

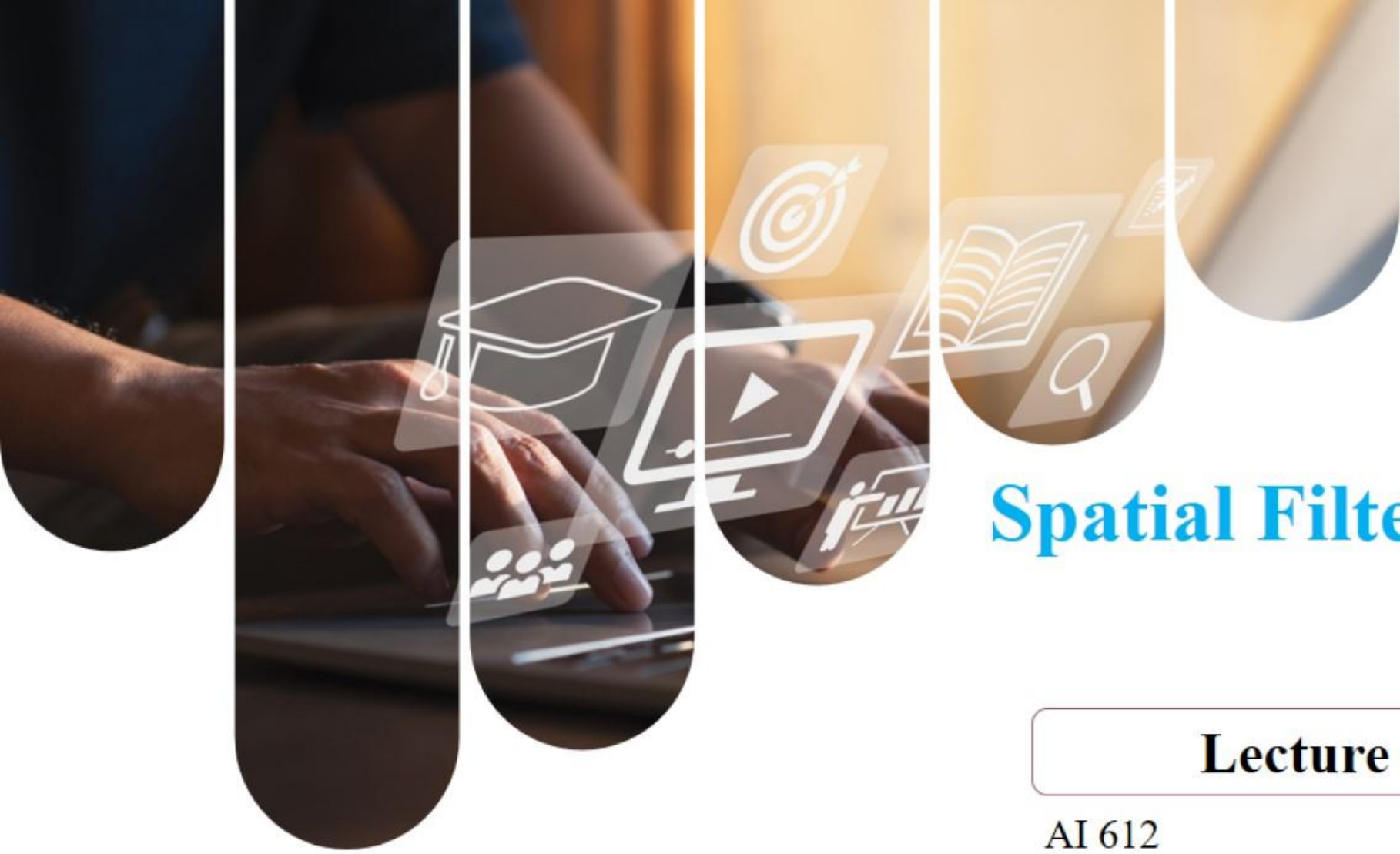
Computer Vision, and Digital Image Processing

Prepared by Dr. Mohamed Mostafa

AI 612

Lecture 4



The background features a series of vertical panels with rounded ends. The first panel shows hands typing on a laptop. The second panel shows a graduation cap icon. The third panel shows a play button icon. The fourth panel shows a target icon. The fifth panel shows an open book icon. The sixth panel shows a magnifying glass icon. The seventh panel shows a group of people icon. The eighth panel shows a bar chart icon. The text "Spatial Filtering" is written in a large, blue, serif font across the middle of the panels.

Spatial Filtering

Lecture 4

AI 612

Prepared by
Dr. Mohamed Mostafa

Chapter
03

Spatial Filtering

Resources for this
lecture

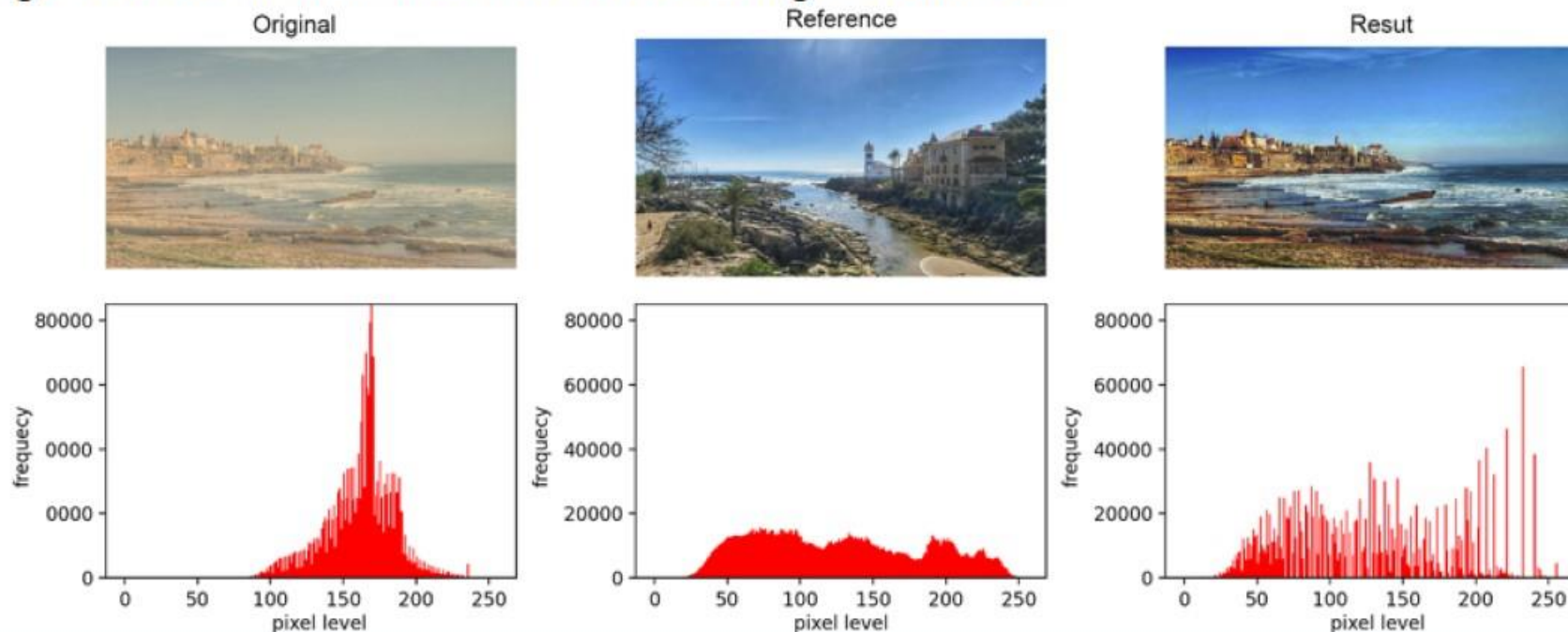
Chapter 3:

