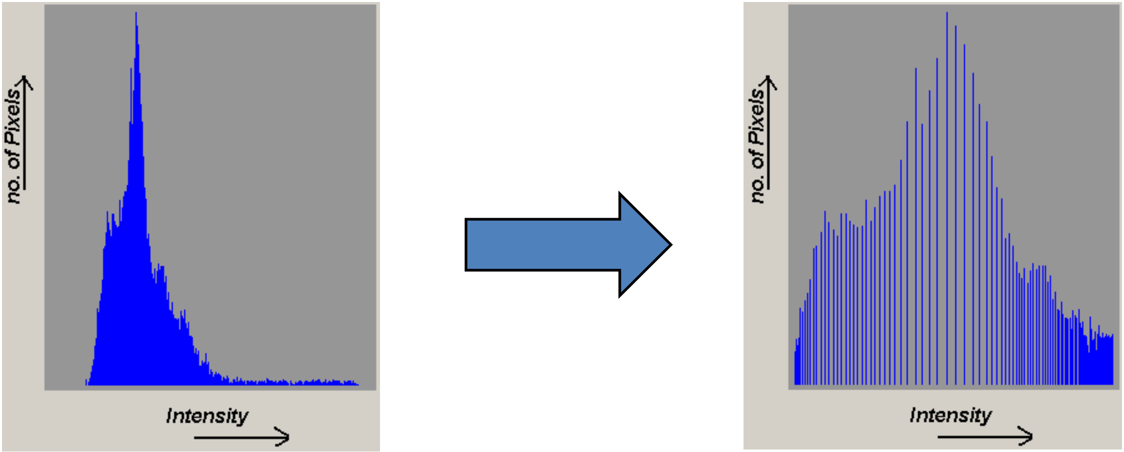
Digital Image Processing

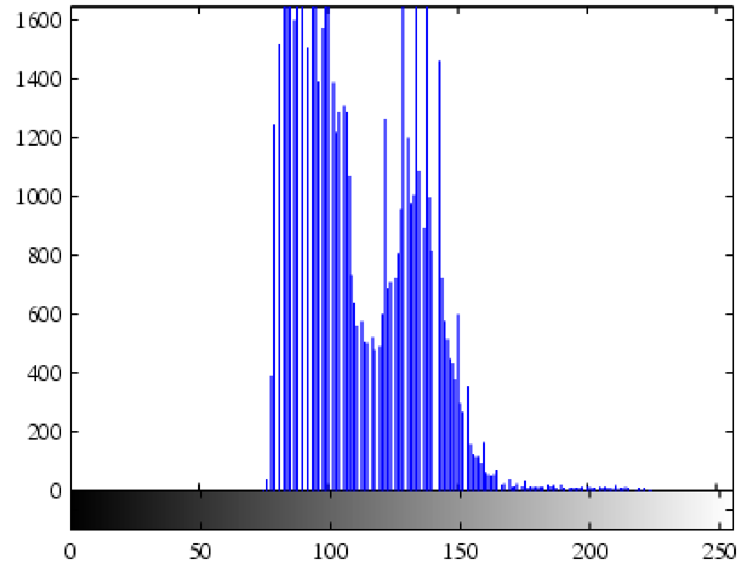
Lecture # 4
Image Enhancement (Histogram & Spatial Filtering)

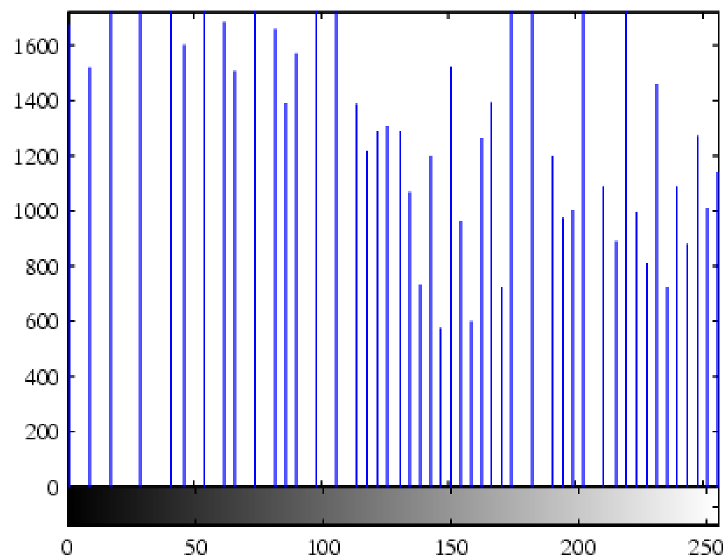
Histogram Equalization

Histogram equalization re-assigns the intensity values of pixels in the input image such that the output image contains a uniform distribution of intensities

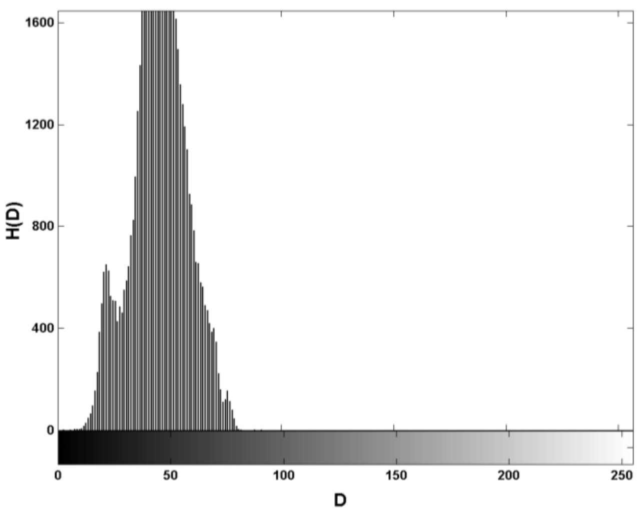


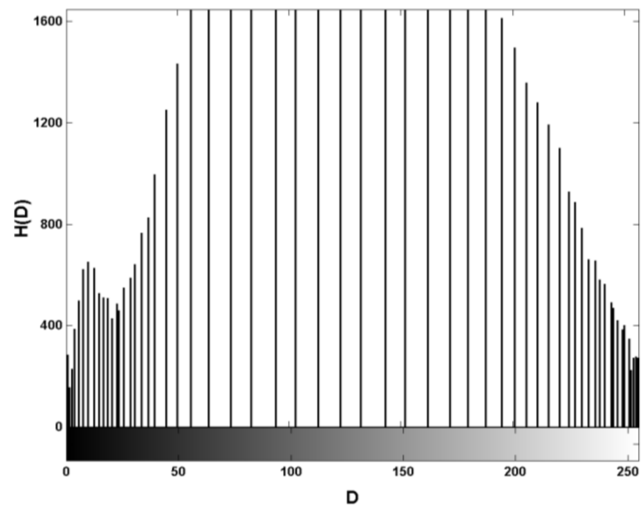
HISTOGRAM EQUALIZATION





AERIAL PHOTOGRAPH OF THE PENTAGON





Resulting image uses more of dynamic range.
Resulting histogram almost, but not completely, flat.

The Probability Distribution Function of an Image

Let $A = \sum_{g=0}^{255} h_I(g)$

Note that since $h_I(g)$ is the number of pixels in I with value g,

A is the number of pixels in I. That is if I is R rows by C columns then $A = R \times C$.

Then,

$$p_I(g) = \frac{1}{A} h_I(g)$$

This is the probability that an arbitrary pixel from *I* has value *g*.

The Probability Distribution Function of an Image

- $p(g)$ is the fraction of pixels in an image that have intensity value g.
- $p(g)$ is the probability that a pixel randomly selected from the given image has intensity value g.
- Whereas the sum of the histogram $h(g)$ over all g from 0 to 255 is equal to the number of pixels in the image, the sum of $p(g)$ over all g is 1.
- p is the normalized histogram of the image

The Cumulative Distribution Function of an Image

Let **q** = *I*(*r*,*c*) be the value of a randomly selected pixel from *I*. Let *g* be a specific gray level. The probability that q ≤ g is given by

$$P_I(g) = \sum_{\gamma=0}^{g} p_I(\gamma) = \frac{1}{A}\sum_{\gamma=0}^{g} h_I(\gamma) = \frac{\sum_{\gamma=0}^{g} h_I(\gamma)}{\sum_{\gamma=0}^{255} h_I(\gamma)},$$

where $h_I(\gamma)$ is the histogram of image *I*.

This is the probability that any given pixel from *I* has value less than or equal to g.

The Cumulative Distribution Function of an Image

Let **q** = $I(r,c)$ be the value of a randomly selected pixel from I. Let g be a specific gray level. The probability that q ≤ g is given by

$$P_I(g) = \sum_{\gamma=0}^{g} p_I(\gamma) = \frac{1}{A}\sum_{\gamma=0}^{g} h_I(\gamma) = \frac{\sum_{\gamma=0}^{g} h_I(\gamma)}{\sum_{\gamma=0}^{255} h_I(\gamma)},$$

where $h_I(\gamma)$ is the histogram of image I.

Also called CDF for "Cumulative Distribution Function".

This is the probability that any given pixel from *I* has value less than or equal to g.

The Cumulative Distribution Function of an Image

- $P(g)$ is the fraction of pixels in an image that have intensity values less than or equal to g.
- $P(g)$ is the probability that a pixel randomly selected from the given band has an intensity value less than or equal to g.
- $P(g)$ is the cumulative (or running) sum of $p(g)$ from 0 through g inclusive.
- $P(0) = p(0)$ and $P(255) = 1$;

Histogram Equalization

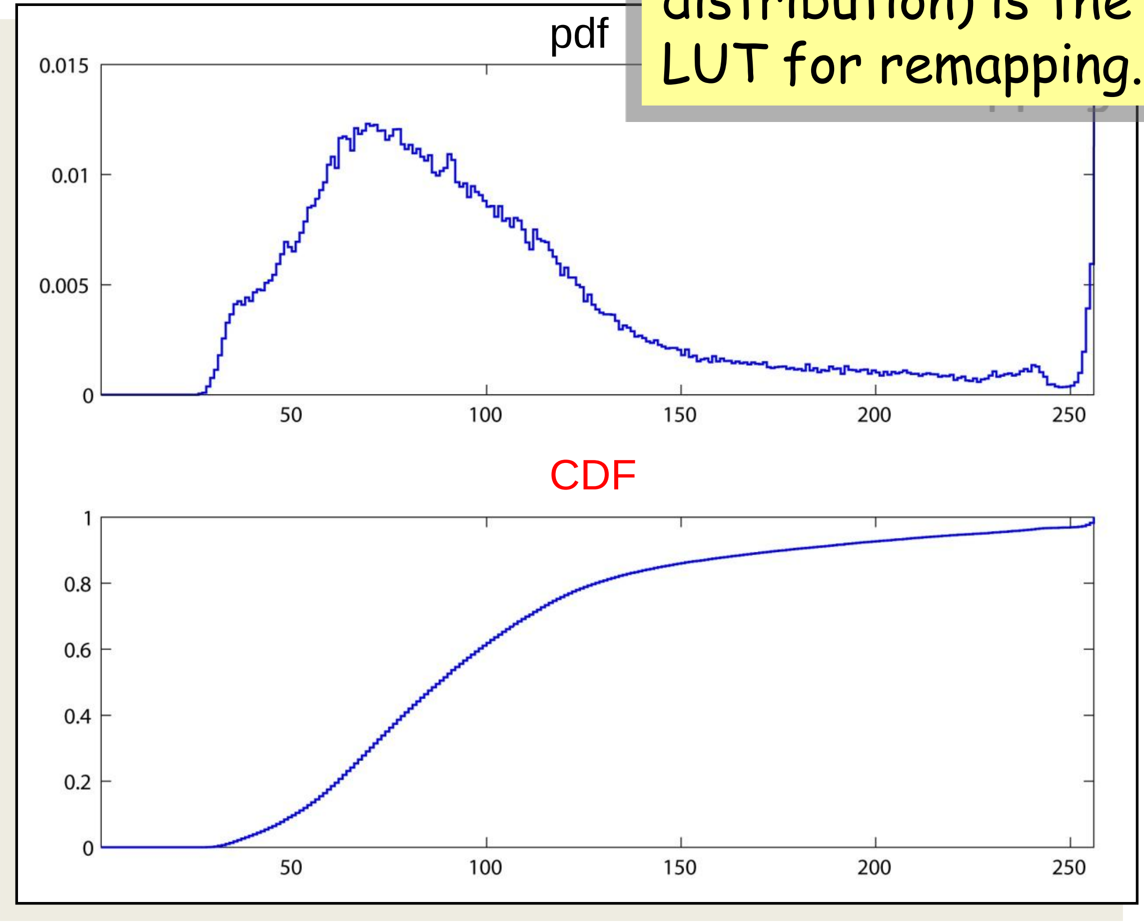
Task: remap image I so that its histogram is as close to constant as possible

Let $P_I(\gamma)$

be the cumulative (probability) distribution function of I.

The CDF itself is used as the LUT.

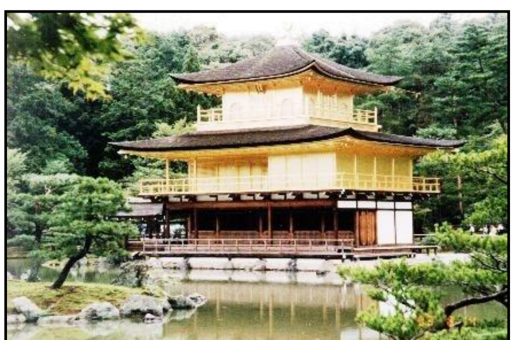
Histogram Equalization

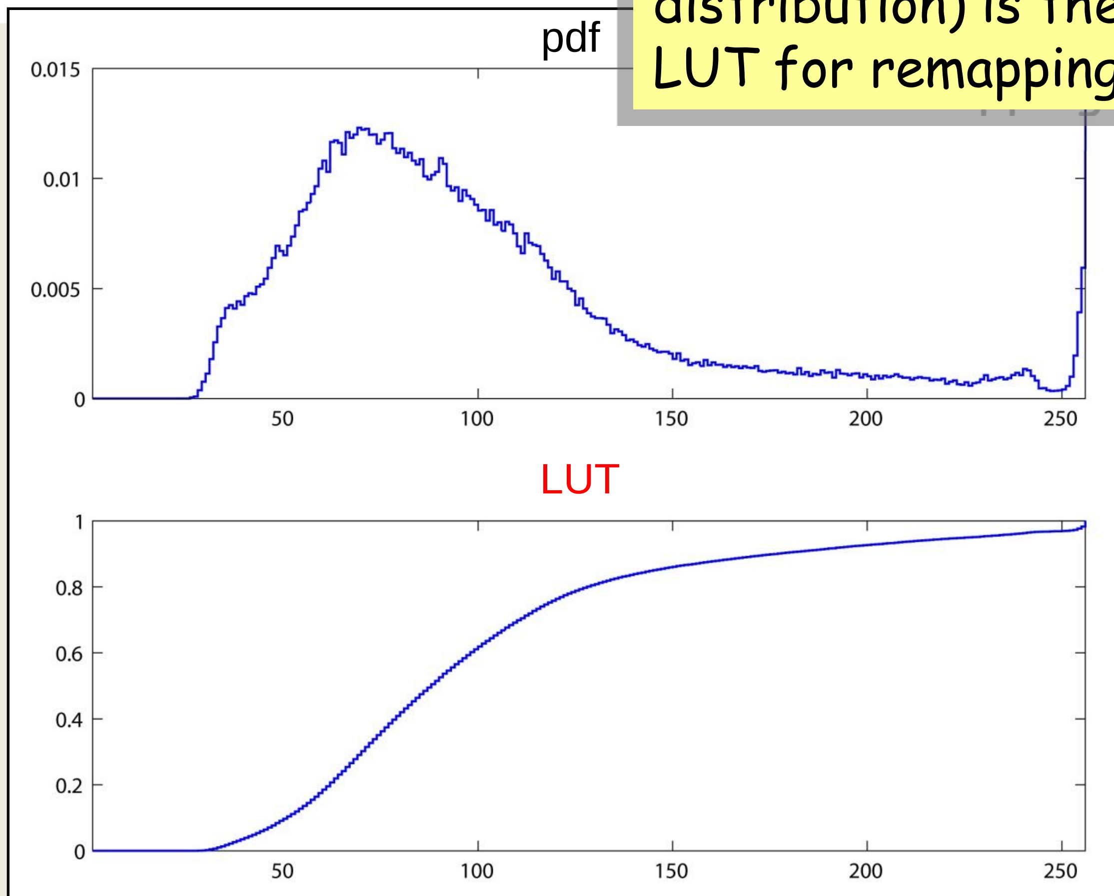


The CDF (cumulative distribution) is the LUT for remapping.

Histogram Equalization

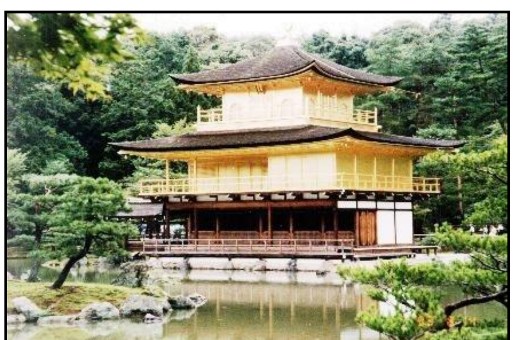
The CDF (cumulative distribution) is the LUT for remapping.

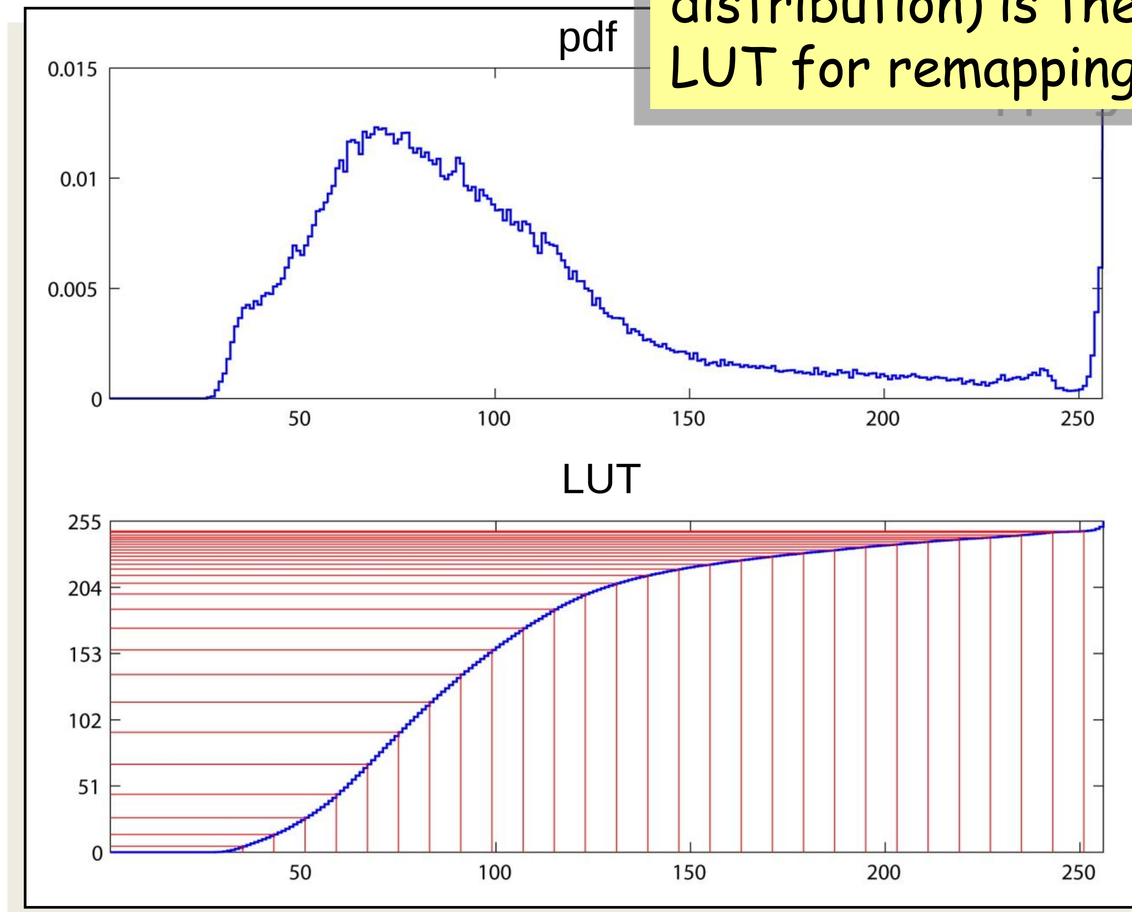




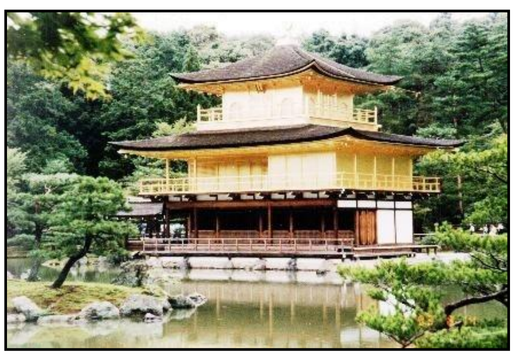
Histogram Equalization

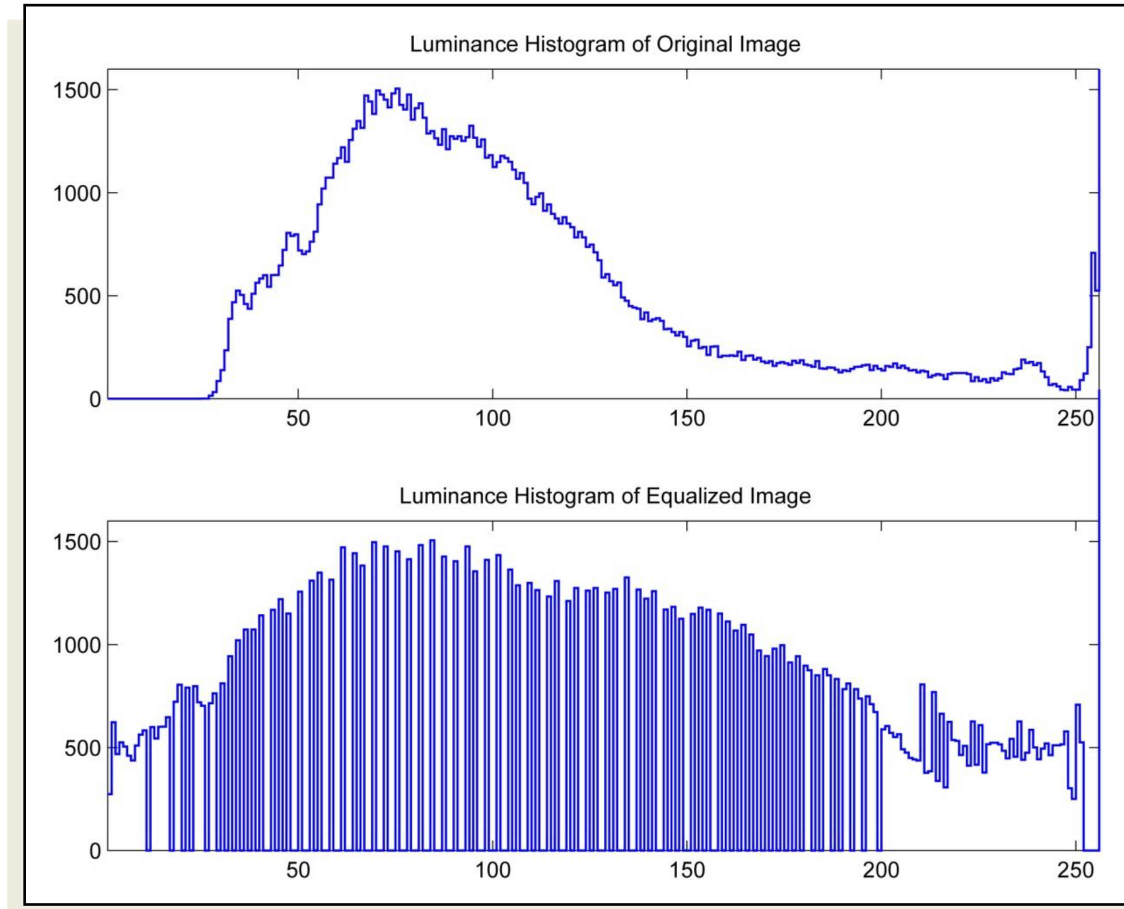
The CDF (cumulative distribution) is the LUT for remapping.



