Intensity Transform and Spatial Filtering

Delivered by

Dr. Subarna Chatterjee
subarna.cs.et@msruas.ac.in

© Ramaiah University of Applied Sciences

Image Enhancement

- **Objective:** To process an image so that the result is more suitable than the original image for a specific application.
- The word specifics is important, because it establishes at the outset; the techniques we will discuss here are very much problem oriented.
- Method that is useful for enhancing **X-ray images** may not necessarily be the best approach for enhancing pictures of **Mars transmitted** by a **space probe**.
- Regardless of the method used, however, image enhancement is one of the most interesting and visually appealing areas of image processing.

© Ramaiah University of Applied Sciences

Intensity Transform and Spatial Filtering

- **Image enhancement** is the process that improves the quality of the image for a specific application. It fall into two broad categories:
  - **spatial domain** methods and
  - **frequency domain** methods
- **Image enhancement methods:**

  1. Spatial domain methods (image plane)

  It refers the image plane itself; Techniques are based on direct manipulation of pixels in the image.

  2. Frequency domain methods

  Techniques are based on modifying the Fourier transform of the image.

  3. Combinations methods:

  There are some enhancement techniques based on various combinations of methods from the two first categories.

© Ramaiah University of Applied Sciences

Cont..

- **Spatial domain** refers to the aggregate of pixels composing an image.
- Spatial domain methods are **procedures** that operate directly on pixels (spatial domain) can be denoted by the expression:

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

  Where: **f(x,y)** is the **input** image, **g(x,y)** is the processed (**output**) image;
- **Mask/Filter:** neighborhood of a point **(x, y)** can be used, by using a **square/ rectangular** (commonly used) or **circular sub image** area centre at **(x, y)** as shown in Fig 3.1**.**
- The **center** of the **sub image** is moved from pixel to pixel starting at the top left of the corner

© Ramaiah University of Applied Sciences

Cont..

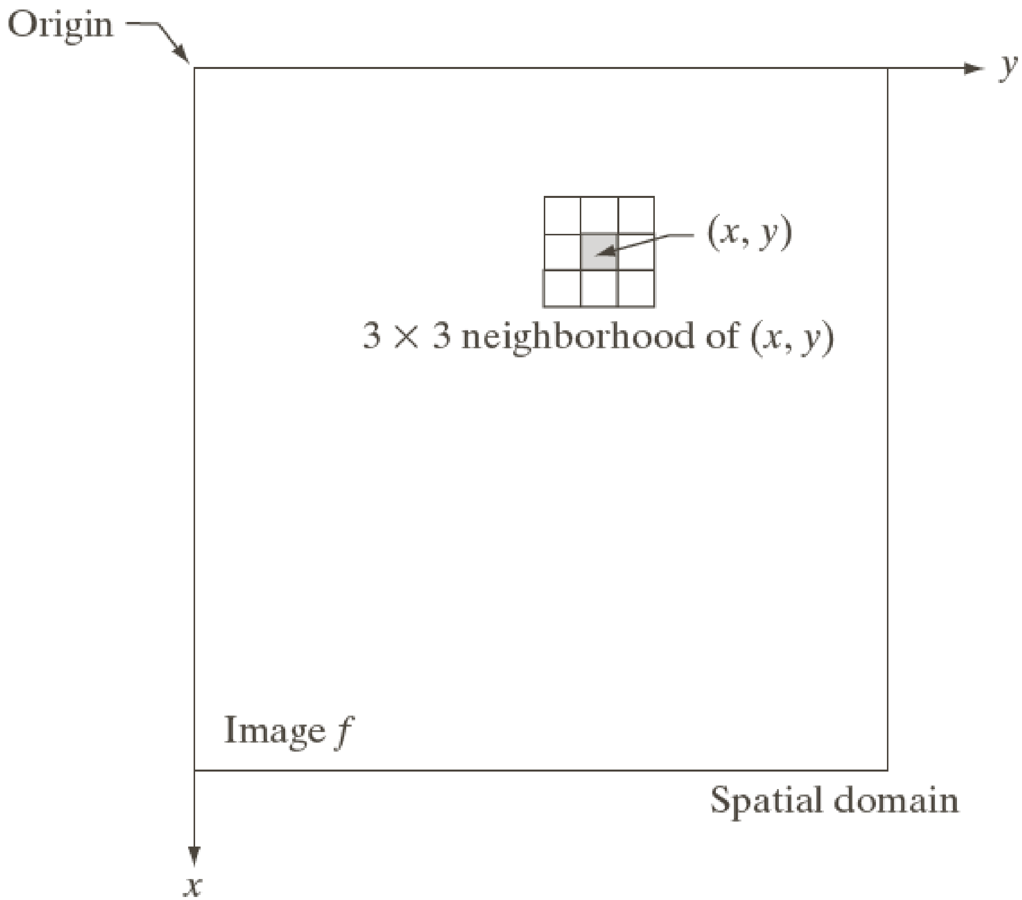


FIGURE 3.1 A 3 × 3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

T-is an operator on **f(x,y)** defined over some neighborhood of **(x,y)**; **T** can operate on a set of I/P images, **example** performing the pixel-by-pixel sum of **K** images for noise reduction.

© Ramaiah University of Applied Sciences

Contrast Stretching and Thresholding

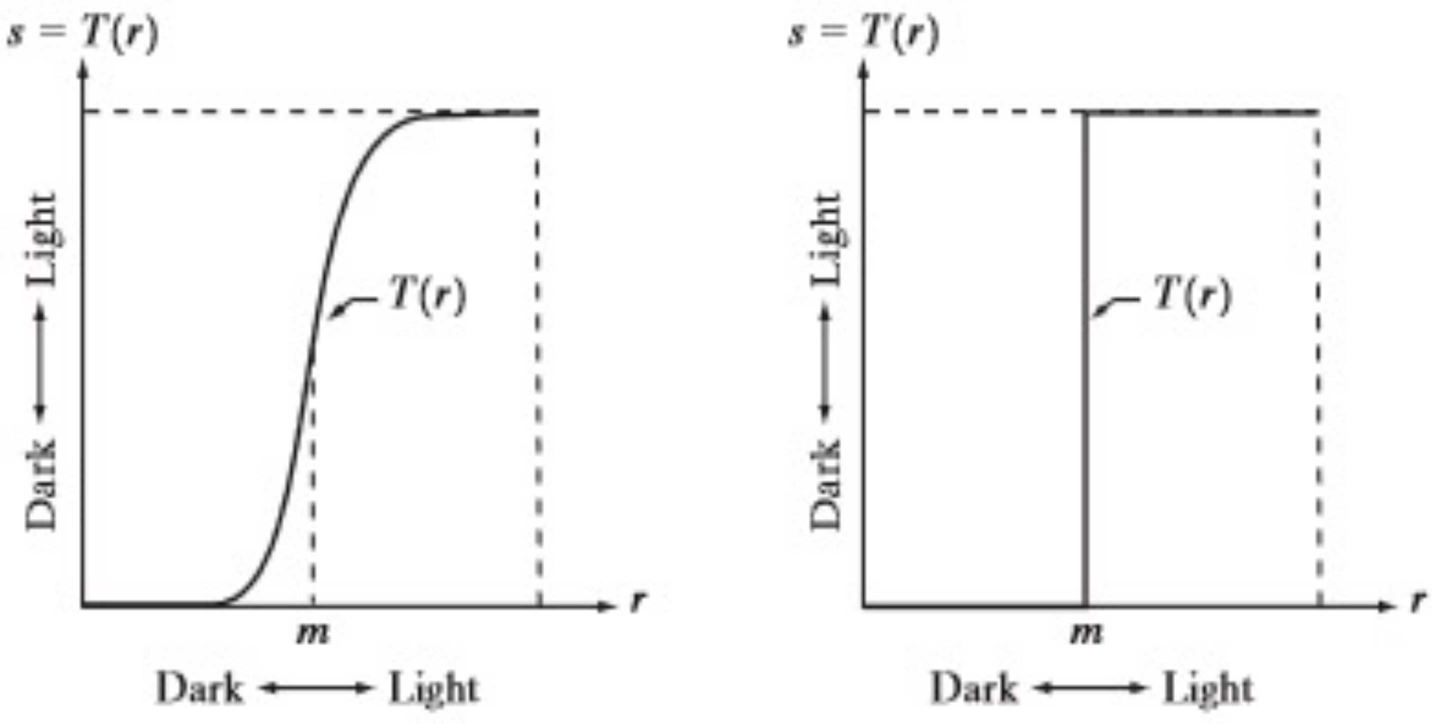
- **Intensity Transformation Functions**
  - **Point processing:**
    - **Neighborhood:** ***1*1 pixel*** (that is, a single pixel). **g** depends on only the value of **f** at **(x,y).**
    - And **T** becomes an **intensity transformation function** (*gray-level or mapping function) of the form:*

      s = T(r)

      Where: **r** = gray level of **f (x,y)** , **s** = gray level of **g(x,y)**
  - **Example**, if T(r) has form as in Fig.3.2(a), then **T** produce an image of higher contrast than the original by **darkening** the levels below **m** and **brightening** the levels above **m** in the original image. This technique is known as **contrast stretching**, the values of **r** below **m** are **compressed** by the transformation function into a **narrow range** of **s**, toward **black**.

© Ramaiah University of Applied Sciences

Cont..



a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

Opposite effect takes place for values of **r** above **m**. Case shown in Fig.3.2(b), T(r) produces a **two-level (binary)** image. A mapping of this form is called a **thresholding function**. Because enhancement at any point in an image depends only on the gray level at that point, techniques in this category often are referred to as point processing.

© Ramaiah University of Applied Sciences

Cont..

