution are two similar operations with weighted sums on sliding windows.

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

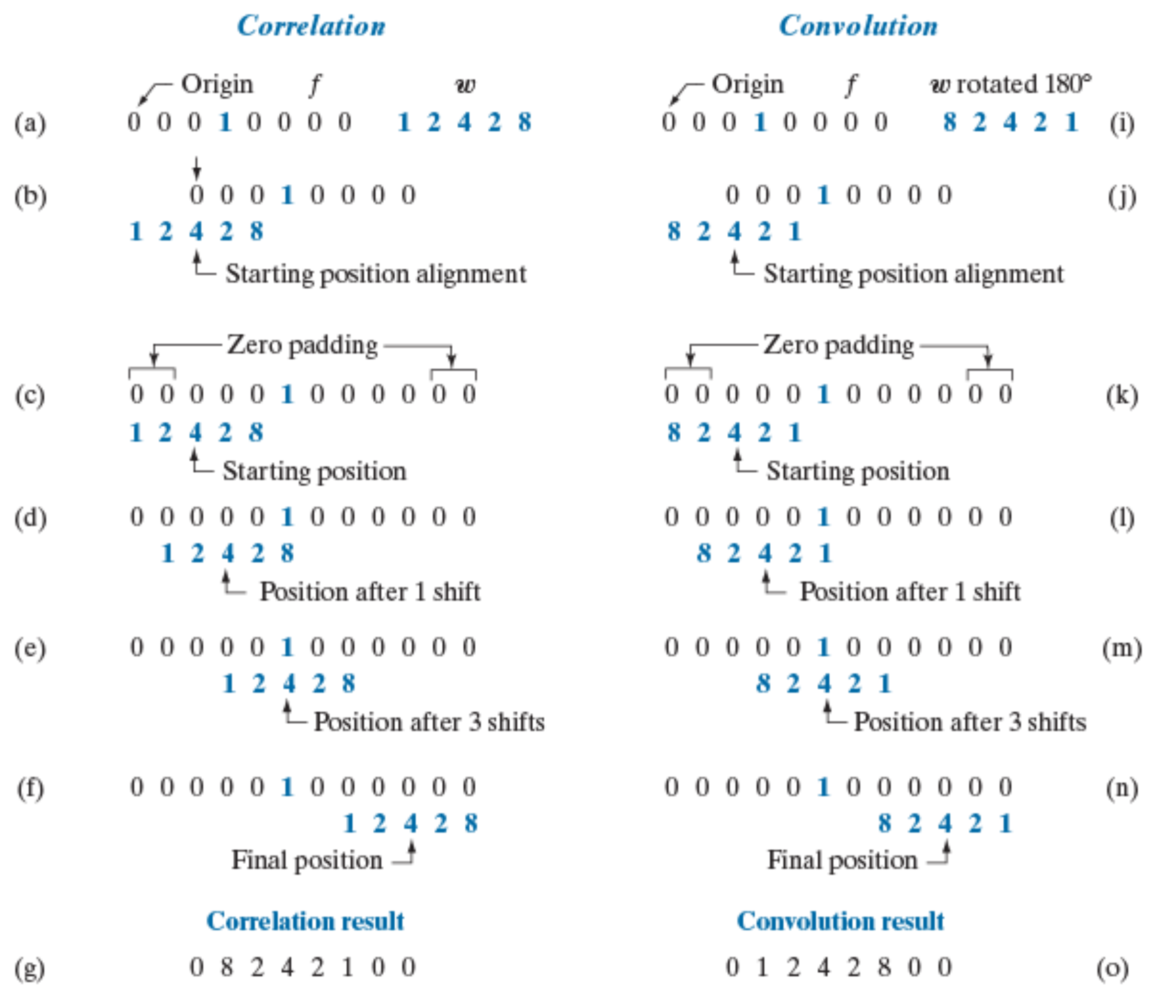
Convolution: $$(w * f)(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)$$

Linear filtering uses convolution unless explicitly said otherwise.

Convolution operates like correlation, but the kernel is rotated 180°.
(no difference for center-symmetric filters)

Chapter 3: Intensity Transformations and Spatial Filtering

Correlation and Convolution in 1-D



Chapter 3: Intensity Transformations and Spatial Filtering

Correlation and Convolution in 2-D

Origin f

0	0	0	0	0
0	0	0	0	0
0	0	**1**	0	0
0	0	0	0	0
0	0	0	0	0

w

1	**2**	**3**
4	**5**	**6**
7	**8**	**9**

(a)

Padded f

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	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

(b)

Initial position for w

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

(c)

Correlation result

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

(d)

Full correlation result

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

(e)

Rotated w

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

(f)

Convolution result

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

(g)

Full convolution result

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

(h)

Difference between Correlation and Convolution

Convolution with an impulse signal produces the filter itself, which does not hold for correlation. The importance of this will be noted in the next chapter.

Properties of correlation and convolution differ.

TABLE 3.5 Some fundamental properties of convolution and correlation. A dash means that the property does not hold.

Property	Convolution	Correlation
Commutative	$f \star g = g \star f$	—
Associative	$f \star (g \star h) = (f \star g) \star h$	—
Distributive	$f \star (g + h) = (f \star g) + (f \star h)$	$f \,☆\, (g + h) = (f \,☆\, g) + (f \,☆\, h)$

Note: the order of linear spatial filters does not matter!

