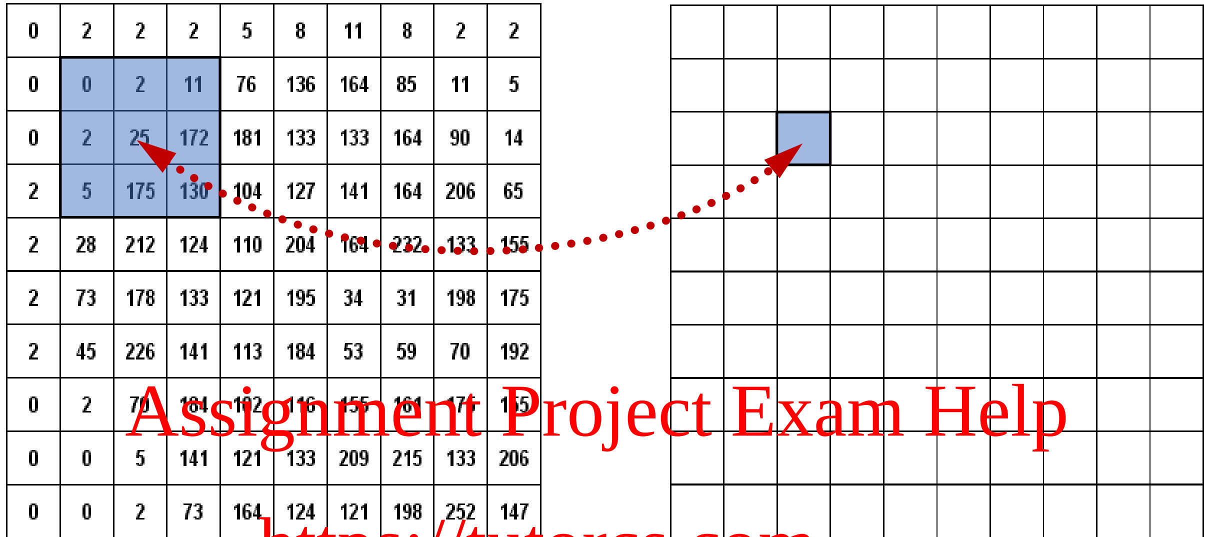


COMP9517: Computer Vision

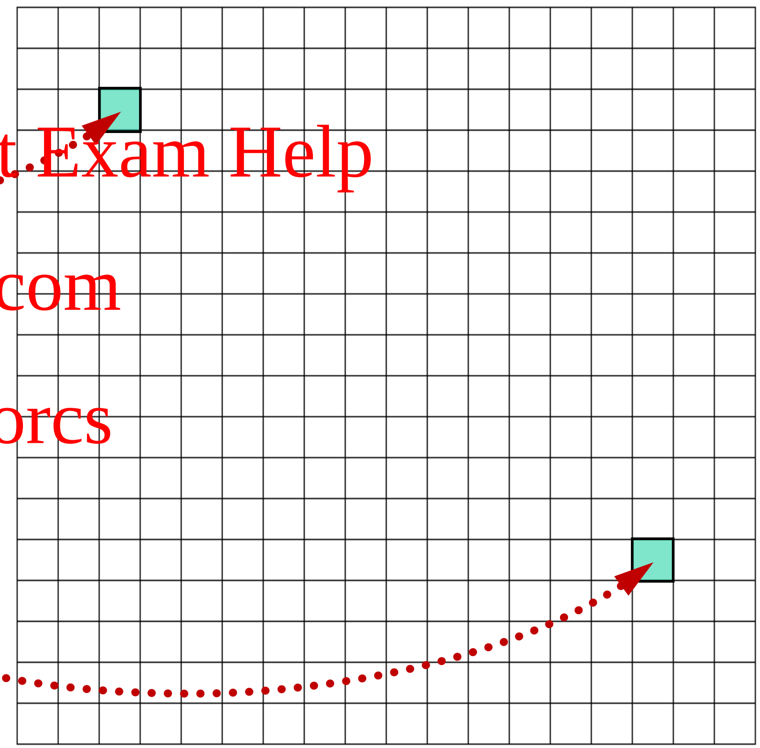
Image Processing II

Types of image processing (recap)

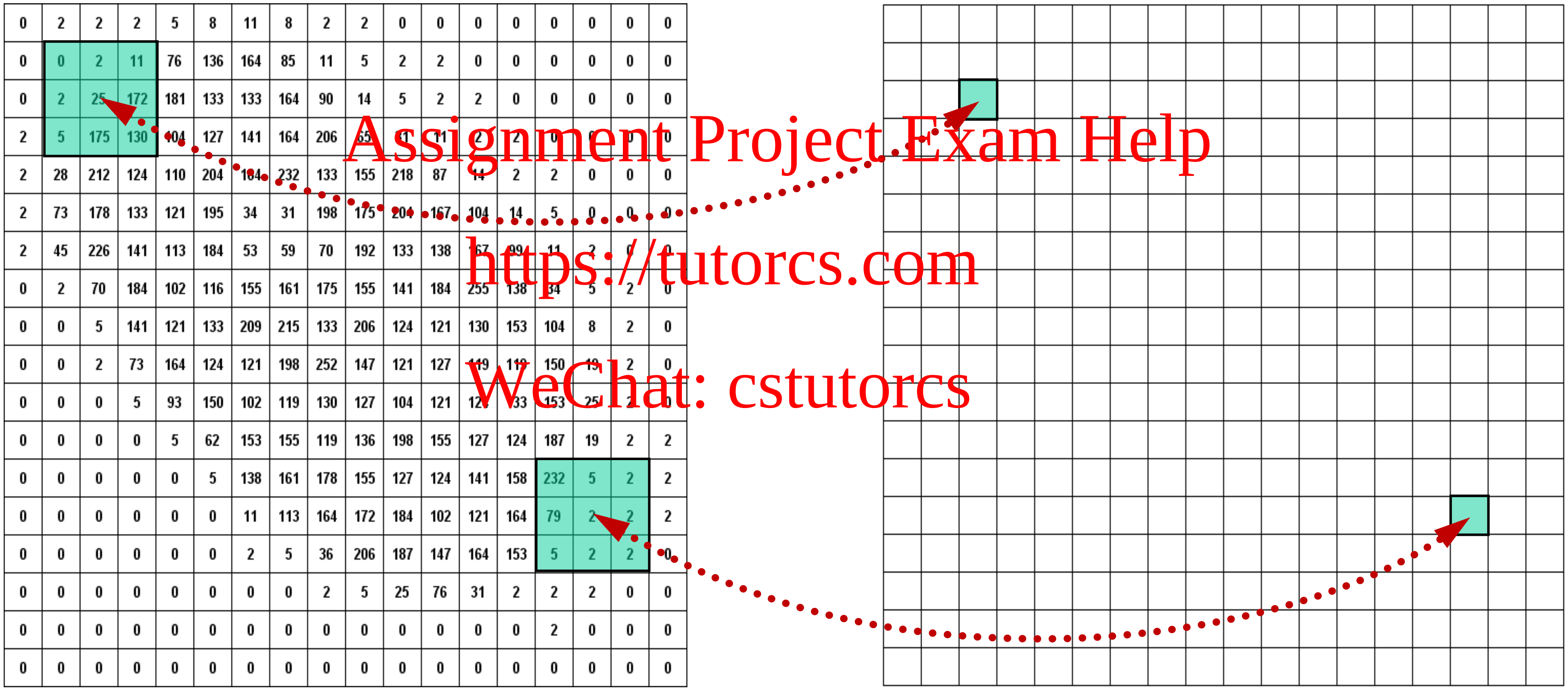
- Two main types of image processing operations:
  - Spatial domain operations (in image space)
  - Frequency domain operations (mainly in Fourier space)
- Two main types of spatial domain operations:
  - Point operations (intensity transformations on individual pixels)
  - Neighbourhood operations (spatial filtering on groups of pixels)

Point operations

0	2	2	2	5	8	11	8	2	2	0	0	0	0	0	0	0	0
0	0	2	11	76	136	164	85	11	5	2	2	0	0	0	0	0	0
0	2	25	172	181	133	133	164	90	14	5	2	2	0	0	0	0	0
2	5	175	130	104	127	141	164	206	65	[illegible]	11	2	2	0	0	0	0
2	28	212	124	110	204	104	232	133	155	218	87	14	2	2	0	0	0
2	73	178	133	121	195	34	31	198	175	204	167	104	14	5	0	0	0
2	45	226	141	113	184	53	59	70	192	133	138	167	99	11	2	0	0
0	2	70	184	102	116	155	161	175	155	141	184	255	138	34	5	2	0
0	0	5	141	121	133	209	215	133	206	124	121	130	153	104	8	2	0
0	0	2	73	164	124	121	198	252	147	121	127	[illegible]	119	150	19	2	0
0	0	0	5	93	150	102	119	130	127	104	121	124	133	153	25	2	0
0	0	0	0	5	62	153	155	119	136	198	155	127	124	187	19	2	2
0	0	0	0	0	5	138	161	178	155	127	124	141	158	232	5	2	2
0	0	0	0	0	0	11	113	164	172	184	102	121	164	79	2	2	2
0	0	0	0	0	0	2	5	36	206	187	147	164	153	5	2	2	0
0	0	0	0	0	0	0	0	2	5	25	76	31	2	2	2	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



Neighbourhood operations



Recap

Spatial domain, intensity transformations (on single pixels)