Gonzalez, Rafael C. *Digital image processing*.
Pearson education, India, 2009.



Histogram Matching (Specification)

- Histogram matching, also known as histogram specification or histogram equalization matching, is a technique used to transform an image's histogram to match a specified or target histogram.
- This process can be useful for various purposes, such as adjusting the contrast of an image to make it similar to a reference image's contrast.



Source:
[medium](#)

Histogram Matching (Specification)

The steps for histogram matching are as follows:

Compute the Histograms:

Source Image: Calculate the histogram of the source image for which you want to adjust the contrast.

Target Histogram: Obtain the histogram of the reference or target image to which you want to match the source image's histogram.

Compute the Cumulative Distribution Function (CDF):

Calculate the cumulative distribution function (CDF) for both the source image's histogram and the target histogram. The CDF represents the cumulative sum of the probabilities of each intensity value.

Normalize the CDFs:

Normalize both CDFs to ensure that they span the full intensity range (usually 0 to 1).

Histogram Matching (Specification)

The steps for histogram matching are as follows:

Create a Mapping Function:

Create a mapping function that relates the source image's normalized CDF to the target image's normalized CDF.

This mapping function helps transform the intensity values of the source image to match the target histogram.

`mapping_function(source_intensity) = target_intensity`

The mapping function assigns a new intensity value for each possible source intensity value.

Apply the Mapping Function:

For each pixel in the source image, apply the mapping function to transform its intensity value. Replace the original intensity with the corresponding target intensity according to the mapping function.

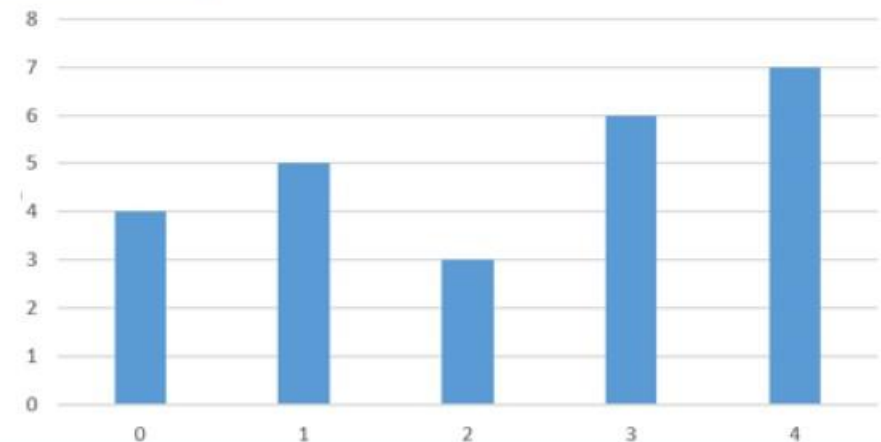
Create the Matched Image:

The result is a new image with the histogram matched to the target histogram. This image should have a similar contrast and distribution of intensity values as the target image.

Histogram Matching Example

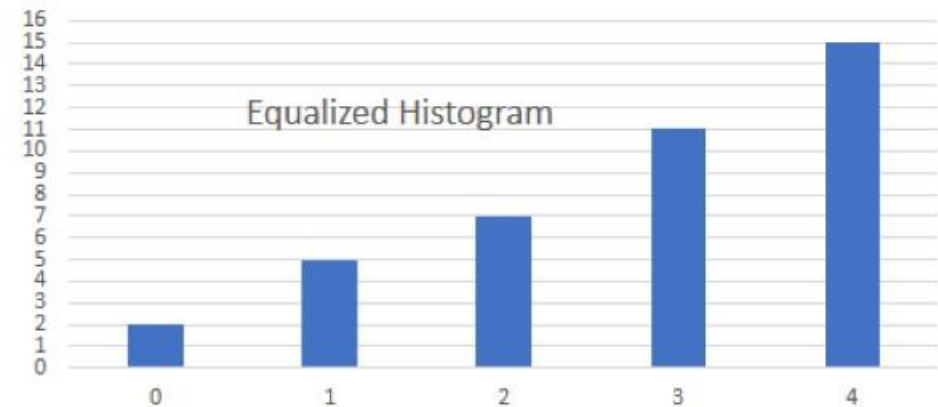
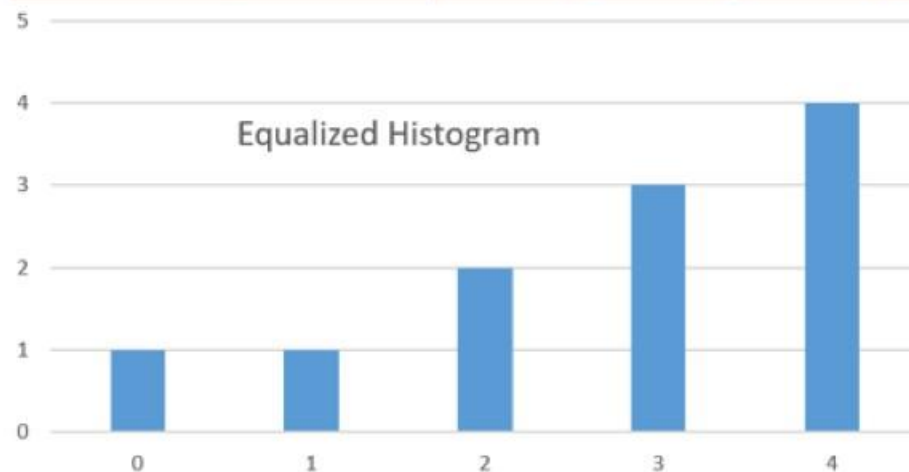
0	2	1	3	4
1	3	4	3	3
0	1	3	1	4
3	1	4	2	0
0	4	2	4	4