Chapter 3: Intensity Transformations and Spatial Filtering

Examples of Linear Filter Kernels

a b

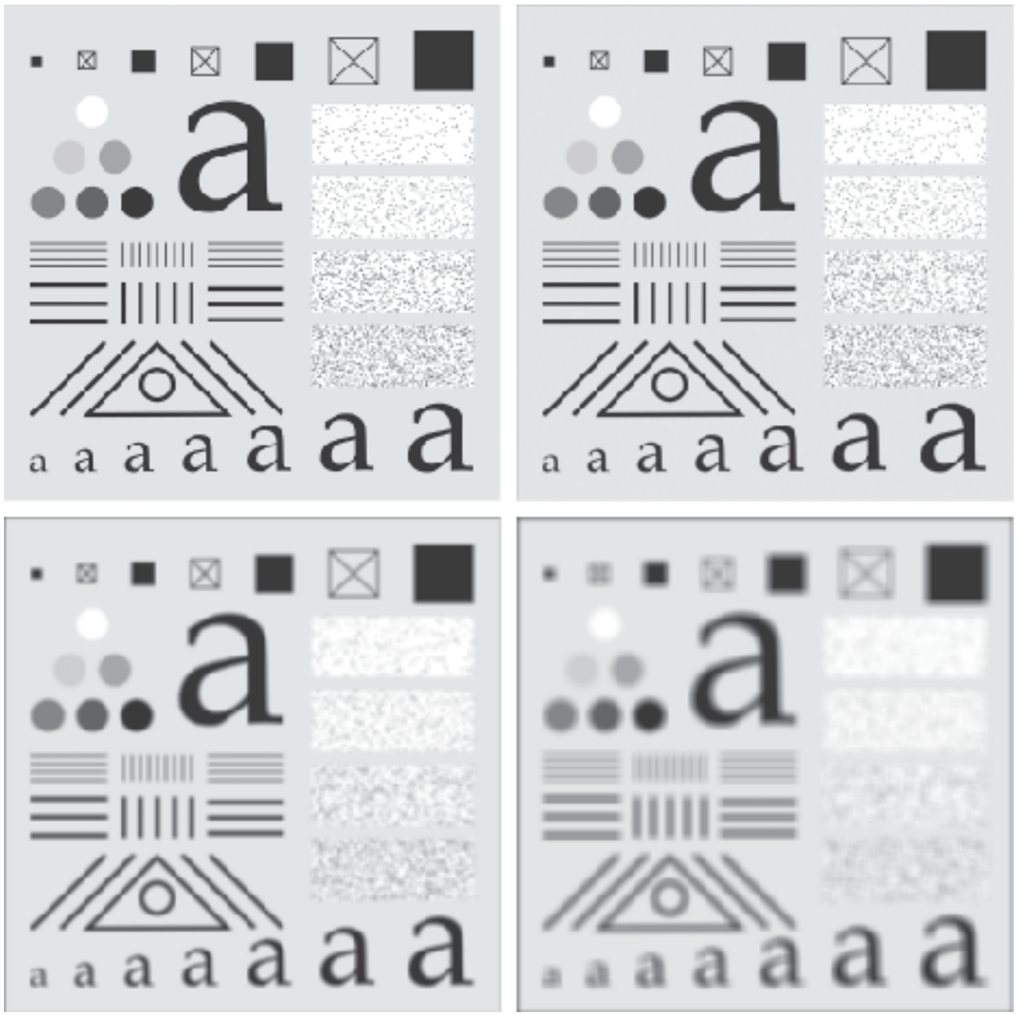
FIGURE 3.37 Examples of smoothing kernels: (a) is a *box* kernel; (b) is a *Gaussian* kernel.

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{4.8976} \times \begin{bmatrix} 0.3679 & 0.6065 & 0.3679 \\ 0.6065 & 1.0000 & 0.6065 \\ 0.3679 & 0.6065 & 0.3679 \end{bmatrix}$$