- Image thresholding
  - Otsu's method
  - Histogram thresholding
  - Multiband thresholding
- Image inversion
- Log transform
- Power-law
- Averaging

Recap

Spatial domain, intensity transformations (on single pixels)

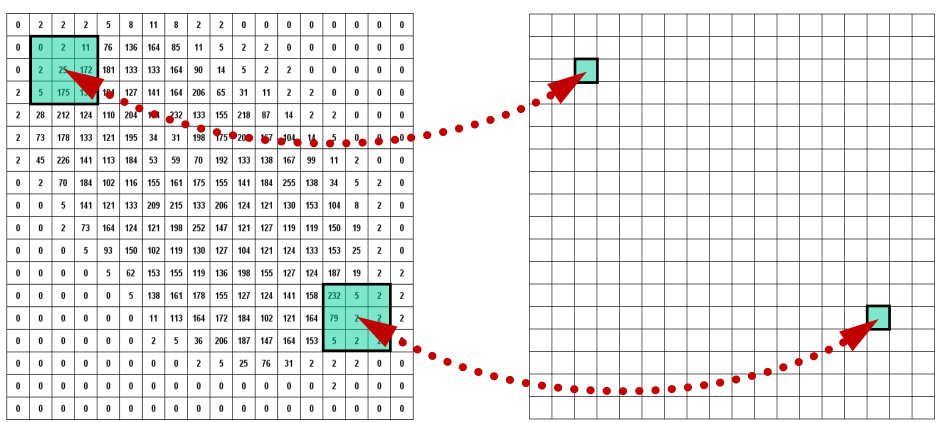
- Piecewise-linear transformation

    - Contrast stretching
    - Gray-level slicing

    - Bit-plane slicing
- Histogram processing
    - Histogram equalization
    - Histogram matching

Spatial Filtering

- These methods use a small **neighbourhood** of a pixel in the input image to produce a new brightnesses value for that pixel
- Also called **filtering** techniques
- Neighbourhood of (x, y) is usually a square or rectangular subimage centred at (x, y).
- **filter / mask / kernel / template / window** is used to indicate the concepts of the subimage or the corresponding operators, in different contexts.



Spatial Filtering

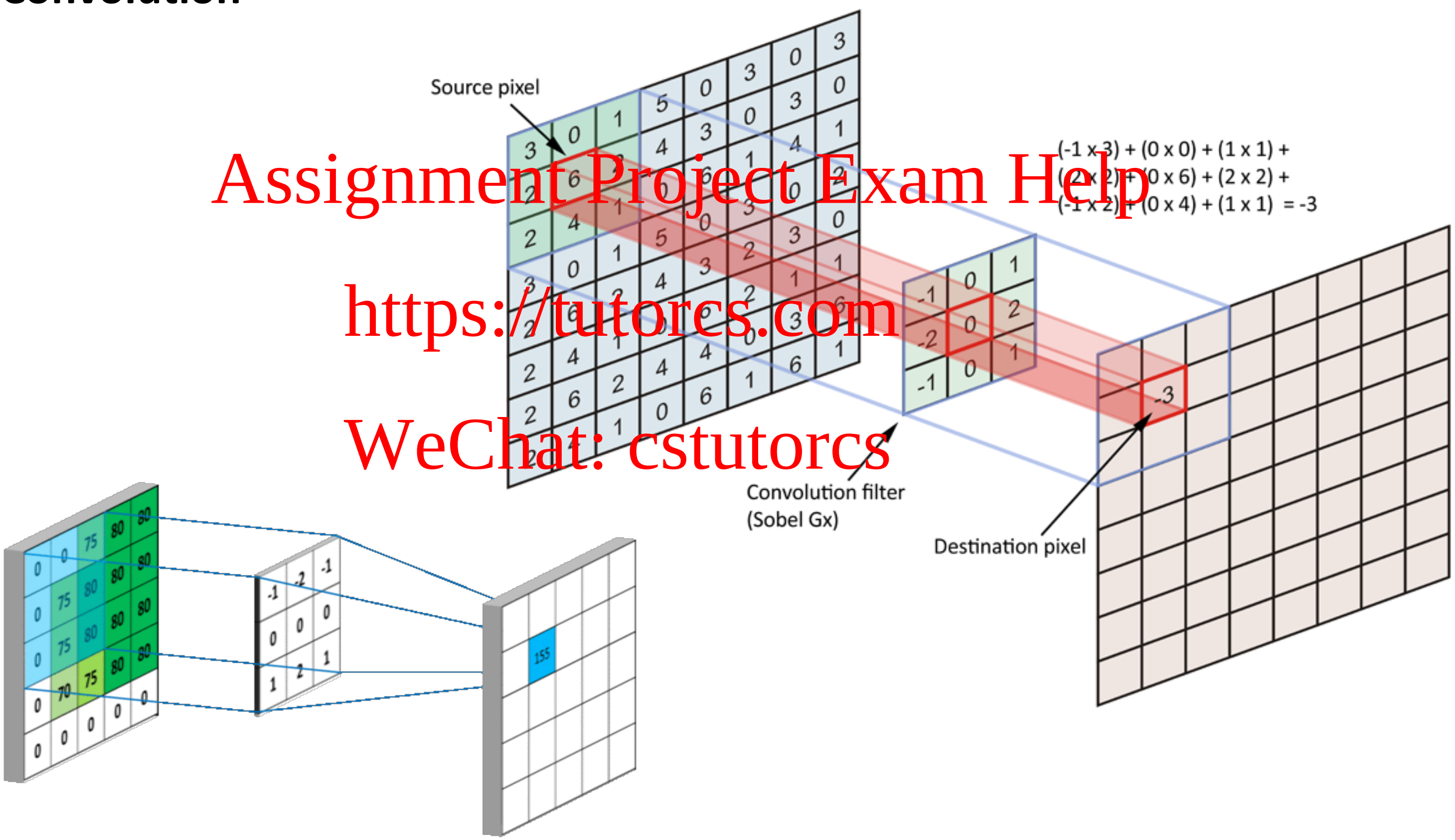
- These methods use a small **neighbourhood** of a pixel in the input image to produce a new brightnesses value for that pixel
- Also called **filtering** techniques
- Neighbourhood of (x, y) is usually a square or rectangular subimage centred at (x, y).
- A **linear transformation** calculates a value in the output image $g(i, j)$ as a linear combination of brightnesses in a local neighbourhood of the pixel in the input image $f(i, j)$ weighted by coefficients h:

$$g(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} h(i, j) f(x - i, y - j)$$

- This is called a **discrete convolution** with the convolution mask/filter/kernel h

Spatial Filtering

Convolution



Smoothing Spatial Filters

Neighbourhood Averaging (Mean Filter)

- The most basic filter, used for image blurring/smoothing and noise reduction

$$g(x, y) = \frac{1}{P} \sum_{(n,m) \in S} f(n, m)$$

- Replace intensity at pixel (x, y) with the **average** of the intensities in a neighbourhood of (x, y)

- We can also use a **weighted average**, giving more importance to some pixels over others in the neighbourhood – can reduce blurring effect
- Neighbourhood averaging blurs edges

Smoothing Spatial Filters - Examples

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

a b

FIGURE 3.34 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to the sum of the values of its coefficients, as is required to compute an average.

Digital Image Processing - Chapter 3, Image Enhancement in the Spatial Domain

Smoothing Spatial Filters - Examples

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.)

Digital Image Processing - Chapter 3, Image Enhancement in the Spatial Domain

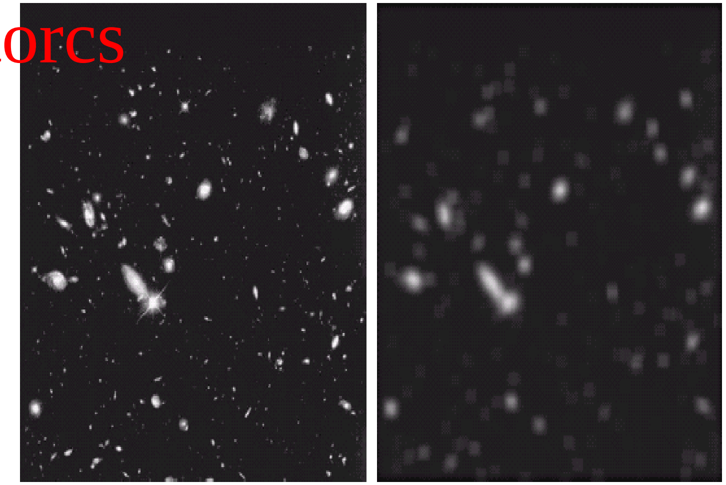
Smoothing Spatial Filters

Smoothing Spatial Filters

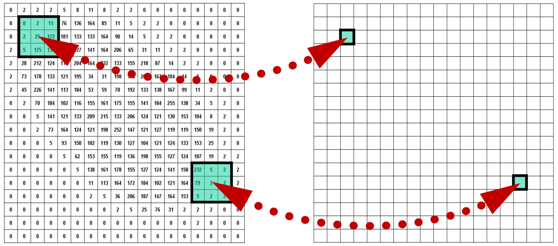
- **Aim:** To suppress noise, other small fluctuations in image- may be result of sampling, quantization, transmission, environment disturbances during acquisition
- Uses redundancy in the image data
- May blur sharp edges, so care is needed

