Digital Image Processing and Pattern Recognition

E1528

Fall 2022-2023

Lecture 4

Histogram & Spatial Filtering Fundamentals

(Correlation & Convolution & Padding)

INSTRUCTOR

DR / AYMAN SOLIMAN

Contents

Objectives

- Understand the meaning of spatial domain processing, and how it differs from transform domain processing.
- Be familiar with the principal techniques used for intensity transformations.
- Understand the physical meaning of image histograms and how they can be manipulated for image enhancement.
- Understand the mechanics of spatial filtering, and how spatial filters are formed.

Objectives (cont.)

- Understand the principles of spatial convolution and correlation.
- Be familiar with the principal types of spatial filters, and how they are applied.
- Be aware of the relationships between spatial filters, and the fundamental role of lowpass filters.
- Understand how to use combinations of enhancement methods in cases where a single approach is insufficient.

Background

- All the image processing techniques discussed in this lecture are implemented in the spatial domain, which is the plane containing the pixels of an image.

- Spatial domain techniques operate directly on the pixels of an image, as opposed, for example, to the frequency domain in which operations are performed on the Fourier transform of an image, rather than on the image itself.

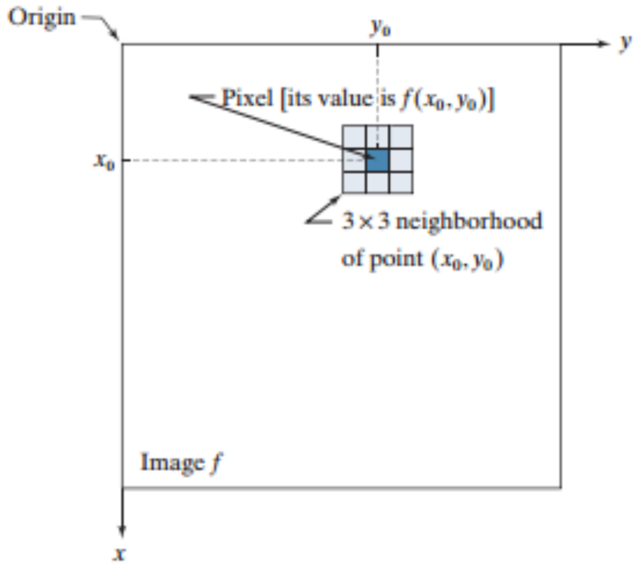
- As you will learn, some image processing tasks are easier or more meaningful to implement in the spatial domain, while others are best suited for other approaches.

➢ Basics of intensity transformations and spatial filtering

➢ The spatial domain processes we discuss are based on the expression

$$g(x,y) = T[f(x,y)]$$

Where $f(x,y)$ is the input image, g(x, y) is the processed image, and T is an operator on f, defined over some neighborhood of (x, y).



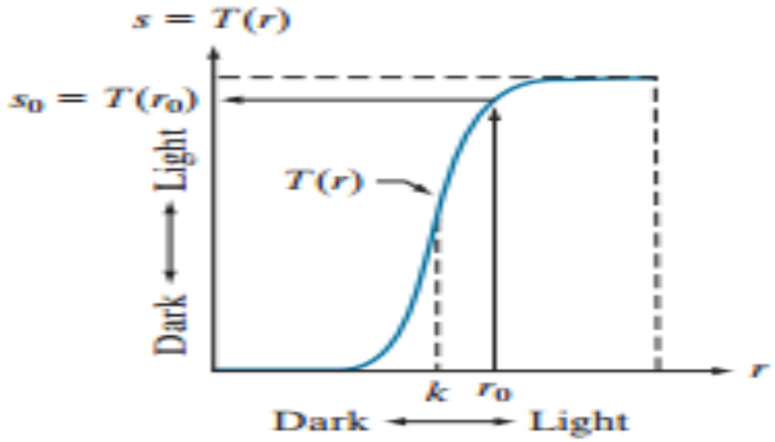
➢ Contrast Stretching

- ➢ The smallest possible neighborhood is of size 1 × 1 . In this case, g depends only on the value of f at a single point (x ,y) and T becomes an intensity (also called a gray-level, or mapping) transformation function of the form

$$s = T(r)$$

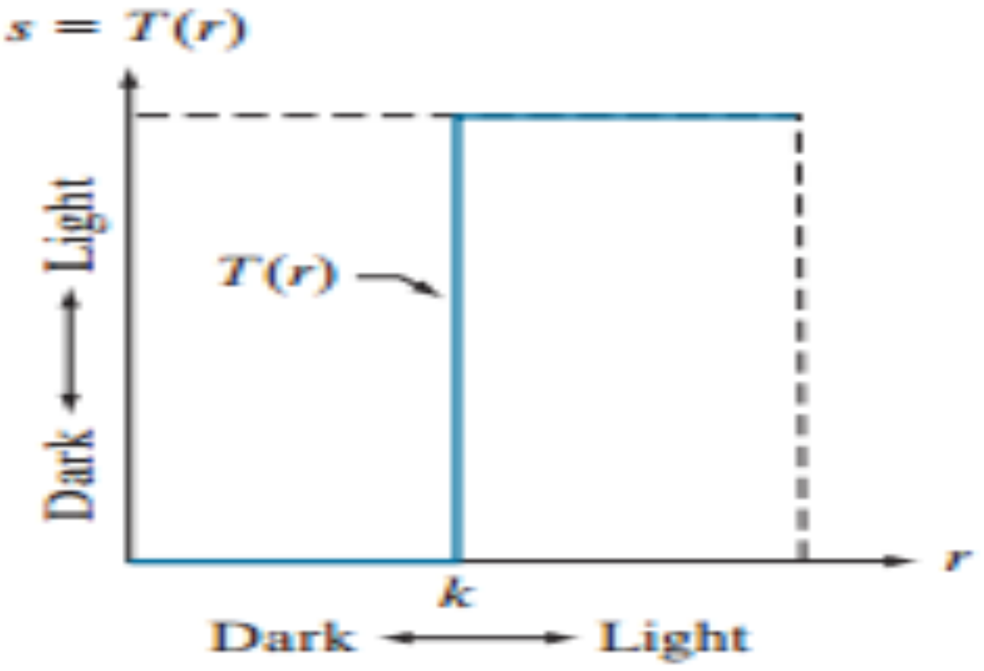
where, for simplicity in notation, we use *s* and *r* to denote, respectively, the intensity of *g* and *f* at any point (x , y).