Note: filter values sum to 1 to preserve average intensity of the original image

Chapter 3: Intensity Transformations and Spatial Filtering

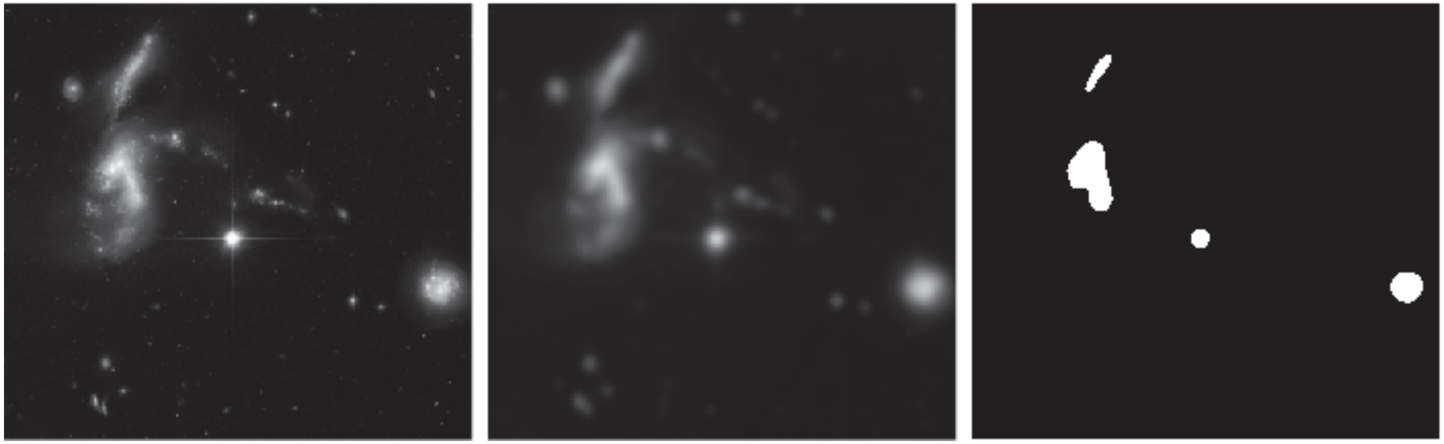
Smoothing (Averaging) Filters



a b
c d

FIGURE 3.39
(a) Test pattern of size 1024×1024 pixels.
(b)-(d) Results of lowpass filtering with box kernels of sizes 3×3, 11×11, and 21×21, respectively.

Chapter 3: Intensity Transformations and Spatial Filtering
Averaging Filter Application



Blobs of the image

a b c

FIGURE 3.47 (a) A 2566 × 2758 Hubble Telescope image of the *Hickson Compact Group*. (b) Result of lowpass filtering with a Gaussian kernel. (c) Result of thresholding the filtered image (intensities were scaled to the range [0, 1]). The Hickson Compact Group contains dwarf galaxies that have come together, setting off thousands of new star clusters. (Original image courtesy of NASA.)

Image from the Hubble space telescope in orbit around the Earth.

Here, we want to blur the image in order to only see large objects.

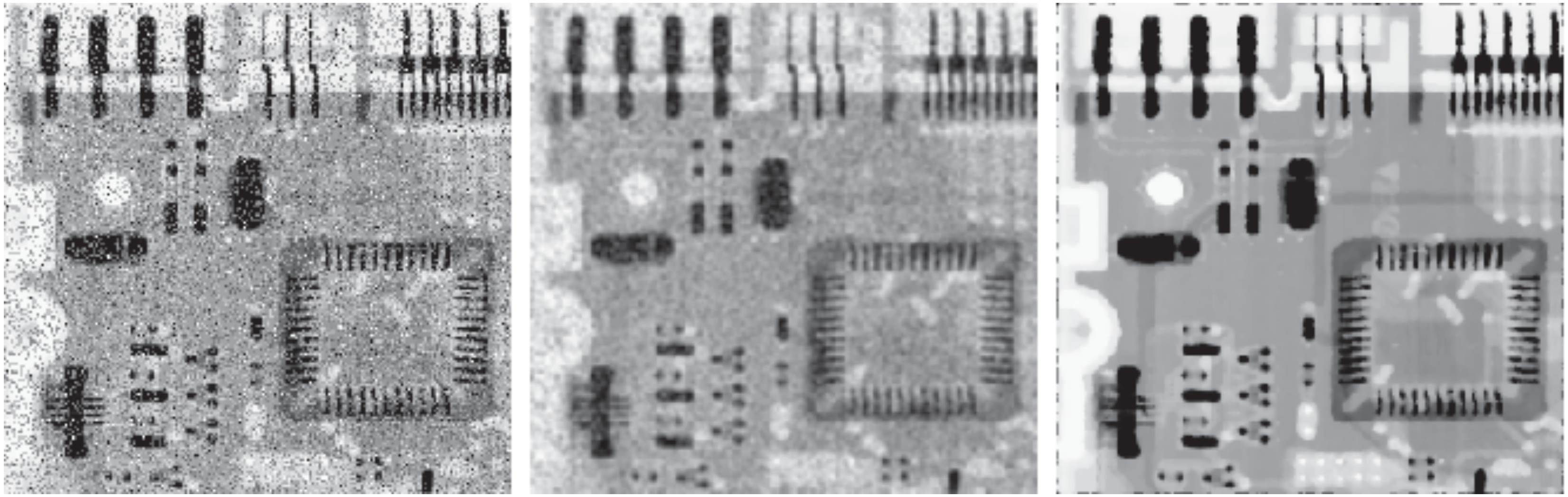
Q: What happens when important details must be preserved or the noise is non-Gaussian?