– **Intensity transformation example:**

1. **Contrast stretching: (*Normalization)*** If ***T(r)*** has the form as shown in figure, the effect of applying the transformation to every pixel of ***f*** to generate the corresponding pixels in ***g*** would:
   - Produce ***higher contrast*** *than the original by:*
   - ***Darkening*** the levels below ***m*** in the original image.
   - ***Brightening*** *the levels above* ***m*** *in the original image.*

   Contrast stretching: simple image enhancement technique that improves the contrast in an image by '***stretching'*** *the range of intensity values it contains to span a desired range of* values; typically, it uses a ***linear scaling function***

2. **Thresholding:**
   - Produce a ***two-level*** *(binary) image, a mapping of this form called* ***thresholding function.***

© Ramaiah University of Applied Sciences

Some Intensity Transform Function

- Procedures that operate directly on pixels (spatial domain) can be denoted by the expression:

- **Some Basic Intensity Transformation Functions:**

- Three basic types of functions used frequently for image enhancement:

  - ***Linear functions:*** ***negative*** *and* ***identity*** *transformations.*

  - ***Logarithmic:*** ***log*** *and* ***inverse*** *log transformations.*

  - ***Power-law functions*** *(* n_{th} ***power*** *and* n_{th} ***root*** *transformations, Gamma correction)*

  - ***Piecewise linear transformations***

© Ramaiah University of Applied Sciences

Some Intensity Transform Function

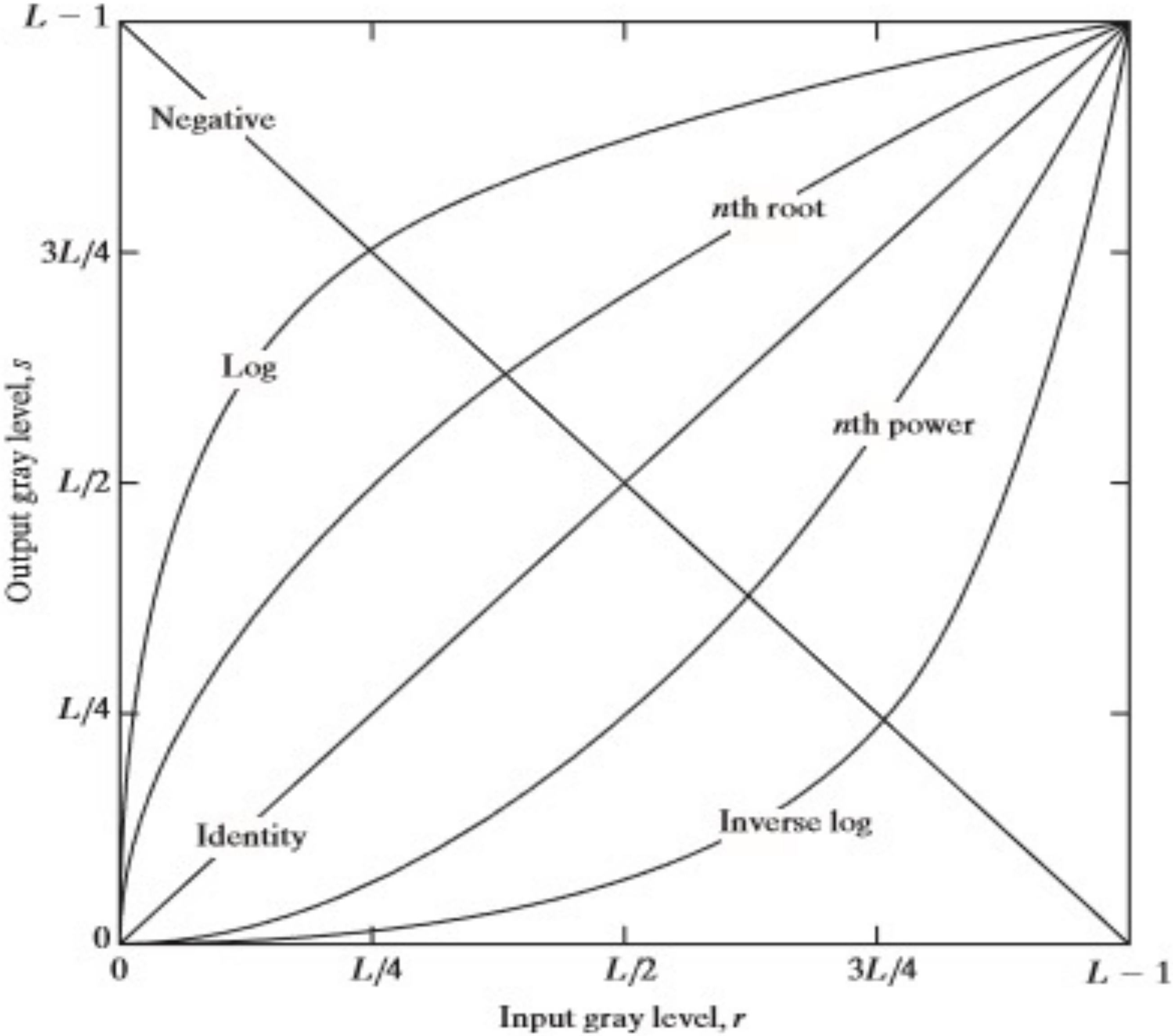


FIGURE 3.3 Some basic intensity transformation functions. All curves were scaled to fit in the range shown.

© Ramaiah University of Applied Sciences

Cont..

1. ***Identity function:***
   - Output intensities to input intensities.
   - Is included in the graph only for completeness.

2. ***Image negatives:***
   - The negative of an image with gray level in the range **[0, L-1],** where **$L = 2^n = 2^8$**, **n=1,2,....** is obtained by using the negative transformation's expression:

     S =L-1-r//

   - That reverses the intensity levels of an input image.
   - The negative transformation is suitable for enhancing white or gray detailed embedded in dark regions of an image, especially when the black area are dominant in size.

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input(M x N)

OUTPUT: Negative gray scale image output(M x N)

STEPS :

1. Read the Input Gray scale Image (M x N)
   - //I=imread('1.jpg'); figure, imshow(I); [m, n]=size(I);

2. Calculate negative of the input image by applying the equation, s = L – 1 – r

```
for (i=0; i<M; i++)
  for (j=0; j<N; j++)
      output[i][j] = L – 1 – input[i][j] ;
      //I2[i][j]=255-I[i][j];
  //figure, imshow(I2);
```