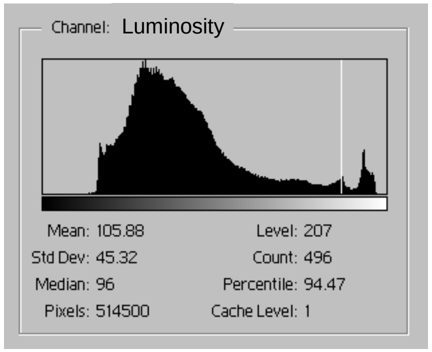


Histogram Equalization

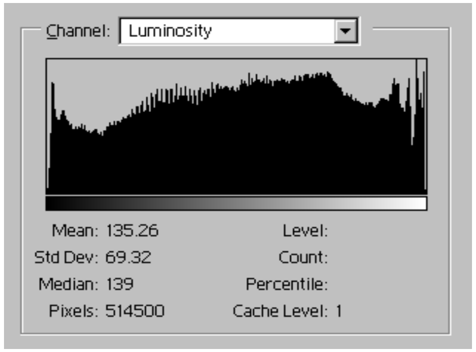




Histogram Equalization



before



after

$$J(r,c) = 255 \cdot P_I\left[I(r,c)\right].$$

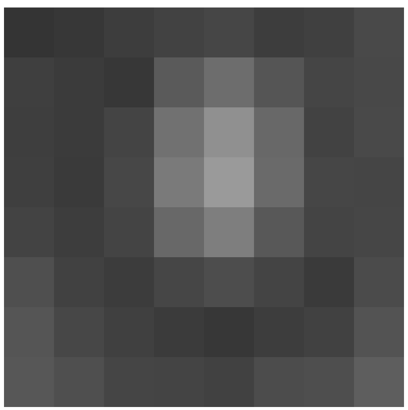
HISTOGRAM EQUALIZATION IMPLEMENTATION

0	0	0	0	0
1	1	1	1	4
4	5	6	6	6
8	8	8	8	9

2	2	2	2	2
4	4	4	4	5
5	5	7	7	7
9	9	9	9	9

Gray levels	0	1	2	3	4	5	6	7	8	9
Counts ($h(r_k)$)	5	4	0	0	2	1	3	0	4	1
	r_0	r_1			r_2	r_3	r_4		r_5	r_6
Normalized h ($P(r_k)$)	5/20	4/20	0	0	2/20	1/20	3/20	0	4/20	1/20
cdf $F(r_k)$	5/20	9/20			**11/20**	**12/20**	**15/20**		**19/20**	**20/20**
$s_k = \text{round}(9 \cdot F(r_k))$	2	4			5	5	7		9	9
	s_0	s_1			s_2	s_3	s_4		s_5	s_6

Histogram Equalization: Example



$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

An 8x8 image

Histogram Equalization: Example

Fill in the following table/histogram

Value	Count	Value	Count	Value	Count	Value	Count	Value	Count
52		64		72		85		113	
55		65		73		87		122	
58		66		75		88		126	
59		67		76		90		144	
60		68		77		94		154	
61		69		78		104			
62		70		79		106			
63		71		83		109			

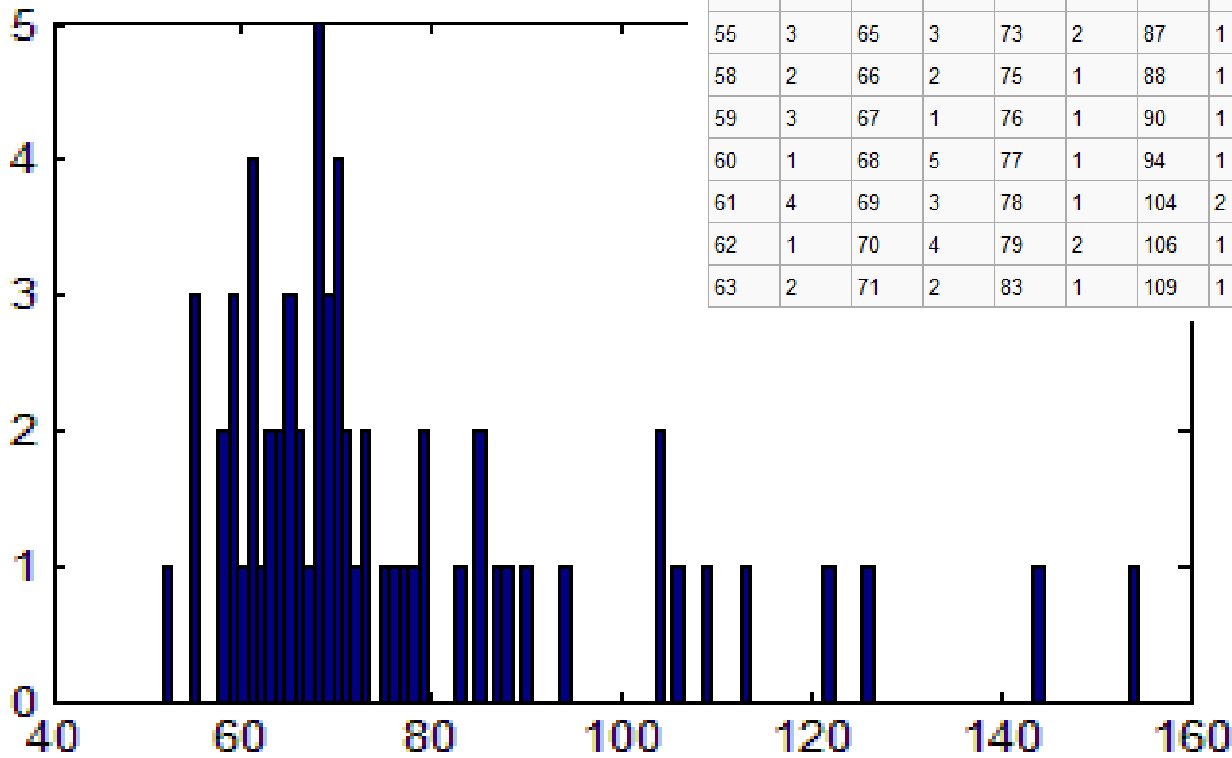
Image Histogram (Non-zero values)

Histogram Equalization: Example

Image Histogram (Non-zero values shown)

Value	Count	Value	Count	Value	Count	Value	Count	Value	Count
52	1	64	2	72	1	85	2	113	1
55	3	65	3	73	2	87	1	122	1
58	2	66	2	75	1	88	1	126	1
59	3	67	1	76	1	90	1	144	1
60	1	68	5	77	1	94	1	154	1
61	4	69	3	78	1	104	2		
62	1	70	4	79	2	106	1		
63	2	71	2	83	1	109	1		

Histogram Equalization: Example



Value	Count	Value	Count	Value	Count	Value	Count	Value	Count
52	1	64	2	72	1	85	2	113	1
55	3	65	3	73	2	87	1	122	1
58	2	66	2	75	1	88	1	126	1
59	3	67	1	76	1	90	1	144	1
60	1	68	5	77	1	94	1	154	1
61	4	69	3	78	1	104	2		
62	1	70	4	79	2	106	1		
63	2	71	2	83	1	109	1		

Histogram Equalization: Example

Image Histogram/Prob Mass Function

Value	Count	Value	Count	Value	Count	Value	Count	Value	Count
52	1	64	2	72	1	85	2	113	1
55	3	65	3	73	2	87	1	122	1
58	2	66	2	75	1	88	1	126	1
59	3	67	1	76	1	90	1	144	1
60	1	68	5	77	1	94	1	154	1
61	4	69	3	78	1	104	2		
62	1	70	4	79	2	106	1		
63	2	71	2	83	1	109	1		

Cumulative Distribution Function (cdf)

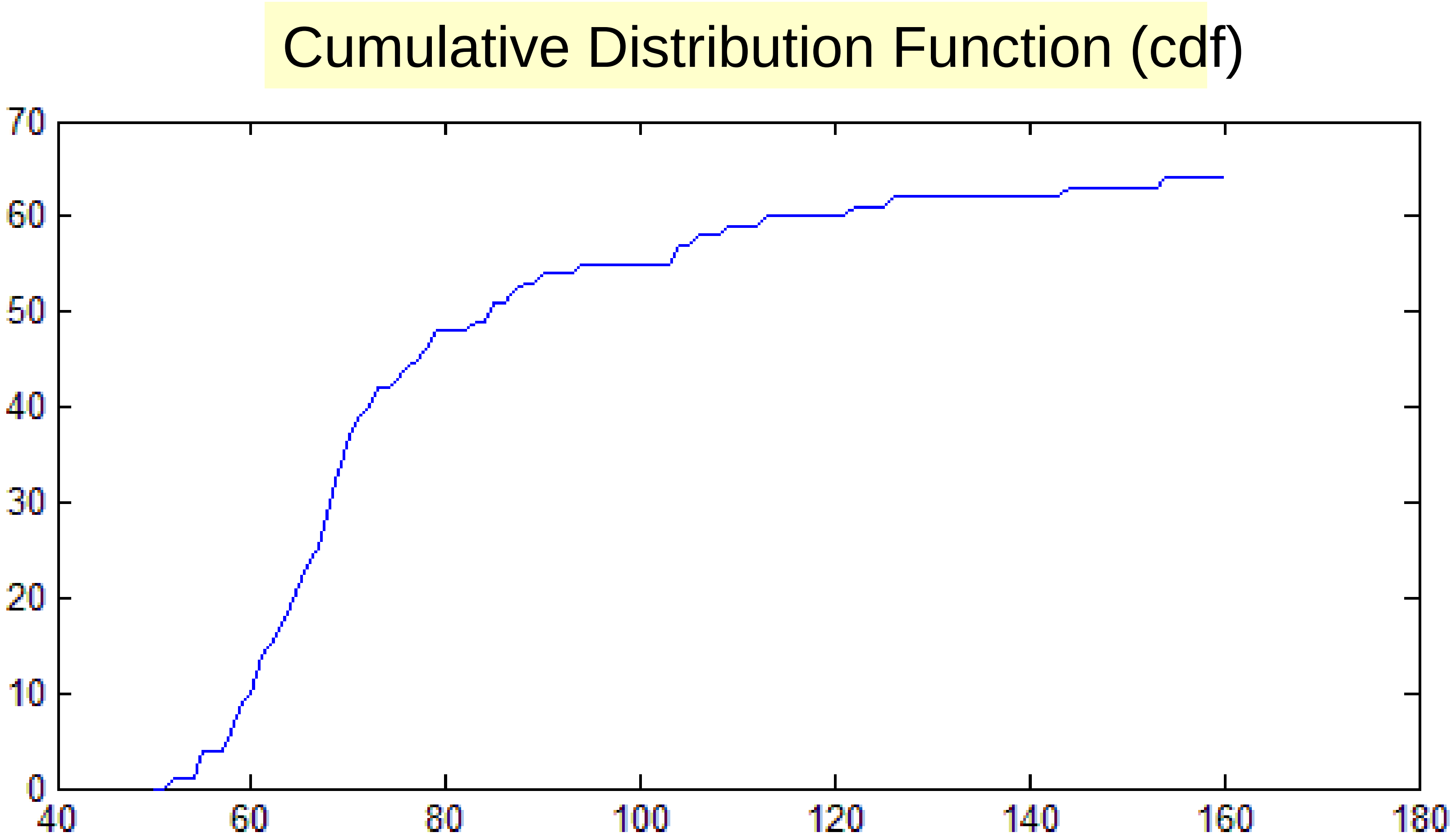
Value	cdf	Value	cdf	Value	cdf	Value	cdf	Value	cdf
52		64		72		85		113	
55		65		73		87		122	
58		66		75		88		126	
59		67		76		90		144	
60		68		77		94		154	
61		69		78		104			
62		70		79		106			
63		71		83		109			

Histogram Equalization: Example

Cumulative Distribution Function (cdf)

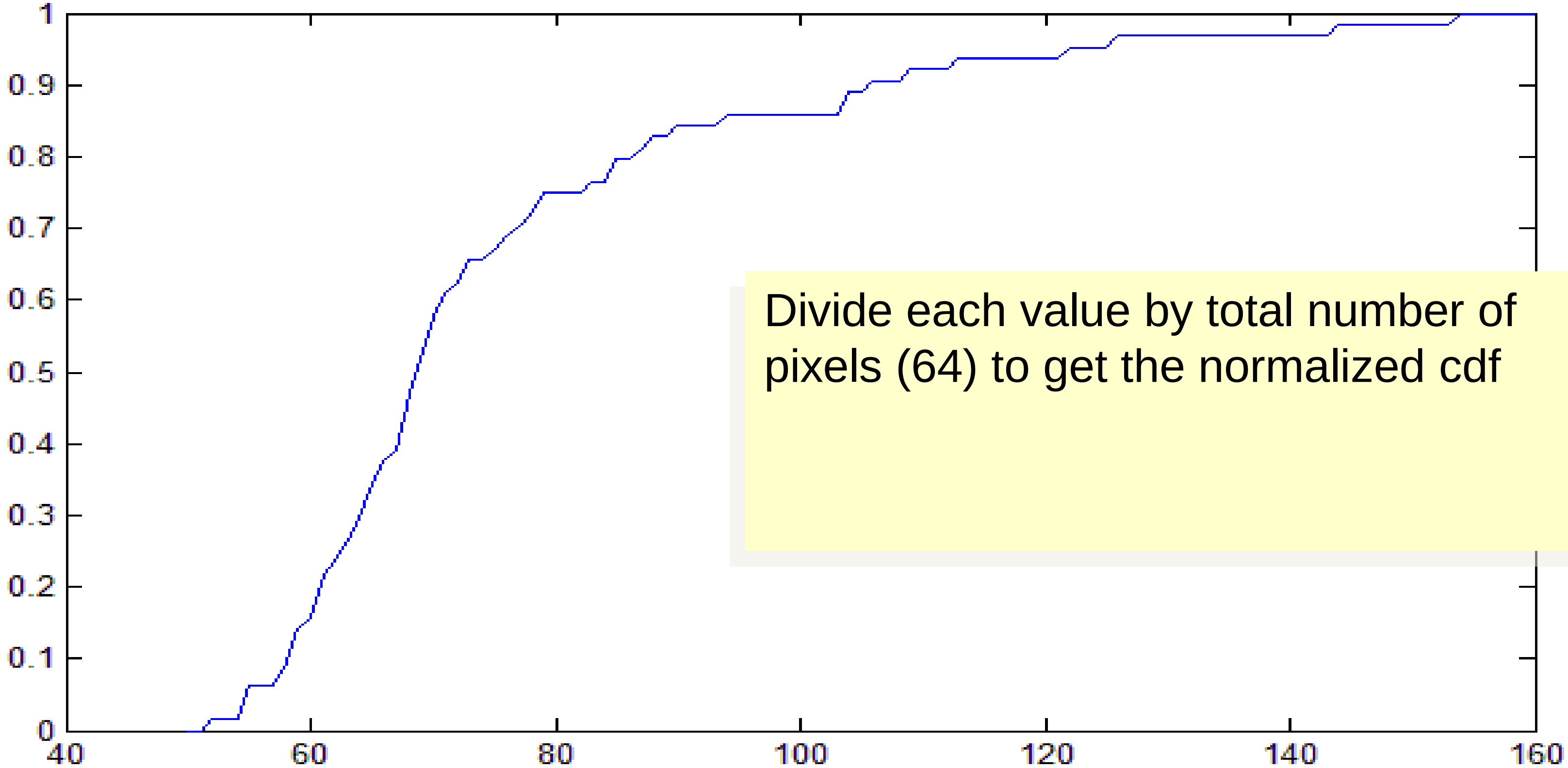
Value	cdf	Value	cdf	Value	cdf	Value	cdf	Value	cdf
52	1	64	19	72	40	85	51	113	60
55	4	65	22	73	42	87	52	122	61
58	6	66	24	75	43	88	53	126	62
59	9	67	25	76	44	90	54	144	63
60	10	68	30	77	45	94	55	154	64
61	14	69	33	78	46	104	57		
62	15	70	37	79	48	106	58		
63	17	71	39	83	49	109	59		

Histogram Equalization: Example



Histogram Equalization: Example

Normalized Cumulative Distribution Function (cdf)