A: may consider a number of techniques:

- median and order statistics (min, max) filtering,
- sharpening spacial filters
  - directional filtering,
- hybrid combinations

Chapter 3: Intensity Transformations and Spatial Filtering

Nonlinear (Order-Statistic) Filtering



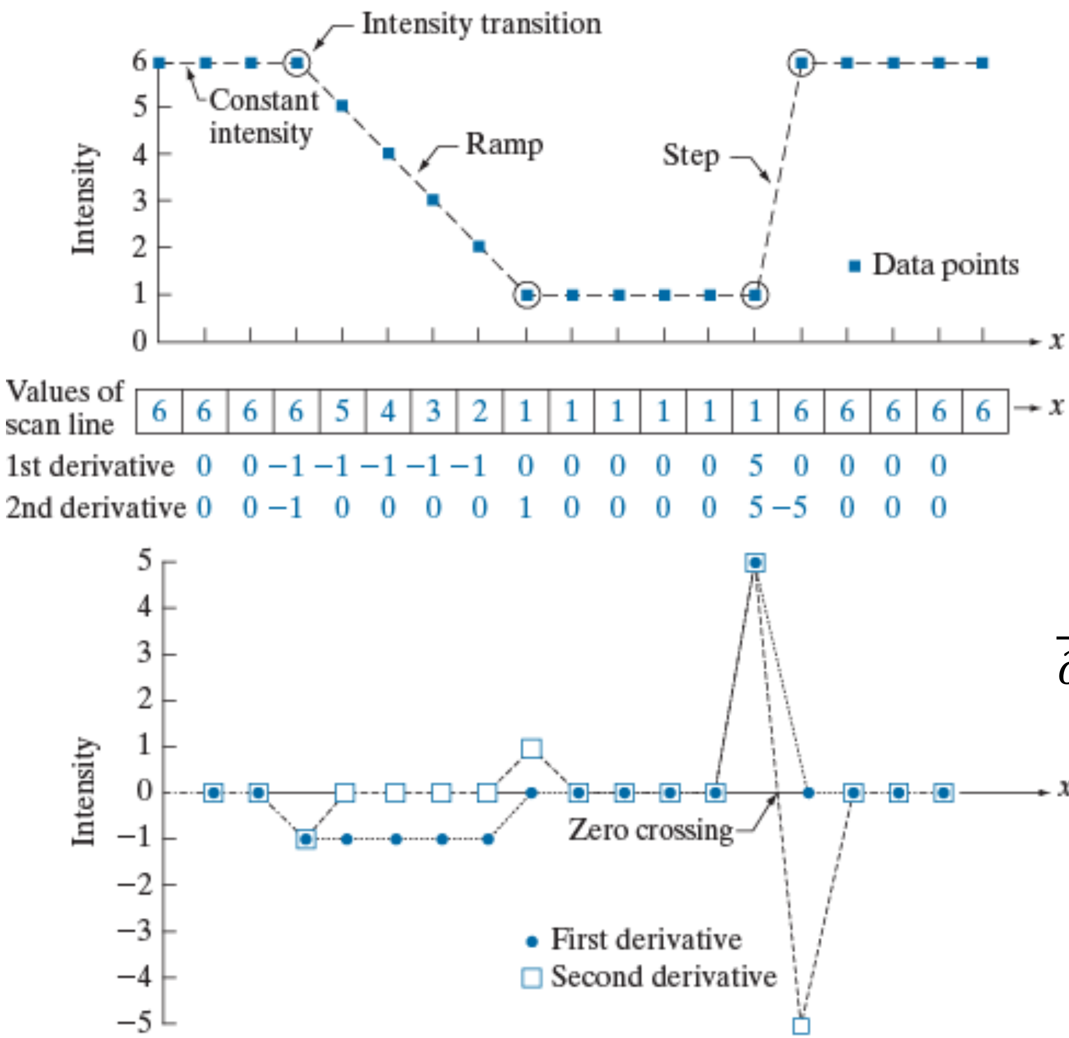
a b c

FIGURE 3.49 (a) X-ray image of a circuit board, corrupted by salt-and-pepper noise. (b) Noise reduction using a 19×19 Gaussian lowpass filter kernel with $\sigma = 3$. (c) Noise reduction using a 7×7 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Median or order statistics filters may perform much better in the presence of *non-Gaussian* noise. (More details in Ch. 5)

Chapter 3: Intensity Transformations and Spatial Filtering

Sharpening Spatial Filters



First derivative:

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

Second derivative:

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

Sharpening Spatial Filters

- **First order derivative**
  - produces thicker edges because the 1st derivative is non-zero along a ”ramp”
  - has strong response to grey-level steps
- **Second order derivative**
  - has a much stronger response to details (e.g. edges and sharp changes)
  - produces a double response at step changes in grey level
  - has a stronger response to a thin line than to a step and to an isolated point than to a thin line.
- **Conclusion**
  - Second derivative is more useful to enhance image fine details (but be careful about noise!!)

Chapter 3: Intensity Transformations and Spatial Filtering

Sharpening Filters: Laplacian

- **Isotropic** 2nd order derivative (Laplacian)

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

- In digital form:

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

in the x-direction and in the y-direction:

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

- Can be implemented using the mask in the next slide

Chapter 3: Intensity Transformations and Spatial Filtering

Sharpening Filters: Laplacian

isotropic for rotations in increments of:

90°

0	1	0
1	−4	1
0	1	0

45°

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

Note: these sum to zero!

negatives of the above two masks

a b
c d

FIGURE ~~**3.39**~~ 3.37 (a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Chapter 3: Intensity Transformations and Spatial Filtering
Sharpening Filters: Laplacian

- Typical use: emphasizing the edges on the image
- Laplacian image is added on top of the original
- Standard enhancement with Laplacian:

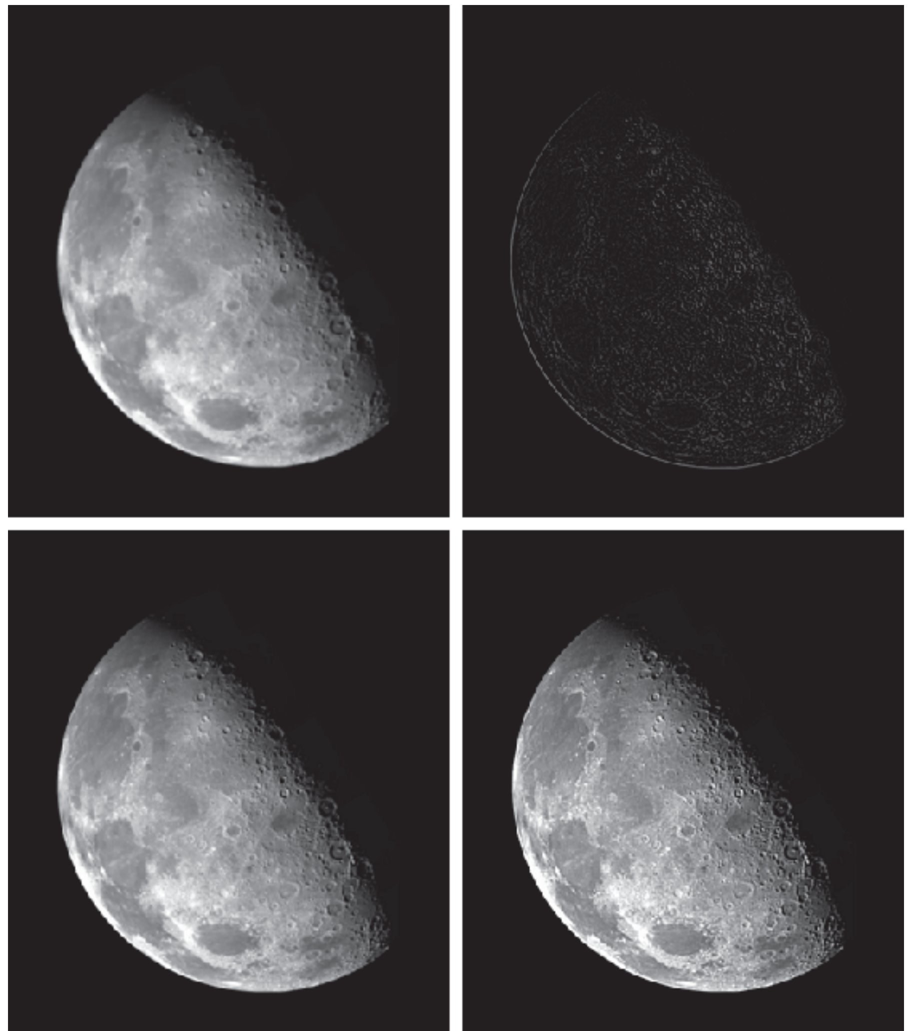
$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) & \textit{if the mask center coeff is negative} \\ f(x,y) + \nabla^2 f(x,y) & \textit{if the mask center coeff is positive} \end{cases}$$

Chapter 3: Intensity Transformations and Spatial Filtering

Laplacian Filtering Example

a b
c d

FIGURE 3.52 (a) Blurred image of the North Pole of the moon. (b) Laplacian image obtained using the kernel in Fig. 3.51(a). (c) Image sharpened using Eq. (3-63) with $c = -1$. (d) Image sharpened using the same procedure, but with the kernel in Fig. 3.51(b). (Original image courtesy of NASA.)



Laplacian image using the 90^{o} mask

Enhanced image using the 90^{o} mask

Enhanced image using the 45^{o} mask

Chapter 3: Intensity Transformations and Spatial Filtering

Composite Mask for Laplacian Filtering

- The previous expression :

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

- The two terms in the summation can be combined and thus simplified:

$$g(x,y)=5f(x,y)-[f(x+1,y)+f(x-1,y)+f(x,y+1)+f(x,y-1)]$$

- This operation can be implemented using the mask in the next slide.

Chapter 3: Intensity Transformations and Spatial Filtering

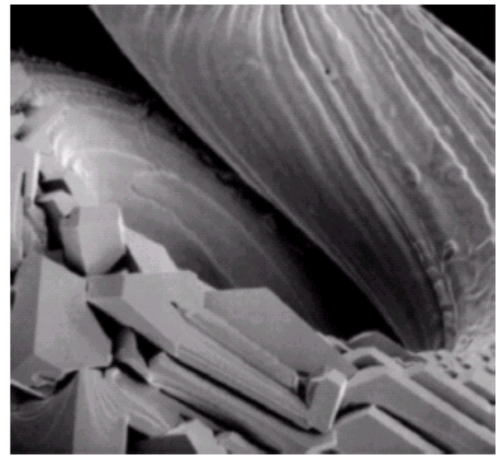
Composite Mask for Laplacian Filtering

0	−1	0
−1	5	−1
0	−1	0

90°

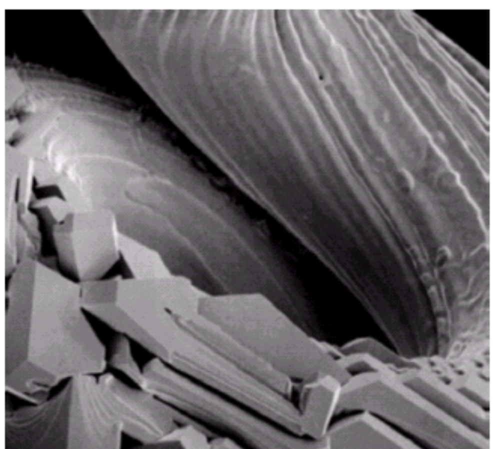
−1	−1	−1
−1	9	−1
−1	−1	−1

45°



SEM image

results for the 90°



results for the 45°

Notice how much sharper it is!