By darkening the intensity levels below k and brightening the levels above k. In this technique, sometimes called contrast stretching



➢ Thresholding Function

- ➢ In the limiting case shown in Fig., T(r) produces a two-level (binary) image.
- ➢ A mapping of this form is called a thresholding function.
- ➢ Some simple yet powerful processing approaches can be formulated with intensity transformation functions.



Intensity-Level Slicing

- There are applications in which it is of interest to highlight a specific range of intensities in an image.

- Some of these applications include enhancing features in satellite imagery, such as masses of water, and enhancing flaws in X-ray images.

- The method, called intensity-level slicing, can be implemented in several ways, but most are variations of two basic themes.

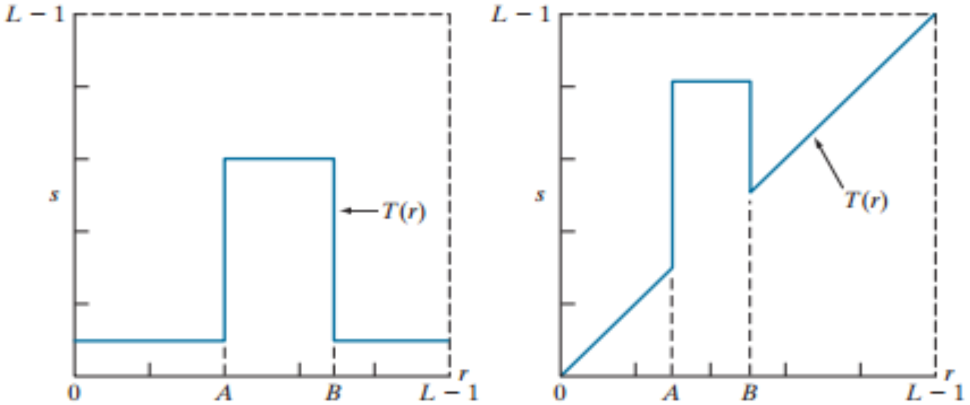
Intensity-Level Slicing (cont.)

- One approach is to display in one value (say, white) all the values in the range of interest and in another (say, black) all other intensities.
- This transformation, shown in Fig(a), produces a binary image.

a b

(a) This transformation function highlights range [A,B] and reduces all other intensities to a lower level

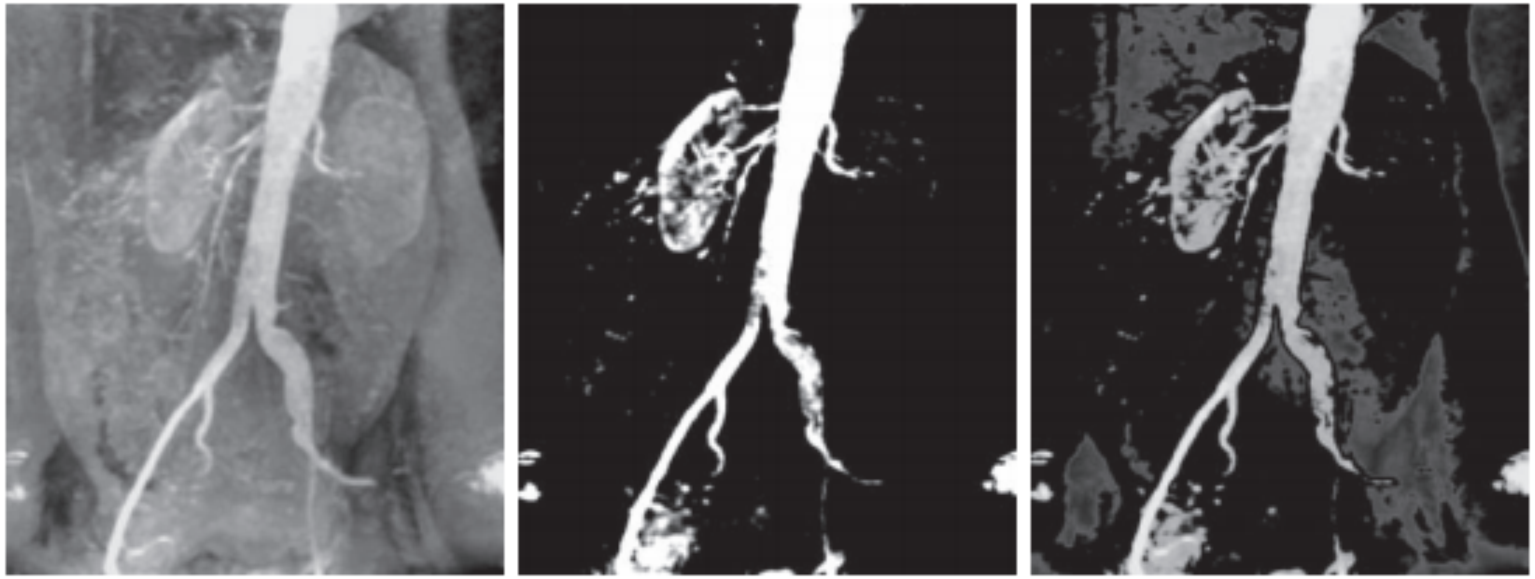
(b) This function highlights range [A,B] and leaves other intensities unchanged.



- The second approach, based on the transformation in Fig(b), brightens (or darkens) the desired range of intensities, but leaves all other intensity levels in the image unchanged.

➢ Intensity-Level Slicing Example

The objective of this example is to use intensity-level slicing to enhance the major blood vessels that appear lighter than the background, as a result of an injected contrast medium.



a b c

(a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig.(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig.(b), with the selected range set near black, so that the grays in the area of the blood vessels and kidneys were preserved.