What if the filter is

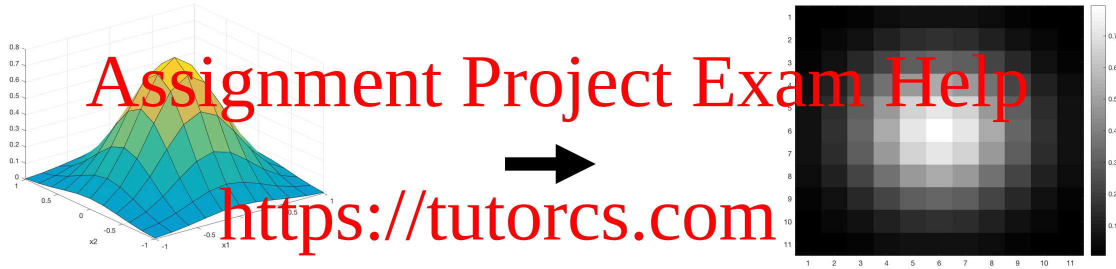
$$\begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix} \quad \text{or} \quad \begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{matrix} \; ?$$



Gaussian Filter

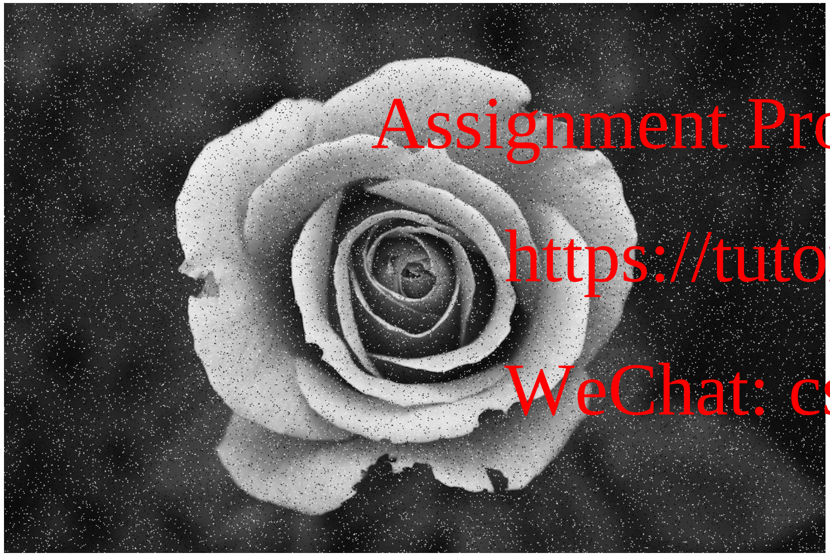


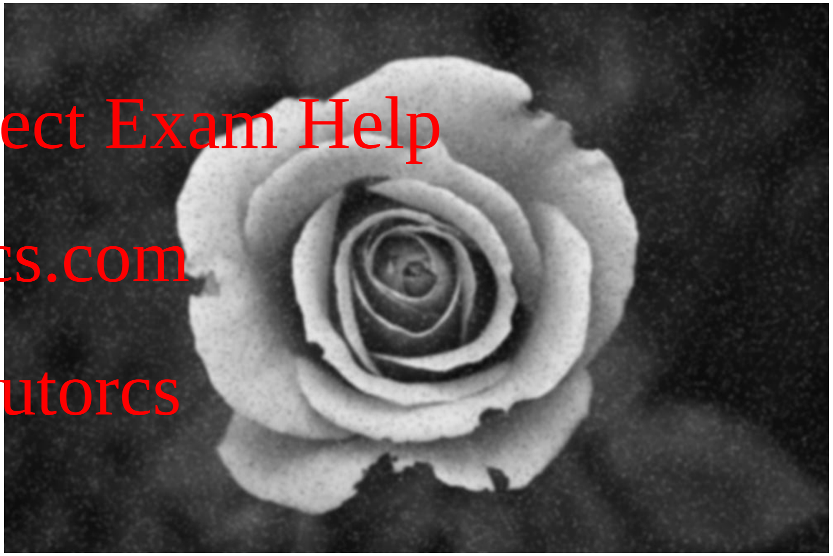
$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\left[\frac{x^2+y^2}{2\sigma^2}\right]}$$



- Replace intensity at pixel (x, y) with the **weighted average** of the intensities in a neighbourhood of (x, y)
- It is a set of weights that approximate the profile of a Gaussian function
- It is very effective in reducing noise and also reducing details (image blurring)

Gaussian Filter



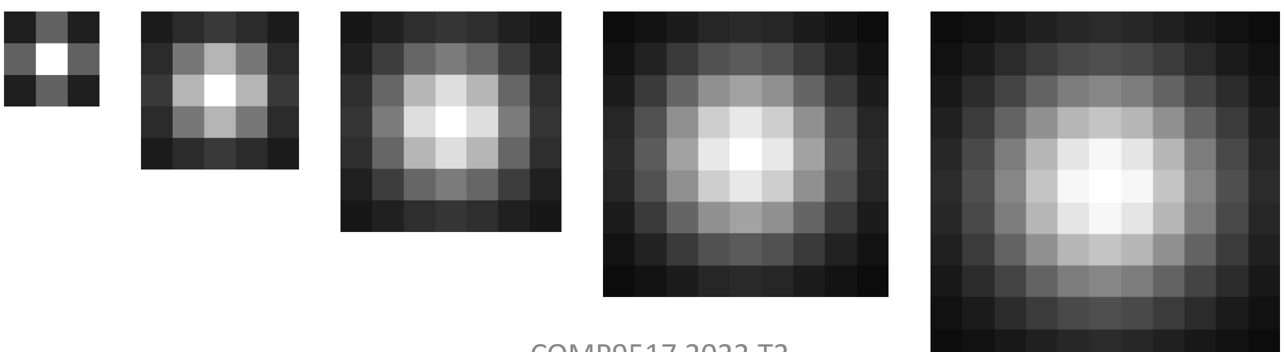


Gaussian Filter

Many nice properties motivate the use of the Gaussian filter

- It is the only filter that is both separable and circularly symmetric
- It has optimal space-frequency localization
- The Fourier transform of a Gaussian is also a Gaussian function
- The *n*-fold convolution of *any* low-pass filter converges to a Gaussian
- It is infinitely smooth so it can be differentiated to any desired degree
- It scales naturally (sigma) and allows for consistent scale-space theory

3 x 3	5 x 5	7 x 7	9 x 9	11 x 11
$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 1.5$	$\sigma = 2.0$	$\sigma = 2.5$



Nonlinear Spatial Filters

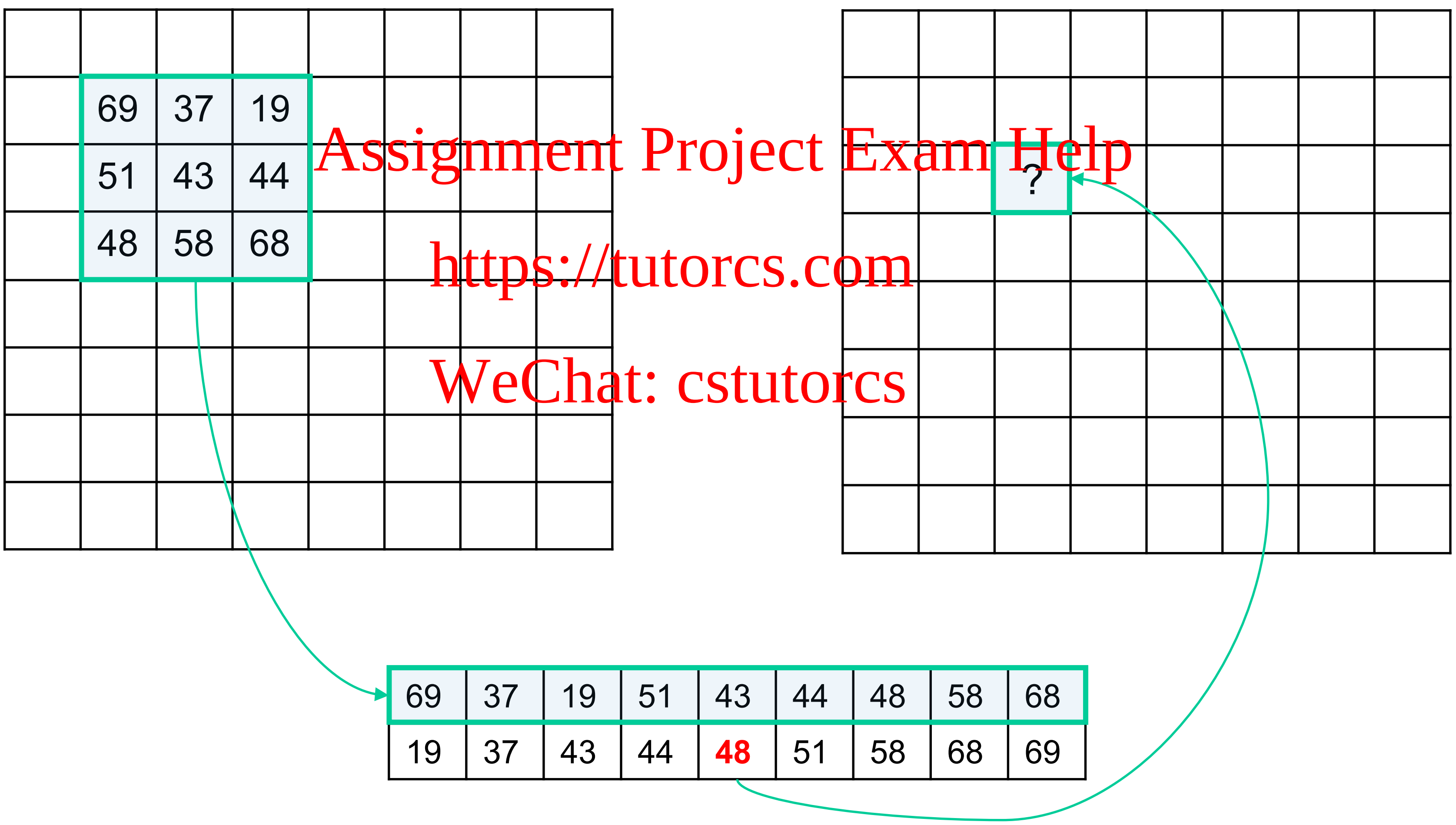
Referring to **order-statistics filters** in many cases: response based on **ordering** the pixels in the neighbourhood and replacing centre pixel with the ranking result