a b c
d e

~~FIGURE 3.41~~ (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Chapter 3: Intensity Transformations and Spatial Filtering

Laplacian with High-Boost Filtering

0	−1	0
−1	$A + 4$	−1
0	−1	0

−1	−1	−1
−1	$A + 8$	−1
−1	−1	−1

a b

~~**FIGURE 3.42**~~ The high-boost filtering technique can be implemented with either one of these masks, with $A \geq 1$.

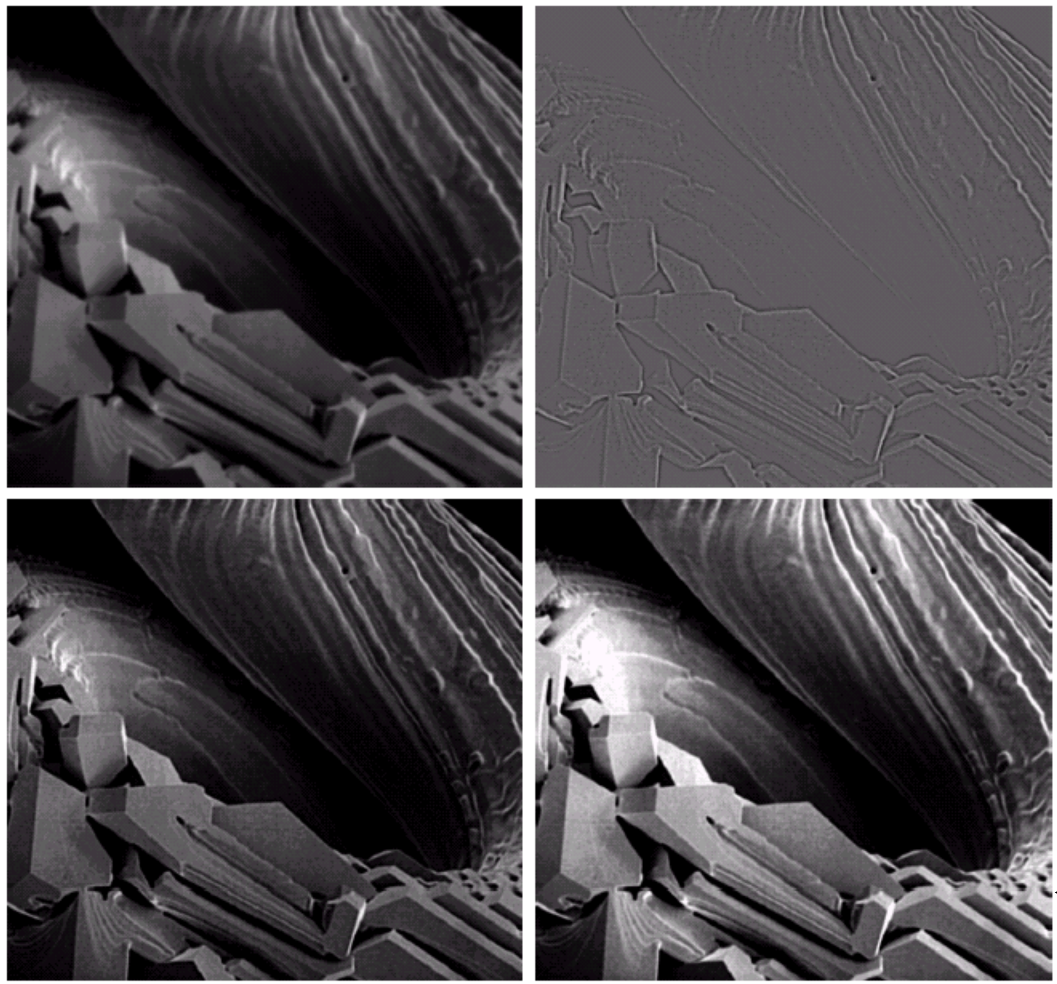
This mask can give better results if the original image is darker than desired, see example in next slide.

Chapter 3: Intensity Transformations and Spatial Filtering

Laplacian with High-Boost Filtering

a	b
c	d

~~FIGURE 3.43~~
(a) Same as Fig. 3.41(c), but darker.
(a) Laplacian of (a) computed with the mask in Fig. 3.42(b) using $A = 0$.
(c) Laplacian enhanced image using the mask in Fig. 3.42(b) with $A = 1$. (d) Same as (c), but using $A = 1.7$.



$$\begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix}$$

$$\begin{matrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{matrix}$$

$$\begin{matrix} -1 & -1 & -1 \\ -1 & 9.7 & -1 \\ -1 & -1 & -1 \end{matrix}$$

Sharpening Filters with First-Order Derivatives

- First derivatives are implemented using the magnitude of the gradient. The gradient is given by:

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

- The magnitude of the gradient is given by:

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

Note that components of the gradient are linear operators, but the magnitude is not! Also the partial derivatives are not isotropic, but the magnitude is!