Intensity	# of Occurrence
0	4
1	5
2	3
3	6
4	7



Histogram Matching Example

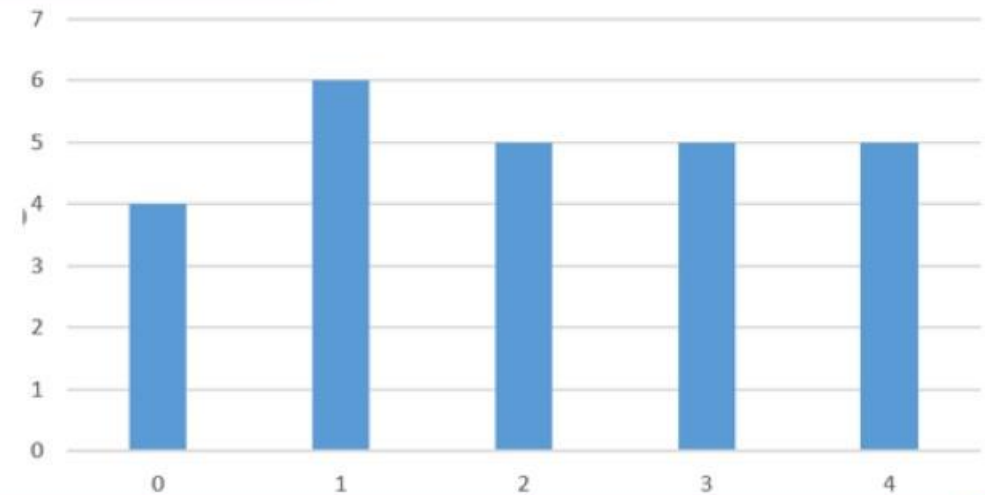
Intensity	# of Occurance	PDF	CDF	CDF*15	Equalized Histogram
0	4	0.16	0.16	2.4	2
1	5	0.2	0.36	5.4	5
2	3	0.12	0.48	7.2	7
3	6	0.24	0.72	10.8	11
4	7	0.28	1	15	15



Histogram Matching Example

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

Intensity	# of Occurrence
0	4
1	6
2	5
3	5
4	5



Histogram Matching Example

Intensity	# of Occurance	PDF	CDF	CDF*15	Equalized Histogram
0	4	0.16	0.16	2.4	2
1	6	0.24	0.4	6	6
2	5	0.2	0.6	9	9
3	5	0.2	0.8	12	12
4	5	0.2	1	15	15

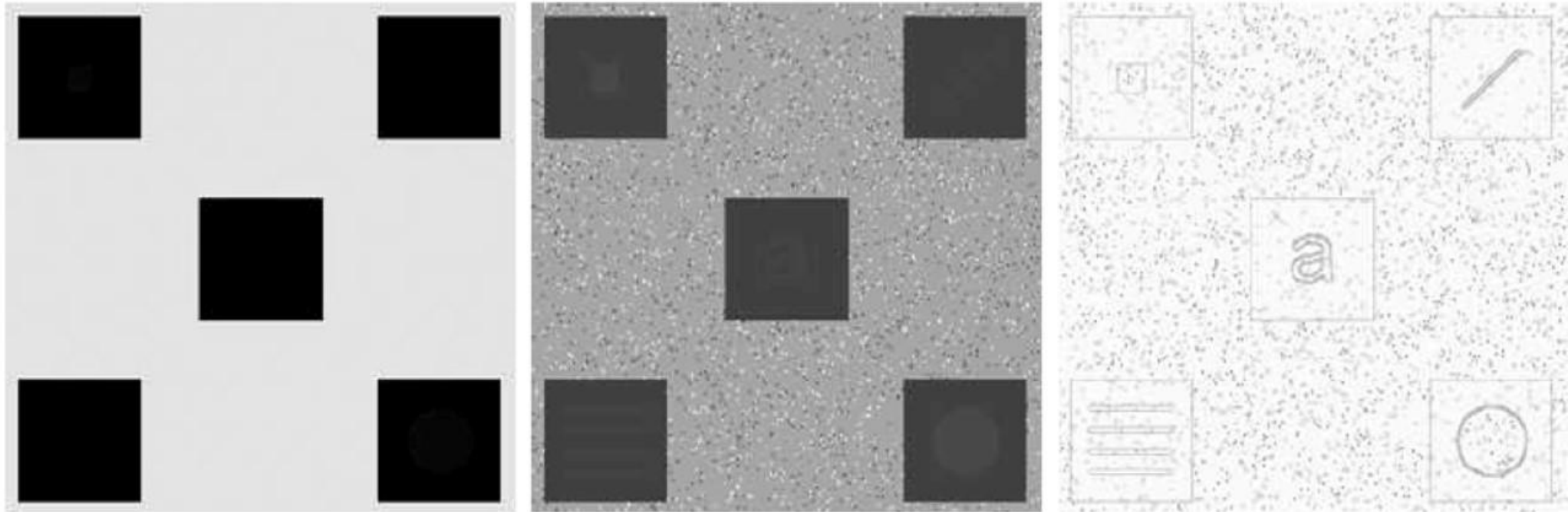
Histogram Matching Example

Intensity	Reference Image Equalized Histogram	Original Image Equalized Histogram	New Intensity
0	2	2	0
1	6	5	0
2	9	7	1
3	12	11	2
4	15	15	4

Local Histogram Processing

- There are cases in which it is necessary to enhance details over small areas in the image.
- The numbers of areas may have negligible influence on the computation of global transformation.
- The solution is to advise transformation function based on the intensity distribution in a neighborhood of every pixel in the image.
- Define transformation functions based on the intensity distribution in a neighborhood of every pixel in the image!
- The procedure is to define a neighborhood and move its center from pixel to pixel.