Median Filter

- Intensity of each pixel is replaced by the *median* of the intensities in neighbourhood of that pixel

- Median M of a set of values is the middle value such that half the values in the set are less than M and the other half greater than M

Median Filter



Nonlinear Spatial Filters

Median Filter

- Intensity of each pixel is replaced by the *median* of the intensities in neighbourhood of that pixel
- Median filtering forces points with distinct intensities to be more like their neighbours, thus eliminating isolated intensity spikes
- Also, isolated pixel clusters (light or dark), whose area is $\leq n^2/2$ are eliminated by an $n \times n$ median filter
- Good for impulse noise (salt-and-pepper noise)
- Other examples of order-statistics filters are max and min filters

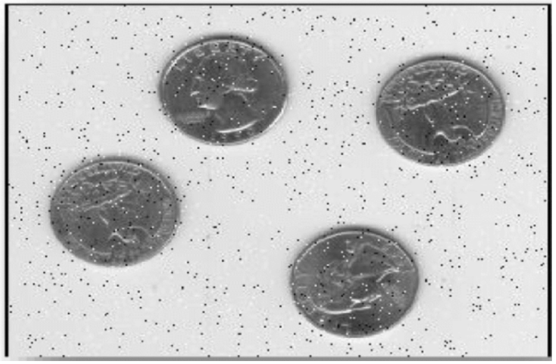
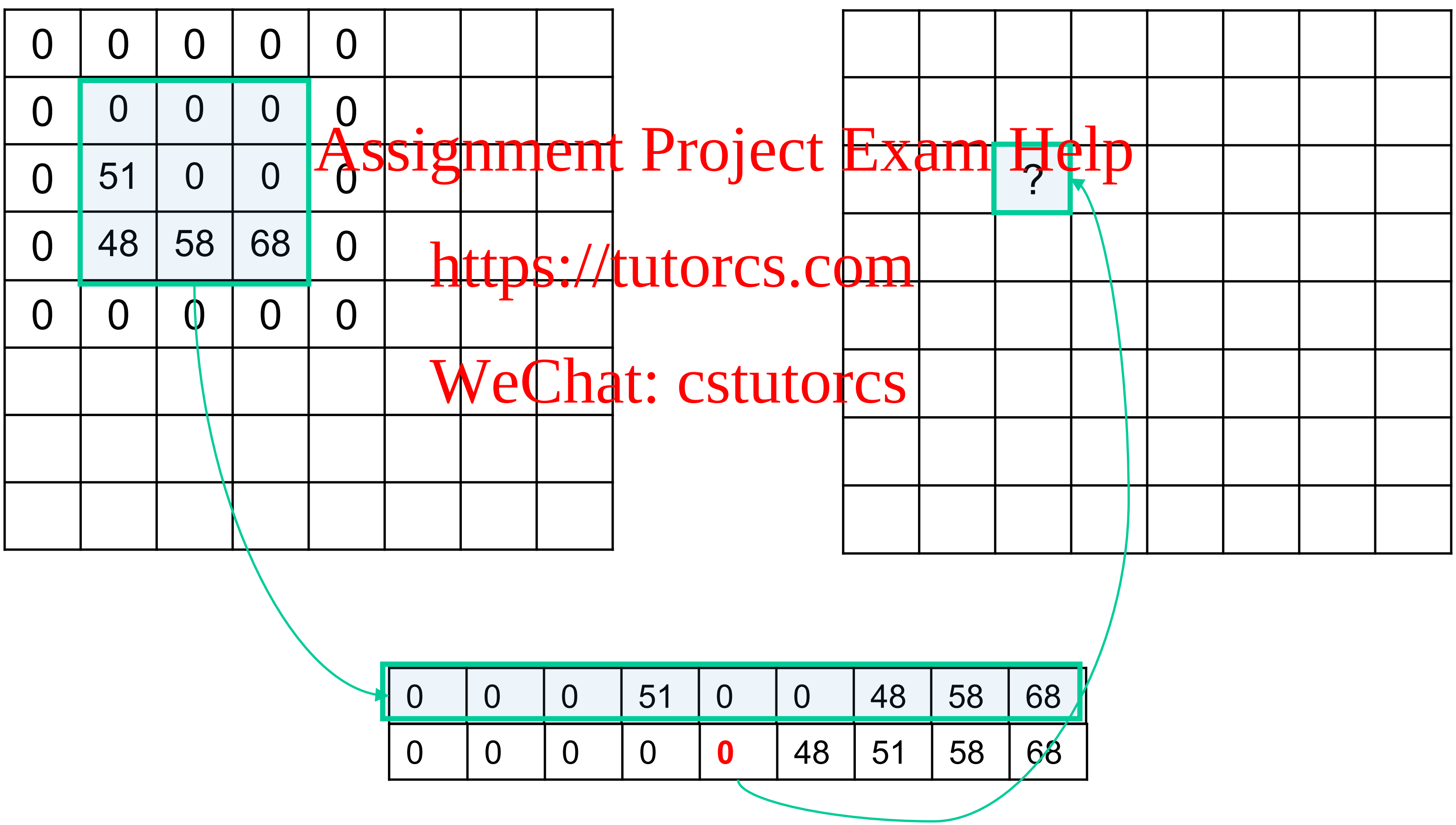
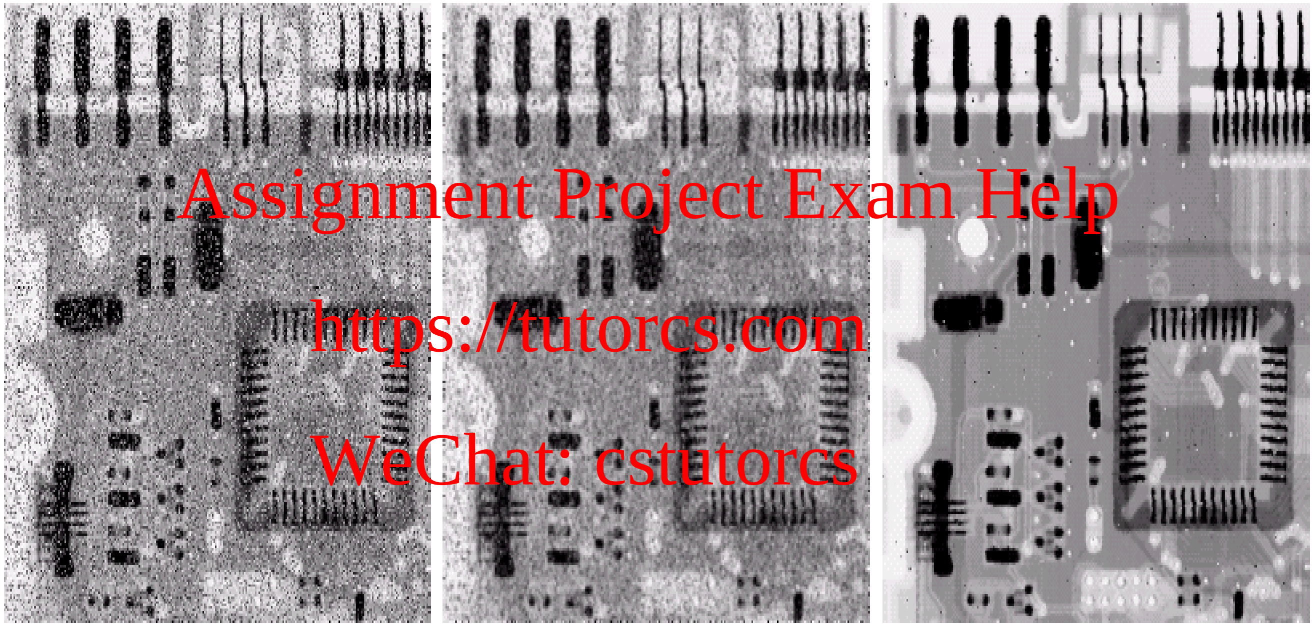


Image with impulse noise (salt-and-pepper noise)