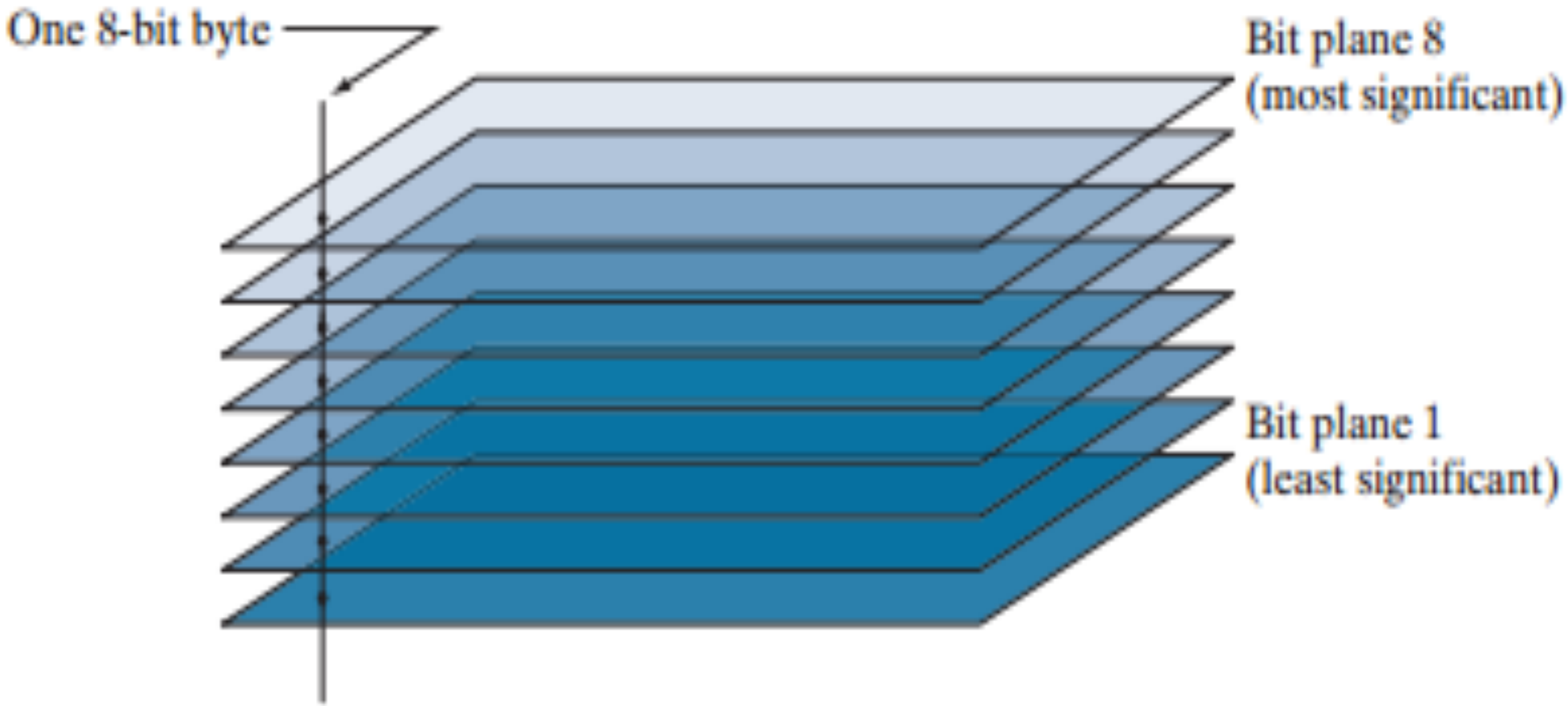
➢ Bit-Plane Slicing

- Pixel values are integers composed of bits. For example, values in a 256-level grayscale image are composed of 8 bits (one byte).

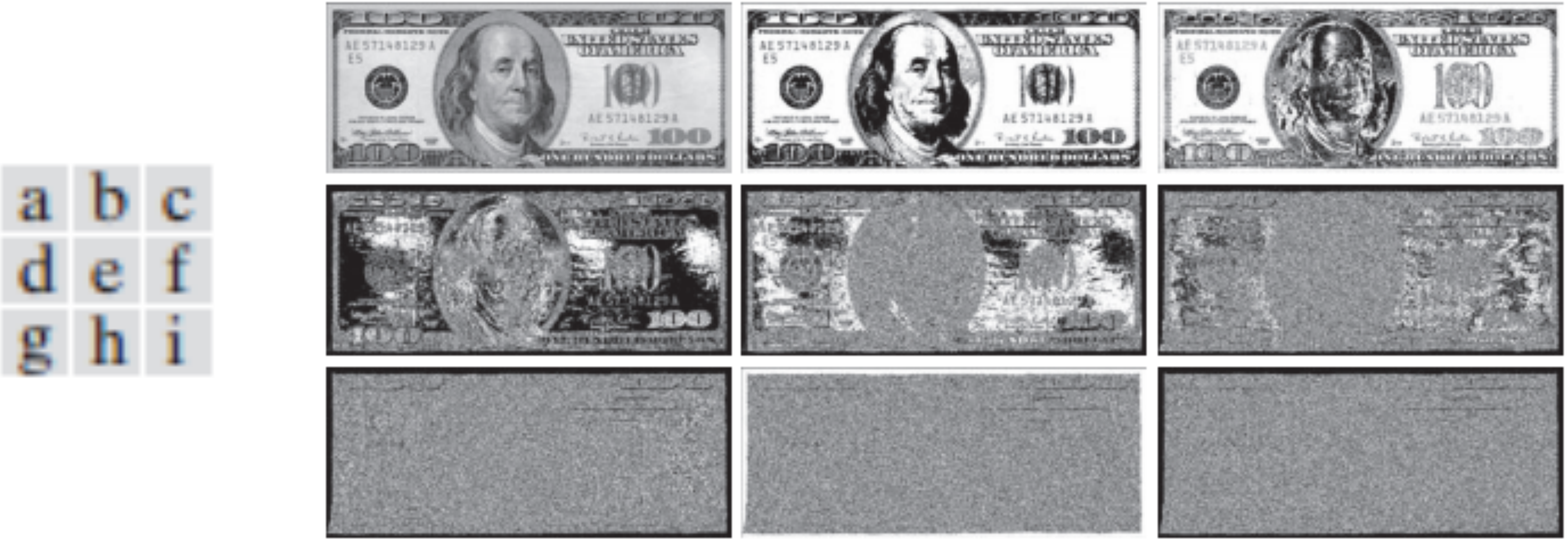
- Instead of highlighting intensity-level ranges, we could highlight the contribution made to total image appearance by specific bits.