Divide each value by total number of pixels (64) to get the normalized cdf

Histogram Equalization: Example

Value	cdf	Value	cdf	Value	cdf	Value	cdf	Value	cdf
52	1	64	19	72	40	85	51	113	60
55	4	65	22	73	42	87	52	122	61
58	6	66	24	75	43	88	53	126	62
59	9	67	25	76	44	90	54	144	63
60	10	68	30	77	45	94	55	154	64
61	14	69	33	78	46	104	57		
62	15	70	37	79	48	106	58		
63	17	71	39	83	49	109	59		

$$J(r,c) = 255 \cdot P_I\left[I(r,c)\right].$$

If cdf is normalized

$$s = round(255.cdf(r))$$

If cdf is NOT normalized

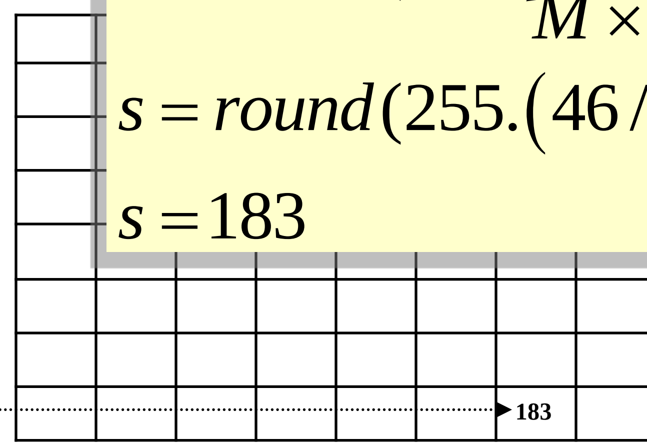
$$s = round(255.\frac{cdf(r)}{M \times N})$$

$$s = round(255.(46/64))$$

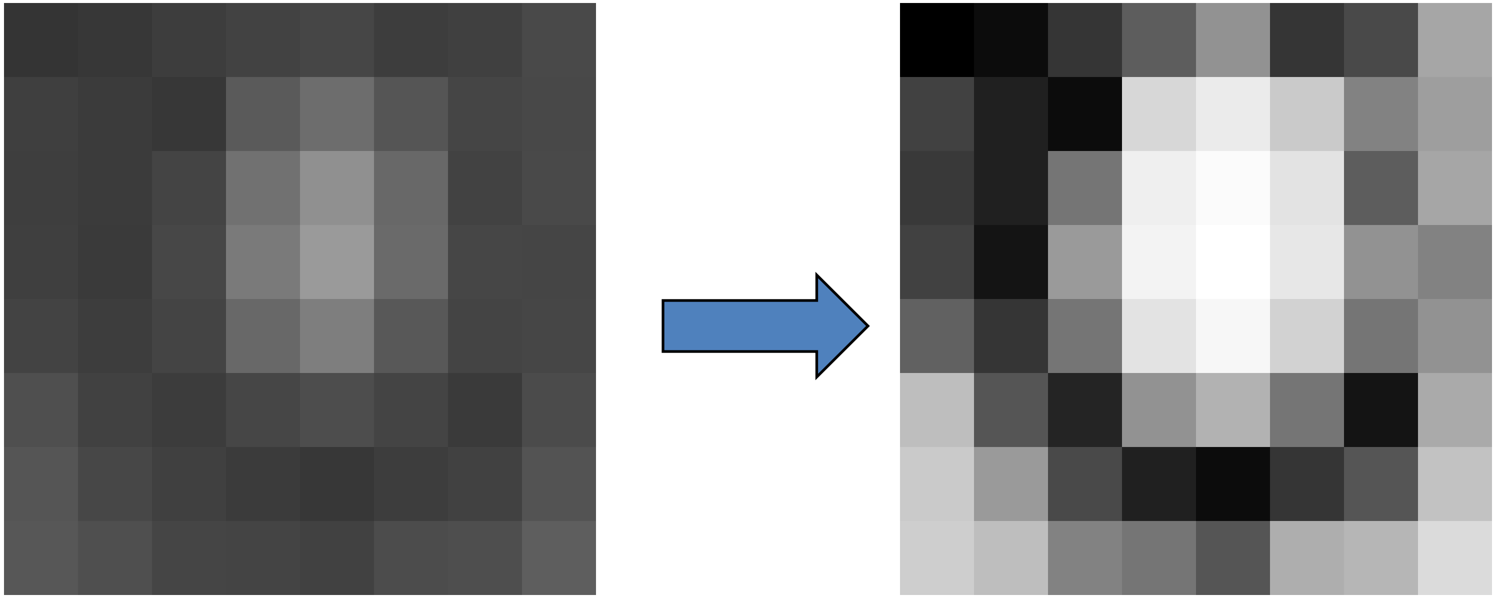
$$s = 183$$

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

Original Image

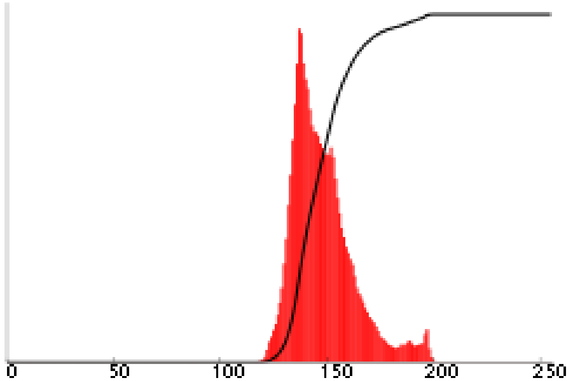


Histogram Equalization: Example



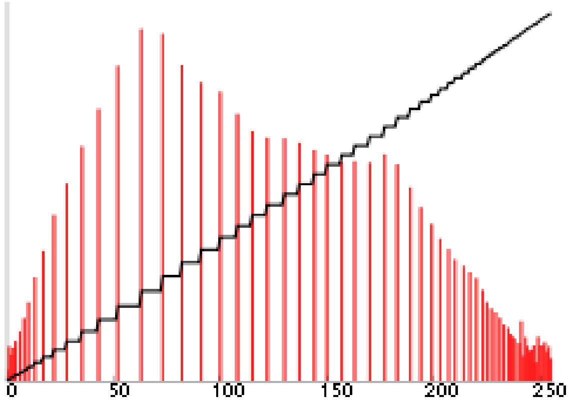
Histogram Equalization: Example

Original Image



Corresponding histogram (red) and cumulative histogram (black)

Image after histogram equalization

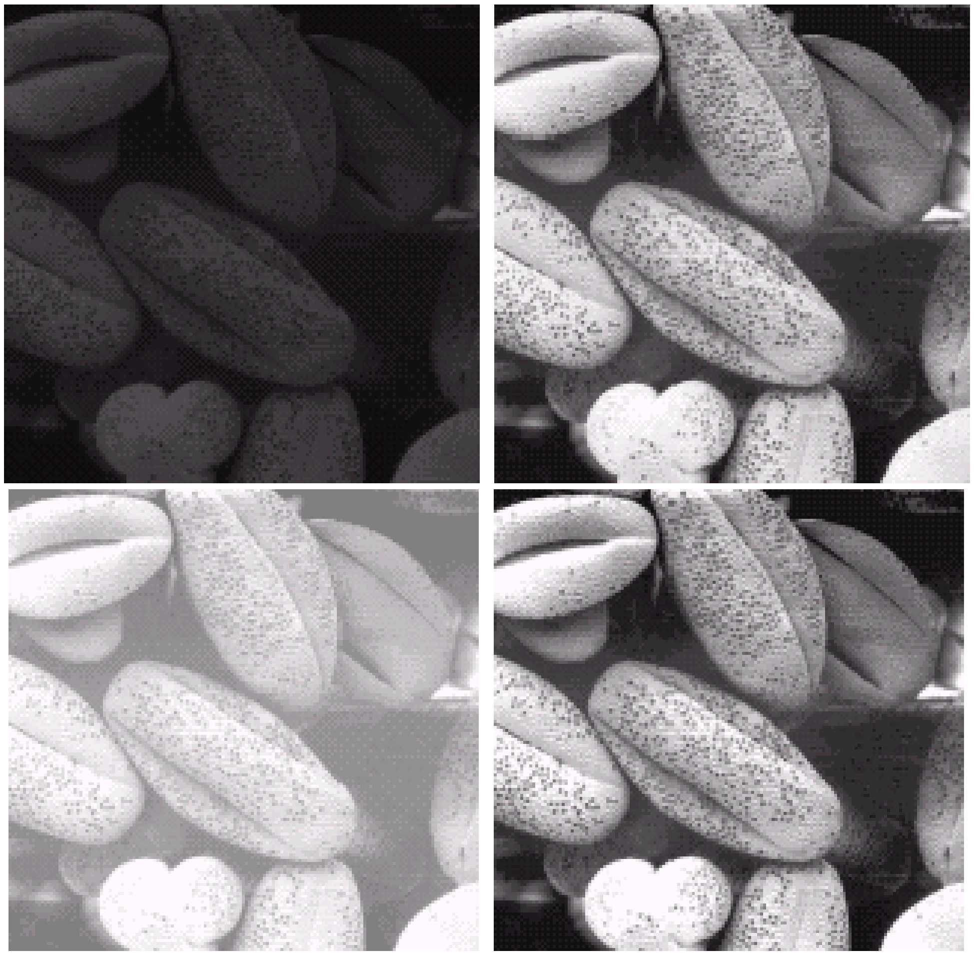


Corresponding histogram (red) and cumulative histogram (black)

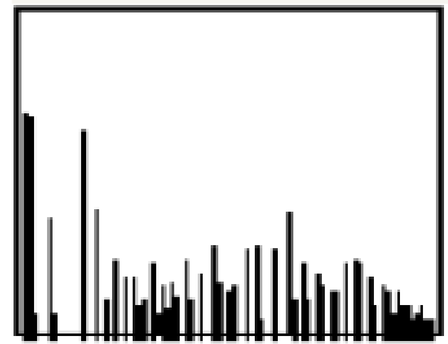
Histogram Equalization: Example

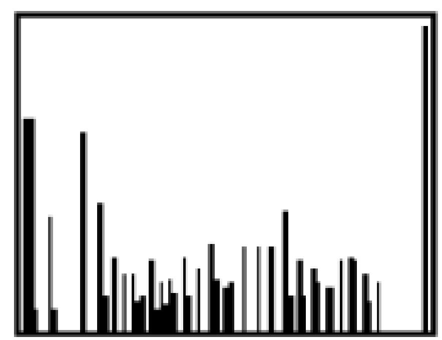
Dark image

Bright image



Equalized Histogram



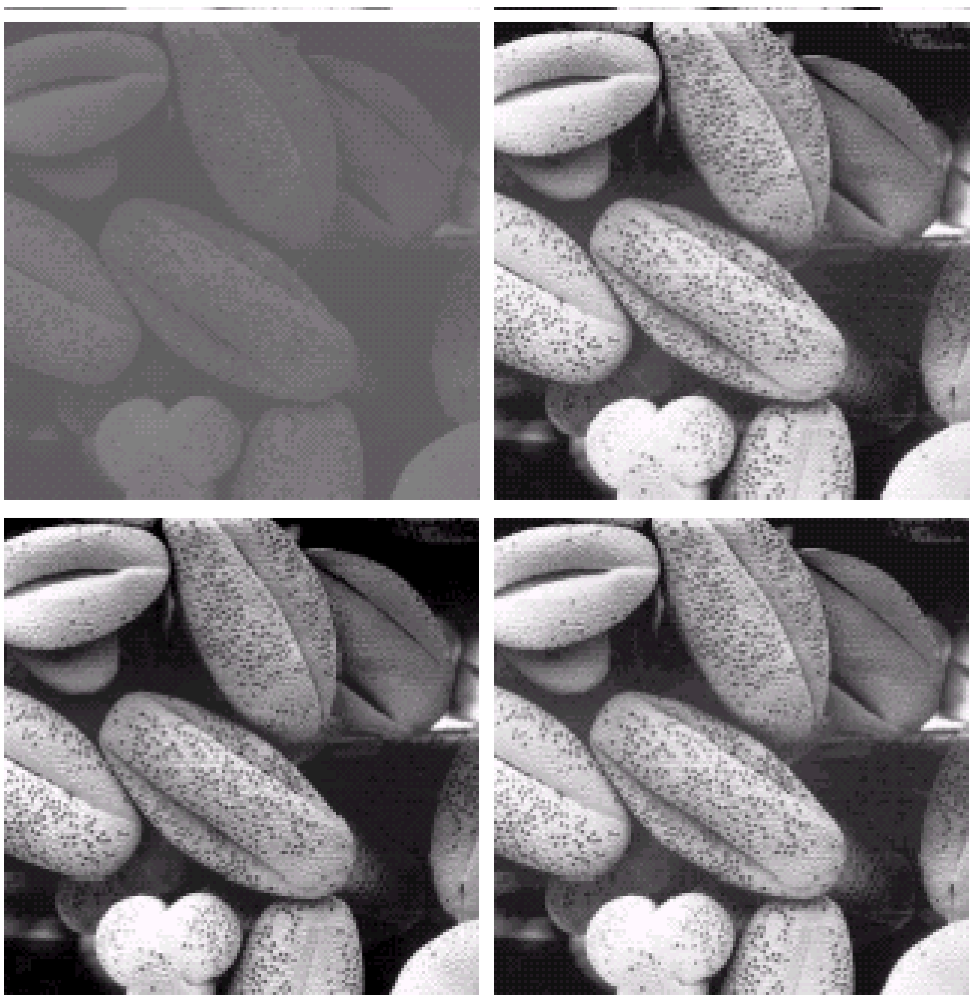


Equalized Histogram

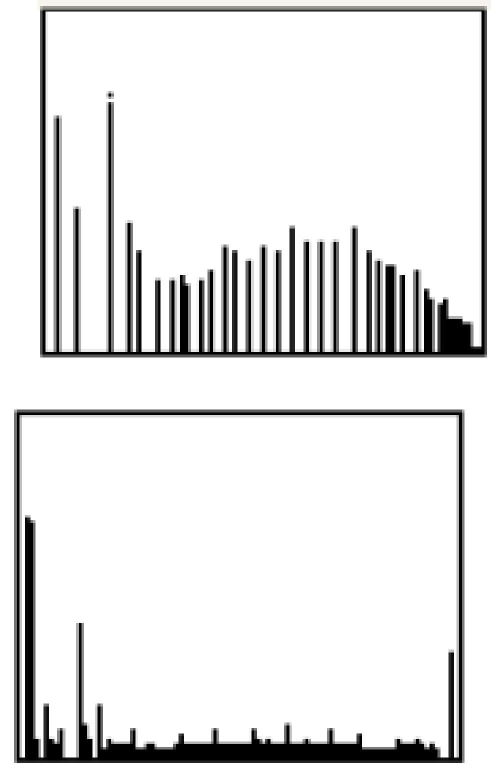
Histogram Equalization: Example

Low contrast

High Contrast



Equalized Histogram



Equalized Histogram

HISTOGRAM MATCHING (SPECIFICATION)

- HISTOGRAM EQUALIZATION DOES NOT ALLOW INTERACTIVE IMAGE ENHANCEMENT AND GENERATES ONLY ONE RESULT: AN APPROXIMATION TO A UNIFORM HISTOGRAM.
- SOMETIMES THOUGH, WE NEED TO BE ABLE TO SPECIFY PARTICULAR HISTOGRAM SHAPES CAPABLE OF HIGHLIGHTING CERTAIN GRAY-LEVEL RANGES.

HISTOGRAM SPECIFICATION

Suppose $s = T(r) = \int_0^r p_r(w)\, dw$

$p_r(r) \rightarrow$ Original histogram ; $p_z(z) \rightarrow$ Desired histogram

Let $v = G(z) = \int_0^z p_z(w)\, dw$ and $z = G^{-1}(v)$

But s and v are identical p.d.f.

$\therefore z = G^{-1}(v) = G^{-1}(s) = G^{-1}(T(r))$

HISTOGRAM SPECIFICATION

- THE PROCEDURE FOR HISTOGRAM-SPECIFICATION BASED ENHANCEMENT IS:
  - EQUALIZE THE LEVELS OF THE ORIGINAL IMAGE USING:

$$s = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n}$$

n: total number of pixels,

nj: number of pixels with gray level rj,

L: number of discrete gray levels

HISTOGRAM SPECIFICATION

– SPECIFY THE DESIRED DENSITY FUNCTION AND OBTAIN THE TRANSFORMATION FUNCTION G(z):

$$v_k = G(z_k) = \sum_{i=0}^{k} p_z(z_i) = s_k$$

p_z: specified desirable PDF for output

HISTOGRAM SPECIFICATION

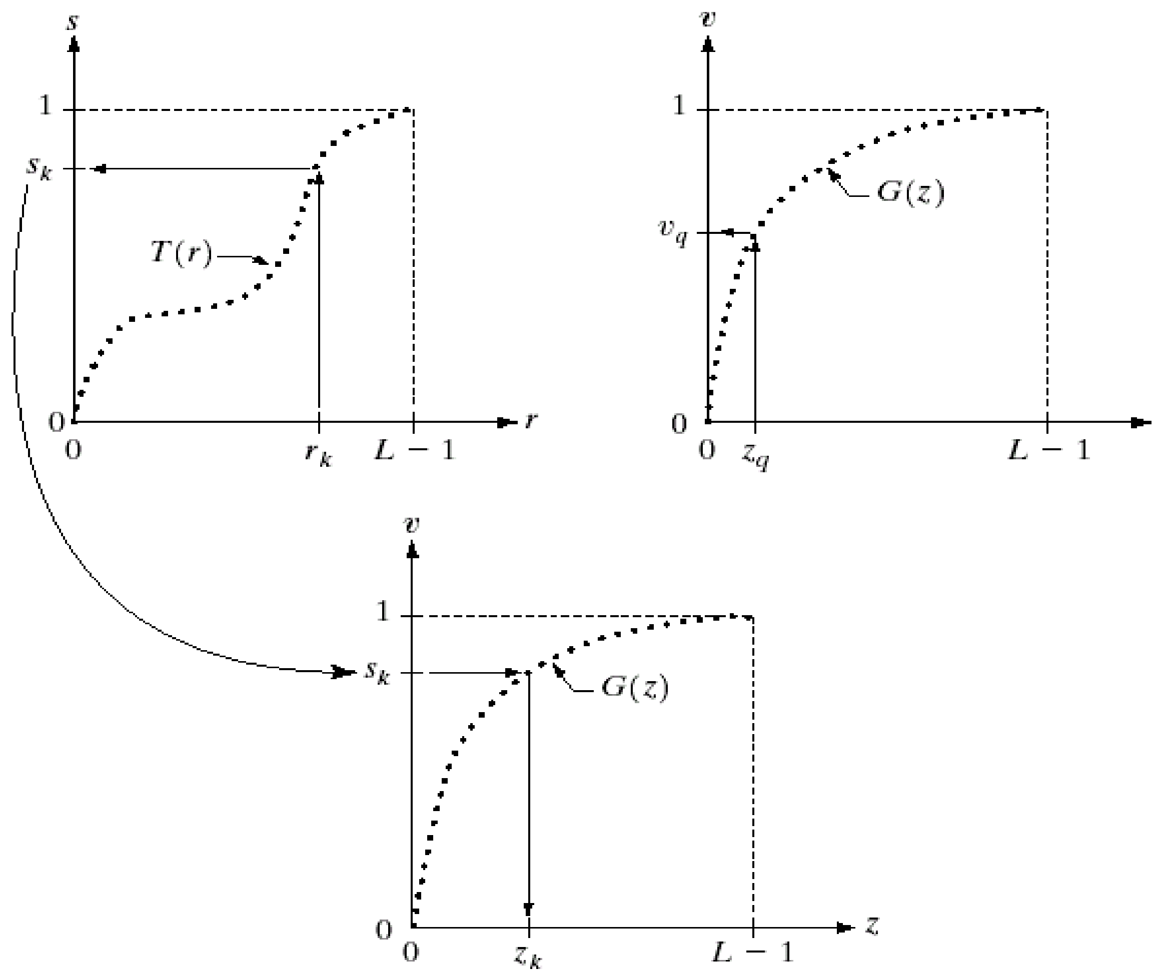
- THE NEW, PROCESSED VERSION OF THE ORIGINAL IMAGE CONSISTS OF GRAY LEVELS CHARACTERIZED BY THE SPECIFIED DENSITY $p_z(z)$.

In essence: $z = G^{-1}(s) \rightarrow \quad z = G^{-1}[T(r)]$

MAPPINGS

a b
c

FIGURE 3.19
(a) Graphical interpretation of mapping from r_k to s_k via $T(r)$.
(b) Mapping of z_q to its corresponding value v_q via $G(z)$.
(c) Inverse mapping from s_k to its corresponding value of z_k.



HISTOGRAM SPECIFICATION

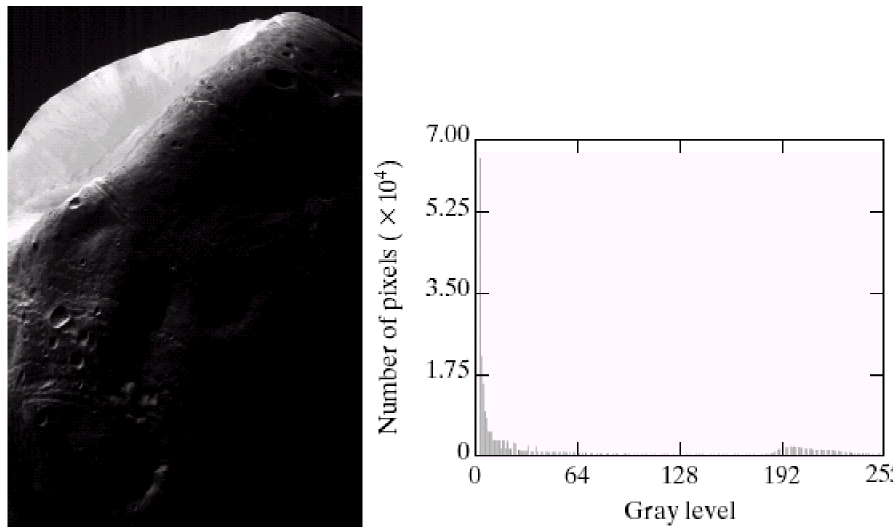
- OBTAIN THE HISTOGRAM OF THE GIVEN IMAGE
- MAP EACH LEVEL r_K TO A LEVEL S_K
- OBTAIN THE TRANSFORMATION FUNCTION G FROM THE GIVEN P_Z (Z)
- PRECOMPUTE Z_K FOR EACH VALUE OF S_K
- FOR EACH PIXEL IN THE ORIGINAL IMAGE, IF THE VALUE OF THAT PIXEL IS r_k MAP THIS VALUE TO ITS CORRESPONDING LEVEL S_K, THEN MAP LEVEL S_K INTO THE FINAL VALUE Z_K

HISTOGRAM SPECIFICATION

k	n_k	$p_r(r_k)$	s_k	$p_z(z_k)$	v_k	n_k
0	790	0.19	0.19	0	0	0
1	1023	0.25	0.44	0	0	0
2	850	0.21	0.65	0	0	0
3	656	0.16	0.81	0.15	0.15	790
4	329	0.08	0.89	0.2	0.35	1023
5	245	0.06	0.95	0.3	0.65	850
6	122	0.03	0.98	0.2	0.85	985
7	81	0.02	1.0	0.15	1.0	448

A 64X64 (4096 PIXELS) IMAGE WITH 8 GRAY LEVELS

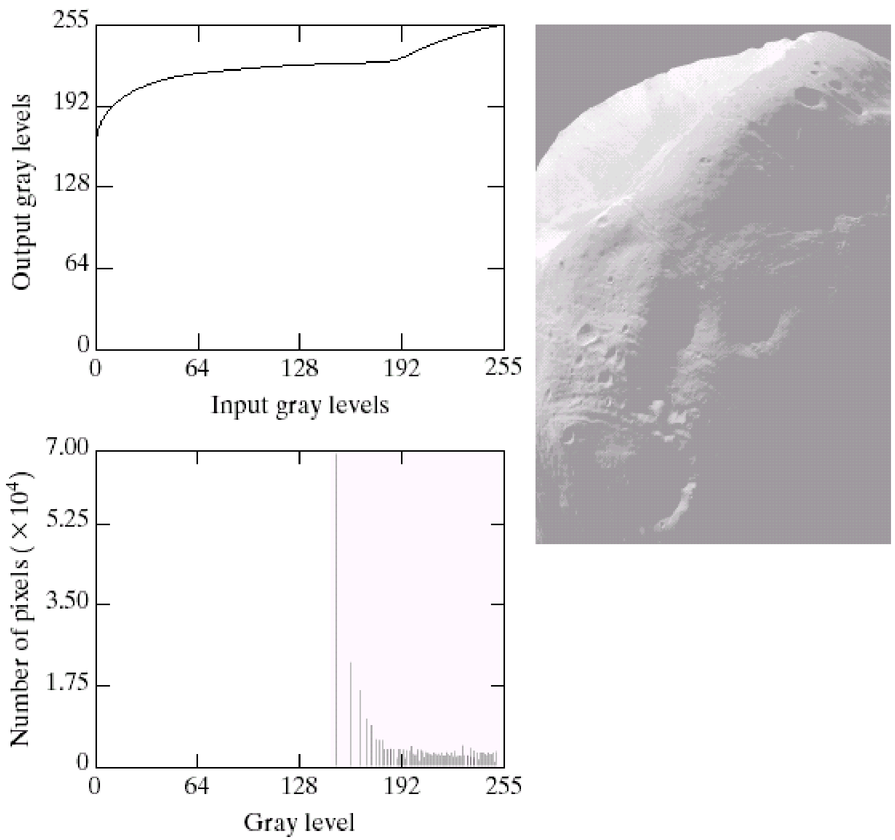
IMAGE ENHANCEMENT IN THE SPATIAL DOMAIN



a b

FIGURE 3.20 (a) Image of the Mars moon Photos taken by NASA's *Mars Global Surveyor.* (b) Histogram. (Original image courtesy of NASA.)