Local Histogram Processing



a b c

FIGURE 3.26 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size 3×3 .

USING HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT

- Statistics obtained directly from an image histogram can be used for image enhancement.
- Let r denote a discrete random variable representing intensity values in the range $[0, L - 1]$, and let $p(r_i)$ denote the normalized histogram component corresponding to intensity value r_i .

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

where m is given by

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

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

Global mean, and
standard deviation

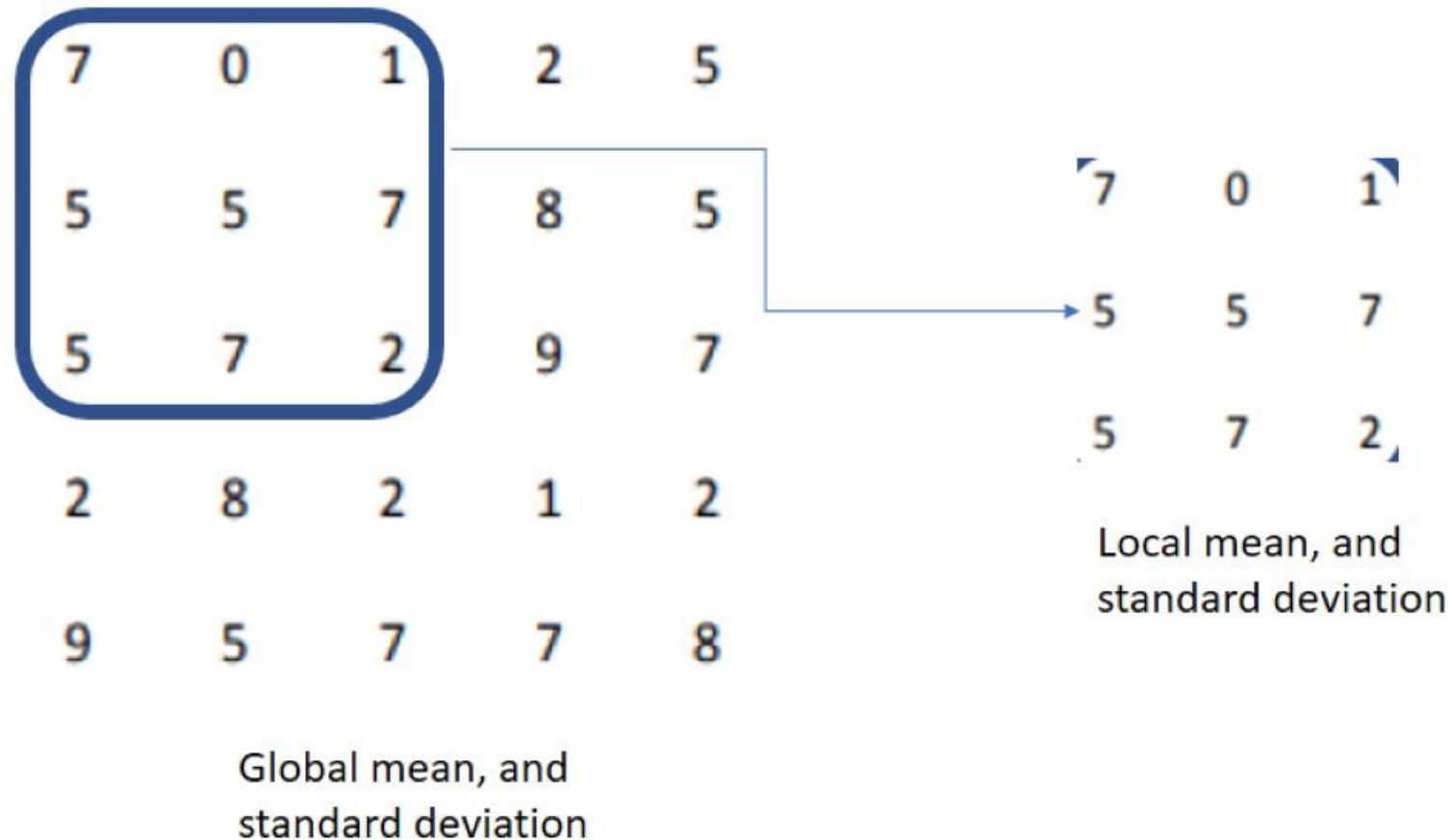
$$m_{S_{xy}} = \sum_{i=0}^{L-1} r_i p_{S_{xy}}(r_i)$$