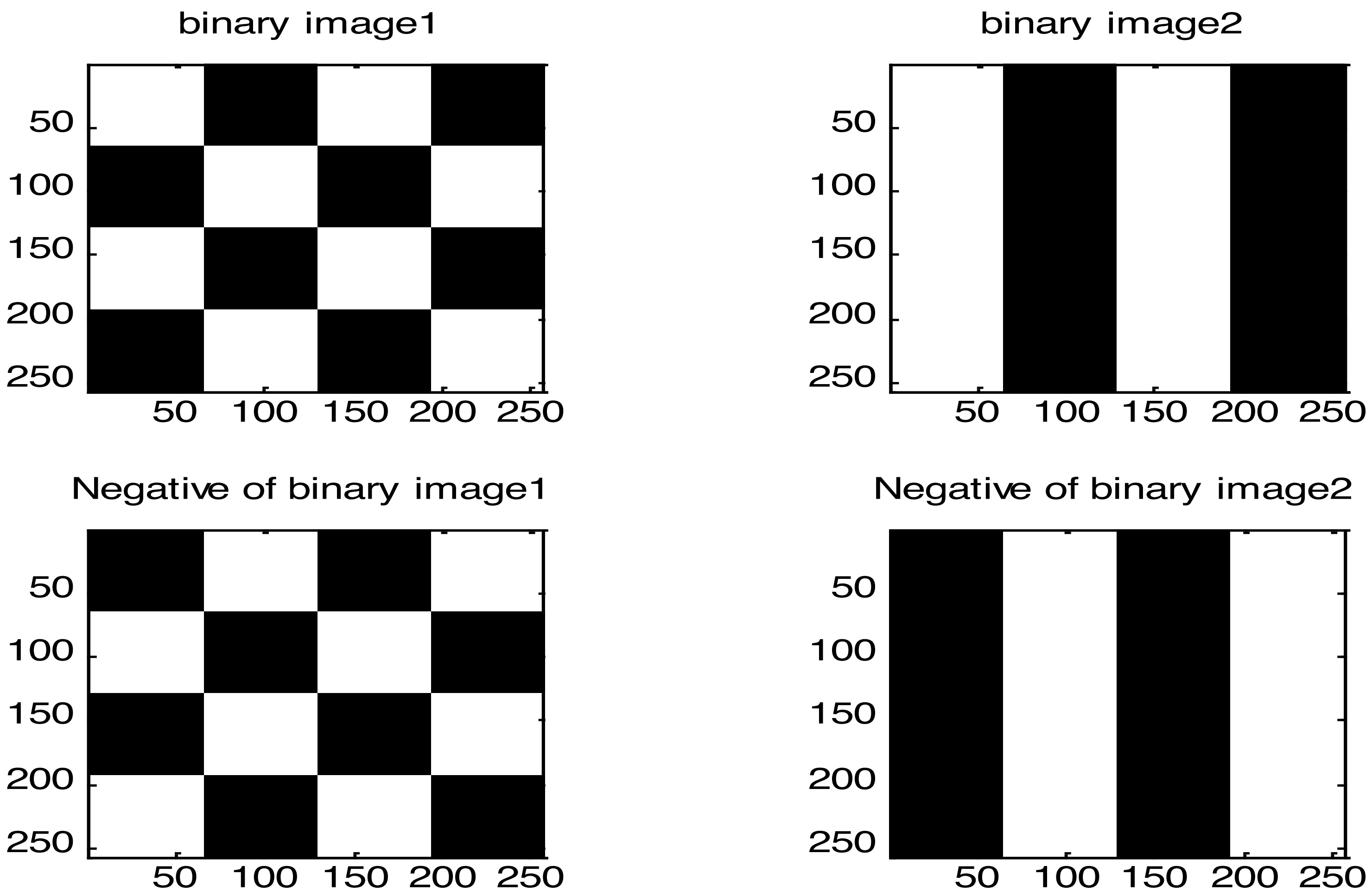
Where **r** is the **input pixel**

s is the **output pixel**

L is the **gray scale level**

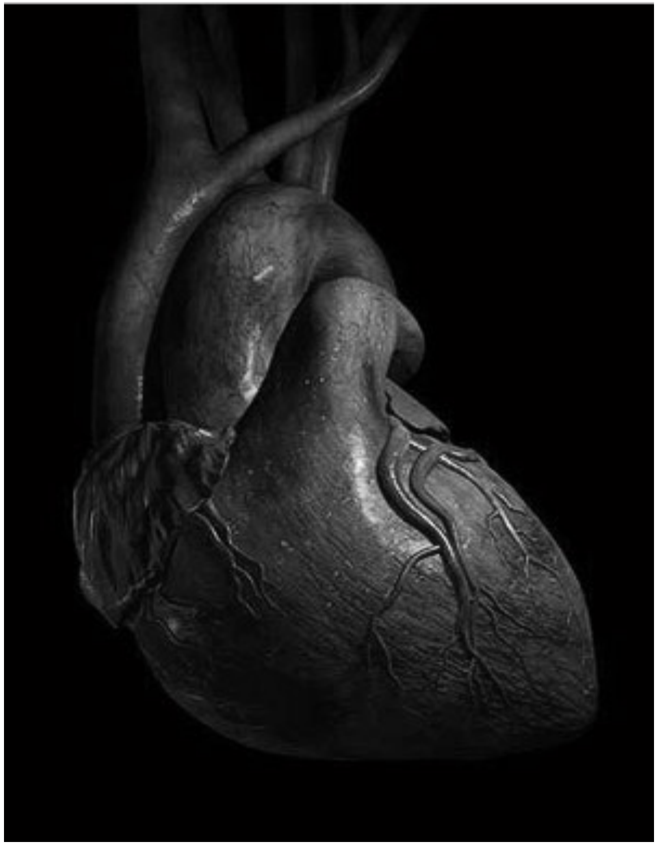
© Ramaiah University of Applied Sciences

Negative of Images

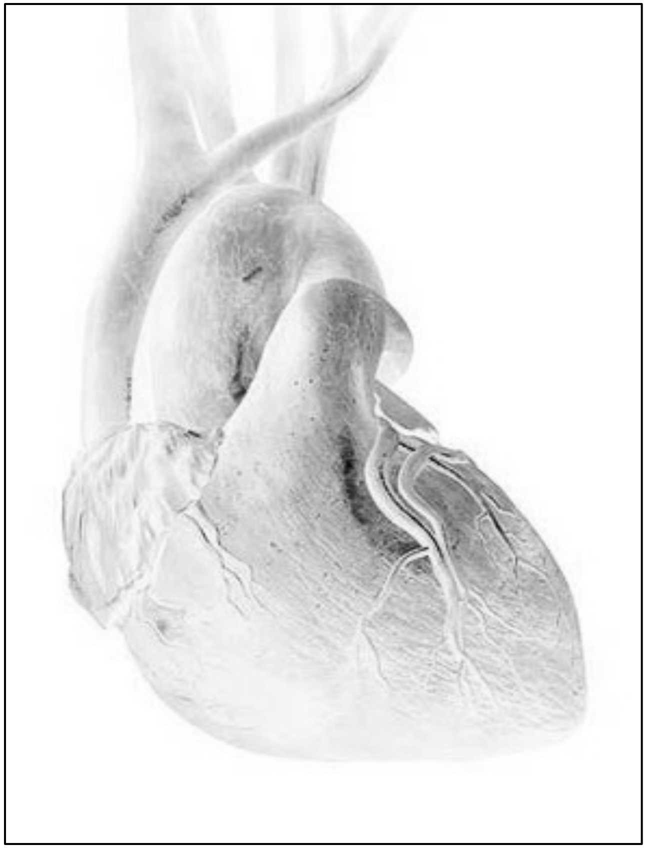


© Ramaiah University of Applied Sciences

Negative of an Image



Original Image



Negative of an Image

© Ramaiah University of Applied Sciences

Negative

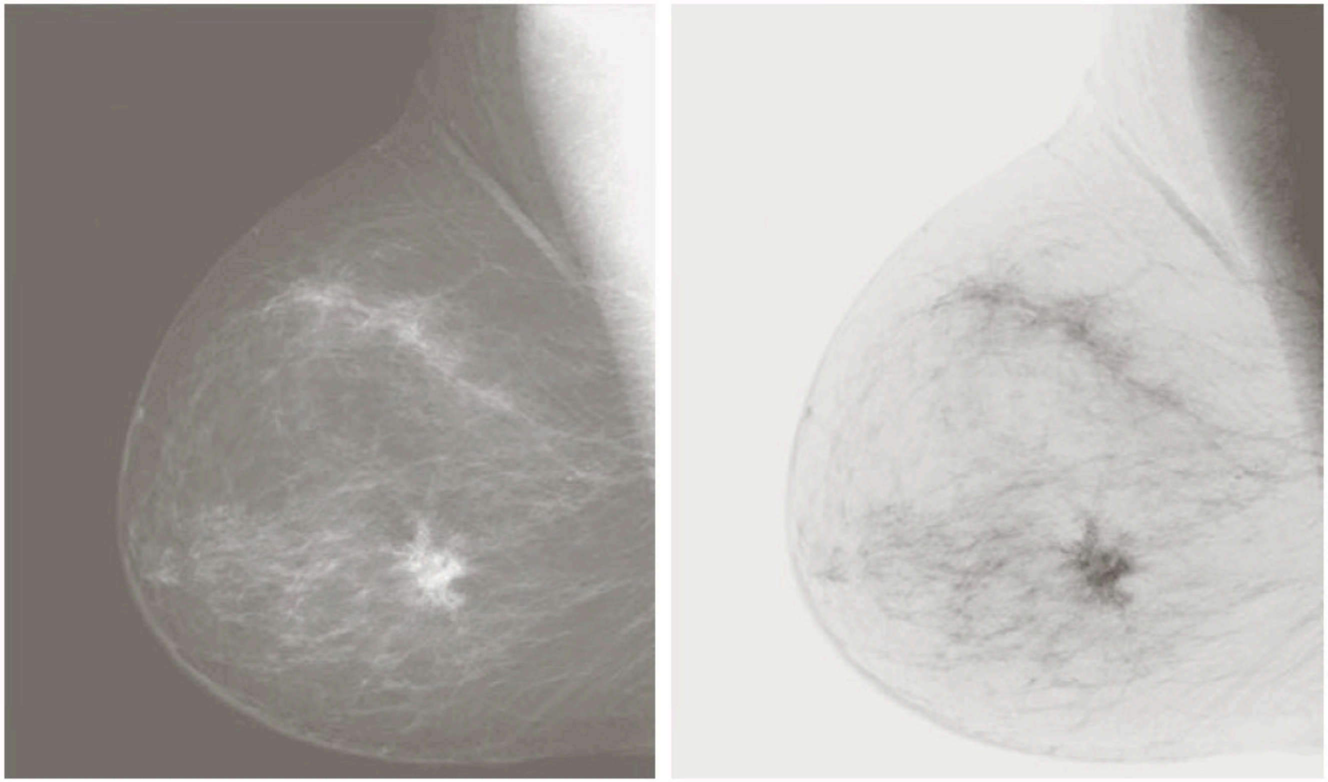


FIGURE 3.4 (a) Original digital mammogram. (b) Negative image obtained using the negative transformation in Eq. (3.2-1). (Courtesy of G.E. Medical Systems.)

© Ramaiah University of Applied Sciences

Cont..

3. Log Transformation:

- The general form of the log transformation:
  - **$S = c. \log (1+r)$** where **c** is a constant, and **$r \geq 0$**
- **Log** transformation *maps a **narrow range** of low gray-level values in the input image* into a ***wider range** of the output levels.*
- Used to ***expand*** *the values of **dark pixels** in an image while **compressing** the **higher-level** values.*
- It ***compresses the dynamic range of images with large variations in pixel values.***
- Example of image with dynamic range: ***Fourier spectrum image.***
- It can have intensity range from 0 to 10^6 or higher.
- We can't see the significant degree of detail as it will be lost in the display.