- The magnitude can be (for faster implementation) approximated by:

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

Chapter 3: Intensity Transformations and Spatial Filtering

Sharpening Filters with First-Order Derivatives

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

f(x,y)

$$\nabla f = [(z_8 - z_5)^2 + (z_6 - z_5)^2]^{\frac{1}{2}}$$

−1	0
0	1

0	−1
1	0

Roberts cross-gradient operator

$$\nabla f \approx |z_9 - z_5| + |z_8 - z_6|$$

−1	−2	−1
0	0	0
1	2	1

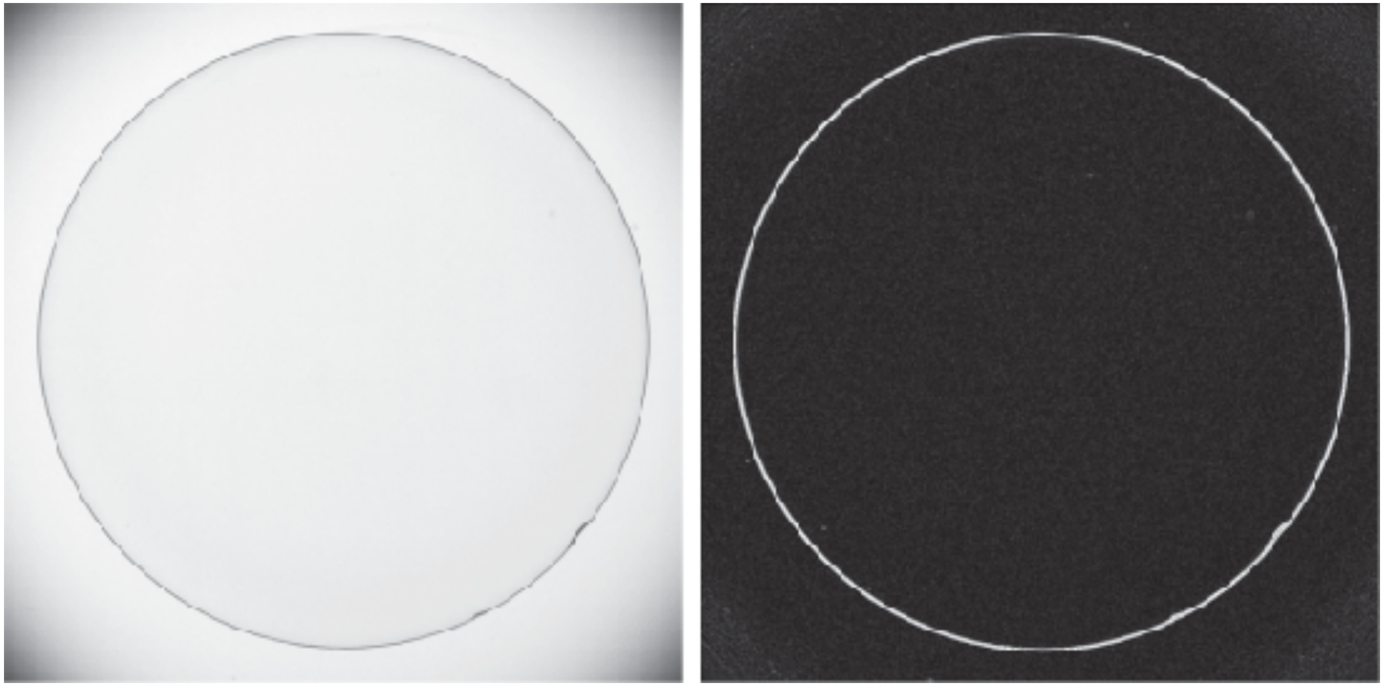
−1	0	1
−2	0	2
−1	0	1

Sobel operators

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

Chapter 3: Intensity Transformations and Spatial Filtering

Gradient Sharpening Filter Example



a b

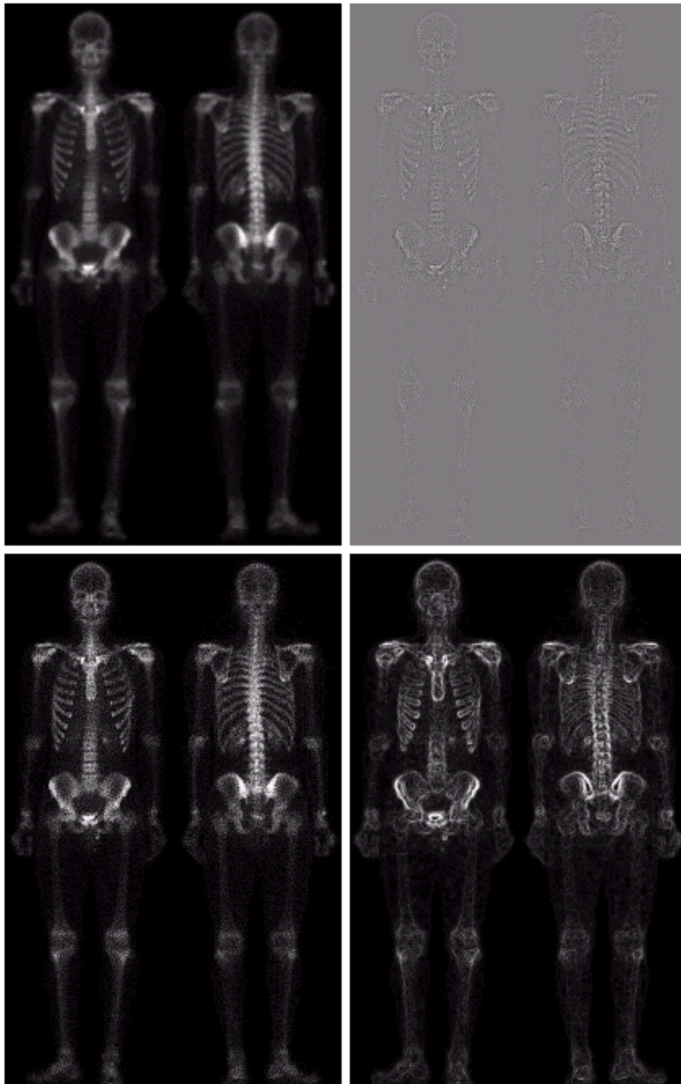
FIGURE 3.57 (a) Image of a contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Perceptics Corporation.)

Chapter 3: Intensity Transformations and Spatial Filtering

Combining Spatial Enhancement Methods

Original

Laplacian of original



Sobel of orig

original+ Laplacian