- As next Figure illustrates, an 8-bit image may be considered as being composed of eight one-bit planes, with plane 1 containing the lowest-order bit of all pixels in the image, and plane 8 all the highest-order bits.

➢ Bit-Plane Slicing (cont.)



➢ Bit-Plane Slicing Example



(a) An 8-bit gray-scale image of size 550 × 1192 pixels. (b) through (i) Bit planes 8 through 1, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image..

Histogram Equalization

➢ Introduction to Histogram Equalization

- Image pre-processing is the term for operations on the images at the lowest level of abstraction. These operations do not increase image information content, but they decrease it if entropy is an information measure.

- The aim of pre-processing is an improvement of the image data that suppresses undesired distortions or enhances some image features relevant for further processing and analysis tasks.

Introduction to Histogram Equalization

- There are four different types of Image Pre-Processing techniques, and they are listed below.
  1) Pixel brightness transformations/ Brightness corrections
  2) Geometric Transformations
  3) Image Filtering and Segmentation
  4) Fourier transform and Image restoration
- Histogram equalization is one of the Pixel brightness transformations techniques. It is a well-known contrast enhancement technique due to its performance on almost all types of image.

Histogram Equalization

- A histogram is a representation of frequency distribution. It is the basis for numerous spatial domain processing techniques. Histogram manipulation can be used for image enhancement.

- Contrast is defined as the difference in intensity between two objects in an image.

- If the contrast is too low, it is impossible to distinguish between two objects, and they are seen as a single object.

➢ Histogram Equalization (cont.)

- ➢ Histogram equalization is a widely used contrast-enhancement technique in image processing because of its high efficiency and simplicity.
- ➢ It is one of the sophisticated methods for modifying the dynamic range and contrast of an image by altering that image such that its intensity histogram has the desired shape.
- ➢ It can be classified into two branches as per the transformation function is used.
    1) Global histogram equalization (GHE)
    2) Local histogram equalization (LHE)

Global histogram equalization (GHE)

- GHE is very simple and fast, but its contrast enhancement power is low.
- Here the histogram of the whole input image is used to compute the histogram transformation function.
- As a result, the dynamic range of the image histogram is flattened and stretched. The overall contrast is improved.

➢ Local histogram equalization (LHE)

- ➢ LHE can enhance the overall contrast more effectively.

- ➢ One of the drawbacks of histogram equalization is that it can change the mean brightness of an image significantly because of histogram flattening and sometimes this is not a desirable property when preserving the original mean brightness of a given image is necessary.

- ➢ Bi-Histogram Equalization was proposed to overcome this problem.

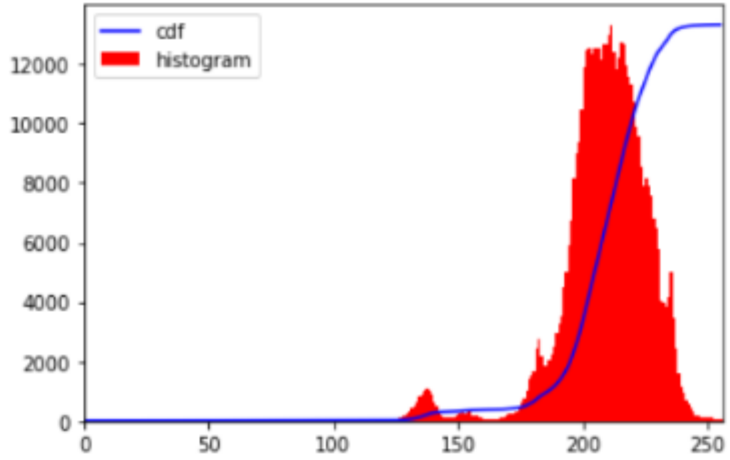
➢ Steps Involved

- ➢ Get the input image
- ➢ Generate the histogram for the image
- ➢ Find the local minima of the image
- ➢ Divide the histogram based on the local minima
- ➢ Have the specific gray levels for each partition of the histogram
- ➢ Apply the histogram equalization on each partition

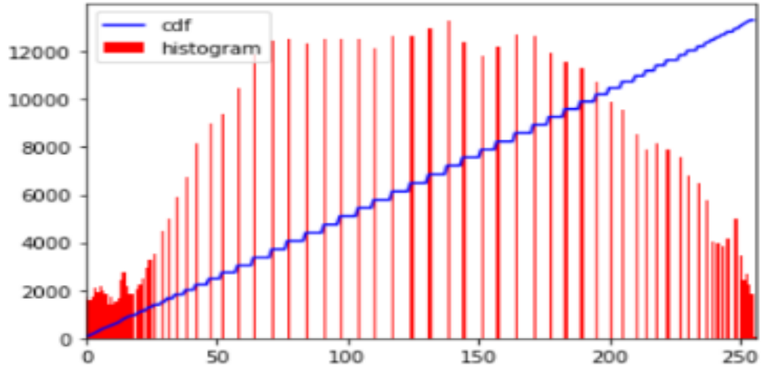
Example

- The below example shows how Histogram equalization modifies the contrast of the Image.

Input Image



output Image



➢ Histogram Equalization wit MATLAB

- ➢ You can adjust the intensity values of image pixels automatically using histogram equalization.

- ➢ Histogram equalization involves transforming the intensity values so that the histogram of the output image approximately matches a specified histogram.