IMAGE ENHANCEMENT IN THE SPATIAL DOMAIN

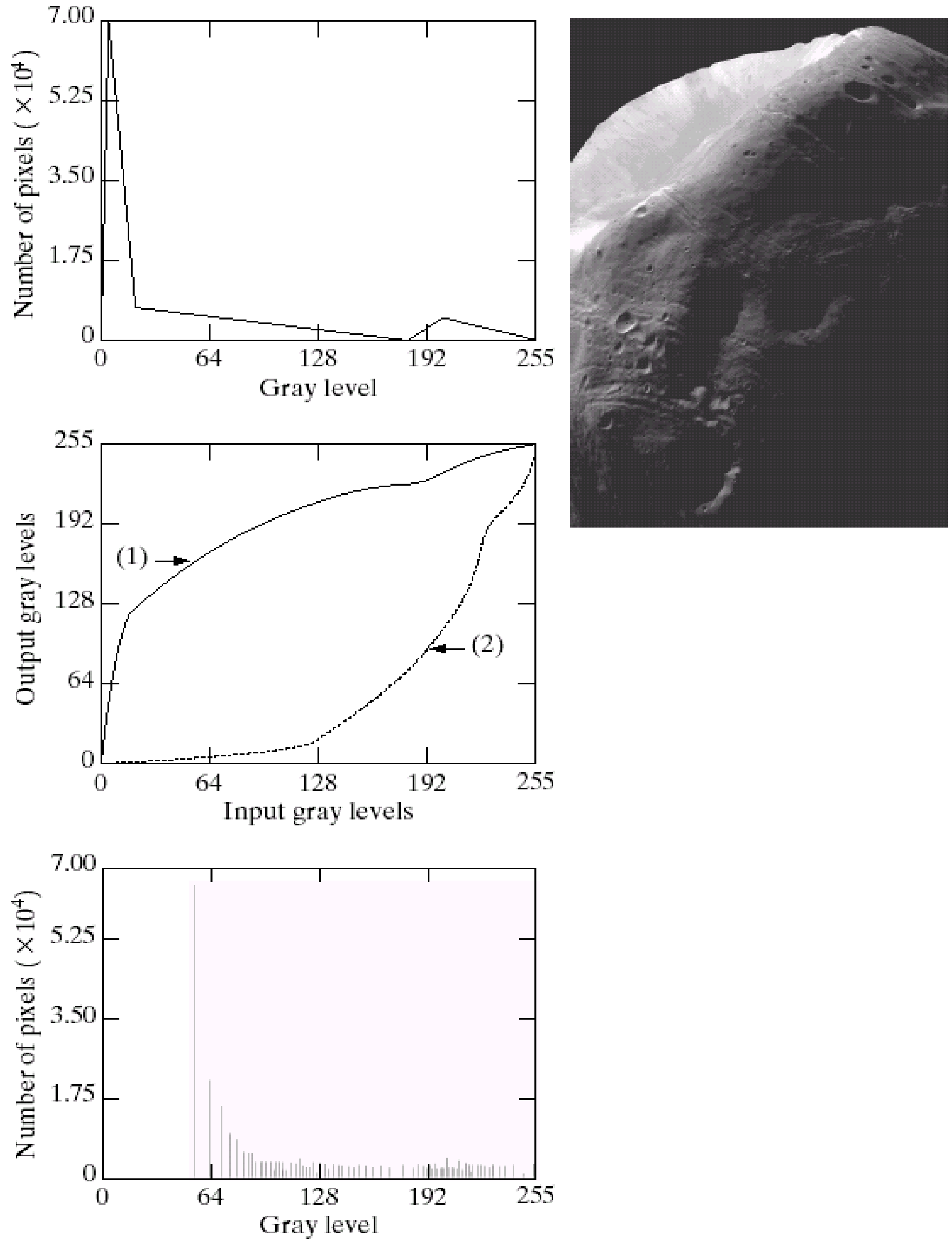


a b
c

FIGURE 3.21
(a) Transformation function for histogram equalization.
(b) Histogram-equalized image (note the washed-out appearance).
(c) Histogram of (b).

a c
b
d

FIGURE 3.22
(a) Specified histogram.
(b) Curve (1) is from Eq. (3.3-14), using the histogram in (a); curve (2) was obtained using the iterative procedure in Eq. (3.3-17).
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).



GLOBAL/LOCAL HISTOGRAM EQUALIZATION

- IT MAY BE NECESSARY TO ENHANCE DETAILS OVER SMALL AREAS IN THE IMAGE
- THE NUMBER OF PIXELS IN THESE AREAS MAY HAVE NEGLIGIBLE INFLUENCE ON THE COMPUTATION OF A GLOBAL TRANSFORMATION WHOSE SHAPE DOES NOT NECESSARILY GUARANTEE THE DESIRED LOCAL ENHANCEMENT
- DEVISE TRANSFORMATION FUNCTIONS BASED ON THE GRAY LEVEL DISTRIBUTION IN THE NEIGHBORHOOD OF EVERY PIXEL IN THE IMAGE
- THE PROCEDURE IS:
  - DEFINE A SQUARE (OR RECTANGULAR) NEIGHBORHOOD AND MOVE THE CENTER OF THIS AREA FROM PIXEL TO PIXEL.
  - AT EACH LOCATION, THE HISTOGRAM OF THE POINTS IN THE NEIGHBORHOOD IS COMPUTED AND EITHER A HISTOGRAM EQUALIZATION OR HISTOGRAM SPECIFICATION TRANSFORMATION FUNCTION IS OBTAINED.
  - THIS FUNCTION IS FINALLY USED TO MAP THE GRAY LEVEL OF THE PIXEL CENTERED IN THE NEIGHBORHOOD.
  - THE CENTER IS THEN MOVED TO AN ADJACENT PIXEL LOCATION AND THE PROCEDURE IS REPEATED.

GLOBAL/LOCAL HISTOGRAM EQUALIZATION

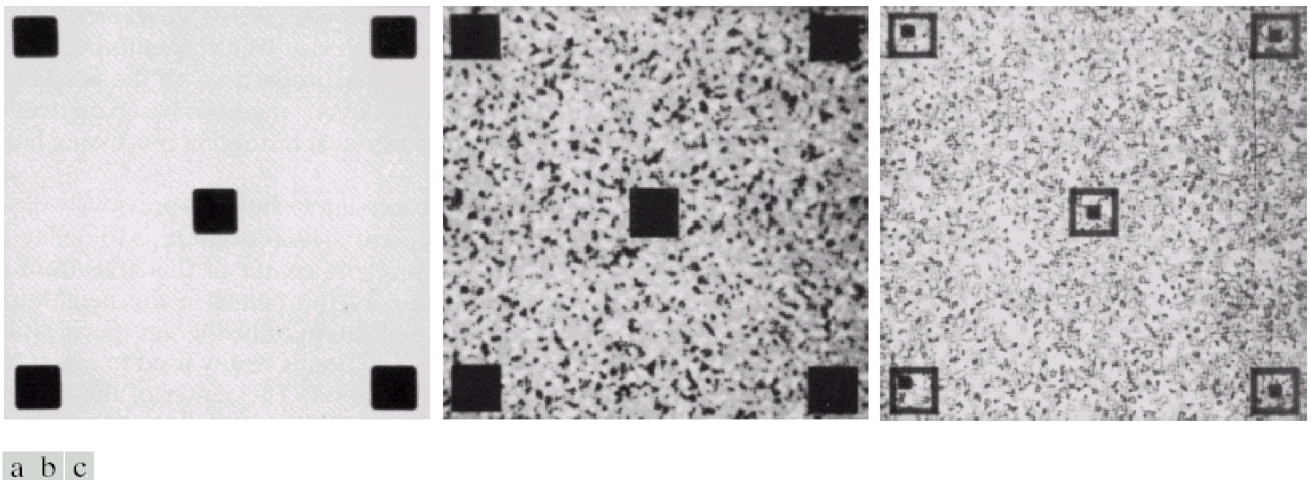


FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7 × 7 neighborhood about each pixel.

USE OF HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT (Global)

- LET r REPRESENT A GRAY LEVEL IN THE IMAGE [0, L-1], AND LET $p(r_i)$ DENOTE THE NORMALIZED HISTOGRAM COMPONENT CORRESPONDING TO THE i^{th} VALUE OF r.
- THE n^{th} MOMENT OF r ABOUT ITS MEAN IS DEFINED AS

$$\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i)$$

- WHERE m IS THE MEAN VALUE OF r (AVERAGE GRAY LEVEL)

$$m = \sum_{i=0}^{L-1} r_i p(r_i)$$

USE OF HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT (Global)

- THE SECOND MOMENT IS GIVEN BY

$$\mu_2(r) = \sum_{i=0}^{L-1} (r_i - m)^2 \, p(r_i)$$

- WHICH IS THE VARIANCE OF r
- MEAN AS A MEASURE OF AVERAGE GRAY LEVEL IN THE IMAGE
- VARIANCE AS A MEASURE OF AVERAGE CONTRAST

USE OF HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT (Local)

- LET (x,y) BE THE COORDINATES OF A PIXEL IN AN IMAGE, AND LET $S_{X,Y}$ DENOTE A NEIGBORHOOD OF SPECIFIED SIZE, CENTERED AT (x,y)
- THE MEAN VALUE m_{SXY} OF THE PIXELS IN $S_{X,Y}$ IS

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

- THE GRAY LEVEL VARIANCE OF THE PIXELS IN REGION $S_{X,Y}$ IS GIVEN BY

$$\sigma_{S_{xy}}{}^{2} = \sum_{(s,t)\in S_{xy}} \left[r_{s,t} - m_{s_{xy}}\right]^2 p(r_{s,t})$$

USE OF HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT

- THE GLOBAL MEAN AND VARIANCE ARE MEASURED OVER AN ENTIRE IMAGE AND ARE USEFUL FOR GROSS ADJUSTMENTS OF OVERALL INTENSITY AND CONTRAST.
- A USE OF THESE MEASURES IN LOCAL ENHANCEMENT IS, WHERE THE LOCAL MEAN AND VARIANCE ARE USED AS THE BASIS FOR MAKING CHANGES THAT DEPEND ON IMAGE CHARACTERISTICS IN A PREDEFINED REGION ABOUT EACH PIXEL IN THE IMAGE.

TUNGSTEN FILAMENT IMAGE

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).

USE OF HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT

- A PIXEL AT POINT (x,y) IS CONSIDERED IF:
  - $m_{SXY} \leq k_0 M_G$, where k0 is a positive constant less than 1.0, and M_G is global mean
  - $\sigma_{sxy} \leq k_2 D_G$, where D_G is the global standard deviation and k_2 is a positive constant
  - $k_1 D_G \leq \sigma_{sxy}$,, with k1 < k2
- A PIXEL THAT MEETS ALL ABOVE CONDITIONS IS PROCESSED SIMPLY BY MULTIPLYING IT BY A SPECIFIED CONSTANT, E, TO INCREASE OR DECREASE THE VALUE OF ITS GRAY LEVEL RELATIVE TO THE REST OF THE IMAGE.
- THE VALUES OF PIXELS THAT DO NOT MEET THE ENHANCEMENT CONDITIONS ARE LEFT UNCHANGED.

IMAGE ENHANCEMENT IN THE SPATIAL DOMAIN

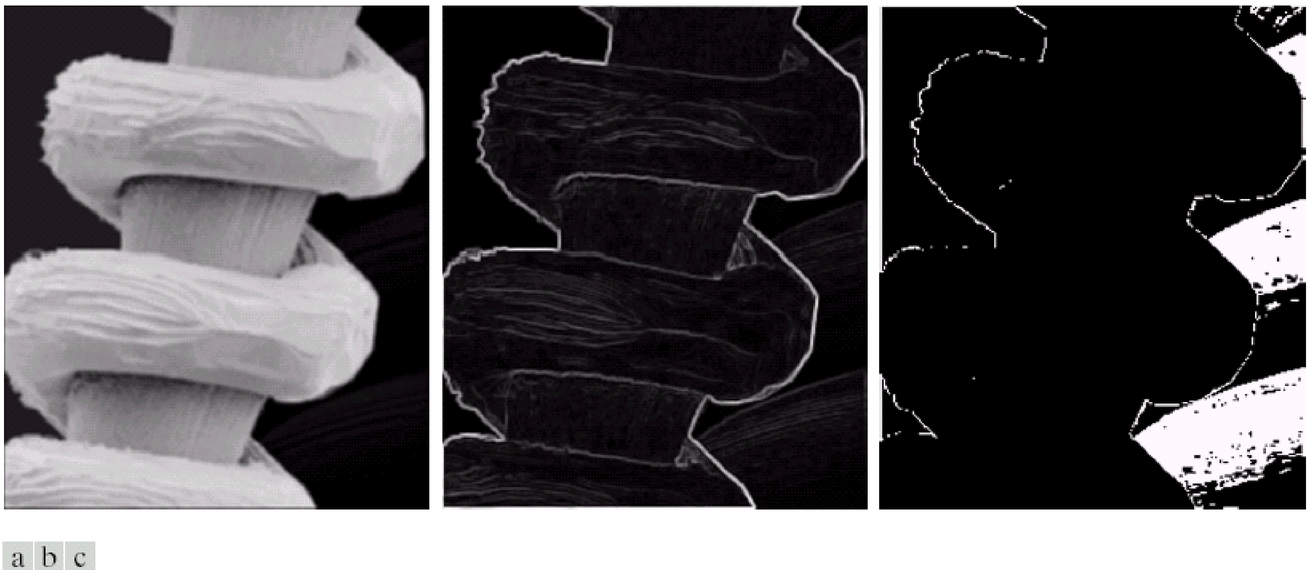


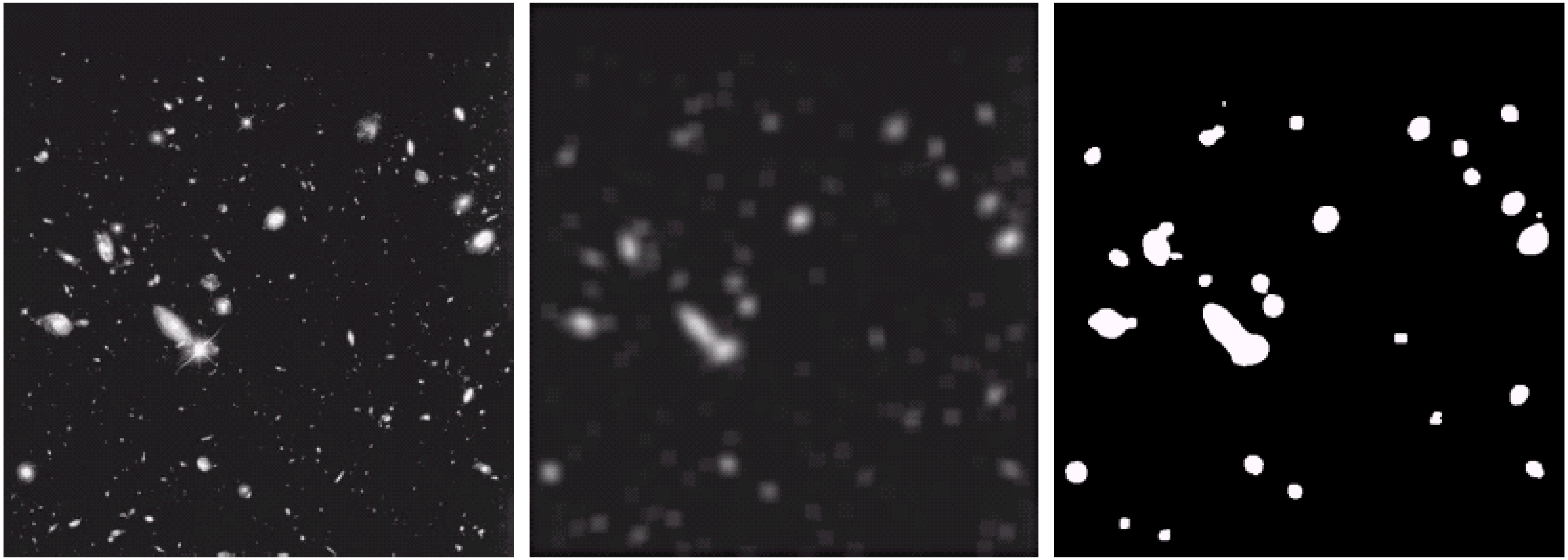
FIGURE 3.25 (a) Image formed from all local means obtained from Fig. 3.24 using Eq. (3.3-21). (b) Image formed from all local standard deviations obtained from Fig. 3.24 using Eq. (3.3-22). (c) Image formed from all multiplication constants used to produce the enhanced image shown in Fig. 3.26.

IMAGE ENHANCEMENT IN THE SPATIAL DOMAIN

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

Spatial Filtering

Spatial Filtering



a b c

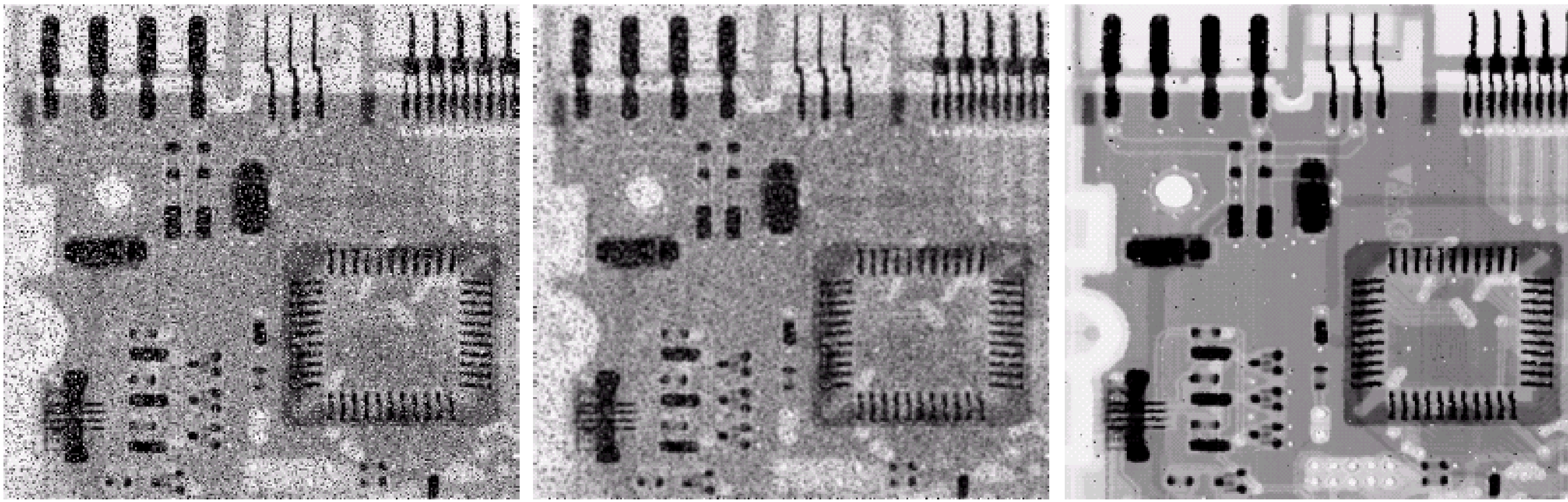
FIGURE 3.36 (a) Image from the Hubble Space Telescope. (b) Image processed by a 15 × 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

Spatial Filtering

a b c

FIGURE 4.50 (a) Original image (784 × 732 pixels). (b) Result of filtering using a GLPF with $D_0 = 100$. (c) Result of filtering using a GLPF with $D_0 = 80$. Note the reduction in fine skin lines in the magnified sections in (b) and (c).