Local mean, and
standard deviation

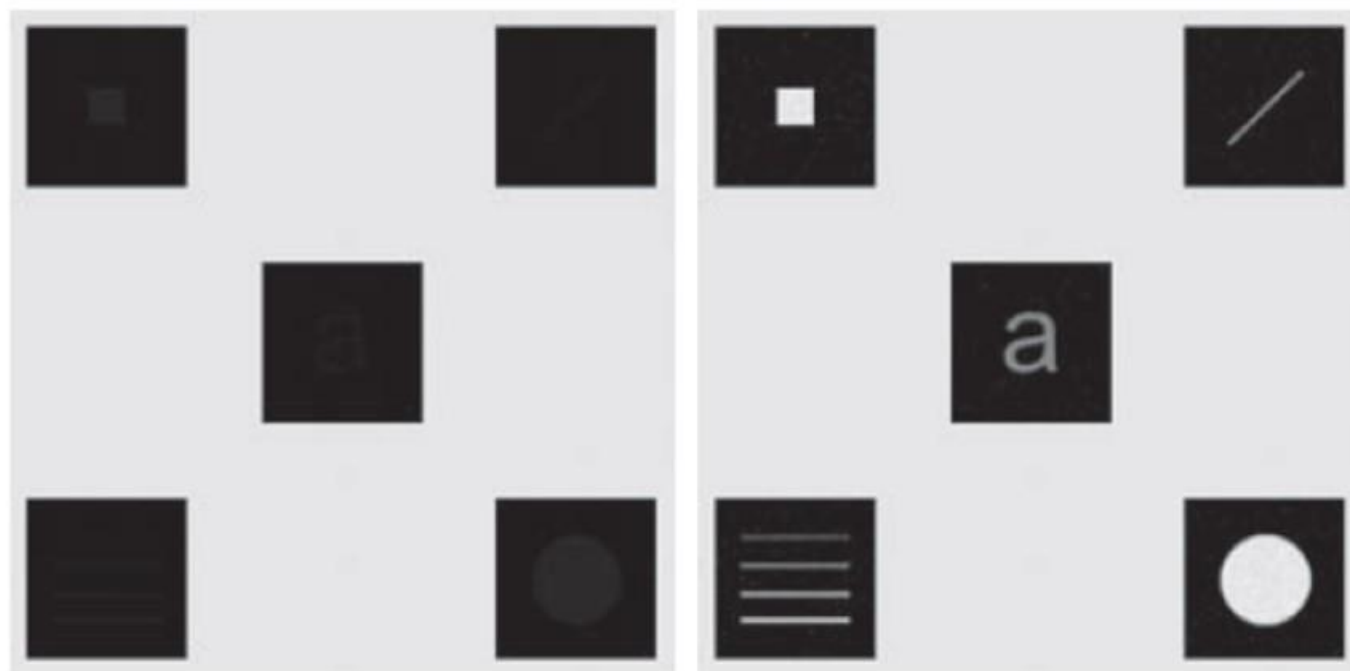
$$\sigma_{S_{xy}}^2 = \sum_{i=0}^{L-1} (r_i - m_{S_{xy}})^2 p_{S_{xy}}(r_i)$$

$$g(x, y) = \begin{cases} Cf(x, y) & \text{if } k_0 m_G \leq m_{S_{xy}} \leq k_1 m_G \text{ AND } k_2 \sigma_G \leq \sigma_{S_{xy}} \leq k_3 \sigma_G \\ f(x, y) & \text{otherwise} \end{cases}$$

USING HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT



USING HISTOGRAM STATISTICS FOR IMAGE ENHANCEMENT



FUNDAMENTALS OF SPATIAL FILTERING

- The name *filter* is borrowed from frequency domain processing, where “filtering” refers to passing, modifying, or rejecting specified frequency components of an image.
- Spatial filtering modifies an image by replacing the value of each pixel by a function of the values of the pixel and its neighbors.
- Spatial Filtering:
 - Linear spatial filtering.
 - Nonlinear spatial filtering.

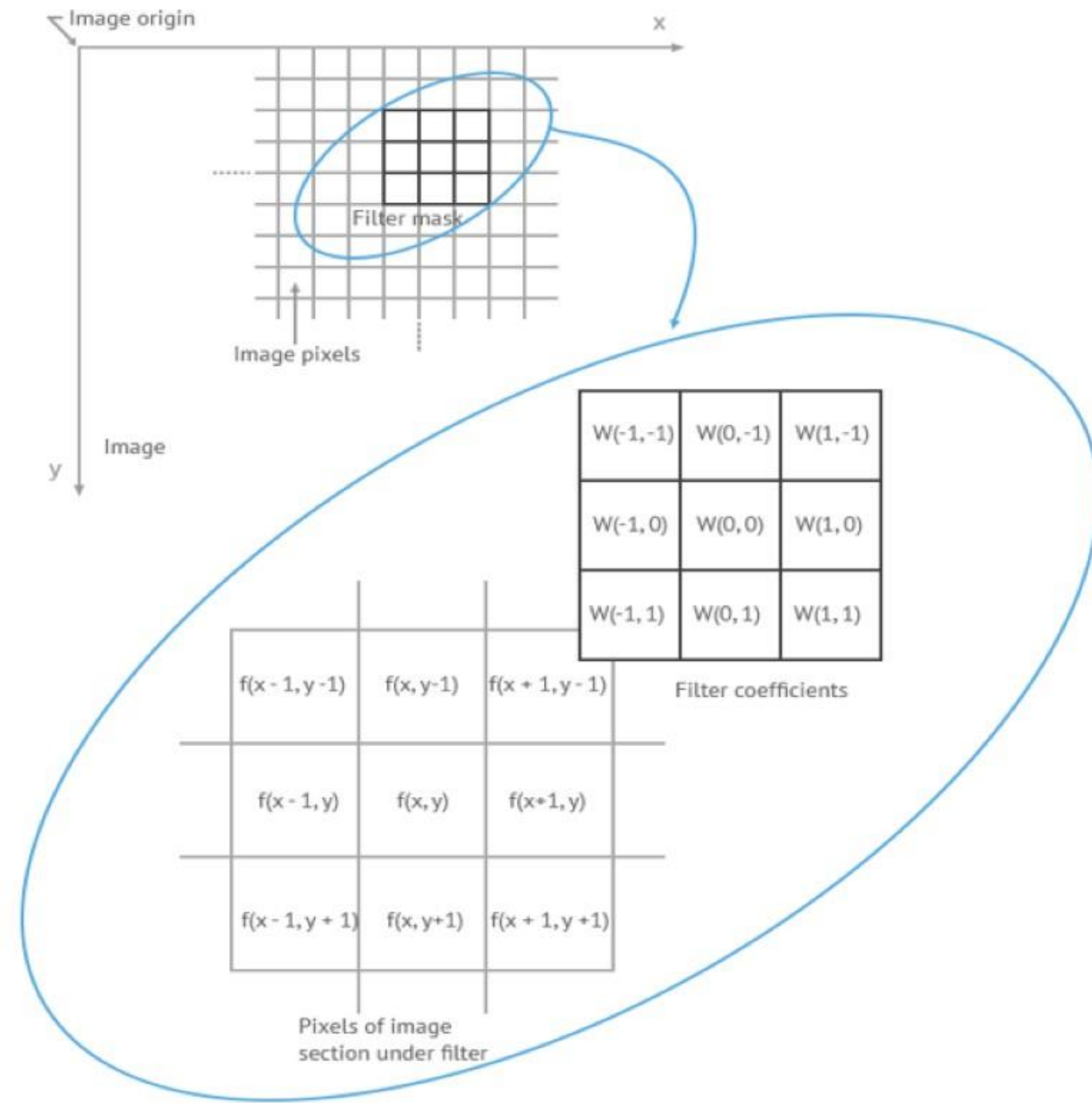
The Mechanics of Spatial Filtering

- The spatial filter consists of:
 - Neighborhood, (typically a small rectangle).
 - Predefined operation that is performed on the image pixels by the neighborhood.
 - Use mask more than 1×1 .
 - It should be odd size mask (3×3 , 5×5 , 1×5 ...)