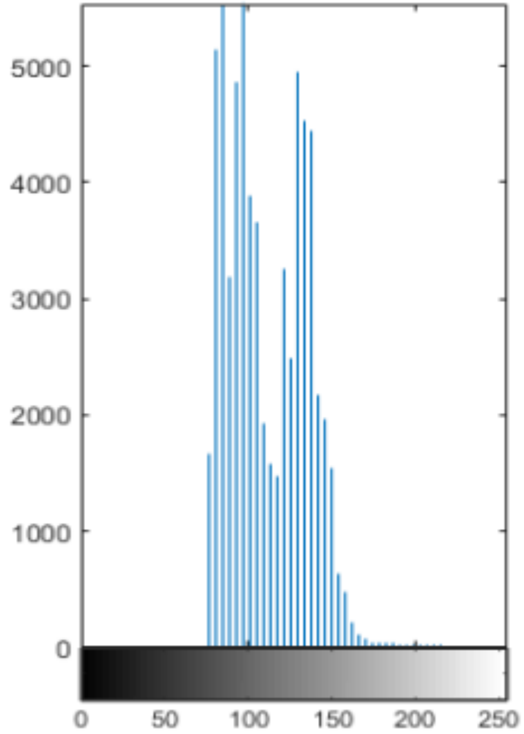
- ➢ By default, the histogram equalization function, histeq, tries to match a flat histogram with 64 bins, but you can specify a different histogram instead.

➢ Histogram Equalization wit MATLAB

- This example shows how to use histogram equalization to adjust the contrast of a grayscale image. The original image has low contrast, with most pixel values in the middle of the intensity range. histeq produces an output image with pixel values evenly distributed throughout the range.

- Read an image into the workspace.

```
I = imread('pout.tif');
figure
subplot(1,2,1)
imshow(I)
subplot(1,2,2)
imhist(I,64)
```



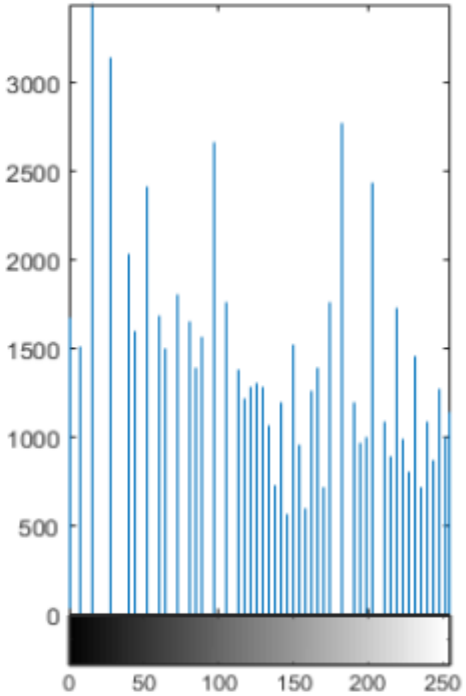
➢ Histogram Equalization wit MATLAB

- ➢ Adjust the contrast using histogram equalization. In this example, the histogram equalization function, histeq, tries to match a flat histogram with 64 bins, which is the default behavior. you can specify a different histogram instead.

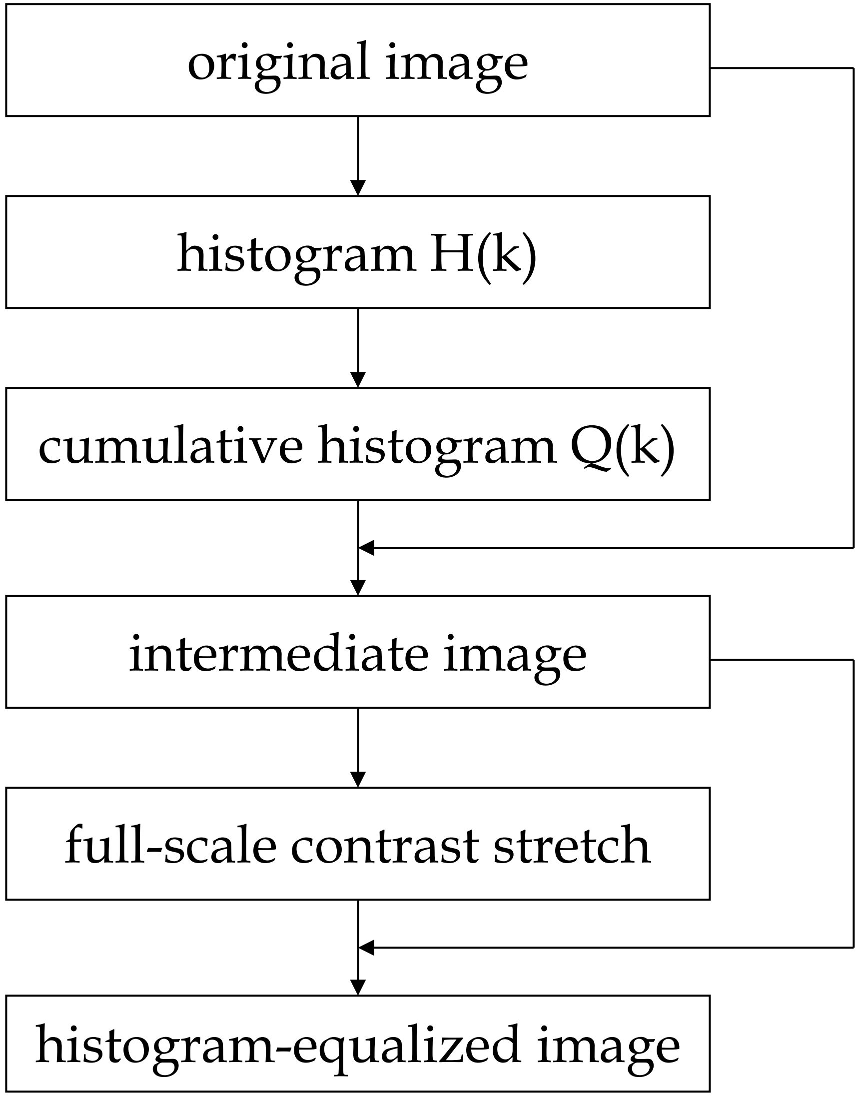
```
J = histeq(I);
```

➢ Display the contrast-adjusted image and its new histogram.

```
figure
subplot(1,2,1)
imshow(J)
subplot(1,2,2)
imhist(J,64)
```