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input (M x N)

OUTPUT : Log transformed gray scale image output (M x N)

STEPS :

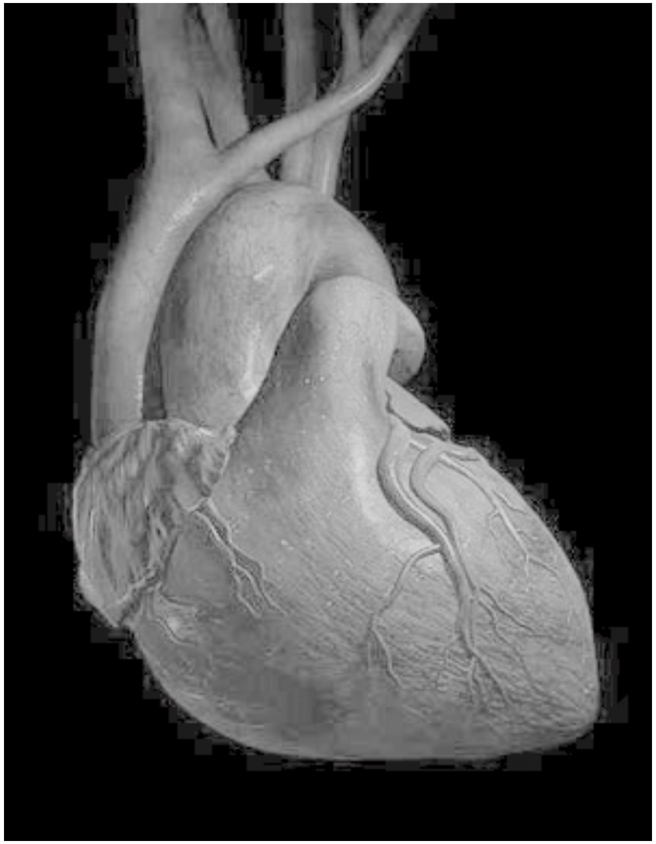
1. Read the Input gray scale Image (M x N)
2. Calculate Log of the input image by applying the Log transform

```
for (i=0; i<M; i++)
    for (j=0; j<N; j++)
        output[i][j] = c * log(1+ input[i][j]);
```

Where **c** is the constant

© Ramaiah University of Applied Sciences

Log Transformations



- Log Transformations maps a narrow range of low intensities values of input levels into a wider range of output levels – expands the values of dark

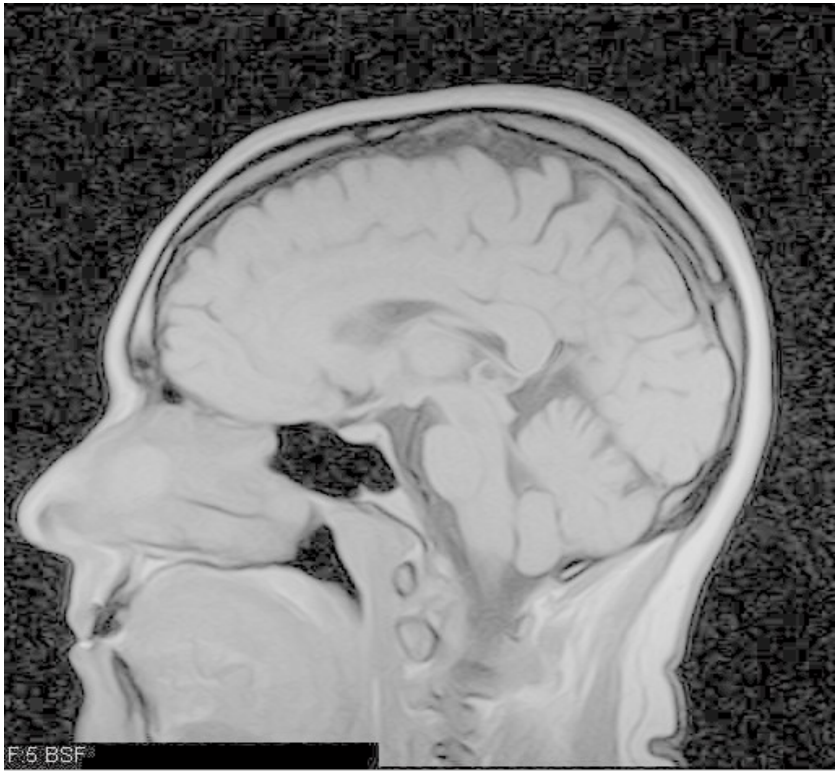
© Ramaiah University of Applied Sciences

Cont..

- A logarithmic transform is appropriate when we want to enhance the low pixel values at the expense of loss of information in the high pixel values
- **Example**, the man in fig 1 is photographed in front of a bright background
- The dynamic range of the film material is too small, so that the gray levels on the subject's face are clustered in a small pixel value range
- A logarithmic transform spreads them over a wider range, while the higher values are compressed

© Ramaiah University of Applied Sciences

Cont..



- On the other hand, applying a logarithmic transform to bright images is less appropriate, because most of its details are contained in the high pixel values
- Applying the logarithmic operator yields the image that shows a lot of information is lost during the transform

© Ramaiah University of Applied Sciences

Cont..

4. ***Inverse logarithm Transformation:***
   - Do opposite to the log transformations.
   - Used to *expand the values of high pixels in an image while compressing the darker-level values.*
5. ***Power-law (Gamma) Transformation:***
   - The general form of the power-law transformation: $\mathbf{S = c.r^{\delta}}$ **,** where **c** and **δ ≥0**
   - Different transformations curves are obtained by varying **δ** (***gamma****).*
   - By convention, the exponent in the power-law equation is referred to as ***gamma***
   - Many image capturing, printing and display devices use **gamma** correction, which enhances the given image by power-law response phenomena.

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input (M x N)

OUTPUT : Log transformed gray scale image output (M x N)

STEPS :

1. Read the Input gray scale Image (M x N)
2. Calculate Log of the input image by applying the Log transform

```
for (i=0; i<M; i++)
    for (j=0; j<N; j++)
        output[i][j] = c * (input[i][j])^δ;
```

Where **c** is the constant and **$\delta \geq 0$**

© Ramaiah University of Applied Sciences

Gamma Correction

- The process used to correct this power-law response phenomena is called ***gamma correction***.
- As in the case of the log transformation, power-law curves with fractional values of **δ** map a **narrow range** of **dark I/P** values into a **wider range** of O/P values, with the opposite being true for higher values of I/P levels.
- Unlike the log function, however, we notice here a family of possible transformation curves obtained simply by varying **δ** .
- We see in Fig. 3.6 that curves generated with values of **$\delta>1$** have exactly the opposite effect as those generated with values of **$\delta < 1$**.
- Finally, we note that **$S = c.r^{\delta}$** reduces to the identity transformation when **$c = \delta = 1$**.

© Ramaiah University of Applied Sciences

Gamma Correction

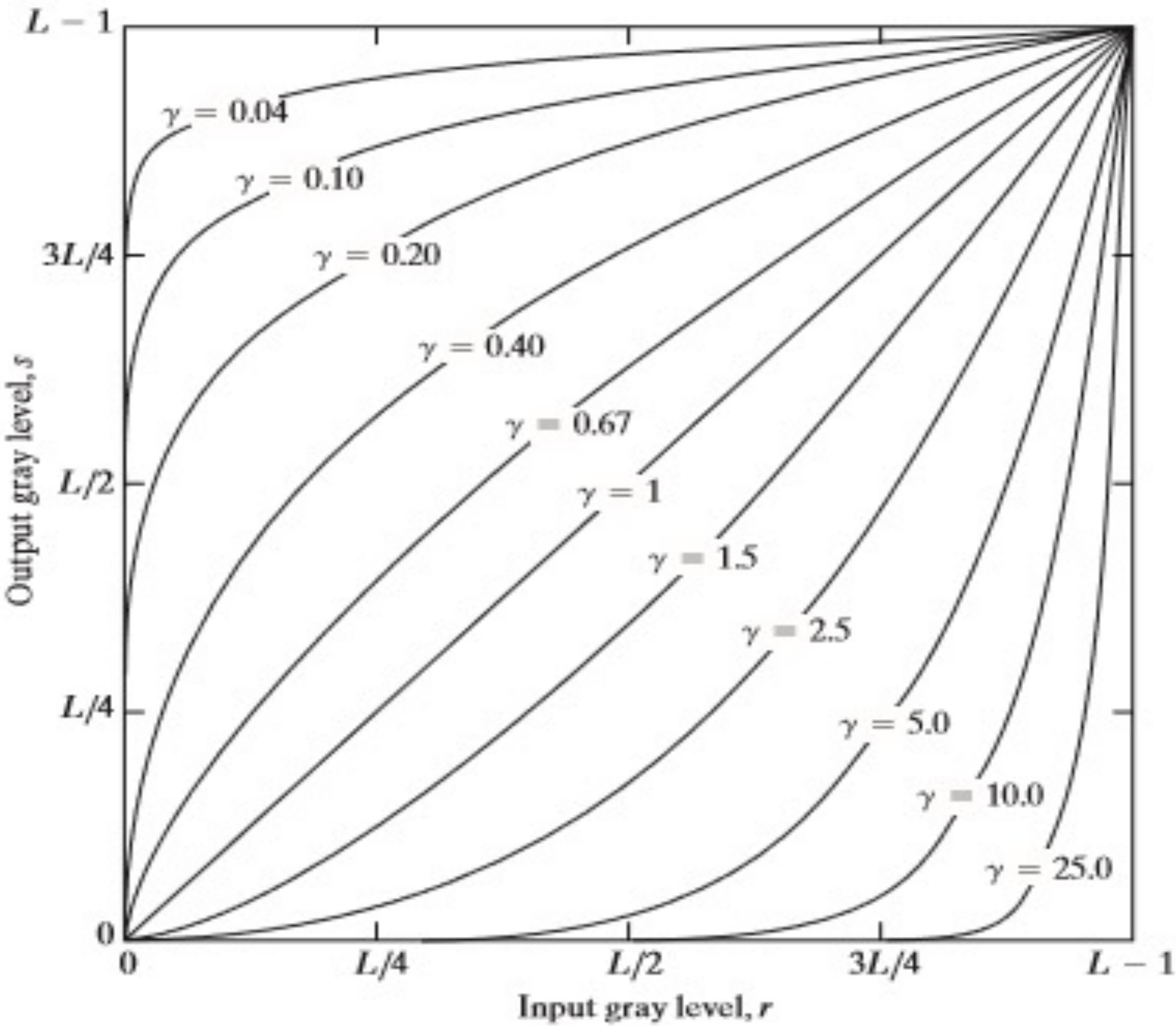


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

Plots of ***s*** versus ***r*** for various values of **δ** or **γ** are shown in Fig. 3.6