The Mechanics of Spatial Filtering

	10	2	11	13	14
	11	13	14	13	13
	10	11	13	11	14
	13	11	14	12	10
	10	14	12	14	14

The Mechanics of Spatial Filtering



$$R = w(-1, -1) * f(x-1, y-1) + w(-1, 0) * f(x-1, y) + \dots + w(0, 0) * f(x, y) + \dots + w(1, 0) * f(x+1, y) + w(1, 1) * f(x+1, y+1)$$

Linear Spatial Filters

- The linear spatial filter for image of $M \times N$, with a filter mask of size $m \times n$ is given by the expression:

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

- Image size: $M \times N$
- Mask size: $m \times n$

- Where x and y are varied so that each pixel in w visits every pixels in f , $a = (m-1)/2$ and $b = (n-1)/2$.
- The spatial filter should be odd size and the smallest being of size 3×3 .

Linear Spatial Filters

The mask falls outside the edge

- **Solutions?**

- Ignore the edges

- The resulting image is smaller than the original

- Pad with zeros

- Introducing unwanted artifacts

	10	2	11	13	14
	11	13	14	13	13
	10	11	13	11	14
	13	11	14	12	10
	10	14	12	14	14

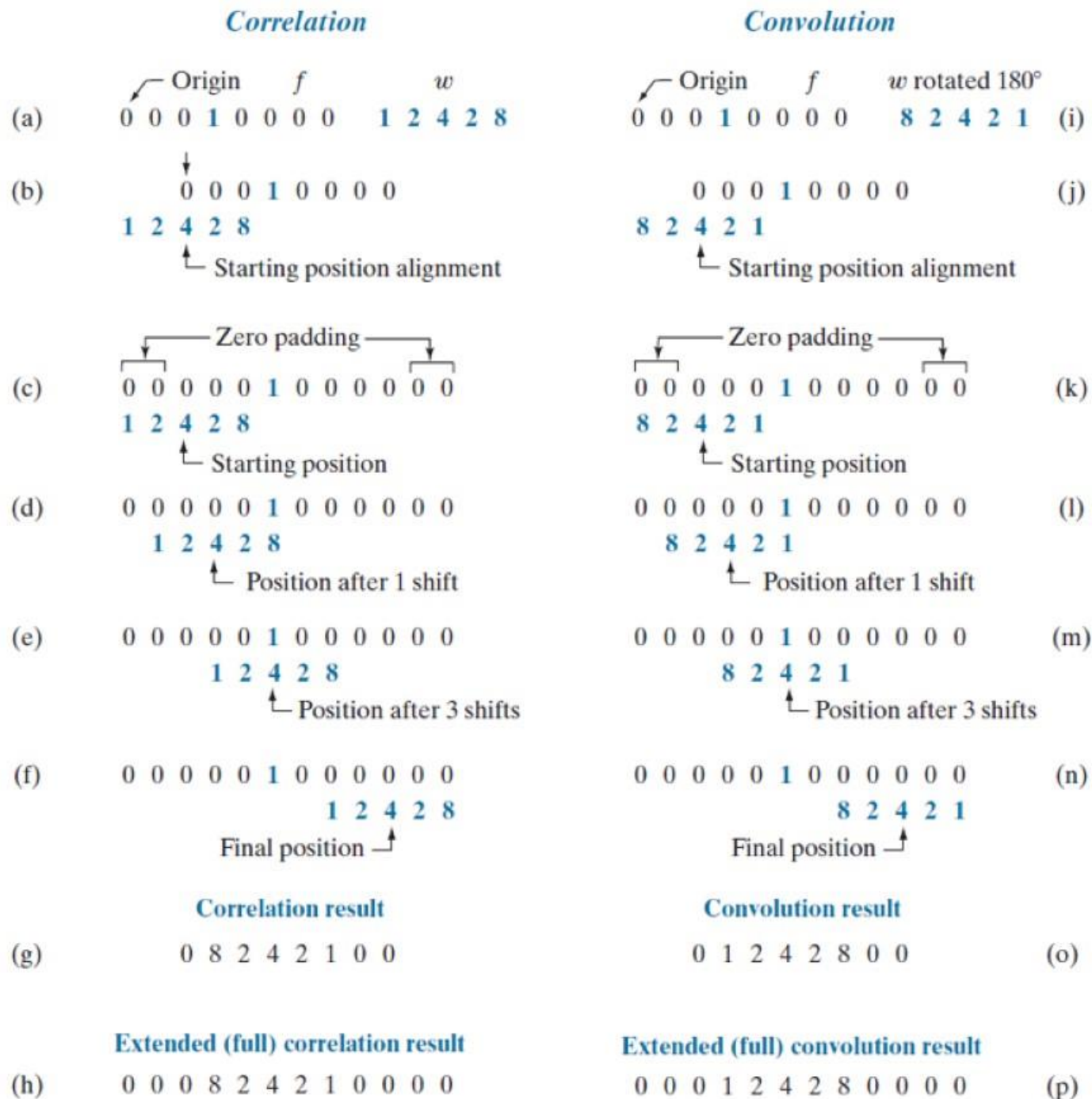
Spatial Correlation and Convolution

- **Correlation** is the process of moving a filter mask over the image and computing the sum of products at each location.
- The mechanics of **Convolution** are the same except the filter first rotate by 180°
- Where the minus signs on the right flip f (i.e. rotate it by 180)

Spatial Correlation and Convolution

Difference Between Correlation & Convolution Filtering		
	Correlation Filtering	Convolution Filtering
Rotating Kernel	The kernel is not rotated	The kernel is rotated
Formula	$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v] F[i + u, j + v]$	$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v] F[i - u, j - v]$
Notation	$G = H \otimes F$	$G = H \star F$

Spatial Correlation and Convolution



Spatial Correlation and Convolution

Diagram illustrating the padding process for a 1D convolution:

(a) Original input f (size 6x1) and kernel w (size 3x1). The padded input is shown with zeros added to the left and right of the original input.