Median Filter



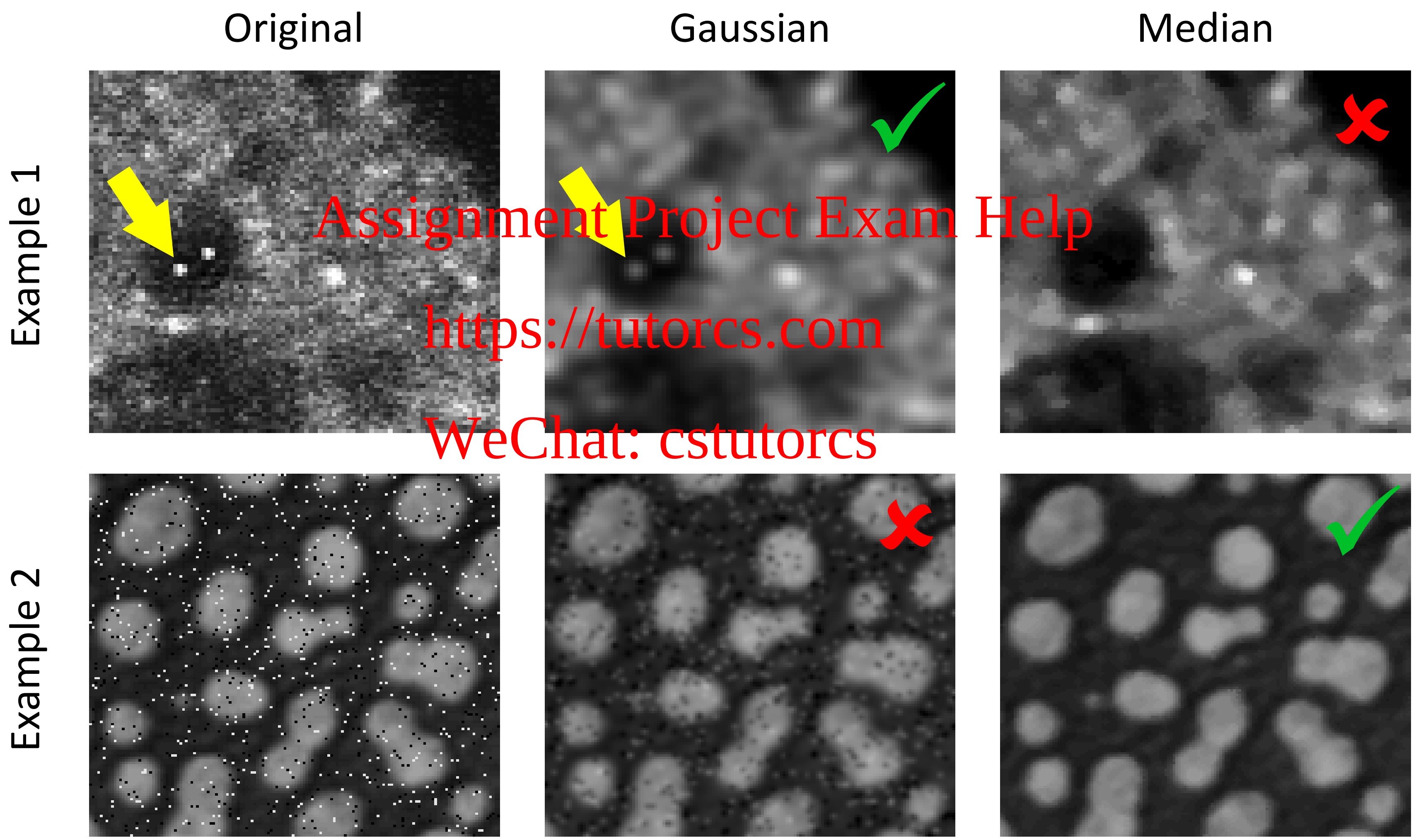
Chapter 3
Image Enhancement in the Spatial Domain



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.)

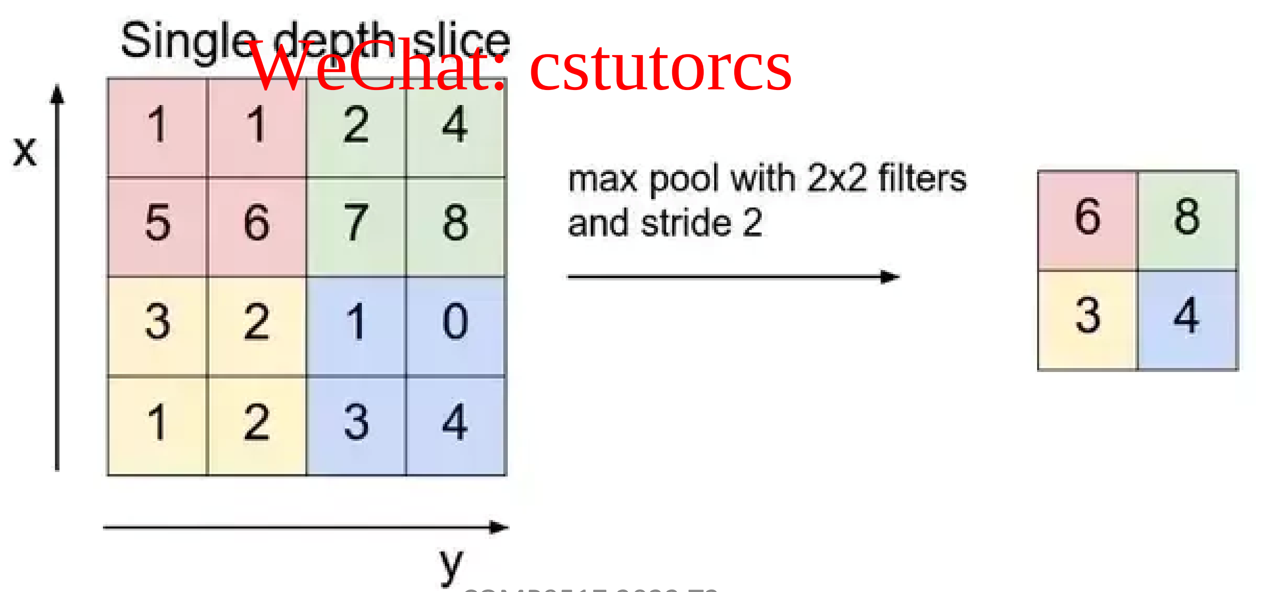
Gaussian Versus Median Filtering



Pooling

Max / average / median pooling

- Provides translation invariance
- Reduces computations
- Popular in deep convolutional neural networks (deep learning)
- Extracting the "most essential/significant" information



Sharpening Spatial Filters
Edge Detection

- Goal is to highlight fine detail, or enhance detail that has been blurred
- Spatial differentiation is the tool; strength of response of derivative operator is proportional to degree of discontinuity of the image at the point where operator is applied
- Image differentiation enhances edges, and de-emphasizes slowly varying gray-level values.

Derivative Definitions

For 1-D function f(x), the first order derivative is approximated as:

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

The second-order derivative is approximated as:

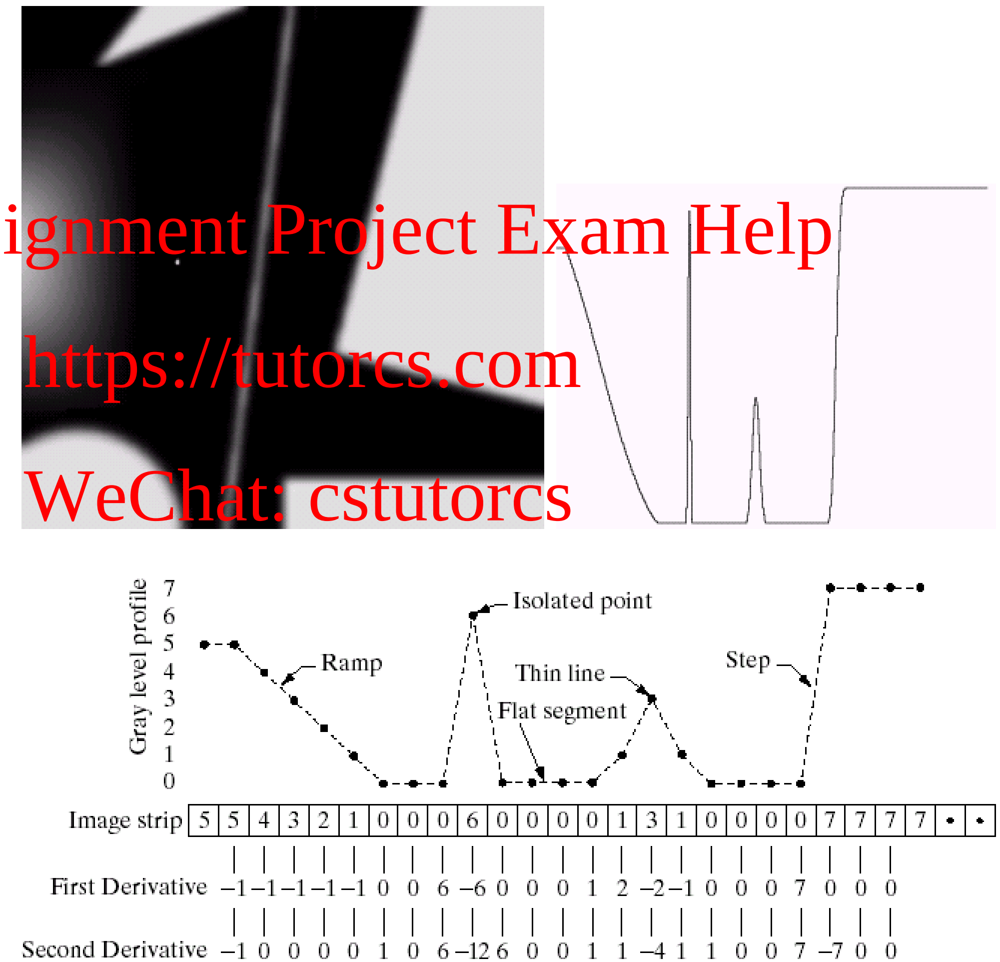
$$\frac{\partial^2 f}{\partial x^2} = f(x+1) - 2f(x) + f(x-1)$$

These are partial derivatives, so extension to 2D is easy

Chapter 3
Image Enhancement in the Spatial Domain

a b
c

FIGURE 3.38
(a) A simple image. (b) 1-D horizontal gray-level profile along the center of the image and including the isolated noise point.
(c) Simplified profile (the points are joined by dashed lines to simplify interpretation).