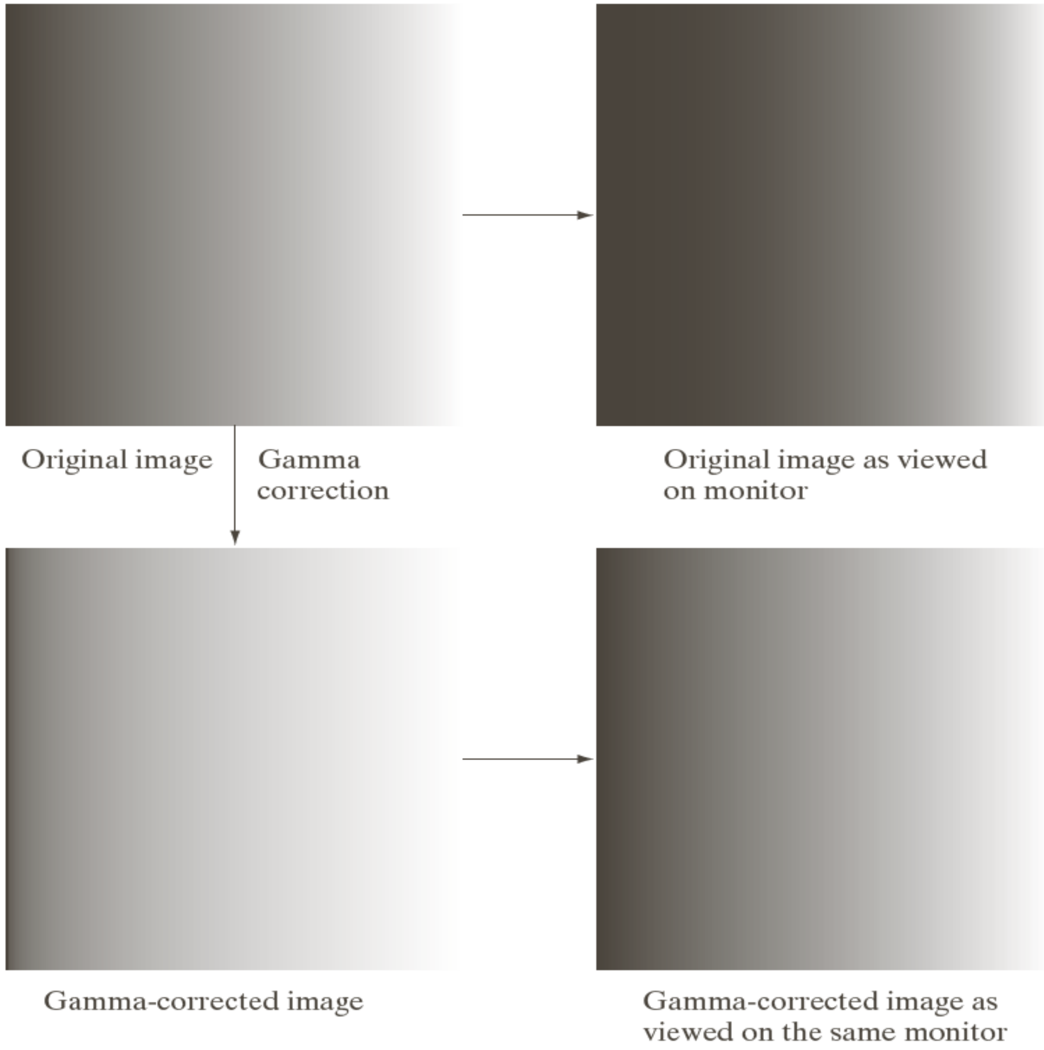
© Ramaiah University of Applied Sciences

Gamma Correction

- **Example** CRT devices have an intensity-to-voltage response i.e. a power function, with ***δ*** *varying from* **1.8** to **2.5**.
- The curve for ***δ*** =2.5 in Fig. 3.6, we see that such display systems would tend to produce **images that are darker than intended**. This effect is illustrated in Fig. 3.7.
- Figure 3.7(a) shows a simple gray-scale linear wedge I/P into a CRT monitor. The O/P of the monitor appears darker than the I/P, as shown in Fig. 3.7(b). Gamma correction in this case is straightforward.
- Now pre-process the I/P image before putting it into the monitor by performing the transformation $s = r^{1/2.5} = r^{0.4}$. The result is in Fig. 3.7(c).
- When I/P into the same monitor, this gamma-corrected I/P produces an O/P that is close in appearance to the original image, as shown in Fig. 3.7(d). A similar analysis would apply to other imaging devices such as scanners and printers.

© Ramaiah University of Applied Sciences

Example: Gamma Transformations

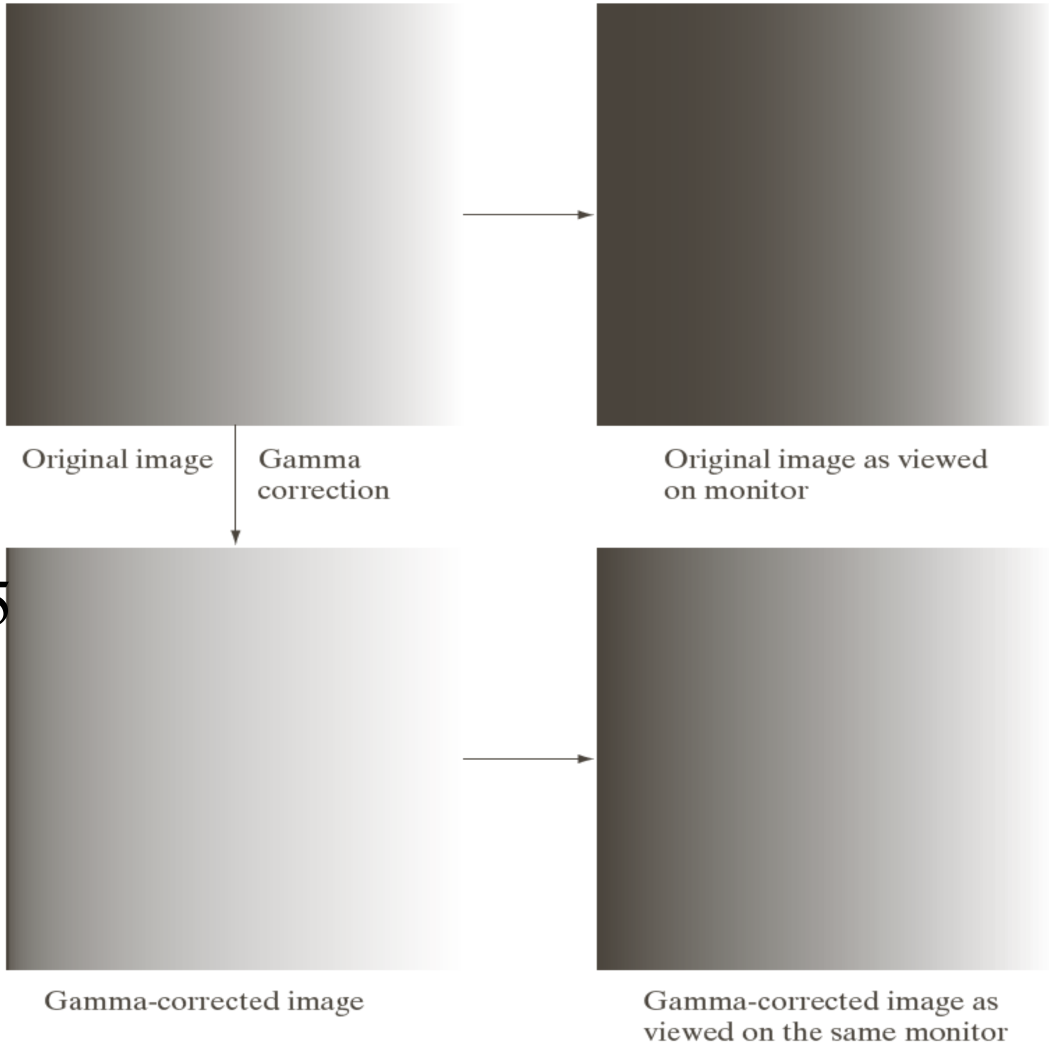


a	b
c	d

FIGURE 3.7
(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).

© Ramaiah University of Applied Sciences

Example: Gamma Transformations



$$s = r^{1/2.5}$$

Cathode ray tube (CRT) devices have an intensity-to-voltage response that is a power function, with exponents varying from approximately 1.8 to 2.5

© Ramaiah University of Applied Sciences

Cont..

γ = ?