Spatial Filtering



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filtering

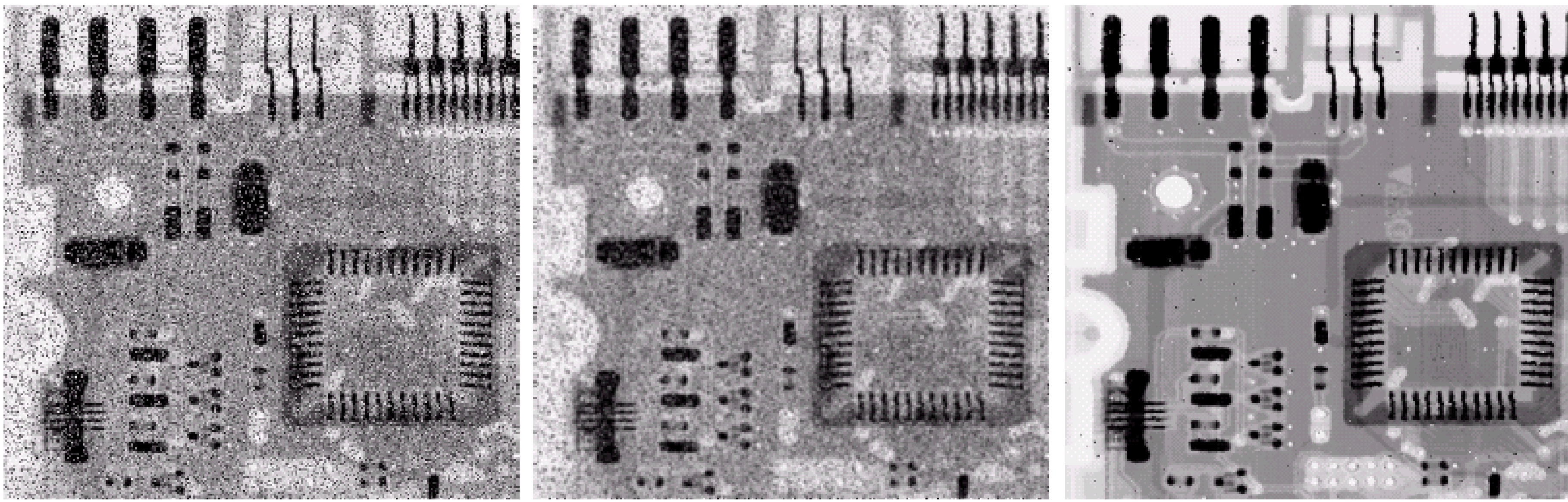


FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Spatial Filtering

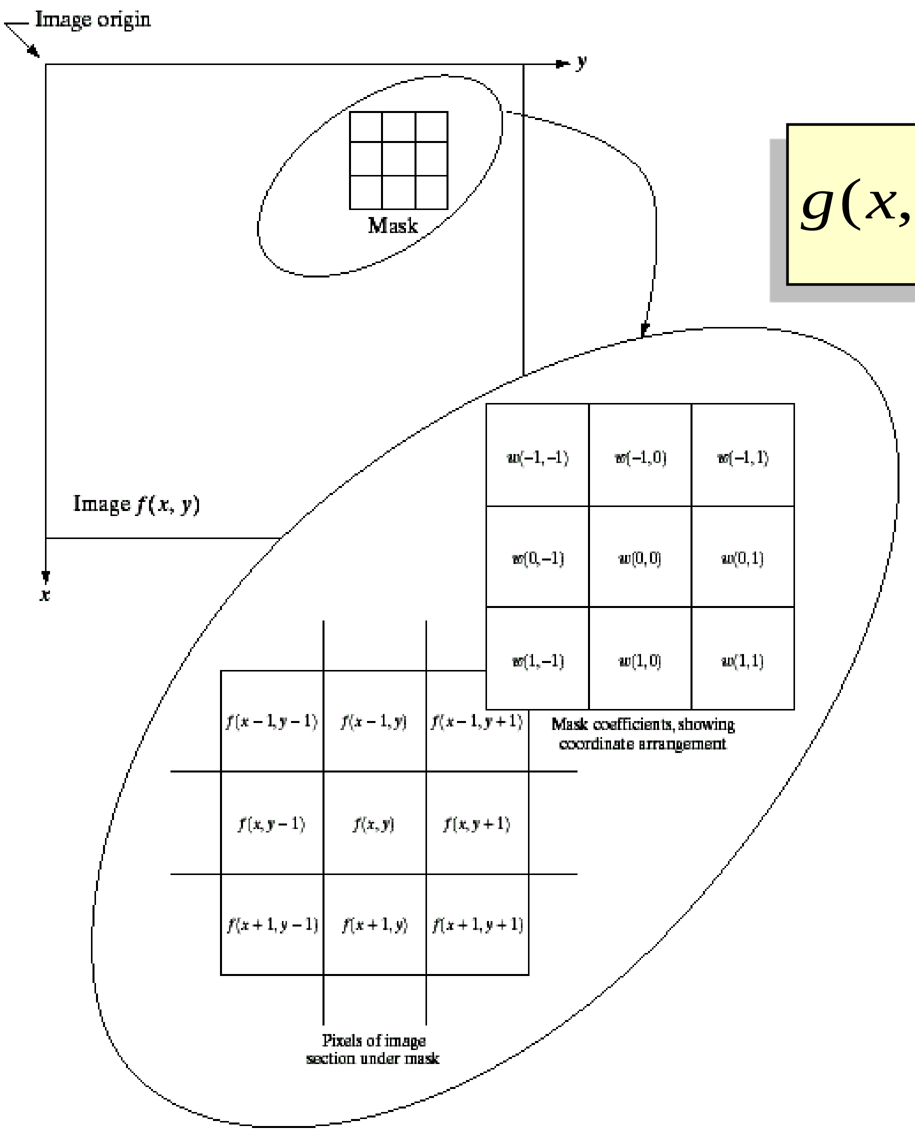
a b

FIGURE 4.58 (a) Original, blurry image. (b) Image enhanced using the Laplacian in the frequency domain. Compare with Fig. 3.38(e).

Spatial Filtering: Basics

- The output intensity value at *(x,y)* depends not only on the input intensity value at *(x,y)* but also on the specified number of neighboring intensity values around *(x,y)*
- Spatial masks (also called window, filter, kernel, template) are used and **convolved** over the entire image for local enhancement (spatial filtering)
- The size of the masks determines the number of neighboring pixels which influence the output value at *(x,y)*
- The values (coefficients) of the mask determine the nature and properties of enhancing technique

Spatial Filtering: Basics



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

where $a = \dfrac{m-1}{2}$, $b = \dfrac{n-1}{2}$

$$x = 0,1,2,....,M-1,\ y = 0,1,2,...,N-1$$

Filtering can be given in equation form as shown above

Spatial Filtering: Basics

- Given the 3×3 mask with coefficients: $w_1, w_2, \ldots, w_9$
- The mask cover the pixels with gray levels: $z_1, z_2, \ldots, z_9$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

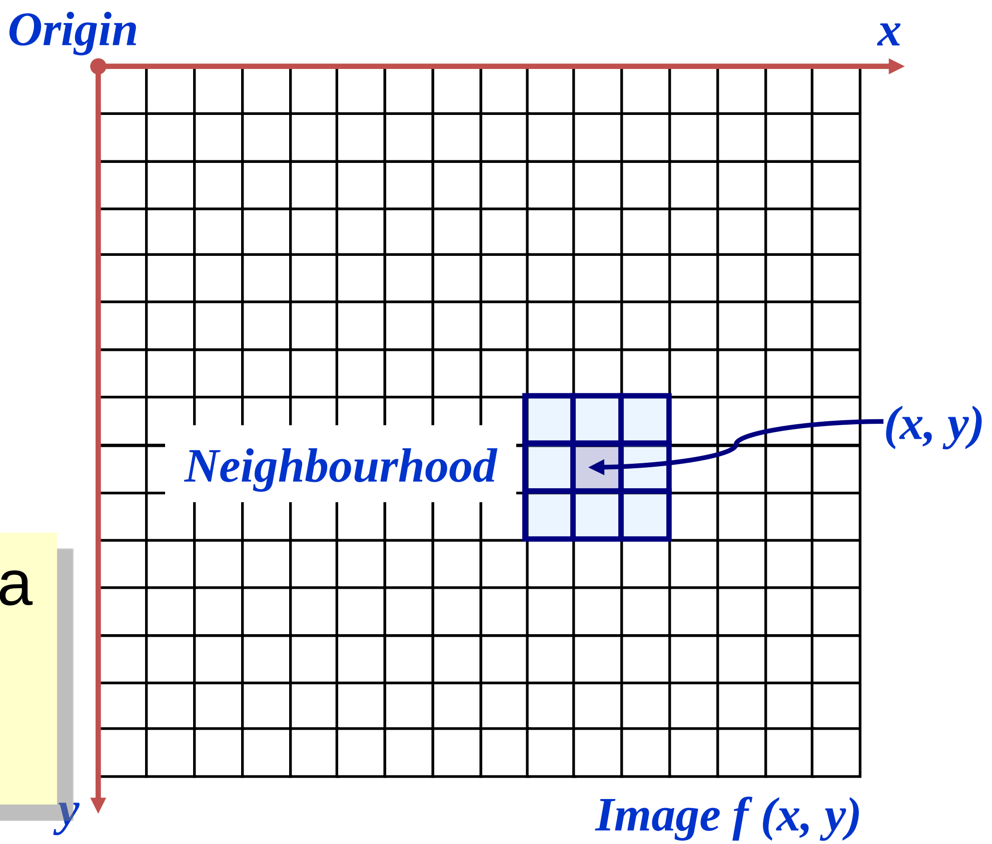
$$z \longleftarrow z_1 w_1 + z_2 w_2 + z_3 w_3 + \cdots + z_9 w_9 = \sum_{i=1}^{9} z_i w_i$$

- z gives the output intensity value for the processed image (to be stored in a new array) at the location of z_5 in the input image

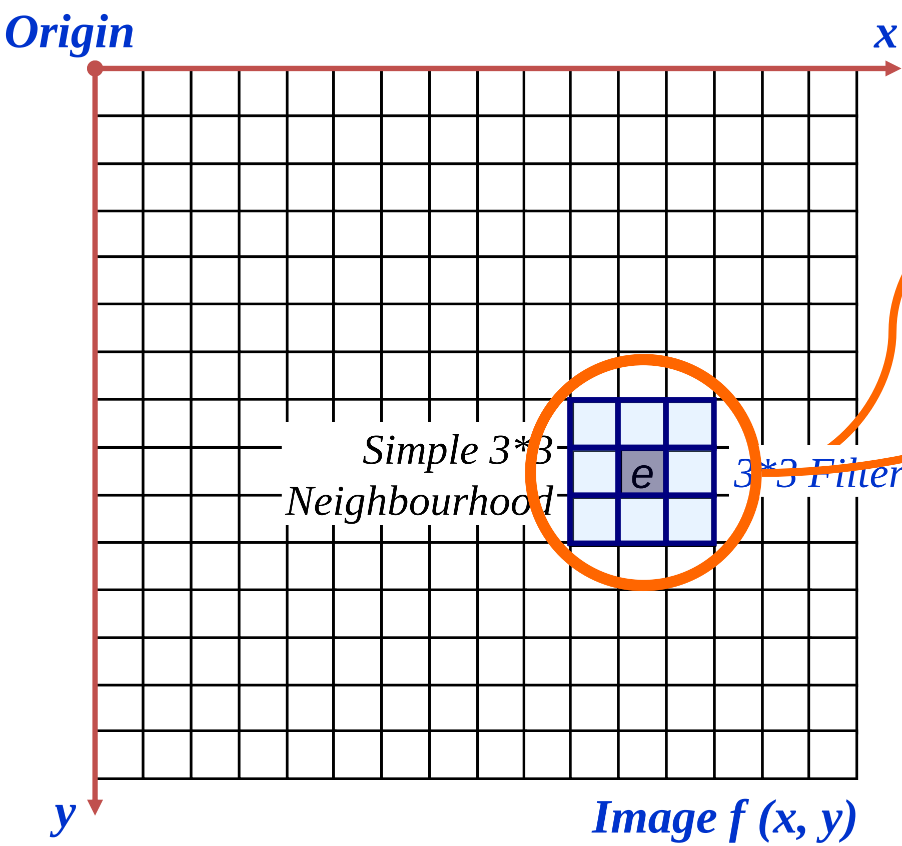
Spatial Filtering: Basics

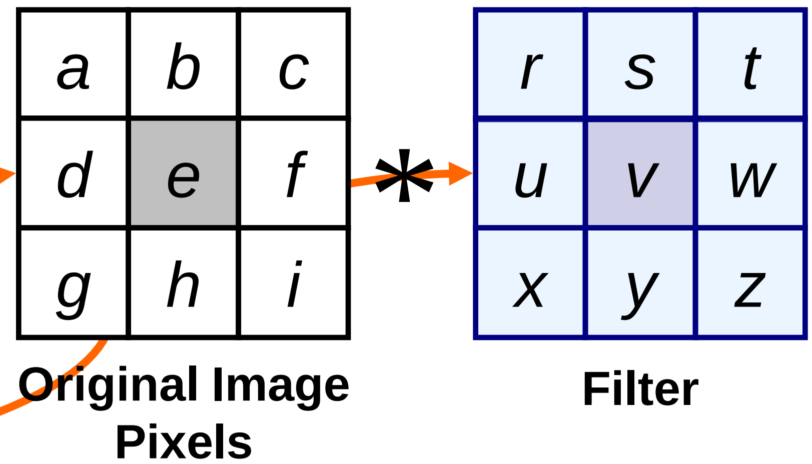
Neighbourhood operations: Operate on a larger neighbourhood of pixels than point operations

Neighbourhoods are mostly a rectangle around a central pixel



Spatial Filtering: Basics





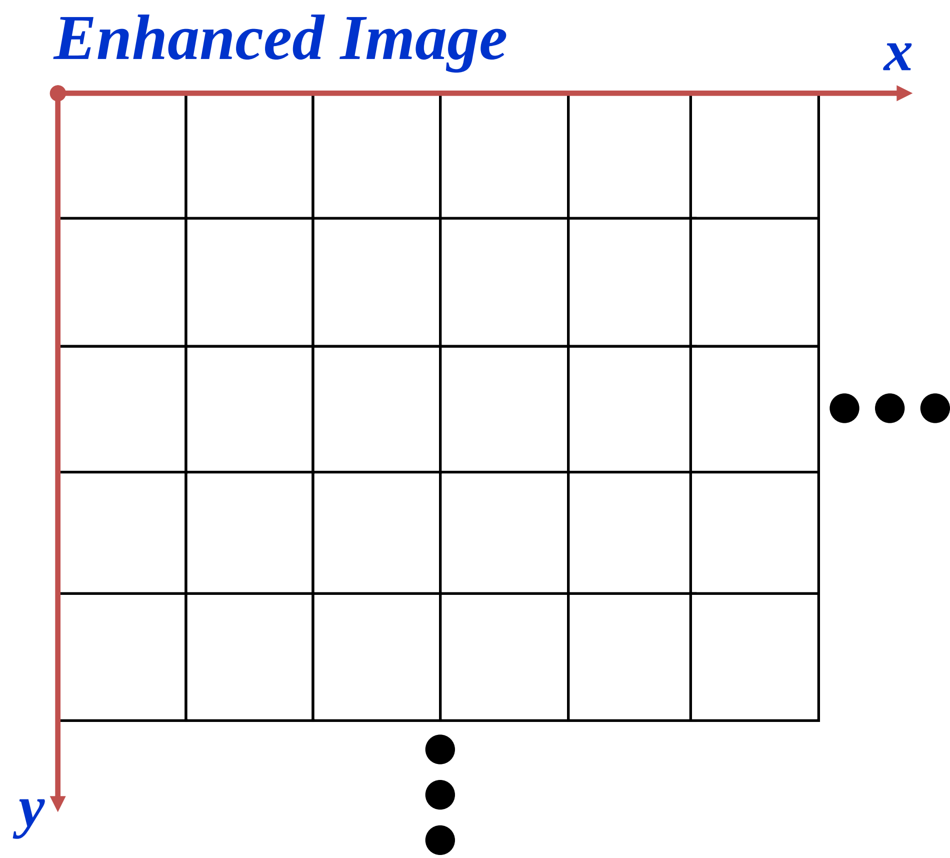
$$
\begin{aligned}
e_{processed} = \; & v*e + \\
& r*a + s*b + t*c + \\
& u*d + w*f + \\
& x*g + y*h + z*i
\end{aligned}
$$

The above is repeated for every pixel in the original image to generate the filtered image

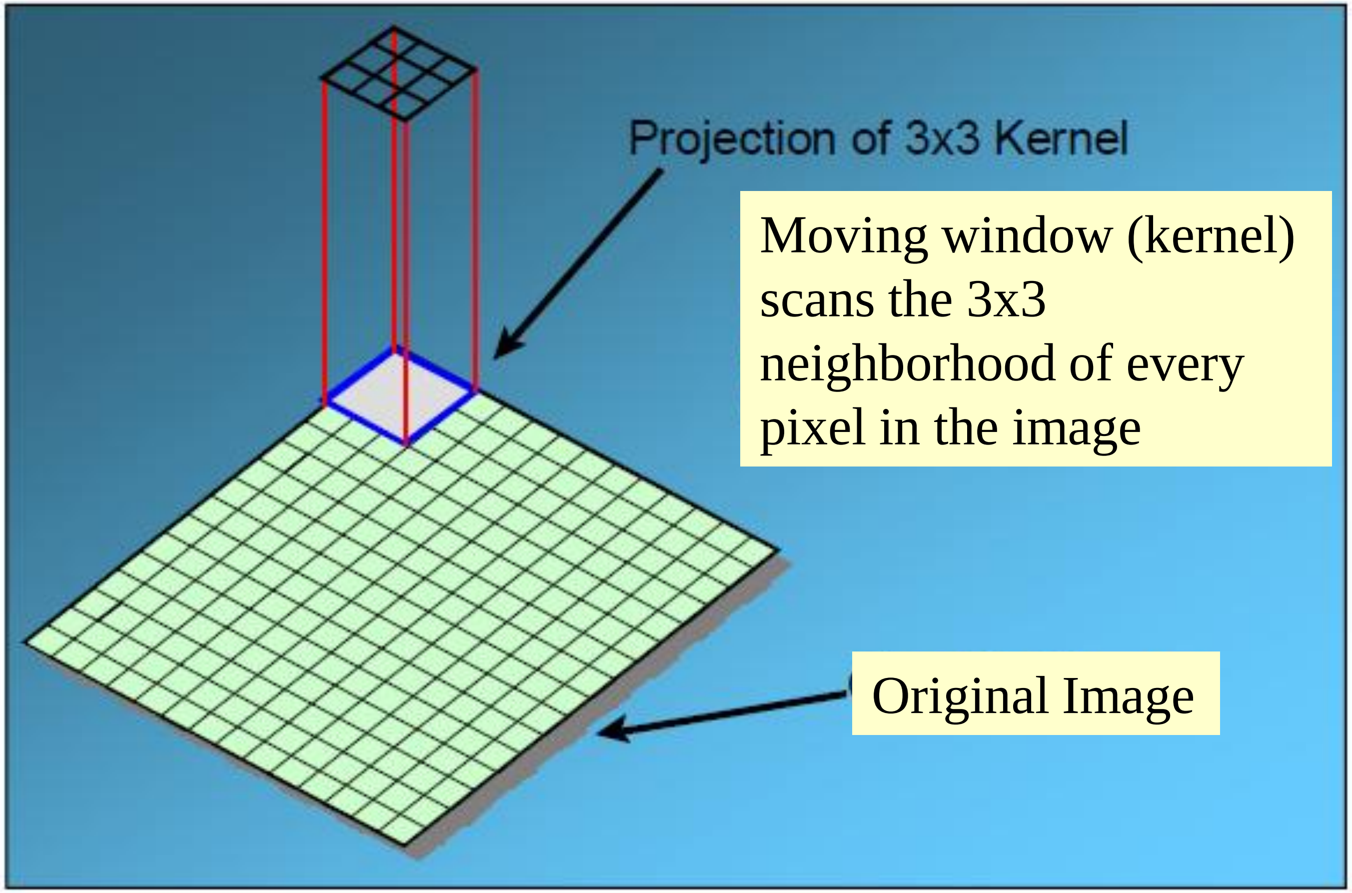
Spatial Filtering: Basics

Original Image

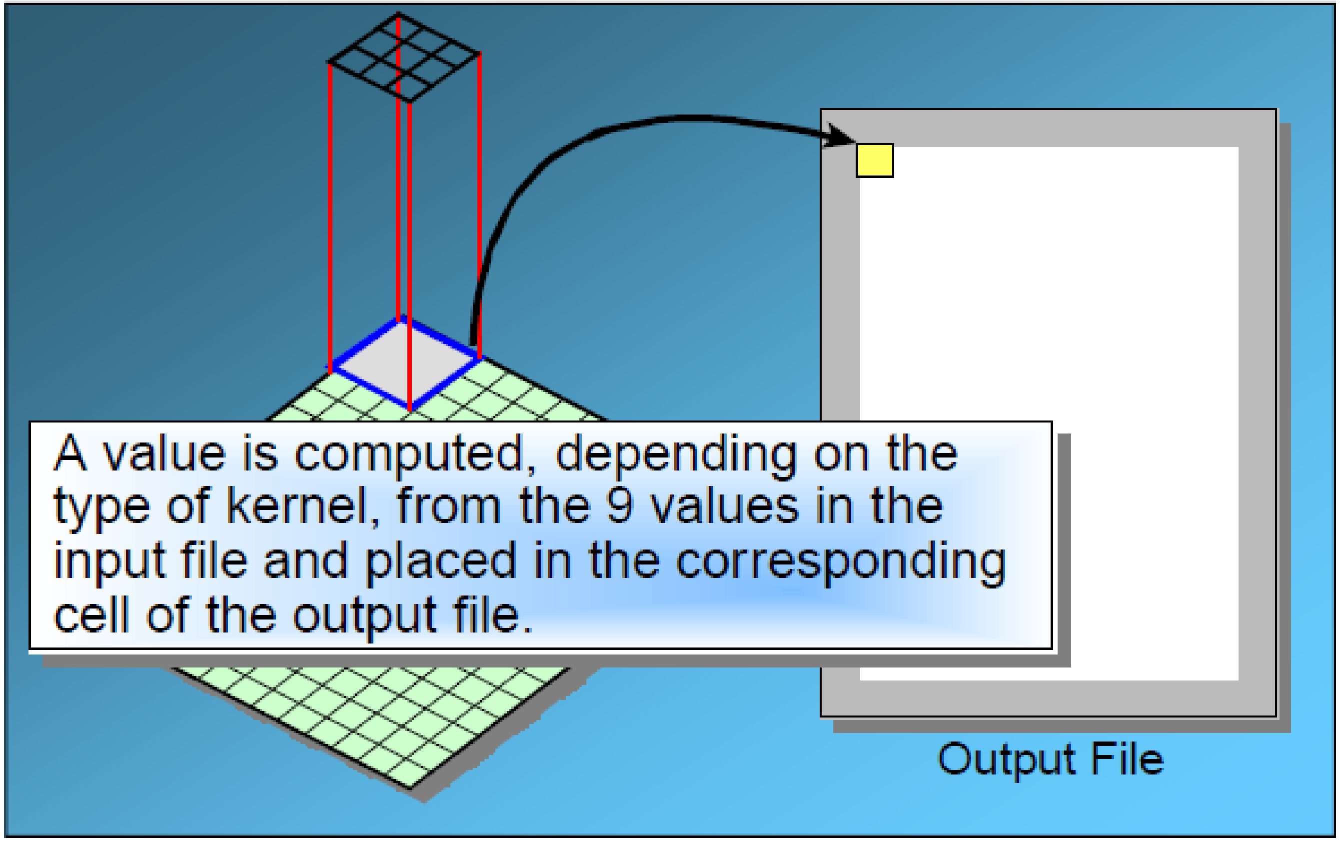
123	127	128	119	115	130
140	145	148	153	167	172
133	154	183	192	194	191
194	199	207	210	198	195
164	170	175	162	173	151