(b) The padded input from (a) is used with a kernel of size 3x1 to produce an output of size 7x1. The output is calculated by sliding the kernel over the padded input and computing the dot product at each position.

Figure 1 illustrates the proposed correlation method. (c) Initial position for w : A 7x7 grid with a 3x3 dashed box at the top-left. (d) Correlation result: A 7x7 grid showing the result of the correlation operation. (e) Full correlation result: A 7x7 grid showing the result of the full correlation operation.

Spatial Filters

- Smoothing Filters – low pass
- Sharpening Filters – high pass

$f(x,y)$



Original Image

$g(x,y)$



Image smoothing
Gaussian Low Pass Filter

$g(x,y)$

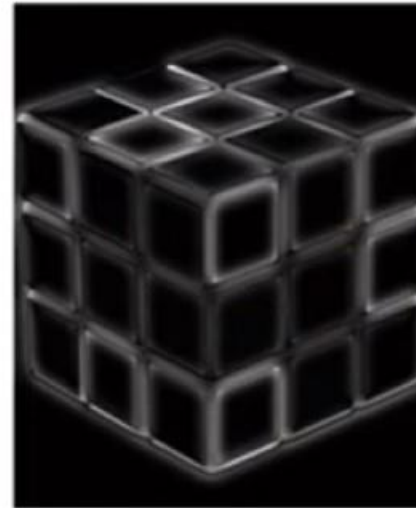


Image sharpening
Gaussian High Pass Filter

Source: [Gaussian Filter in Python](#)

Smoothing Spatial Filters

- Smoothing filters are used for blurring and for noise reduction.
- Blurring is used in preprocessing tasks such as removal of small details from an image.
- Smoothing (i.e., low-pass filters)
 - Reduce noise and small details.
 - The elements of the mask must be positive.
 - Sum of mask elements is 1
- Noise reduction can be accomplished by:
 - Smoothing linear filter.
 - Smoothing nonlinear filter.

Smoothing Spatial Filters

- A **Low - Pass** filter allows low spatial frequencies to pass unchanged , but suppress high frequencies. The low pass filter smoothes or blurs the image.
 - This tends to reduce noise, but also obscures fine detail.
 - **How it works:** the value of each pixel is replaced by the average of the gray levels in the neighborhood.
1. A spatial averaging filter in which all coefficients are equal sometimes is called a **box filter (standard average)**.
 2. A spatial averaging filter in which the coefficients are different sometimes is called a **Weighted Average**.
 3. **Median filter: order statistics**, the center pixel of a $M \times M$ neighborhood is replaced by the median value of the corresponding window. .

Smoothing Spatial Filters

$1/9 \times$

1	1	1
1	1	1
1	1	1

box filter (standard average).

$1/18 \times$

1	2	1
2	4	3
3	1	1

Weighted Average.

Smoothing Spatial Filters

$1/9 \times$

1	1	1
1	1	1
1	1	1

box filter (standard average).

10	2	11	13	14
11	13	14	13	13
10	11	13	11	14
13	11	14	12	10
10	14	12	14	14



Box filter for $r=13$

$$= (1 \cdot 13 + 1 \cdot 14 + 1 \cdot 13 + 1 \cdot 11 + 1 \cdot 13 + 1 \cdot 11 + 1 \cdot 11 + 1 \cdot 14 + 1 \cdot 12) / 9 = \mathbf{12.44}$$

Smoothing Spatial Filters

10	2	11	13	14
11	13	14	13	13
10	11	13	11	14
13	11	14	12	10
10	14	12	14	14

$1/9 \times$

1	1	1
1	1	1
1	1	1

box filter (standard average).

$1/18 \times$

1	2	1
2	4	3
3	1	1

Weighted Average.

Box filter for $r=13$

$$= (1 \cdot 13 + 1 \cdot 14 + 1 \cdot 13 + 1 \cdot 11 + 1 \cdot 13 + 1 \cdot 11 + 1 \cdot 11 + 1 \cdot 14 + 1 \cdot 12) / 9 = 12.44$$

Weighted filter for $r=13$

$$= (1 \cdot 13 + 2 \cdot 14 + 1 \cdot 13 + 2 \cdot 11 + 4 \cdot 13 + 3 \cdot 11 + 3 \cdot 11 + 1 \cdot 14 + 1 \cdot 12) / 18 = 12.22$$

Smoothing Spatial Filters

6	6	100	100
5	5	100	100
5	5	100	100
6	6	100	100

$1/9 \times$

1	1	1
1	1	1
1	1	1



2.4	24.7	45.7	44.4
3.5	36.9	46.2	66.7

box filter (standard average).

Smoothing Spatial Filters

6	6	100	100
5	5	100	100
5	5	100	100
6	6	100	100

$1/18 \times$

1	2	1
2	4	3
3	1	1



2.9	25.3	51.5	55.5
3.5	31.5	68.5	72.2

Weighted Average.

Smoothing Spatial Filters

Original Image



Box filter 3x3



Box filter 11x11



Box filter 21x21



Order-Statistic (Nonlinear)Filters

- Order-statistic filters are nonlinear spatial filters whose response is based on ranking the pixels contained in the image area encompassed by the filter and then replacing the value of center pixel with the value determined by the ranking result.
- The best-known filter in this category is the median, min and max filters.
- Median filters are effective in the presence of impulse(salt and pepper) noise because of its appearance as white and black dots superimposed on an image.

A 4x4 grid of pixel values illustrating salt and pepper noise. The grid contains the following values:

67	66	55	0
57	255	56	70
57	50	60	70
67	60	75	80

The value 255 is highlighted with a blue box and labeled "salt" with an arrow. The value 0 is highlighted with a blue box and labeled "pepper" with an arrow.