FIGURE 3.63 (a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel gradient of image (a). (Original image courtesy of G.E. Medical Systems.)

Problem:

Nuclear whole body scan want to detect diseases, e.g. bone infection and tumors

Challenge:

•Single technique may not produce desirable results

•Must thus devise a strategy for the given application at hand

Strategy:

•use Laplacian to highlight details,

•gradient to enhance edges,

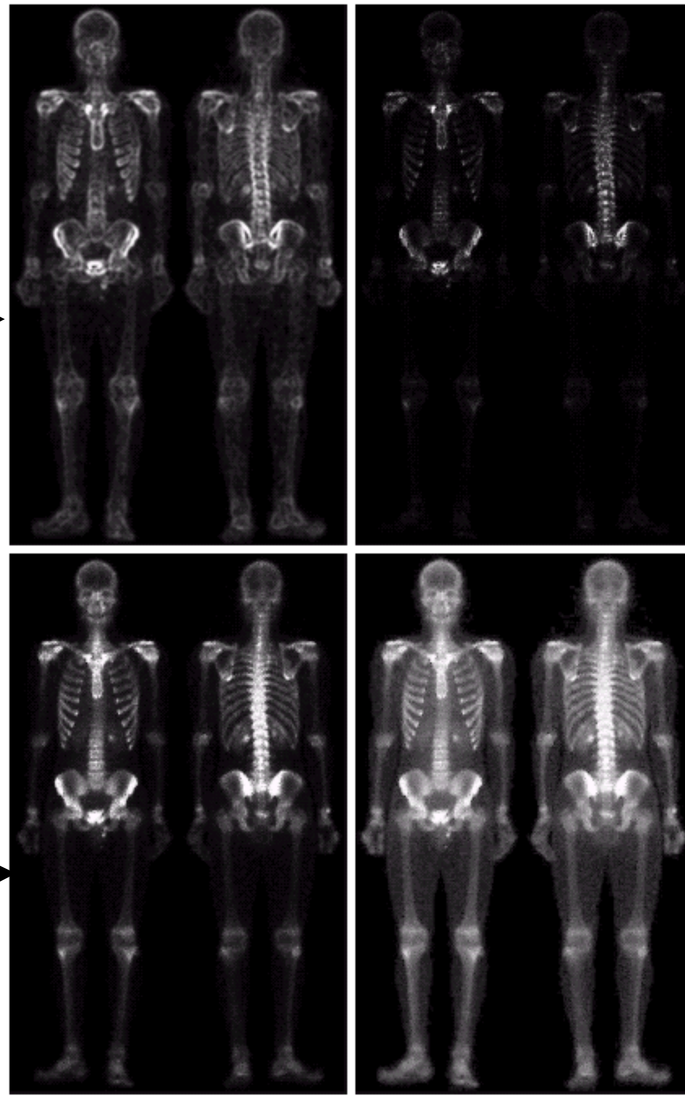
•grey-level transformation to increase dynamic range

Chapter 3: Intensity Transformations and Spatial Filtering

Combining Spatial Enhancement Methods

smoothed Sobel

orig + mask



Mask obtained by:
(orig + Lap) x (smoothed Sobel)

e f
g h

FIGURE 3.63 *(Continued)* (e) Sobel image smoothed with a 5 × 5 box filter. (f) Mask image formed by the product of (b) and (e). (g) Sharpened image obtained by the adding images (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare images (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

Final result power law to orig. + mask