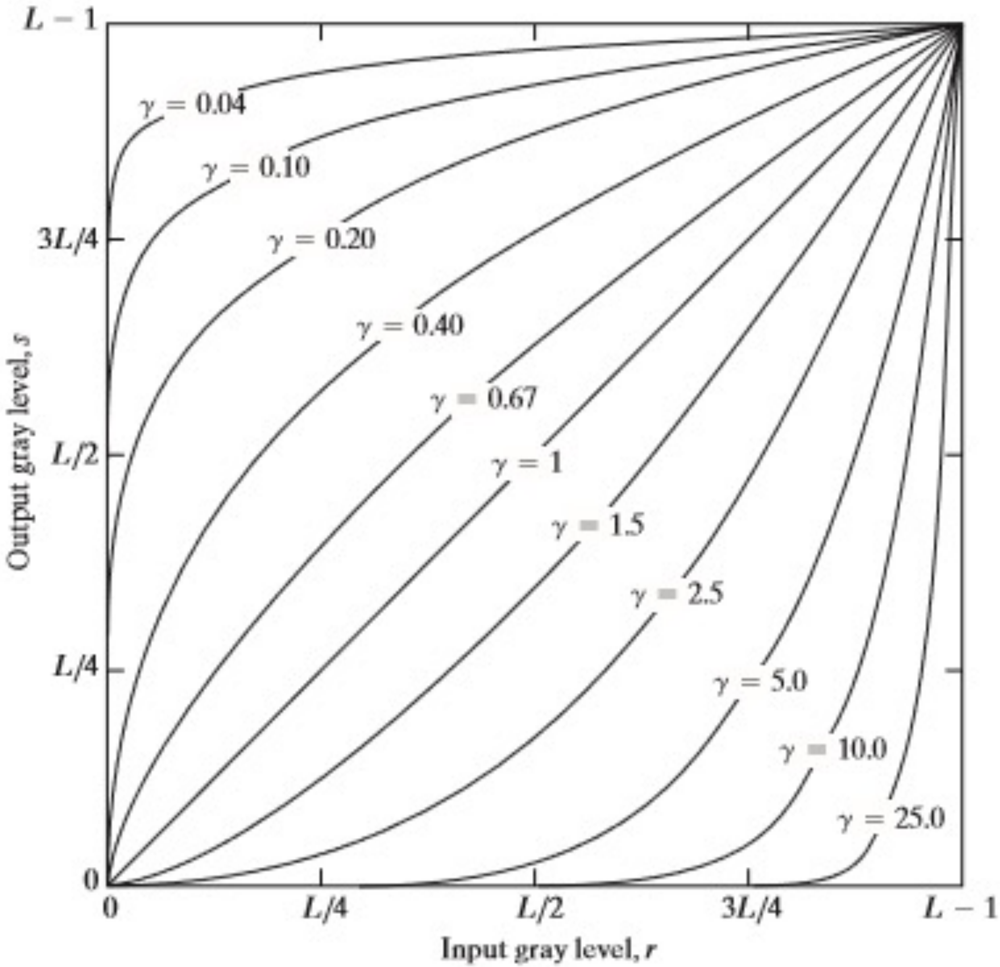


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

© Ramaiah University of Applied Sciences

Gamma Correction

γ = ?

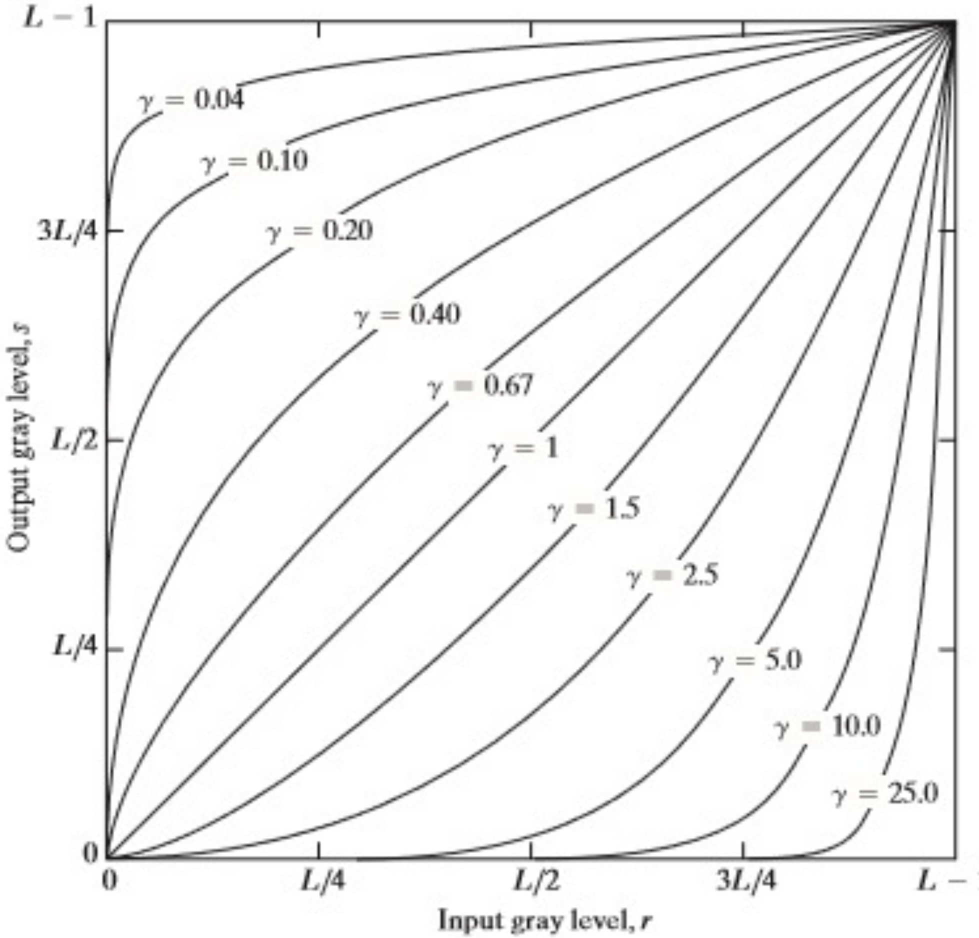
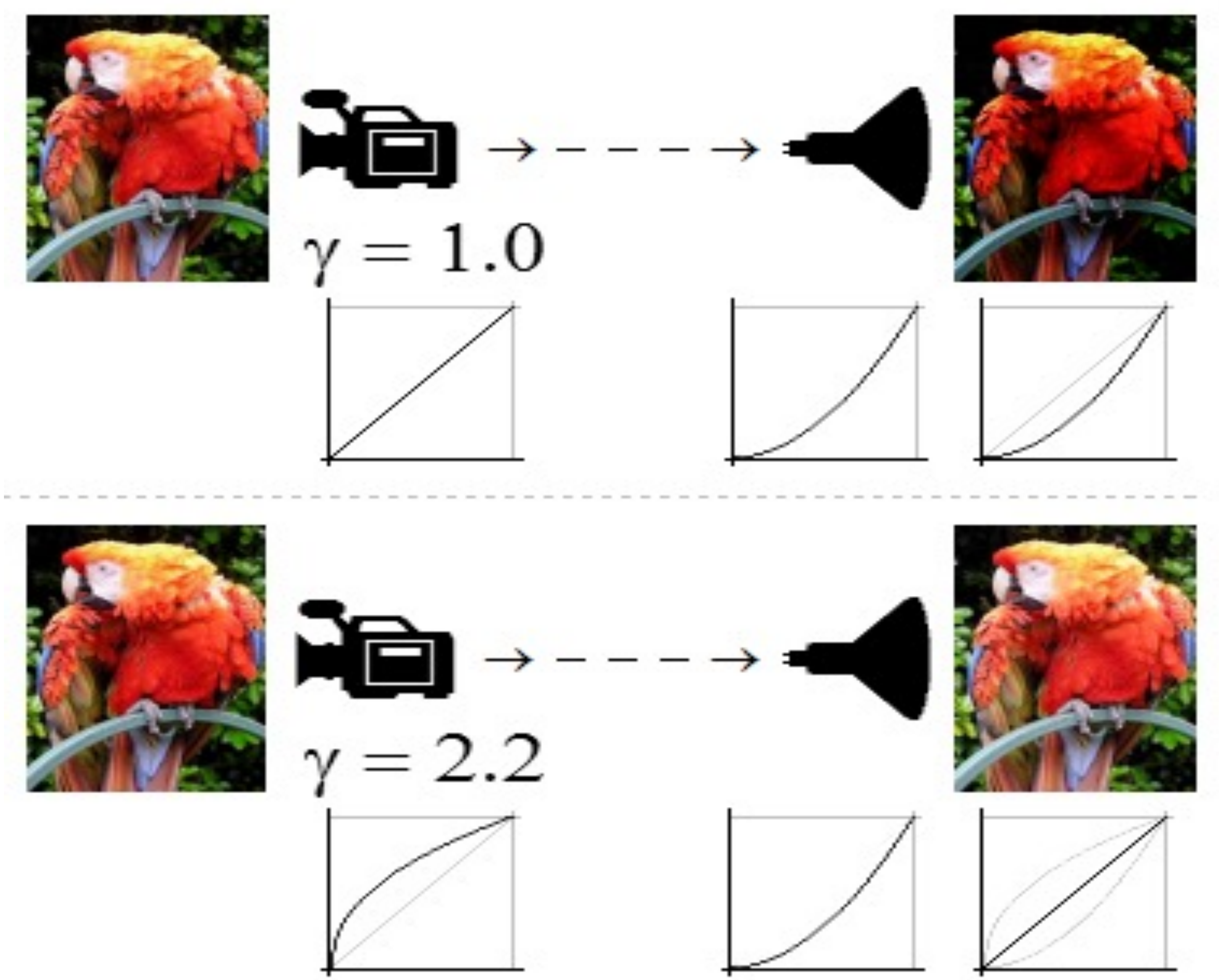