Median Filters

	67	66	55	0
	57	255	56	70
	57	50	60	70
	67	60	75	80

Median Filters

67	67	66		
67	67	66	55	0
57	57	255	56	70
	57	50	60	70
	67	60	75	80

67			

Pixels near 67 without ordering: { 67, 67, 66, 67, 67, 66, 57, 57, 255 }

Ordered pixels: { 57, 57, 66, 66, **67**, 67, 67, 67, 255 }

Median Filters

67	66	55	
67	66	55	0
57	255	56	70
57	50	60	70
67	60	75	80

67	66		

Pixels near 66 without ordering: { 67, 66, 55, 67, 66, 55, 57, 255, 56 }

Ordered pixels: { 55 , 55 , 56, 57, **66**, 66, 67, 67, 255 }

Median Filters

	66	55	0
67	66	55	0
57	255	56	70
57	50	60	70
67	60	75	80

67	66	56	

Pixels near 66 without ordering: { 66, 55, 0, 66, 55, 0, 255, 56, 70 }

Ordered pixels: { 0, 0, 55, 55, **56**, 66, 66, 70, 255 }

Median Filters

		55	0	0
67	66	55	0	0
57	255	56	70	70
57	50	60	70	
67	60	75	80	

67	66	56	55

Pixels near 66 without ordering: {55, 0, 0, 55, 0, 0, 56, 70, 70}

Ordered pixels: {0, 0, 0, 0, 55, 55, 56, 70, 70}

Median Filters

67	67	66	55	0
57	57	255	56	70
57	57	50	60	70
	67	60	75	80

67	66	56	55
57			

Pixels near 66 without ordering: {67, 67, 66, 57, 57, 255, 57, 57, 50 }

Ordered pixels: {50, 57, 57, 57, **57**, 66, 67, 67, 255 }

Median Filters

67	66	55	0
57	255	56	70
57	50	60	70
67	60	75	80

67	66	56	55
57	57		

Pixels near 66 without ordering: {67, 66, 55, 57, 255, 56, 57, 50, 60 }

Ordered pixels: {50, 55, 56, 57, 57, 60 , 66, 67, 255}

Median Filters

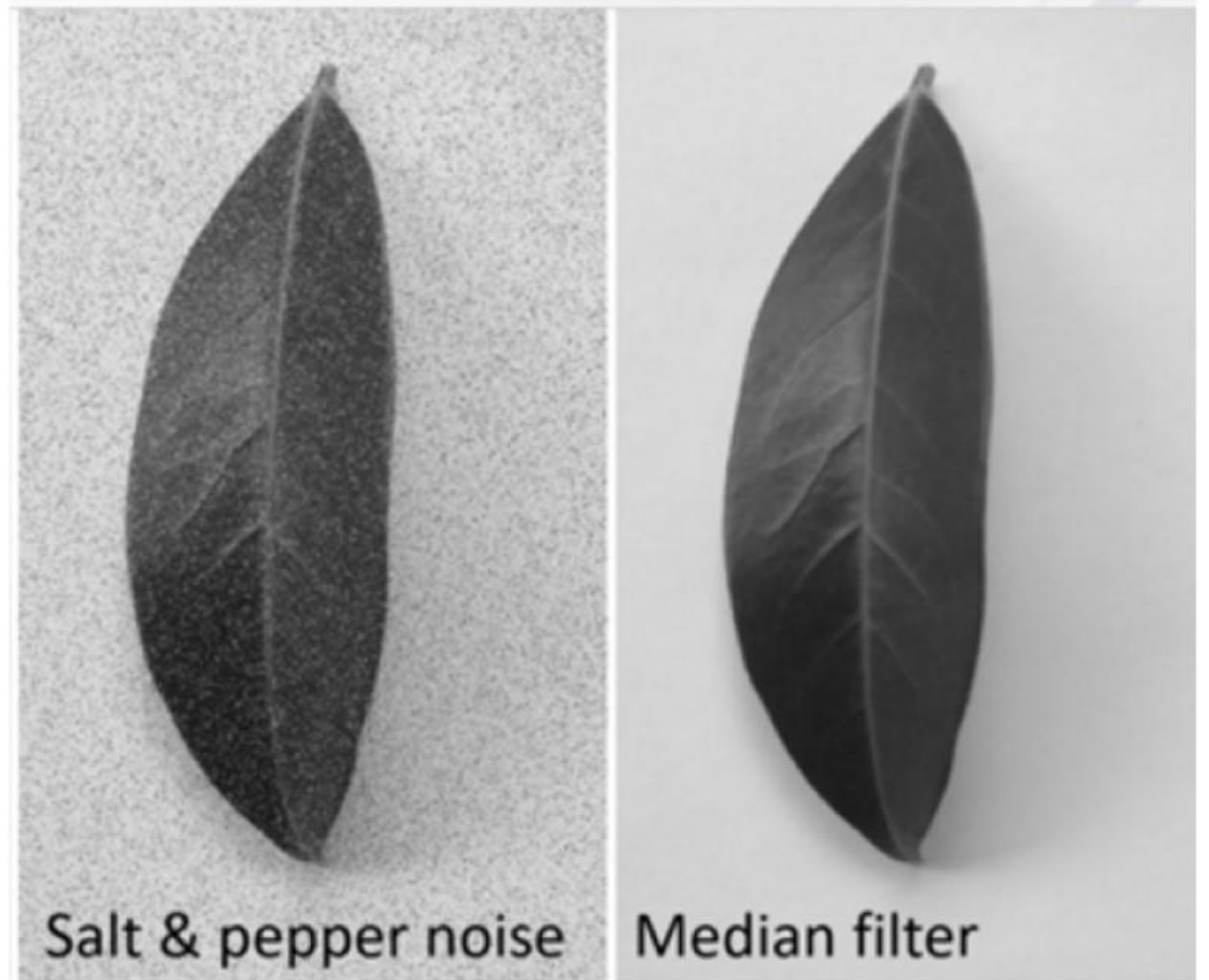
67	66	55	0
57	255	56	70
57	50	60	70
67	60	75	80

67	66	56	55
57	57		
		70	

Pixels near 66 without ordering: {255, 56, 70, 50, 60, 70, 60, 75, 80 }

Ordered pixels: {50, 56, 60, 60, **70**, 70, 75, 80, 255}

Median Filters



Order-Statistic (Nonlinear)Filters

- **Assignment:**

- Explain how the following filters are working?
 - Max filter
 - Min filter





Thank You