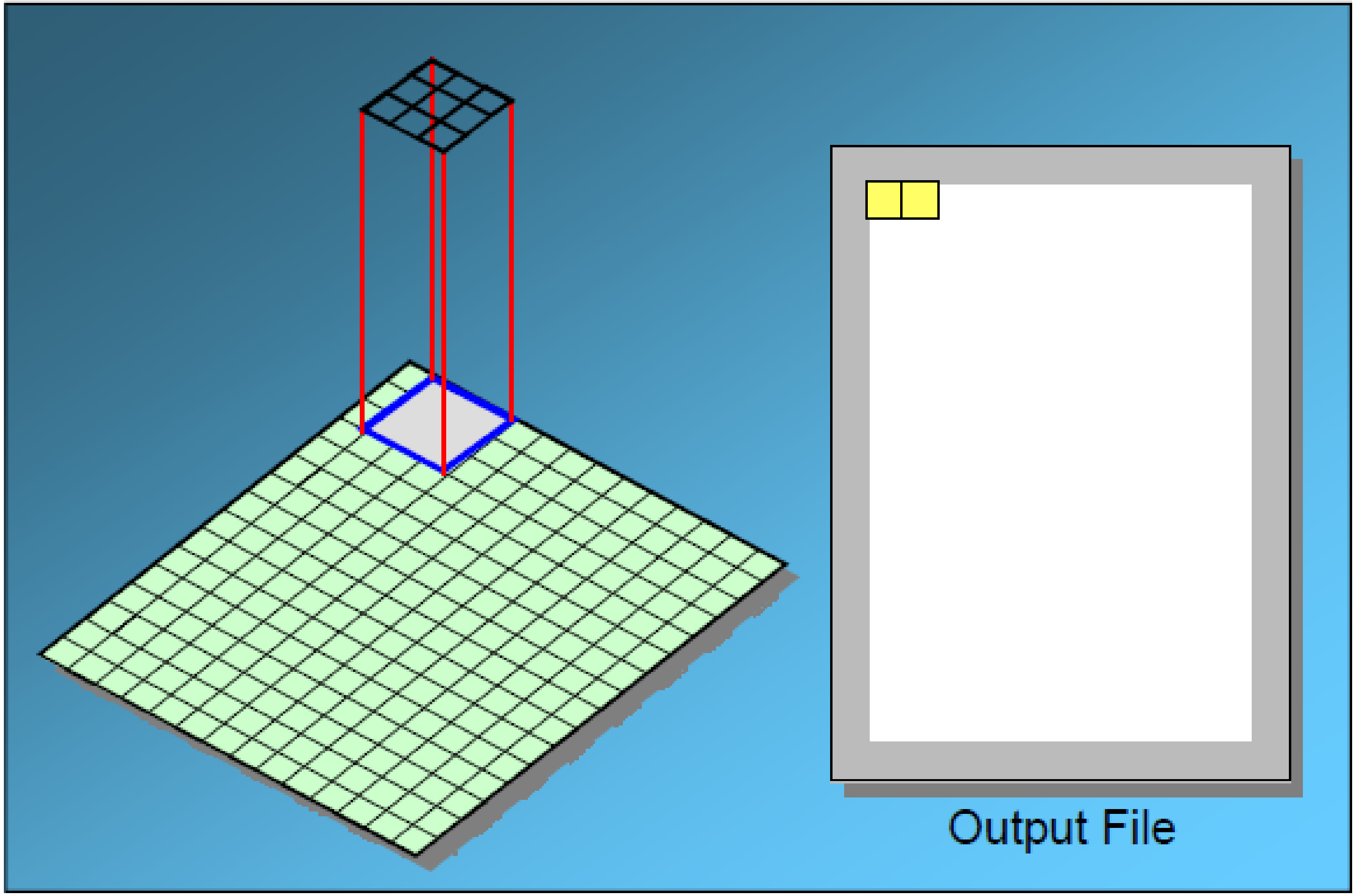
Spatial Filtering: Basics



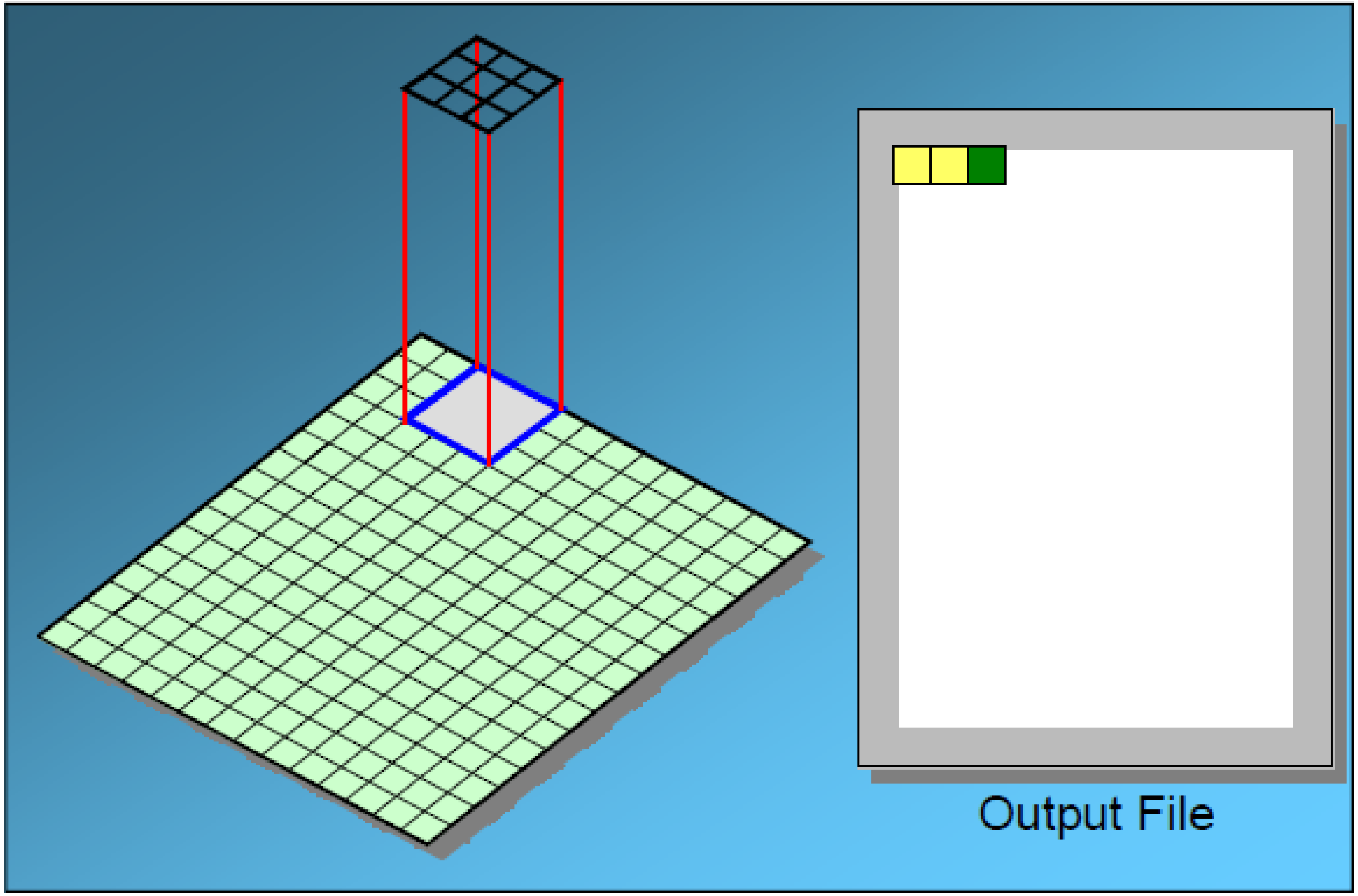
Spatial Filtering: Basics



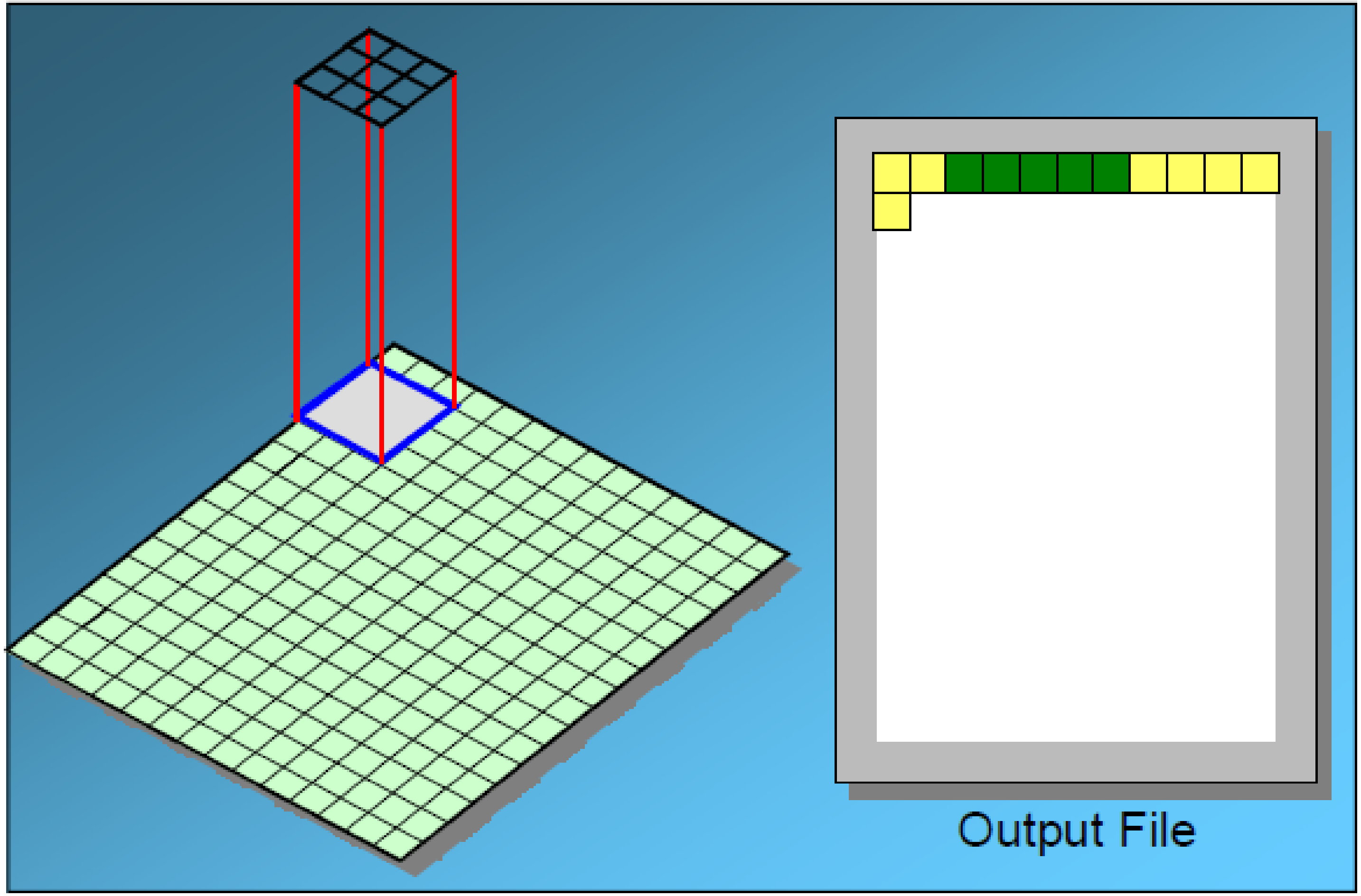
Spatial Filtering: Basics



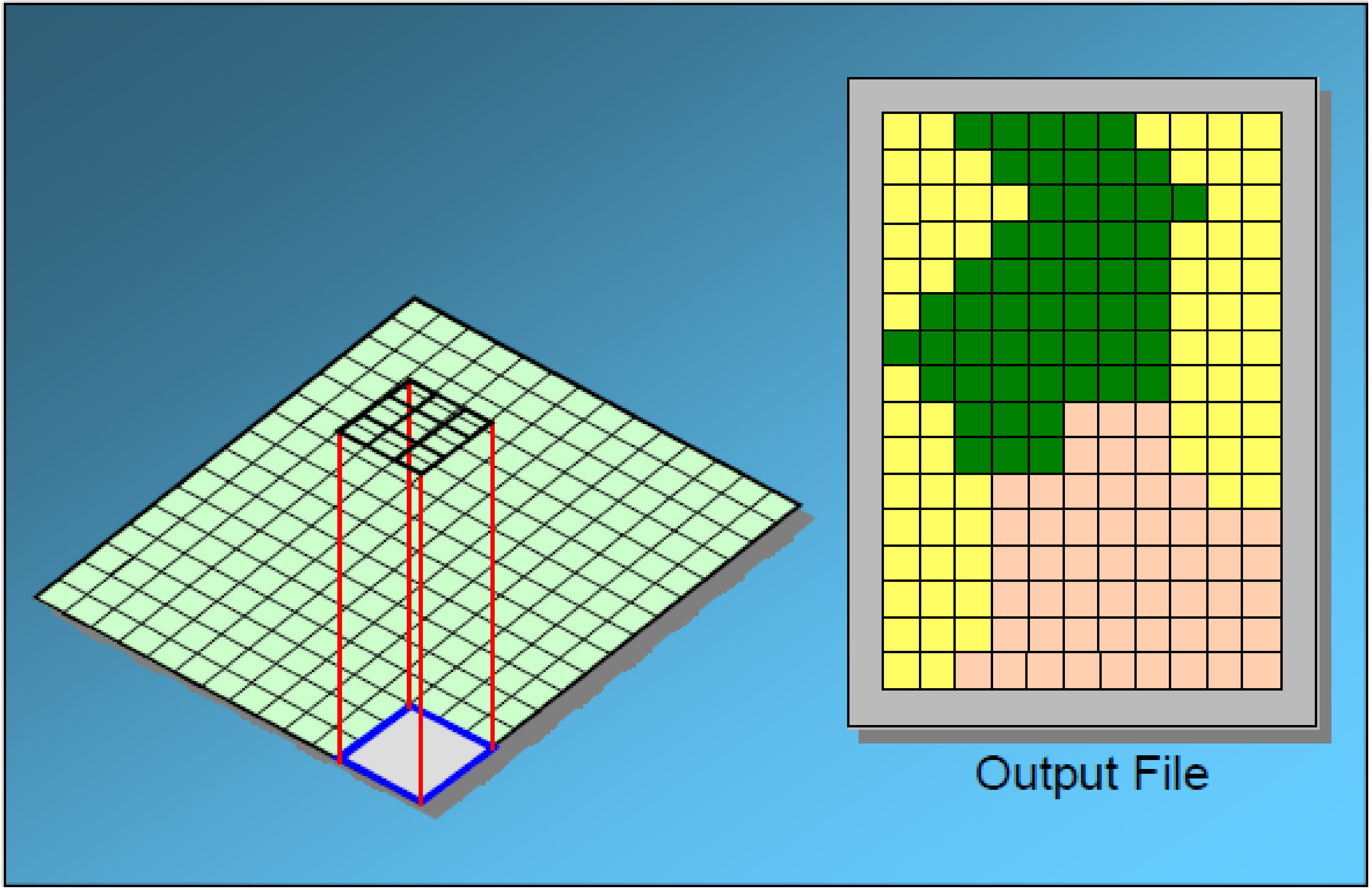
Spatial Filtering: Basics



Spatial Filtering: Basics



Spatial Filtering: Basics



Spatial Filtering: Basics

Mask operation near the image border: Problem arises when part of the mask is located outside the image plane

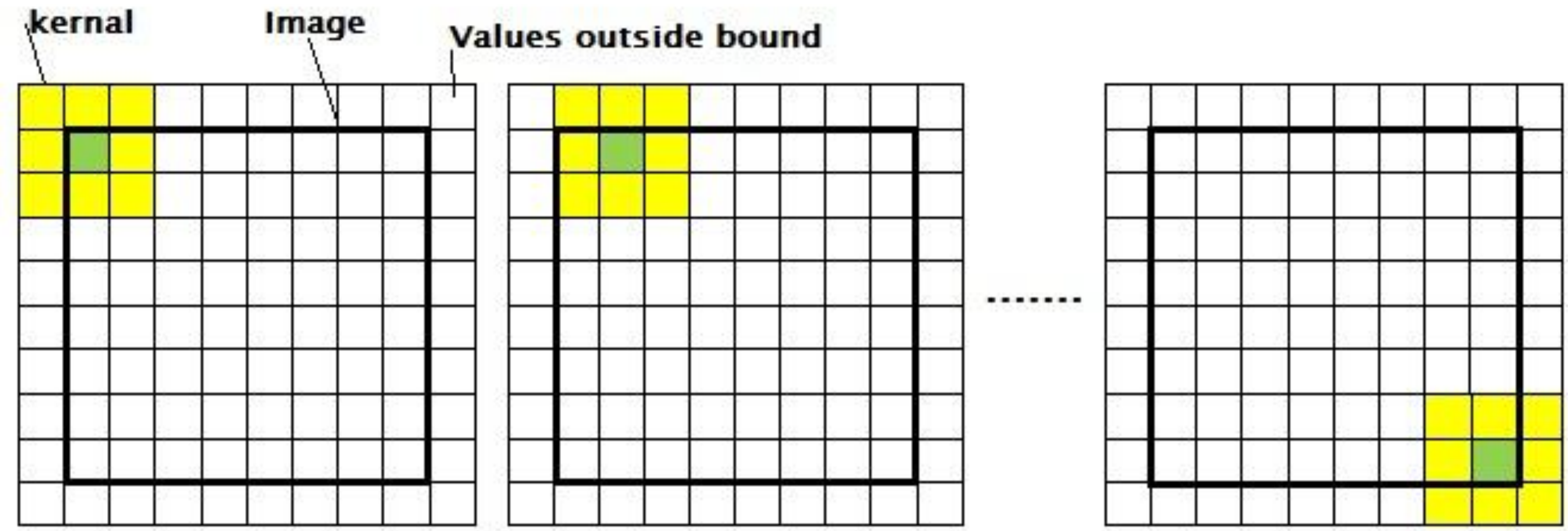
Discard the problem pixels (e.g. 512x512 input 510x510 output if mask size is 3x3)

Zero padding: Expand the input image by padding zeros (512x512 original image, 514x514 padded image, 512x512 output)

Zero padding is not recommended as it creates artificial lines or edges on the border

Pixel replication: We normally use the gray levels of border pixels to fill up the expanded region (for 3x3 mask). For larger masks a border region equal to half of the mask size is mirrored on the expanded region.