➢ Summary of the Histogram Equalization Algorithm



Spatial Filtering

Introduction to spatial filtering

- Spatial filtering is used in a broad spectrum of image processing applications, so a solid understanding of filtering principles is important.
- The name filter is borrowed from frequency domain processing where "filtering" refers to passing, modifying, or rejecting specified frequency components of an image.
- For example, a filter that passes low frequencies is called a lowpass filter. The net effect produced by a lowpass filter is to smooth an image by blurring it. We can accomplish similar smoothing directly on the image itself by using spatial filters.

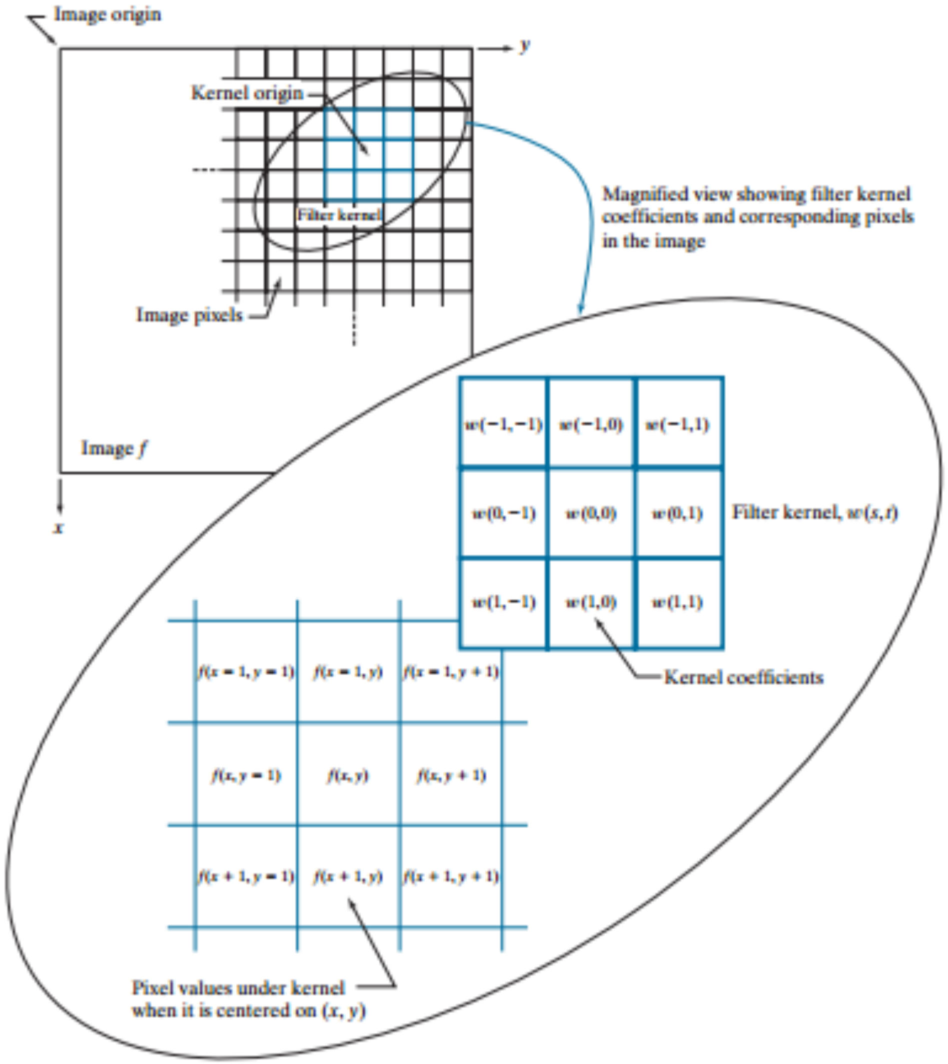
Spatial Filters

- Spatial filtering modifies an image by replacing the value of each pixel by a function of the values of the pixel and its neighbors.
- If the operation performed on the image pixels is linear, then the filter is called a linear spatial filter. Otherwise, the filter is a nonlinear spatial filter.
- We will focus attention first on linear filters and then introduce some basic nonlinear filters.

The Mechanics of Linear Spatial Filtering

- A linear spatial filter performs a sum-of-products operation between an image f and a filter kernel, w.
- The kernel is an array whose size defines the neighborhood of operation, and whose coefficients determine the nature of the filter.
- Other terms used to refer to a spatial filter kernel are mask, template, and window. We use the term filter kernel or simply kernel.

➢ The Mechanics of Linear Spatial Filtering



➢ The Mechanics of Linear Spatial Filtering

- Last figure illustrates the mechanics of linear spatial filtering using a 3×3 kernel. At any point (x,y) in the image, the response, g(x,y), of the filter is the sum of products of the kernel coefficients and the image pixels encompassed by the kernel:

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

- As coordinates x and y are varied, the center of the kernel moves from pixel to pixel, generating the filtered image, g, in the process.

➢ The Mechanics of Linear Spatial Filtering

- Observe that the center coefficient of the kernel, $w(0,0)$, aligns with the pixel at location (x,y). For a kernel of size $m \times n$, we assume that $m = 2a+1$ and $n = 2b+1$, where a and b are nonnegative integers.

- This means that our focus is on kernels of odd size in both coordinate directions.

- In general, linear spatial filtering of an image of size $M \times N$ with a kernel of size $m \times n$ is given by the expression

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t) \qquad (1)$$

Spatial Correlation and Convolution

- Spatial correlation is illustrated graphically in the figure, and it is described mathematically by equation (1).
- Correlation consists of moving the center of a kernel over an image and computing the sum of products at each location.
- The mechanics of spatial convolution are the same, except that the correlation kernel is rotated by 180°. Thus, when the values of a kernel are symmetric about its center, correlation and convolution yield the same result. The reason for rotating the kernel will become clear in the following discussion. The best way to explain the differences between the two concepts is by example.