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

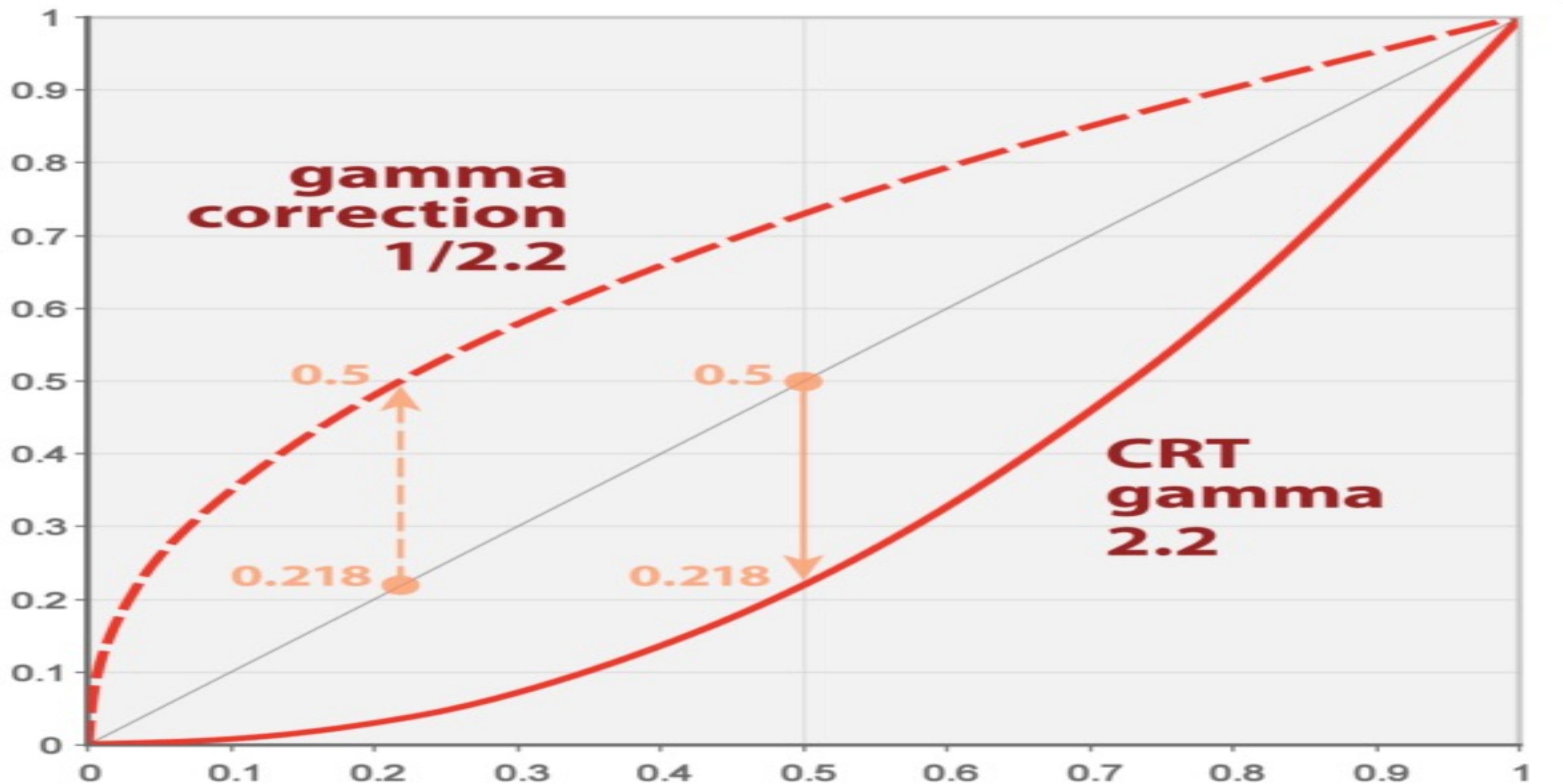
© Ramaiah University of Applied Sciences

Gamma Compression (gamma encoding)

- Systems with linear and gamma-corrected cameras
- The dashes in the middle represent the storage and transmission of image signals or data files
- The 3 curves represent I/P –O/P functions of the camera, the display, and the overall system, respectively.



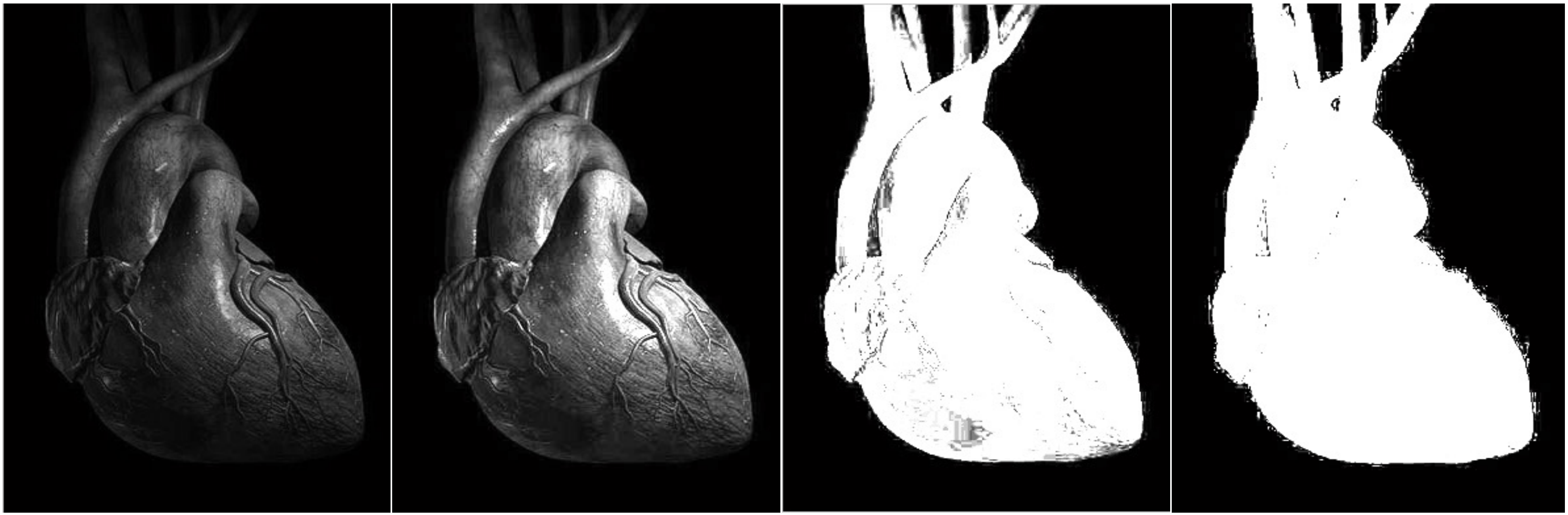
Example of CRT gamma correction



© Ramaiah University of Applied Sciences

Gamma Correction

- Each panel shows the display gamma that the pixel values have been adjusted for



Input Image | Gamma(γ) = 1, c=2 | Gamma(γ) = 2, c=2 | Gamma(γ) = 3, c=2

What is gamma correction ?

- Some of the images on the server, especially the older images, are not gamma corrected
- In other words, they are not corrected for the nonlinear relationship between pixel value and displayed intensity that is typical for a color television monitor
- This nonlinear relationship is roughly a power function, i.e. *displayed_intensity = pixel_value^gamma*
- Most monitors have a gamma between 1.7 and 2.7. Gamma correction consists of applying the inverse of this relationship to the image before display, i.e. by computing *new_pixel_value = old_pixel_value^(1.0/gamma)*.

© Ramaiah University of Applied Sciences

Why is gamma correction is important?

- Displaying an image accurately on a computer screen is of concern. Images that are not corrected properly can look either bleached out, or too dark.
- Trying to reproduce colors accurately also requires some knowledge of gamma correction because varying the value of gamma correction changes not only the brightness, but also the ratios of red to green to blue.
- Gamma correction has become increasingly important in the past few years, as use of DI for commercial purposes has increased.
- It is not unusual that images created for millions of people, the majority have different monitors or monitor settings. Some computer systems even have partial gamma correction built in.

© Ramaiah University of Applied Sciences

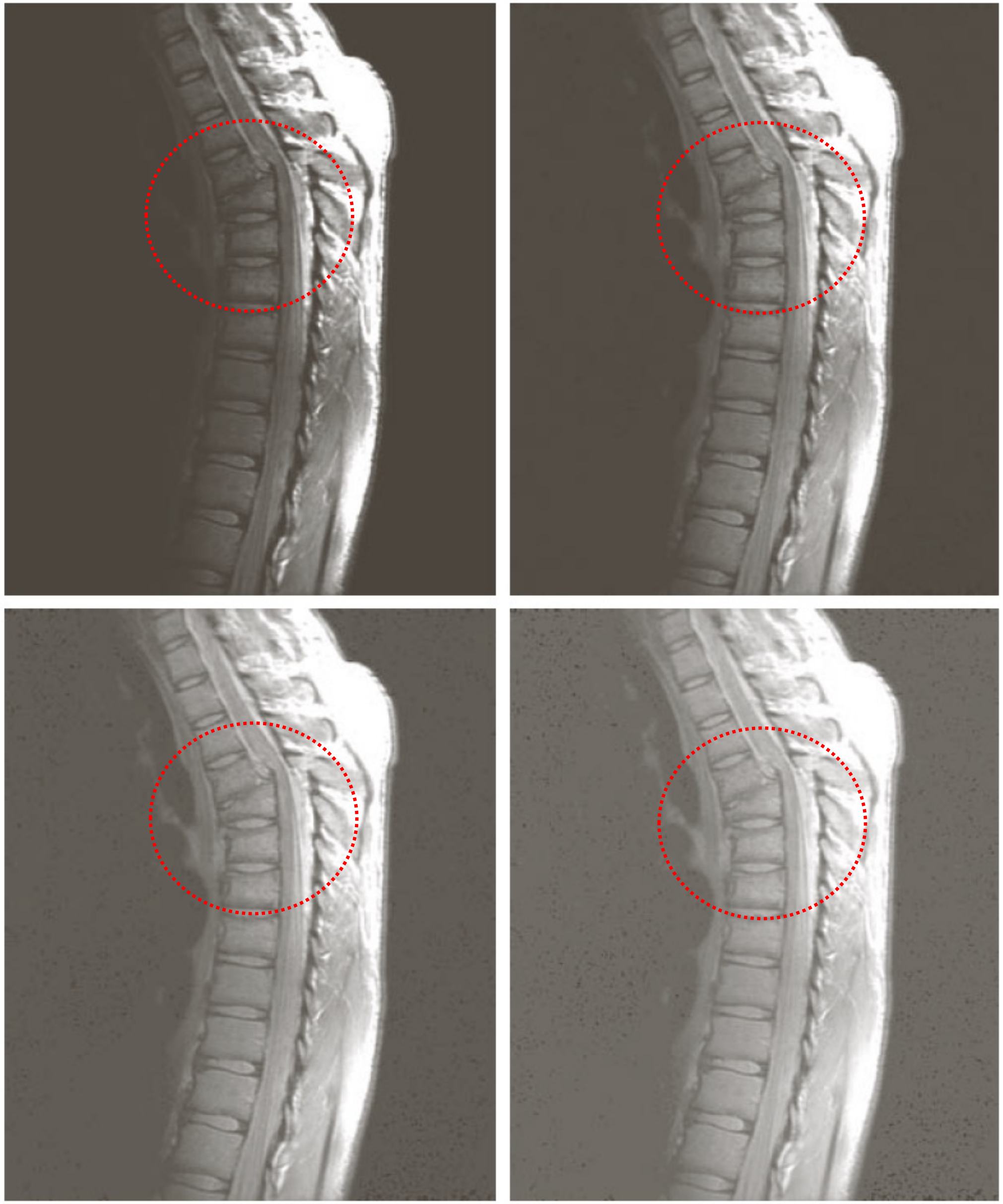
Cont..