Spatial Filtering: Basics



Mask operation near the border: Pixel replication

102	102	130	143	123	115	...	...
102	102	130	143	123	115	...	...
93	93	...	...	...			
98	98	...					
82	82	...					
65	65						
...	...						
...	...						

Expanded area

Original image size (shaded area)

Smoothing Spatial Filters

Simply average all of the pixels in a neighbourhood around a central value

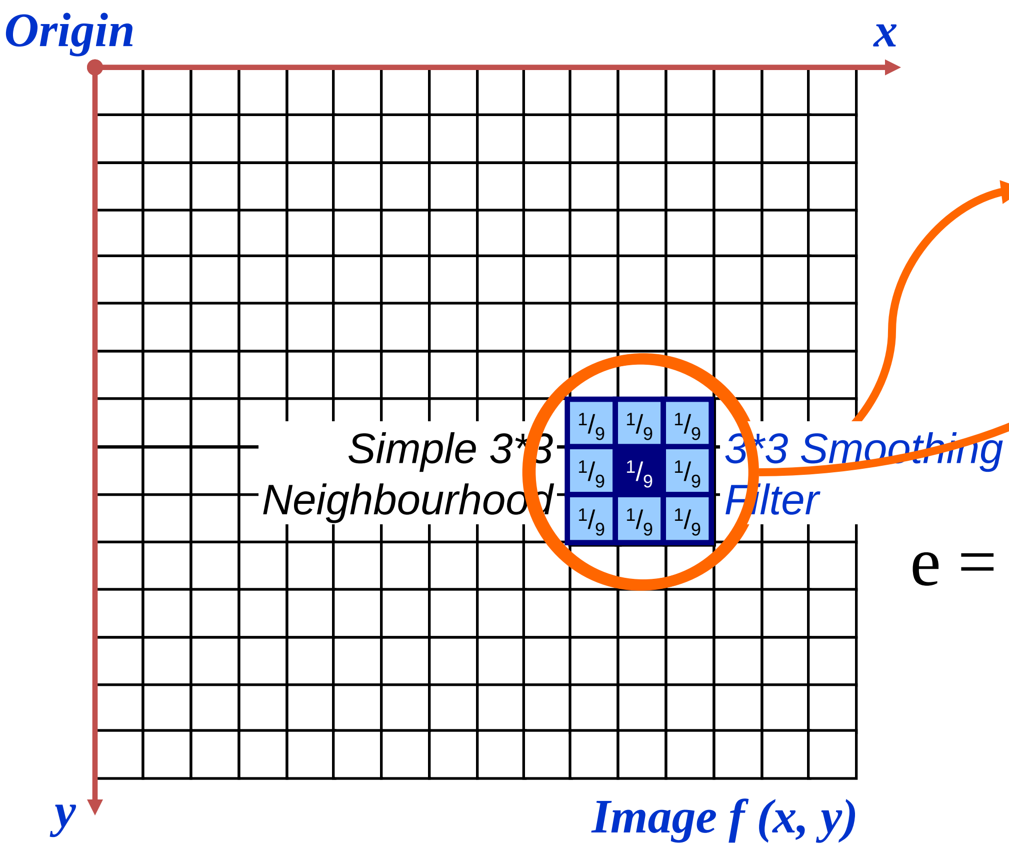
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

Simple averaging filter

Smoothing Spatial Filters

- For blurring/noise reduction
- Blurring is usually used in preprocessing steps, e.g., to remove small details from an image prior to object extraction, or to bridge small gaps in lines or curves
- Equivalent to Low-pass spatial filtering in frequency domain because smaller (high frequency) details are removed based on neighborhood averaging (averaging filters)

Smoothing Spatial Filters



104	100	108
99	106	98
95	90	85

Original Image Pixels

*

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

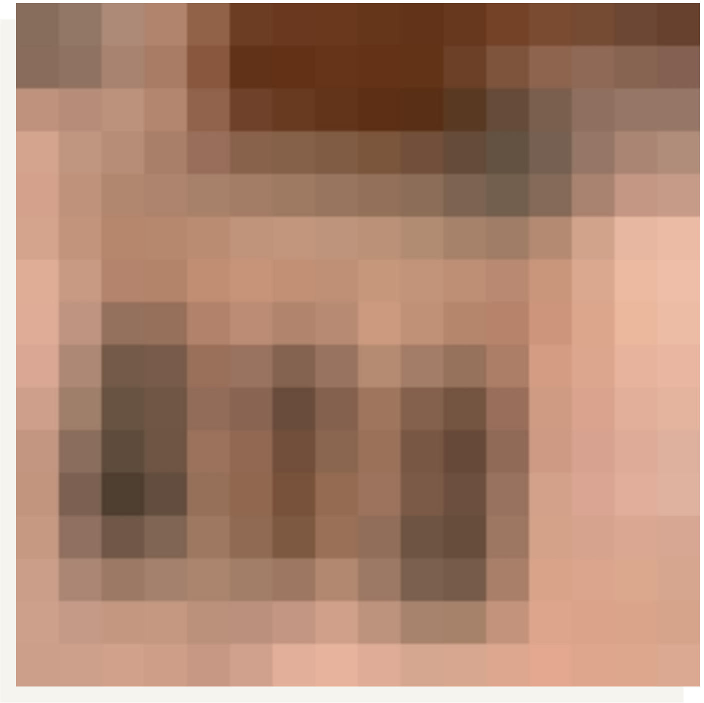
Filter

$$
\begin{aligned}
e = \; & {}^1/_9*106 + \\
& {}^1/_9*104 + {}^1/_9*100 + {}^1/_9*108 + \\
& {}^1/_9*99 + {}^1/_9*98 + \\
& {}^1/_9*95 + {}^1/_9*90 + {}^1/_9*85 \\
= \; & 98.3333
\end{aligned}
$$

The above is repeated for every pixel in the original image to generate the smoothed image

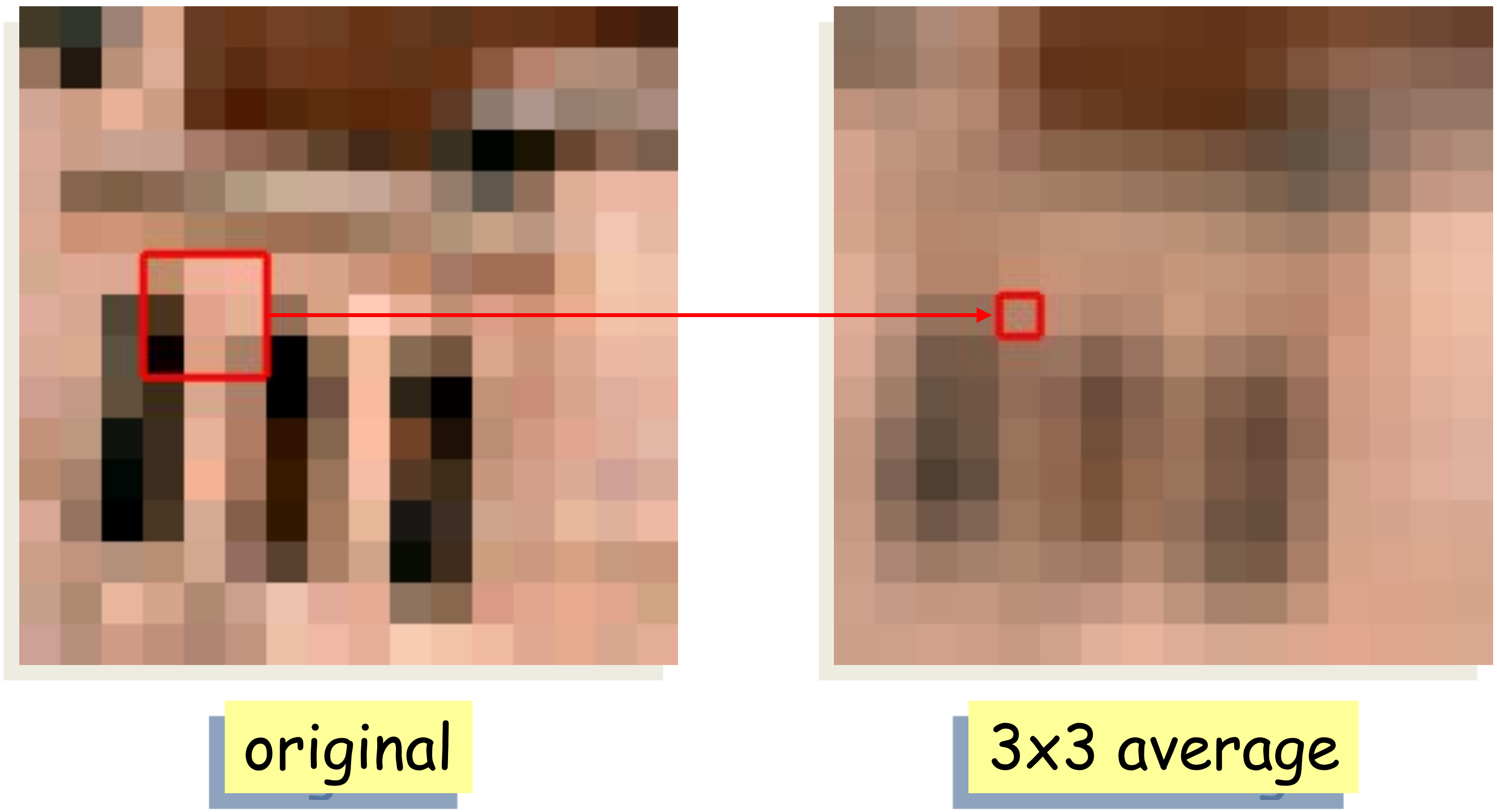
Smoothing Filter: Example

original

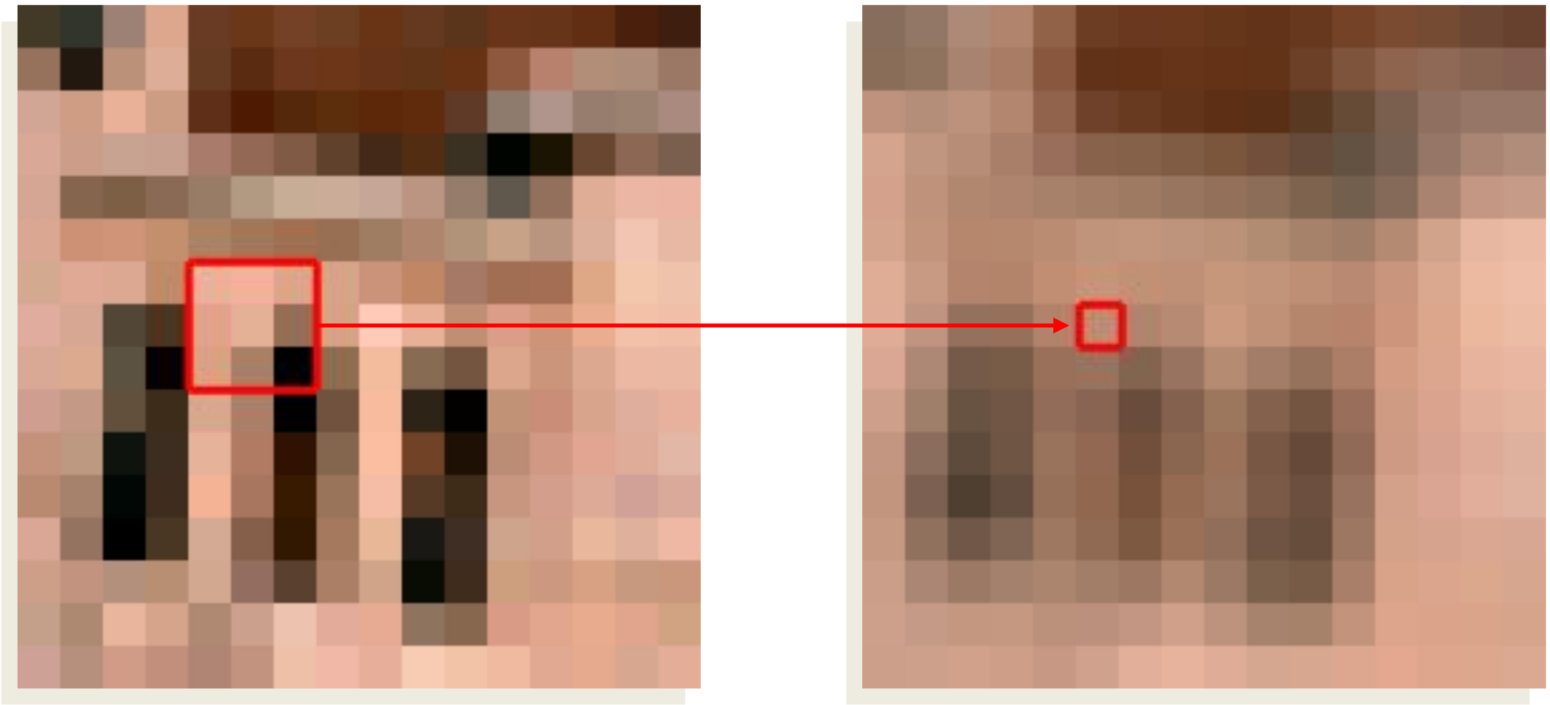


3x3 average

Smoothing Filter: Example



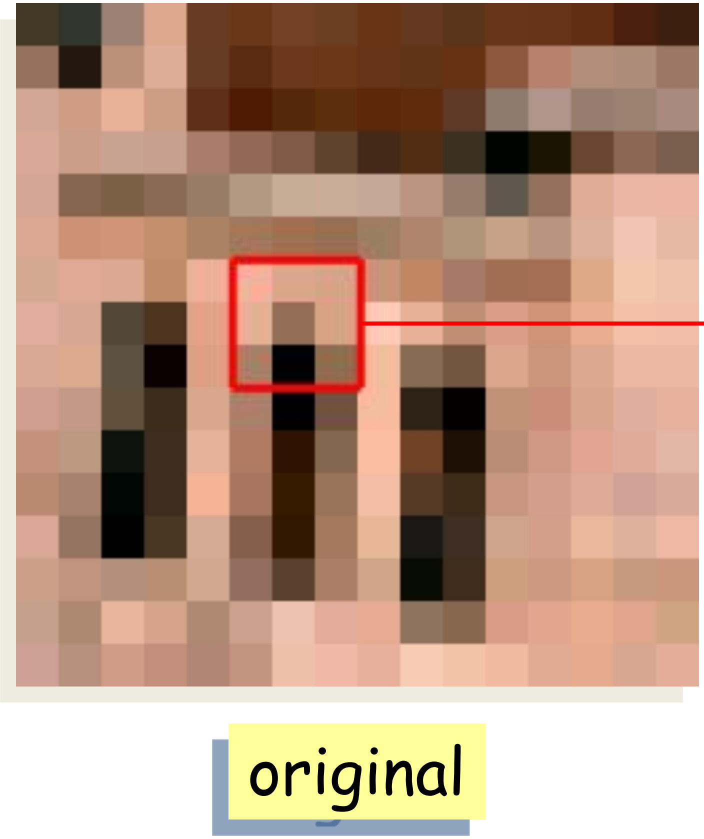
Smoothing Filter: Example



original

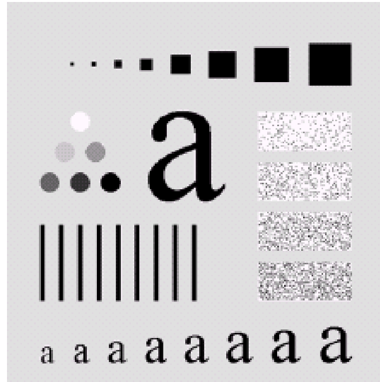
3x3 average

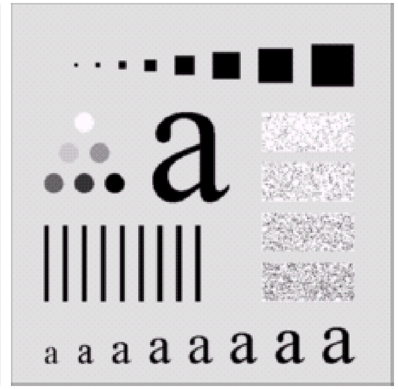
Smoothing Filter: Example



Original image

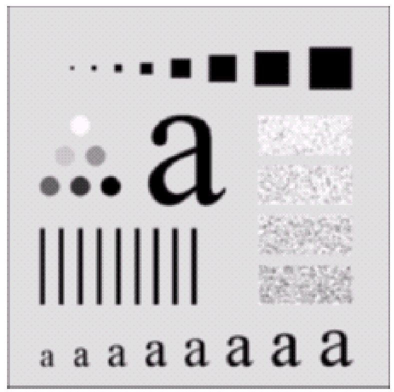
Size: 500x500

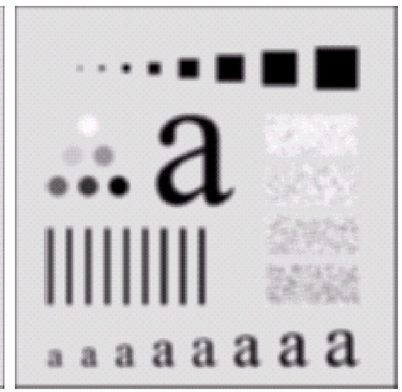




Smooth by 3x3 box filter

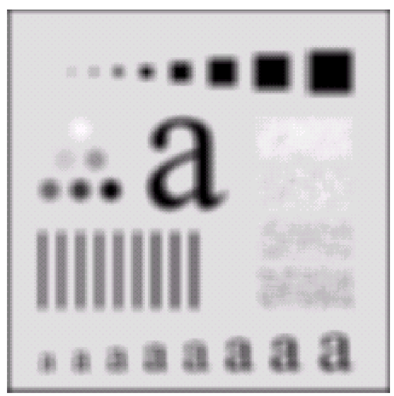
Smooth by 5x5 box filter

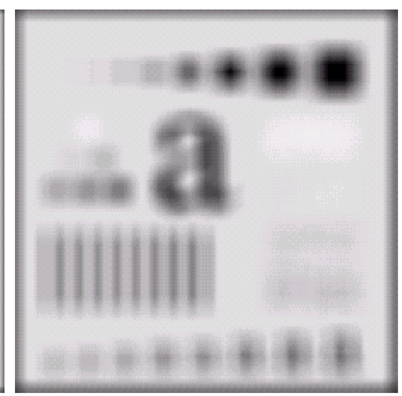




Smooth by 9x9 box filter

Smooth by 15x15 box filter





Smooth by 35x35 box filter

Notice how detail begins to disappear

Smoothing Spatial Filters

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

Consider the output pixel is positioned at the center

Box Filter all coefficients are equal

Weighted Average give more (less) weight to near (away from) the output location

Order-Statistic Filtering

- Output is based on order of gray levels in the masked area
- Some simple neighbourhood operations include:
  - **Min:** Set the pixel value to the minimum in the neighbourhood
  - **Max:** Set the pixel value to the maximum in the neighbourhood
  - **Median:** The median value of a set of numbers is the midpoint value in that set

Median Filter

- For an image, mask symmetric: 3x3, 5x5, etc.