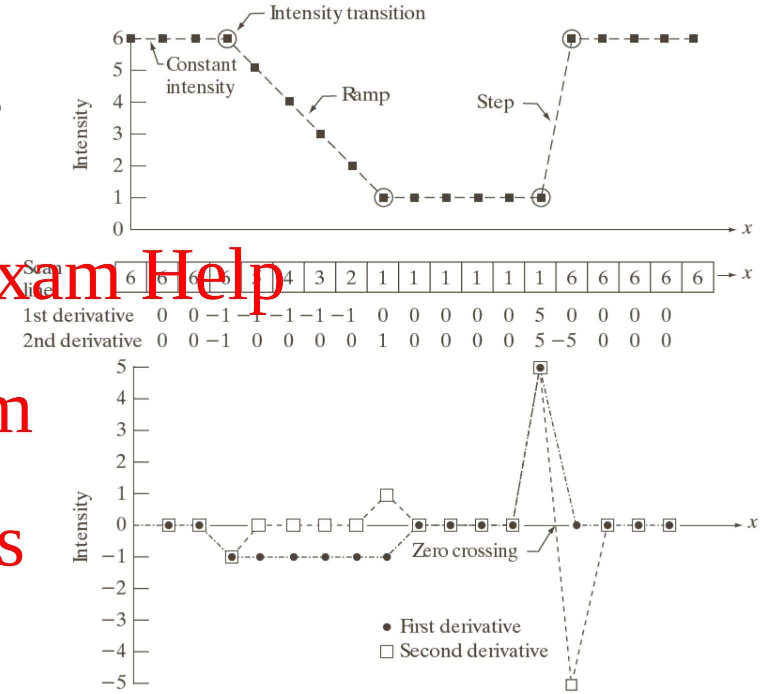


Basic Idea

- Horizontal scan of the image
- Edge modelled as a ramp to represent blurring due to sampling
- First derivative is
  - Non-zero along ramp
  - zero in regions of constant intensity
  - constant during an intensity transition
- Second derivative is
  - Nonzero at onset and end of ramp
  - Stronger response at isolated noise point
  - zero everywhere except at onset and termination of intensity transition
- Thus, magnitude of first derivative can be used to detect the presence of an edge, and sign of second derivative to determine whether a pixel lies on dark or light side of an edge.

Summary

- First-order derivatives produce thicker edges, have stronger response to gray-level step
- Second-order derivatives produce stronger response to fine detail (thin lines, isolated points), produce double response at step changes in gray level



a
b
c

FIGURE 3.36 Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image. In (a) and (c) data points are joined by dashed lines as a visualization aid.

Gradient Operator

First-order derivatives implemented using magnitude of the gradient

For function f the gradient ∇f at (x, y) has components $f_x = \frac{\partial f}{\partial x}, f_y = \frac{\partial f}{\partial y}$

The magnitude of the gradient vector is

$$\|\nabla f\| = \sqrt{f_x^2 + f_y^2}$$

This is sometimes approximated as $\|\nabla f\| = |f_x| + |f_y|$

f_x and f_y are linear and may be obtained by using masks

We use numerical techniques to compute these, giving rise to different masks, e.g. Roberts' 2 x 2 cross-gradient operators, Sobel's 3 x 3 masks

Chapter 3
Image Enhancement in the Spatial Domain

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

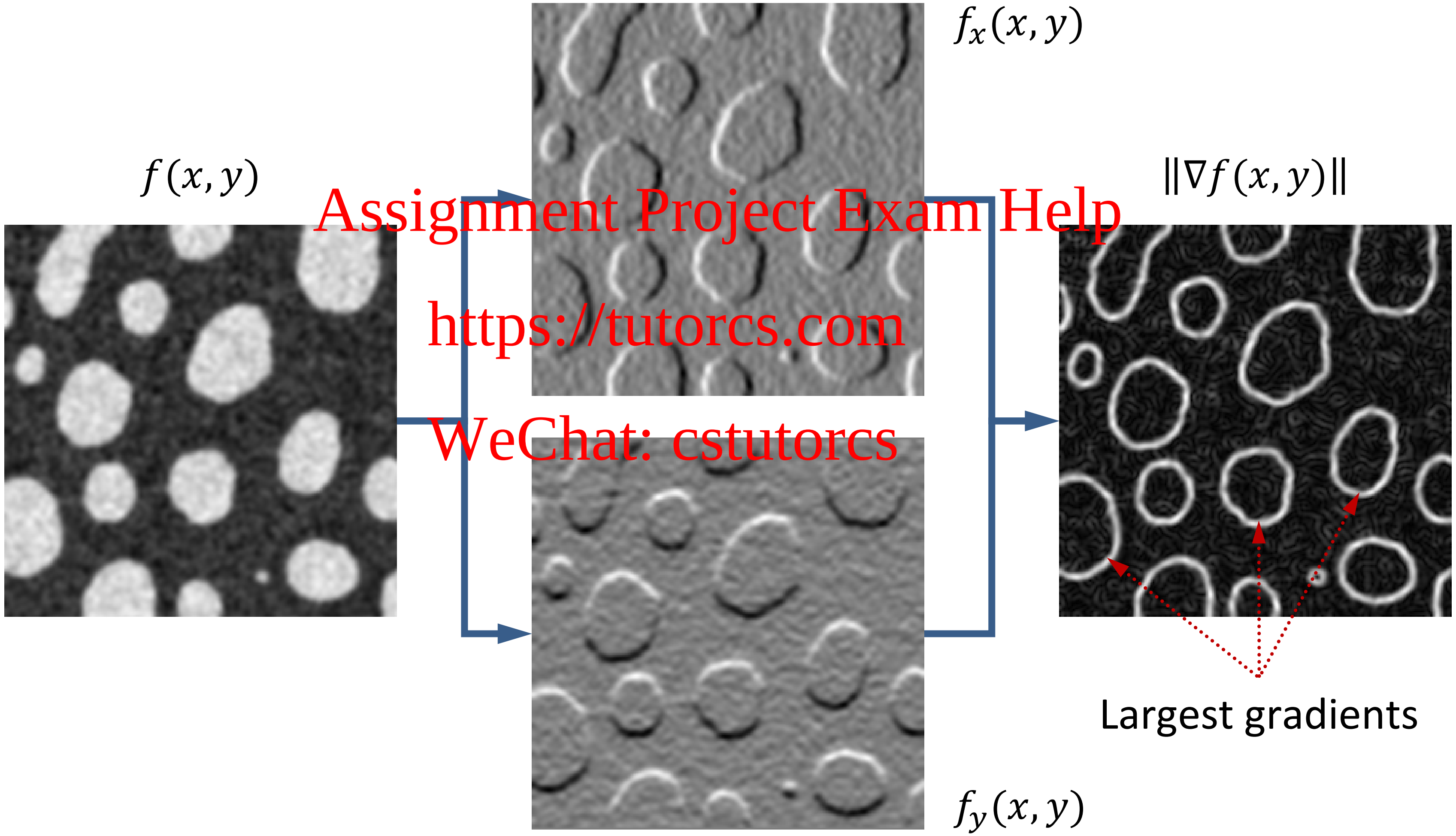
−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

a
b c
d e

FIGURE 3.44 A 3×3 region of an image (the z's are gray-level values) and masks used to compute the gradient at point labeled z_5. All masks coefficients sum to zero, as expected of a derivative operator.

Gradient Operator



Laplacian Operator

Second order derivatives based on the Laplacian.

For a function $f(x, y)$ the Laplacian is defined by

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

This is a linear operator, as all derivative operators are.

In discrete form:

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

and similarly in *y* direction.