Spatial Correlation and Convolution Dimensions

- 1-D
- 2-D

➢ Spatial Correlation and Convolution (cont.)

- We begin with a 1-D illustration, in which case equation(1) becomes

$$g(x) = \sum_{s=-a}^{a} w(s) f(x+s) \qquad (2)$$

Next figure (a) shows a 1-D function, f, and a kernel, w. The kernel is of size 1 × 5 , so a = 2 and b = 0 in this case. Figure (b) shows the starting position used to perform correlation, in which w is positioned so that its center coefficient is coincident with the origin of f.

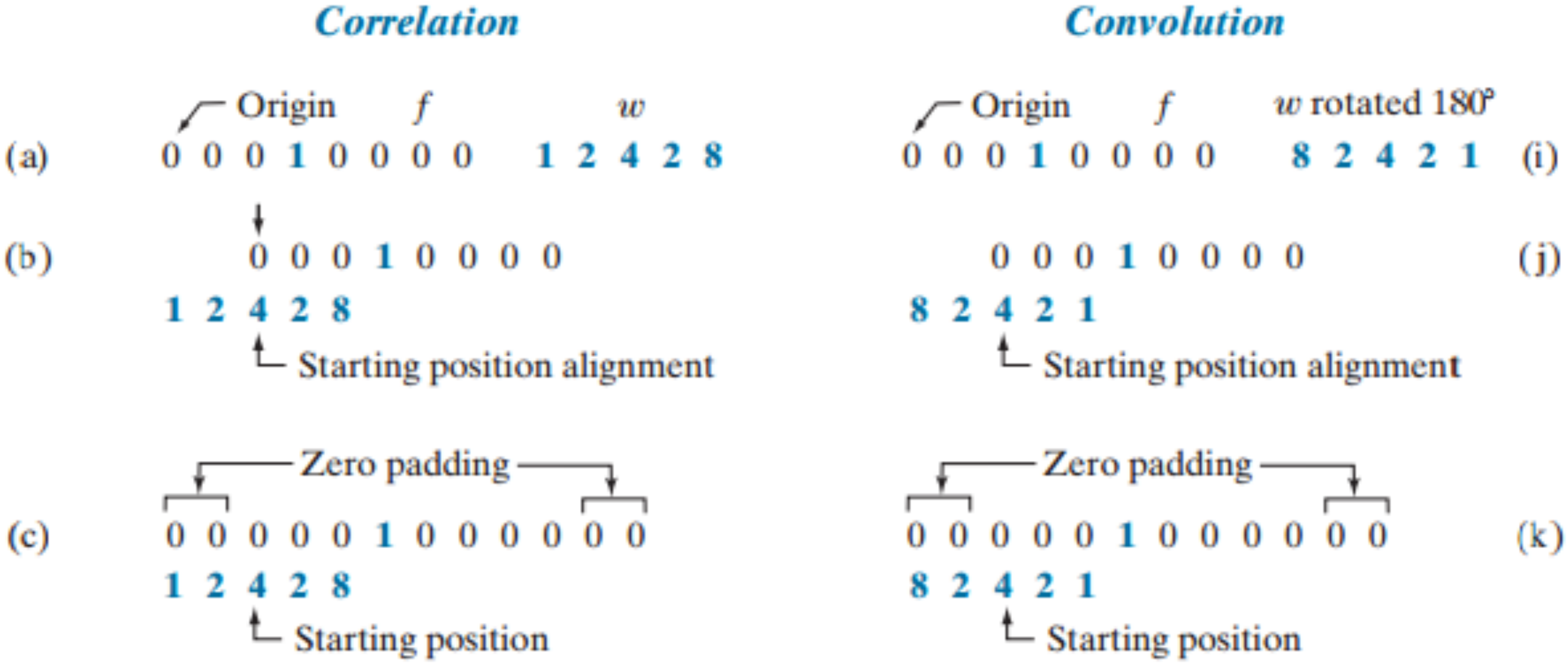
Spatial Correlation and Convolution (cont.)

- The first thing we notice is that part of w lies outside f, so the summation is undefined in that area.

- A solution to this problem is to pad function f with enough 0's on either side. In general, if the kernel is of size $1 \times m$, we need $(m-1)/2$ zeros on either side of f in order to handle the beginning and ending configurations of w with respect to f.

- Figure(c) shows a properly padded function. In this starting configuration, all coefficients of the kernel overlap valid values.

➢ Spatial Correlation and Convolution (cont.)



➢ Spatial Correlation and Convolution (cont.)