Sorted: 0,0,1,1,1,2,2,2,4

Input

1	2	0	1	3	
2	2	4	2	2	
1	0	1	0	1	
1	2	1	0	2	
2	5	3	1	2	

Output

	1				

Median Filtering

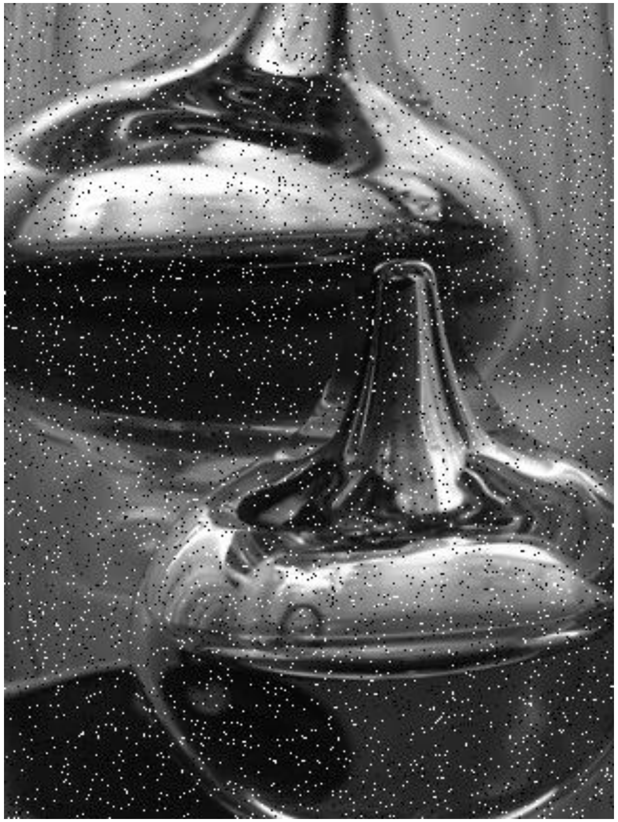
10	20	20
20	15	20
20	25	100

Sort the values
Determine the median

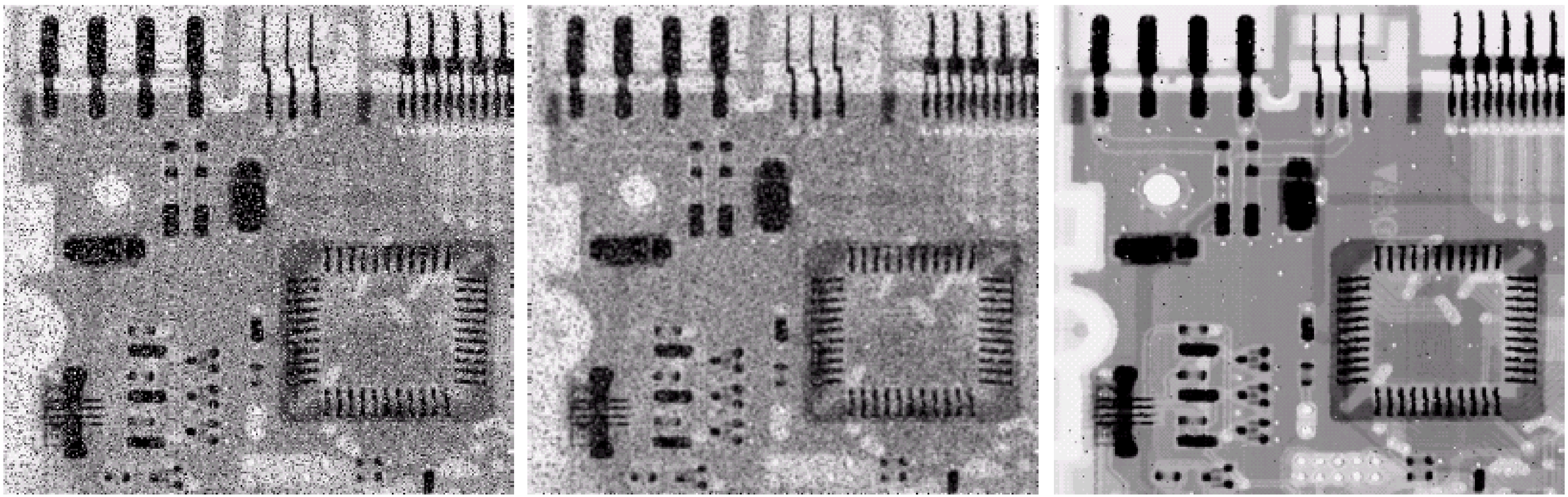
Median = ? 20

- Particularly effective when
  - The noise pattern consists of strong impulse noise (salt-and-pepper)

Salt and Pepper Noise



Median Filtering



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Min/Max Filtering

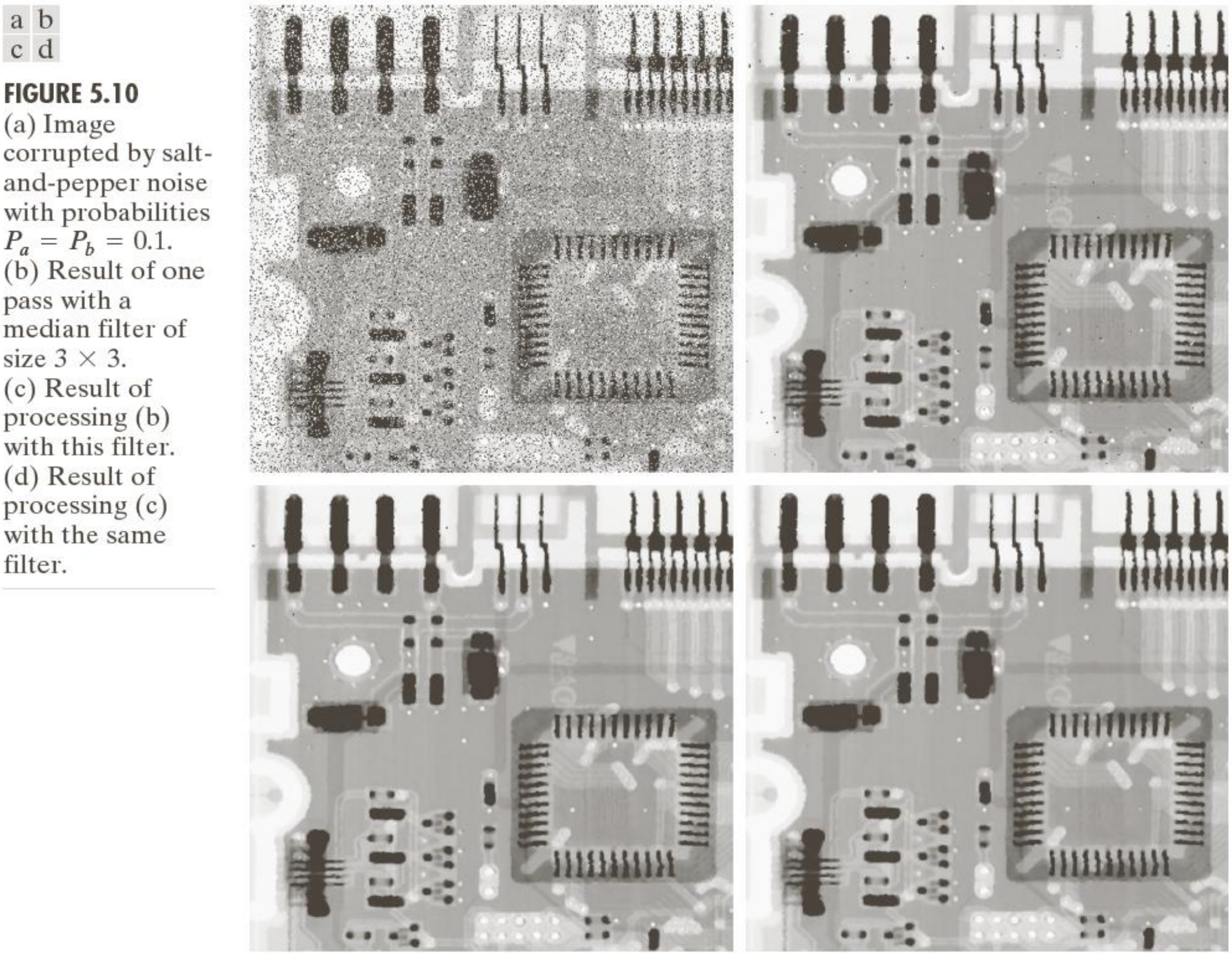


FIGURE 5.10 (a) Image corrupted by salt-and-pepper noise with probabilities $P_a = P_b = 0.1$. (b) Result of one pass with a median filter of size 3×3. (c) Result of processing (b) with this filter. (d) Result of processing (c) with the same filter.

Sharpening Spatial Filters

Previously we have looked at smoothing filters which remove fine detail

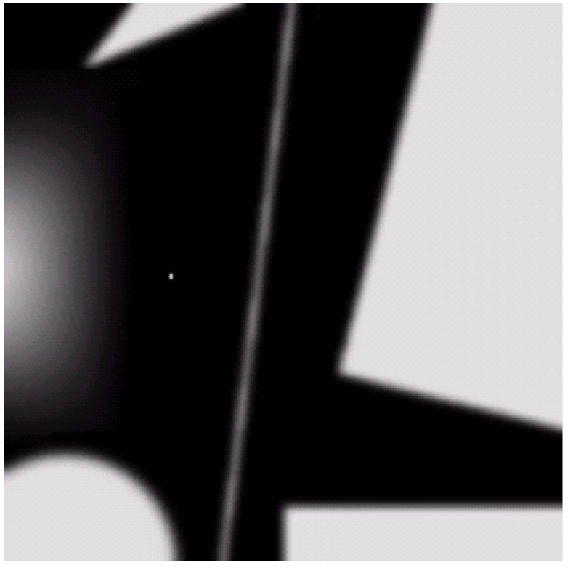
Sharpening spatial filters seek to highlight fine detail

- Remove blurring from images
- Highlight edges

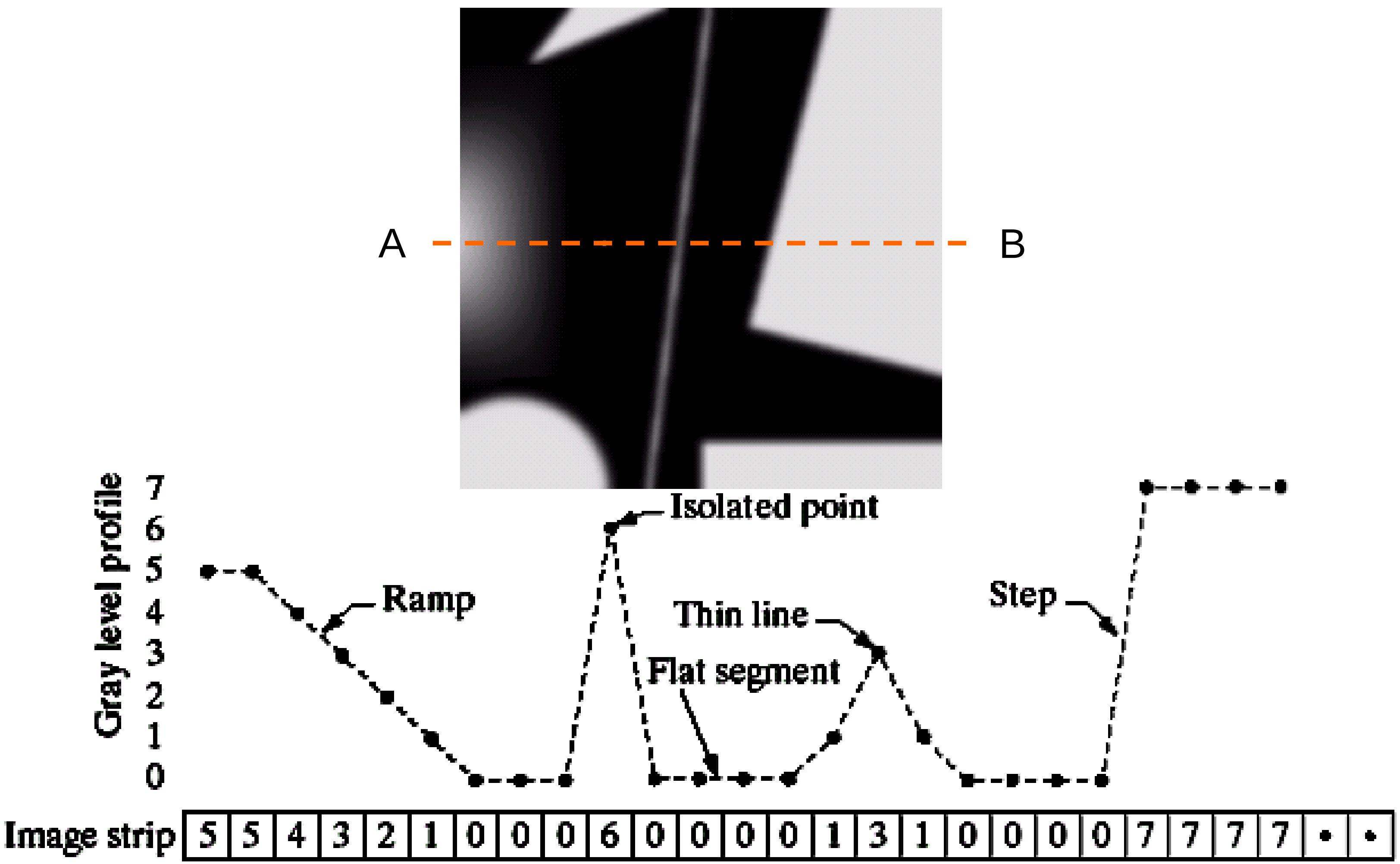
Sharpening filters are based on *spatial differentiation*

Spatial Differentiation

- Let’s consider a simple 1 dimensional example



Spatial Differentiation



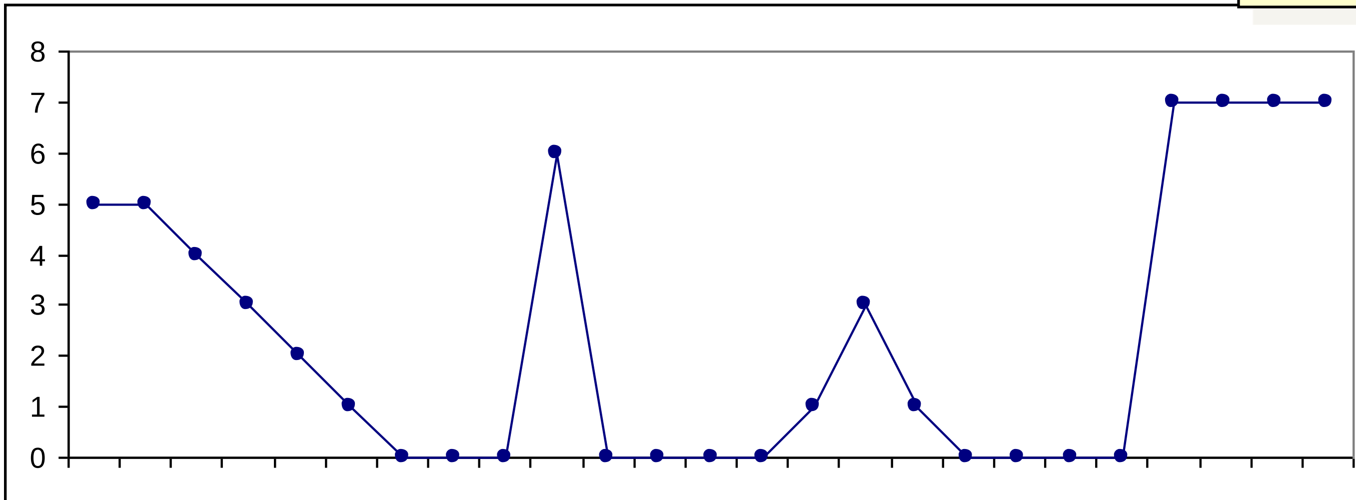
1^{st} Derivative

The 1^{st} derivative of a function is given by:

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

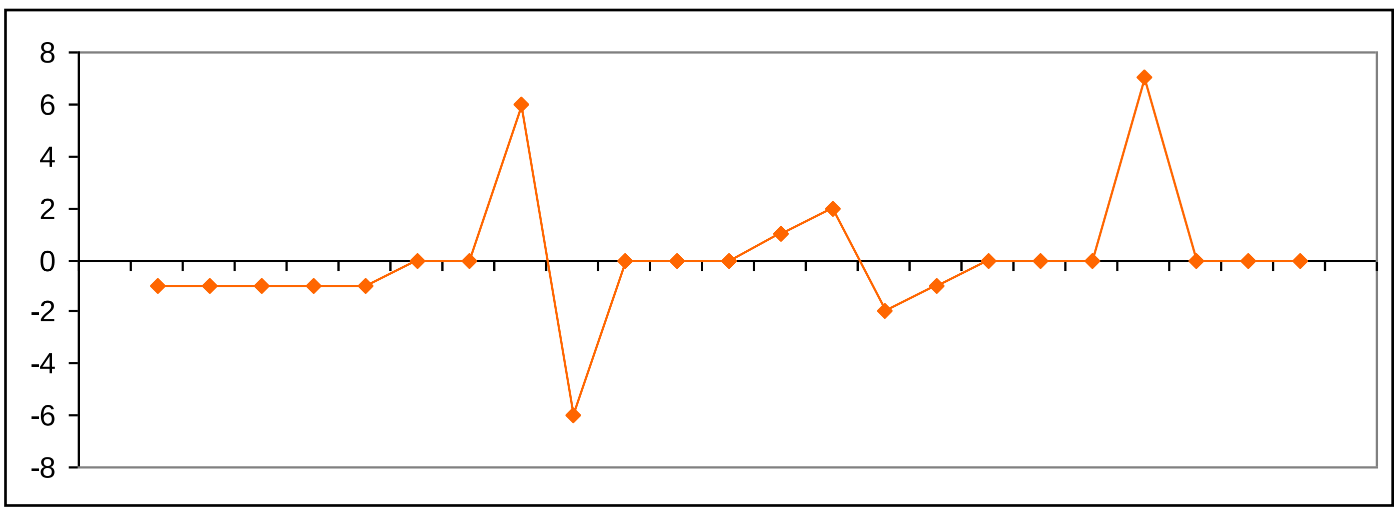
Its just the difference between subsequent values and measures the rate of change of the function

1st Derivative



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	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	0	0	



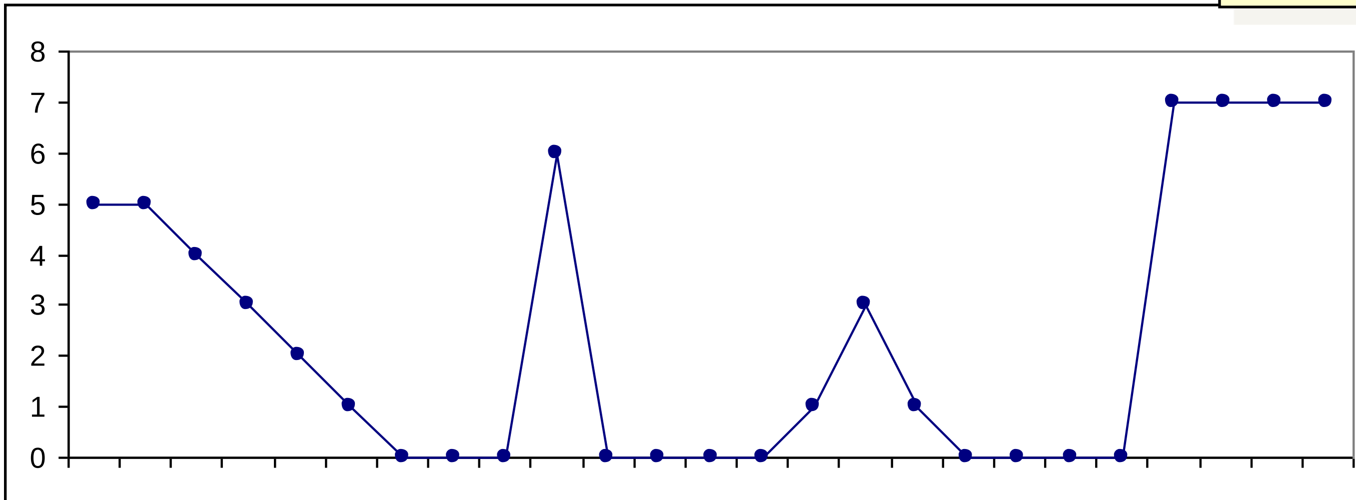
2^{nd} Derivative

The 2nd derivative of a function is given by:

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

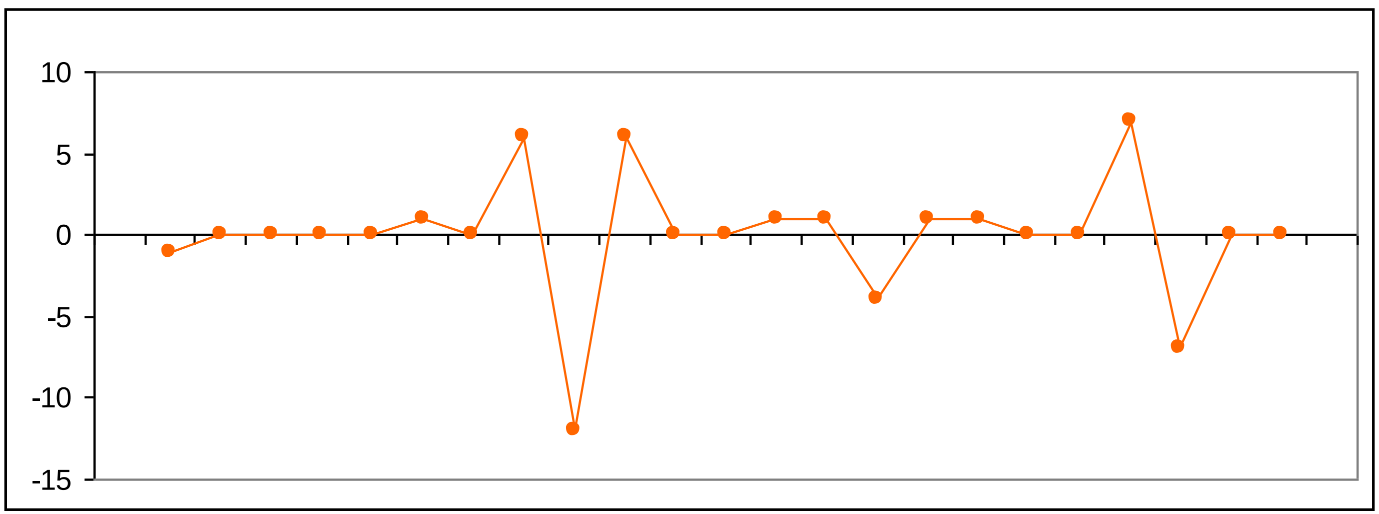
$$\frac{\partial^2 f}{\partial^2 x} = f(x+1) + f(x-1) - 2f(x)$$

2nd Derivative



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	



2^{nd} Derivative for Image Enhancement

The 2nd derivative is more useful for image enhancement than the 1st derivative - *Stronger response to fine detail*

We will come back to the 1st order derivative later on

The first sharpening filter we will look at is the *Laplacian*

Readings from Book (3^{rd} Edn.)

- 3.3 Histogram
- 3.5 Spatial filtering

Acknowledgements

- Digital Image Processing”, Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Peters, Richard Alan, II, Lectures on Image Processing, Vanderbilt University, Nashville, TN, April 2008
- Brian Mac Namee, Digitial Image Processing, School of Computing, Dublin Institute of Technology
- Computer Vision for Computer Graphics, Mark Borg

Material in these slides has been taken from, the following resources