Summing them gives us

$$\Delta^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) \ - 4f(x, y)$$

Chapter 3
Image Enhancement in the Spatial Domain

(a)

0	1	0
1	–4	1
0	1	0

(b)

1	1	1
1	–8	1
1	1	1

(c)

0	–1	0
–1	4	–1
0	–1	0

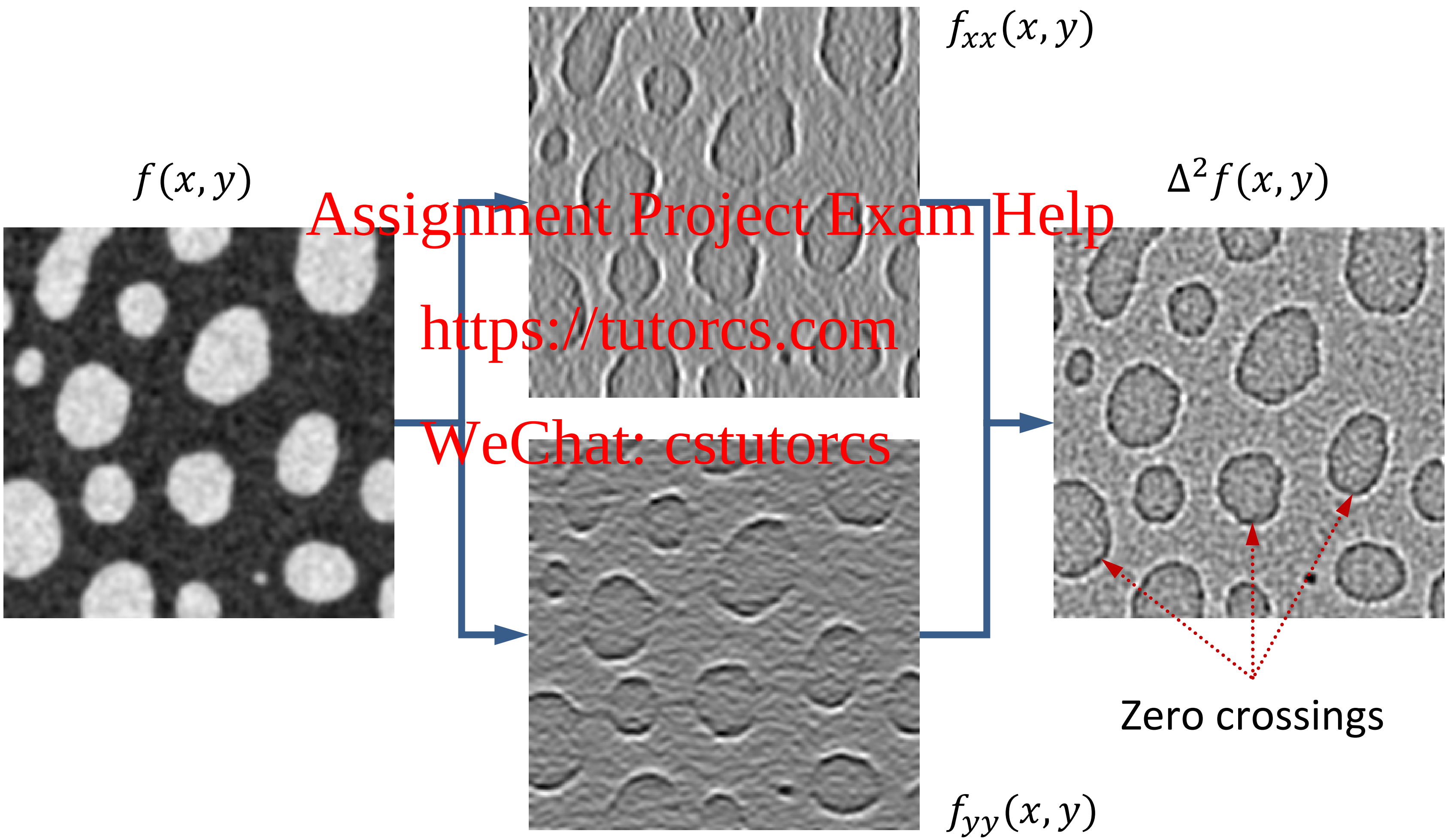
(d)

–1	–1	–1
–1	8	–1
–1	–1	–1

a b
c d

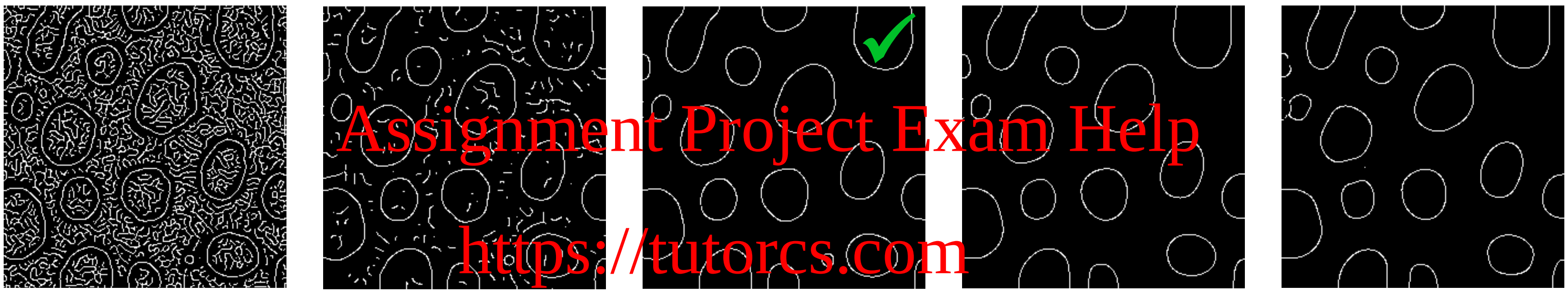
FIGURE 3.39 (a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Laplacian Operator



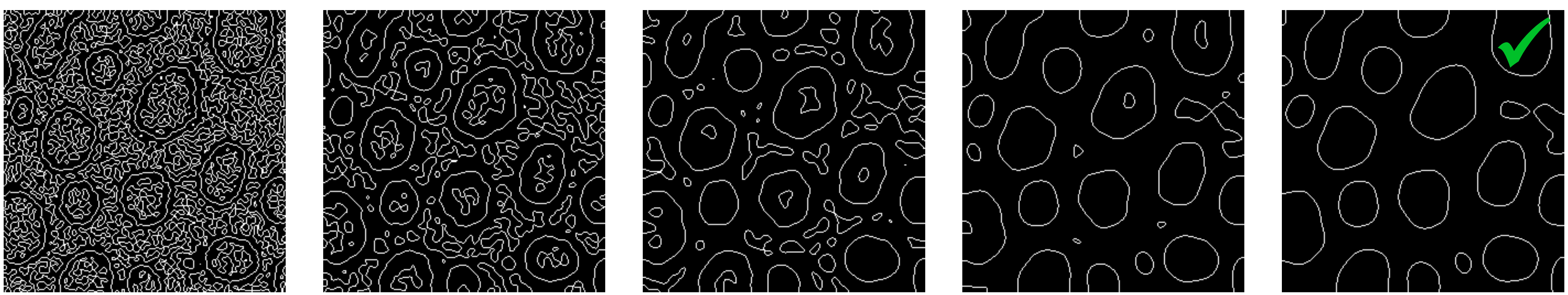
Gradient Versus Laplacian Edge Detection

Edges from thresholding local maxima of the gradient magnitude image



$\sigma = 1$ $\sigma = 3$ $\sigma = 5$ $\sigma = 7$ $\sigma = 9$

Edges from finding the zero-crossings of the Laplacian image



$\sigma = 1$ $\sigma = 3$ $\sigma = 5$ $\sigma = 7$ $\sigma = 9$

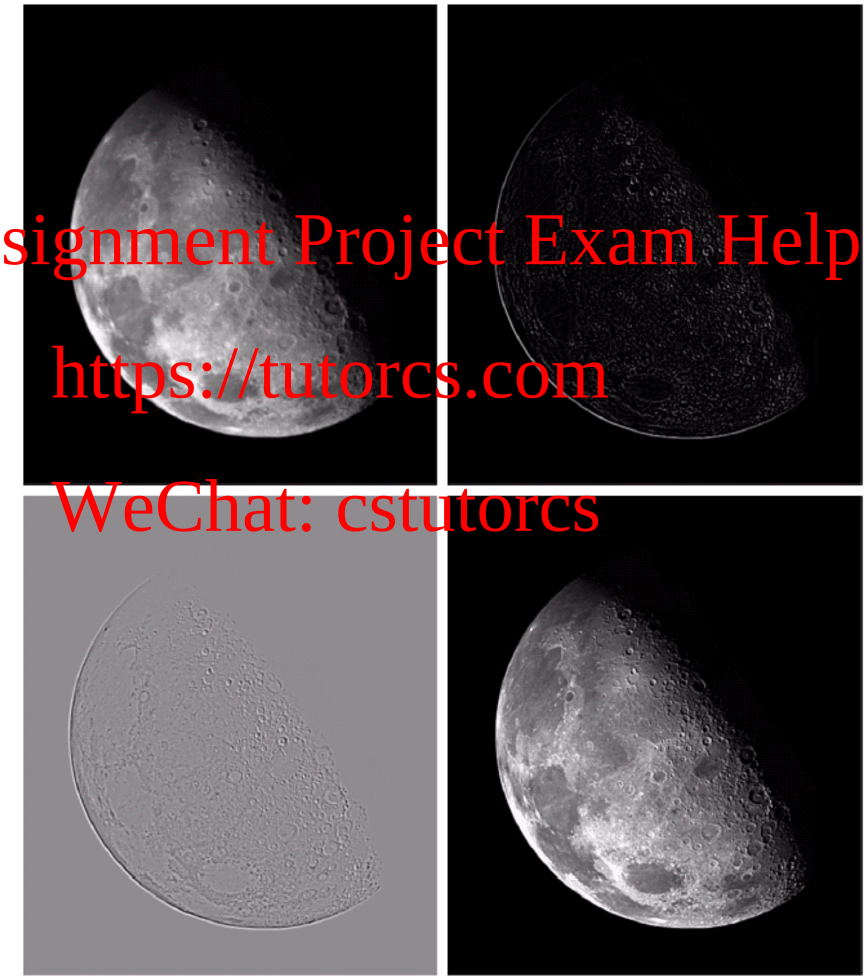
The Laplacian

- There are other forms of the Laplacian, which can include diagonal directions, for example