(d)
0 0 0 0 0 1 0 0 0 0 0 0
1 2 4 2 8
Position after 1 shift

(e)
0 0 0 0 0 1 0 0 0 0 0 0
1 2 4 2 8
Position after 3 shifts

(f)
0 0 0 0 0 1 0 0 0 0 0 0
1 2 4 2 8
Final position

Correlation result

(g)
0 8 2 4 2 1 0 0

Extended (full) correlation result

(h)
0 0 0 8 2 4 2 1 0 0 0 0

0 0 0 0 0 1 0 0 0 0 0 0 (l)
8 2 4 2 1
Position after 1 shift

0 0 0 0 0 1 0 0 0 0 0 0 (m)
8 2 4 2 1
Position after 3 shifts

0 0 0 0 0 1 0 0 0 0 0 0 (n)
8 2 4 2 1
Final position

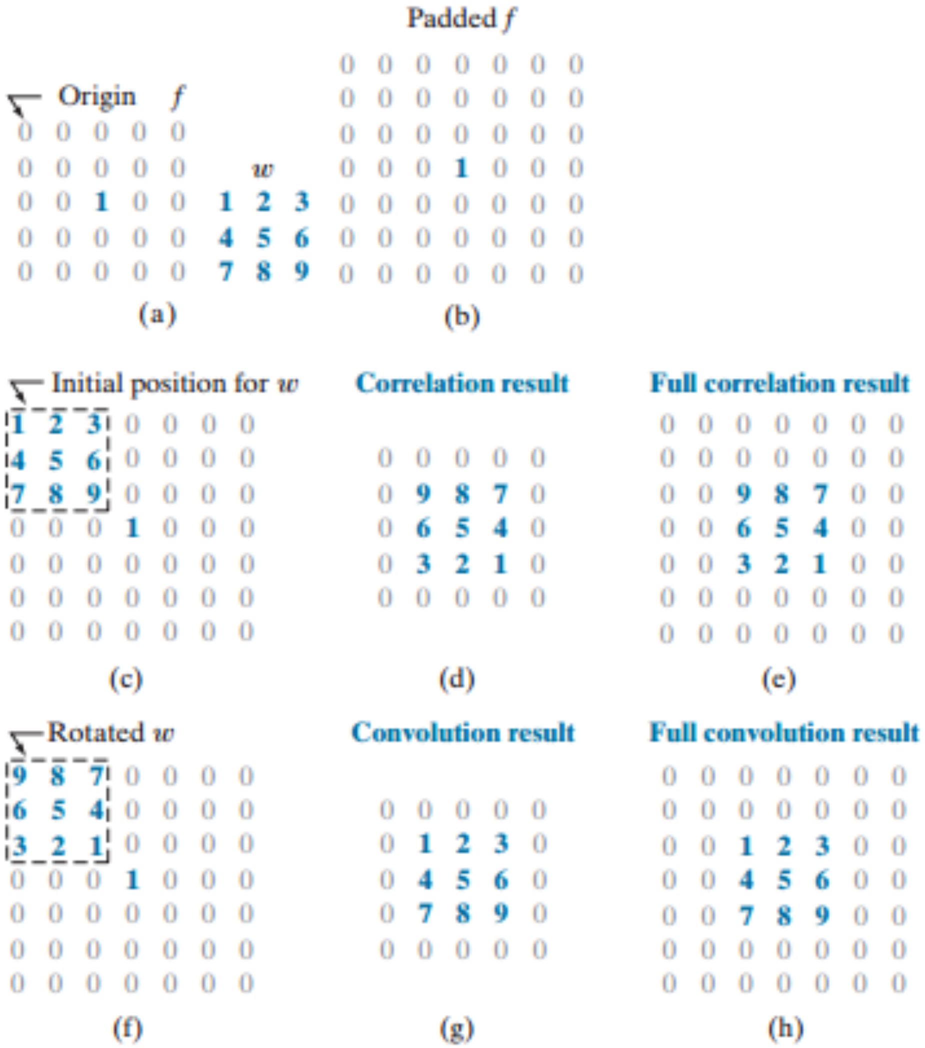
Convolution result

0 1 2 4 2 8 0 0 (o)

Extended (full) convolution result

0 0 0 1 2 4 2 8 0 0 0 0 (p)

➢ Spatial Correlation and Convolution 2-D



➢ Properties

➢ Some fundamental properties of convolution and correlation.

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

➢ Sometimes an image is filtered (i.e., convolved) sequentially, in stages say Q stage. Because of the commutative property of convolution, this multistage filtering can be done in a single filtering operation, as following.

➢ Note that We cannot do the same reduction for correlation because it is not commutative.

Padding

➢ Padding

- ➢ What is Padding?
- ➢ Types of padding.
- ➢ Comparison

What is Padding?

- Padding extends the boundaries of an image to avoid undefined operations when parts of a kernel (mask) lie outside the border of the image during filtering.

- In general, if the kernel is of size m*n, we need (m-1)/2 rows at the top and bottom and (n-1)/2 columns at the right and left and the values these new cells will take depends on the padding technique.

➢ Types of padding

- ➢ Zero padding
- ➢ Mirror (or symmetric) padding
- ➢ Replicate padding

zero padding

- In signal processing, zero padding refers to the practice of adding zeroes to a time-domain signal.
- Zero-padding is often done before performing a fast Fourier transform on the time-domain signal.

$$\mathbf{X} = \begin{bmatrix} a & b & c \\ d & f & g \\ h & j & k \end{bmatrix},$$

$$\mathrm{Pad}(1, \mathbf{X}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & a & b & c & 0 \\ 0 & d & f & g & 0 \\ 0 & h & j & k & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

➢ Mirror padding

- ➢ Values outside the boundary of the image are obtained by mirror-reflecting the image across its border.

13	12	11	12	13	14	15	14	13
8	7	6	7	8	9	10	9	8
3	2	1	2	3	4	5	4	3
8	7	6	7	8	9	10	9	8
13	12	11	12	13	14	15	14	13
18	17	16	17	18	19	20	19	18
13	12	11	12	13	14	15	14	13
8	7	6	7	8	9	10	9	8

➢ Replicate padding

- ➢ Values outside the boundary are set equal to the nearest image border value. It is useful when the areas near the border of the image are constant.

1	1	1	2	3	4	5	5	5
1	1	1	2	3	4	5	5	5
1	1	1	2	3	4	5	5	5
6	6	6	7	8	9	10	10	10
11	11	11	12	13	14	15	15	15
16	16	16	17	18	19	20	20	20
16	16	16	17	18	19	20	20	20
16	16	16	17	18	19	20	20	20

Comparison

- Zero Padding: When zero (black) padding is used, the net result of smoothing at or near the border is a dark gray border that arises from including black pixels in the averaging process.
- Using the 11*11 kernel resulted in a more prominent dark border. The result with the 21*21 the kernel shows significant blurring of all components of the image, including a darker boarder.
- The two other approaches to padding tend to solve the dark-border problem that results from zero padding.

➢ Comparison

- ➢ Mirror Padding: mirror padding does duplicate the details of the image on the edges which makes it more applicable when the areas near the border contain image details.

- ➢ Replicate Padding: It duplicate the only pixel on the edge which makes it useful when the areas near the border of the image are constant.


Thank you