- Laplacian highlights grey-level discontinuities and produces dark featureless backgrounds

- The background can be recovered by adding or subtracting the Laplacian image to the original image

Chapter 3
Image Enhancement in the Spatial Domain



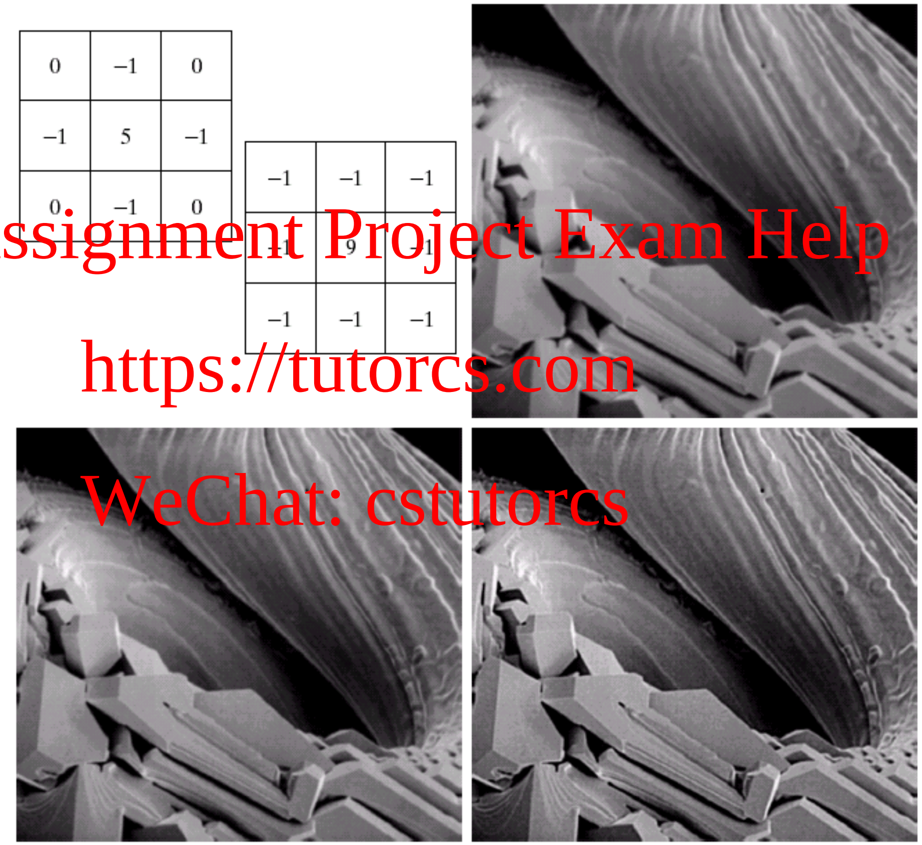
a b
c d

FIGURE 3.40 (a) Image of the North Pole of the moon. (b) Laplacian-filtered image. (c) Laplacian image scaled for display purposes. (d) Image enhanced by using Eq. (3.7-5). (Original image courtesy of NASA.)

Chapter 3
Image Enhancement in the Spatial Domain

0	−1	0
−1	5	−1
0	−1	0

−1	−1	−1
−1	9	−1
−1	−1	−1



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
Image Enhancement in the Spatial Domain

a b
c d

FIGURE 3.46 (a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel of (a).

Chapter 3
Image Enhancement in the Spatial Domain

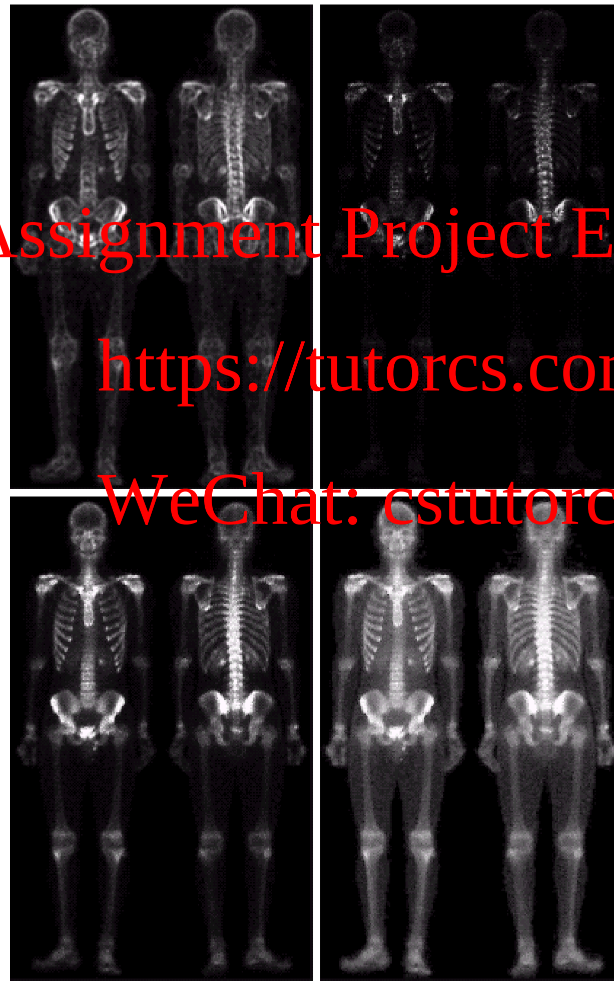


FIGURE 3.46 *(Continued)* (e) Sobel image smoothed with a 5 × 5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

Padding

- When we use spatial filters for pixels on the boundary of an image, we do not have enough neighbours

- To get an image with the same size as input image

  - ***Zero***: set all pixels outside the source image to 0
  - ***Constant***: set all pixels outside the source image to a specified border value

  - ***Clamp***: repeat edge pixels indefinitely
  - ***Wrap***: copy pixels from opposite side of the image
  - ***Mirror***: reflect pixels across the image edge

Padding Example

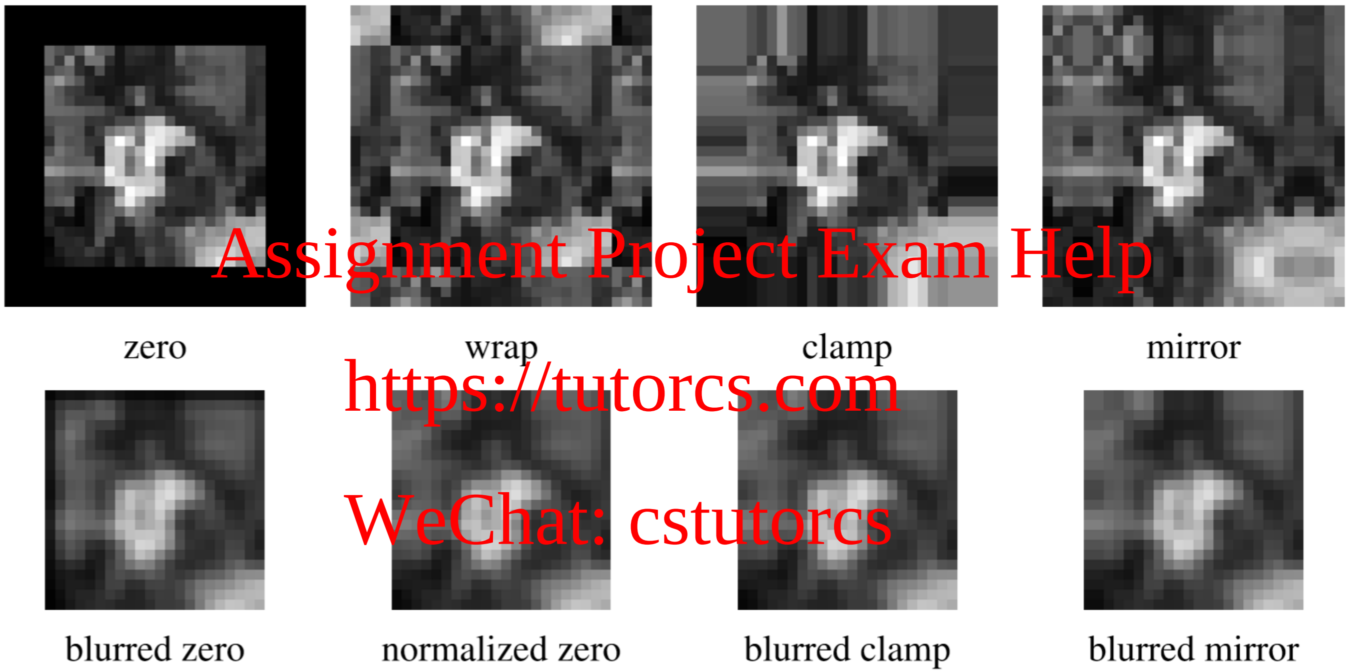


Figure 3.13 Border padding (top row) and the results of blurring the padded image (bottom row). The normalized zero image is the result of dividing (normalizing) the blurred zero-padded RGBA image by its corresponding soft alpha value.

Szeliski, "Computer Vision", Chapter 3

References and Acknowledgement

- *Gonzalez* and Woods, 2002, *Chapter 3.5-3.8*
- *Gonzalez* and Woods, 2002, *Chapter 4.1-4.4, 7.1*
- *Szeliski Chapter 3.1-3.5*
- Some material, including images and tables, drawn from the textbook, *Digital Image Processing* by Gonzalez and Woods, and P.C. Rossin’s presentation