- Current image standards do not contain the value of gamma with which an image was created, thus complicating the issue. Given these constraints, a reasonable approach when storing images in a Web site is to preprocess the images with a gamma that represents an “average” of the types of monitors and CS that one expects in the open market at any given point in time.
- It is useful for GP contrast manipulation. Figure 3.8(a) shows a MRI of an upper thoracic human spine with a fracture dislocation and spinal cord impingement. The fracture is visible near the vertical center of the spine, approximately one-fourth of the way down from the top of the picture.
- Since the given image is dark, details are desirable. This can be done by power-law transformation with a fractional exponent.

© Ramaiah University of Applied Sciences

Example: Gamma Transformations

as γ decreased from 0.6 to 0.4, more detail became visible.



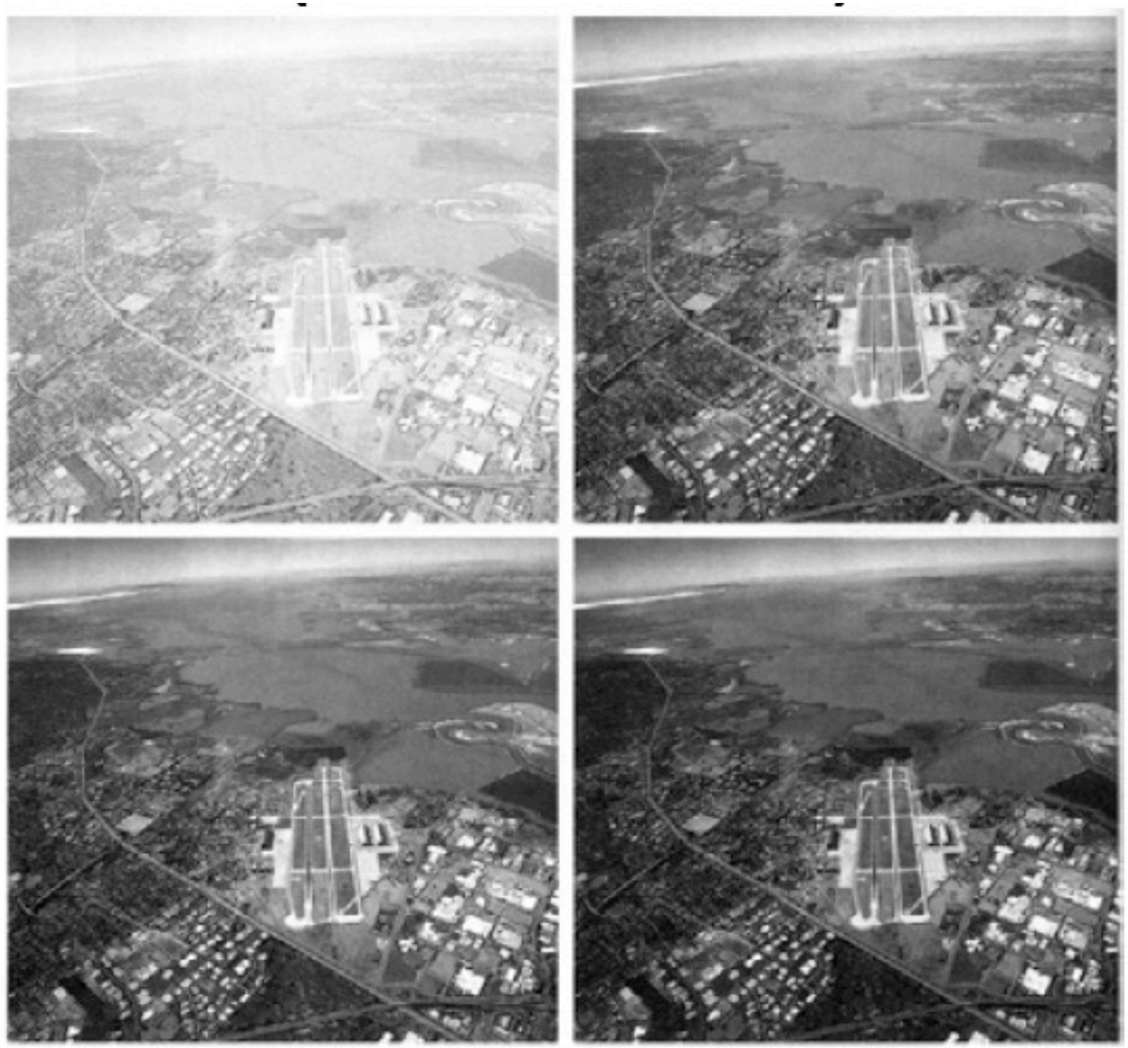
a	b
c	d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

© Ramaiah University of Applied Sciences

Power-Law Transformations

The original "washed-out" image out and gamma transformed with $c = 1$ and $\gamma = 3.0, 4.0$, and 5.0 respectively. The last image may appear having areas that are too dark



Cont..

- The difference between the **log function** and the **power-law functions** is that using the power-law function a family of possible transformation curves can be obtained just by varying the λ.

- Power-law transformations are much more **versatile** for spreading/compressing of gray levels in an image than the log transformation.

- Log function has the important characteristic that it compresses the dynamic range of images with large variations in pixel values.

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input(M x N)

OUTPUT: Power-Law transformed Gray scale image output(M x N)

STEPS :

1. Read the Input Gray scale Image (M x N)
2. Calculate Power-Law of the input image by applying the equation $s = cr^{\gamma}$

   Where s is input and r is output image

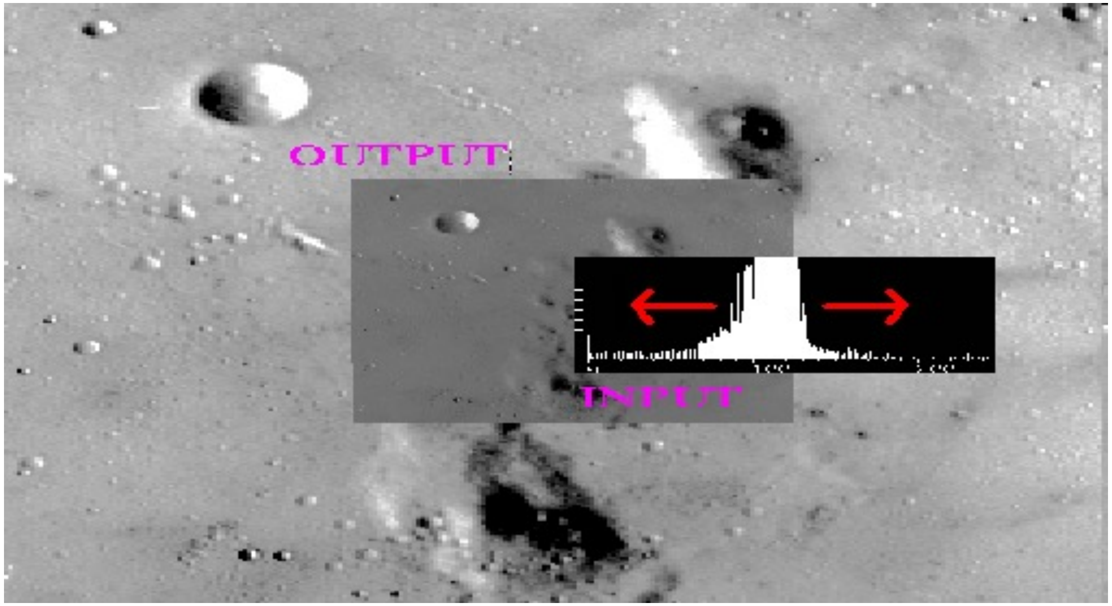
```
for (i=0; i<M; i++)
        for (j=0; j<N; j++)
                Calculate    output[i][j] = c * pow( input[i][j], gamma );
```

   Where **c** is the constant gamma is the power of the input pixel

© Ramaiah University of Applied Sciences

Piecewise Linear Transformations

- Another approach is to use a piecewise linear function of arbitrary complexity
  - **Contrast stretching**



  - **Intensity-level slicing**

© Ramaiah University of Applied Sciences

Cont..

6. Piecewise-linear Transformation Functions

- A complementary approach to the methods discussed is to use piecewise linear functions.
  - **Advantage:** the form of piecewise function is *arbitrarily complex* (more options to design), some important transformations can be formulated only as piecewise functions.
  - **Disadvantage:** their specification requires more user input.

A. **Contrast stretching:** One of the most (simplest) piecewise functions is the ***contrast stretching, which is*** used to enhance the low contrast images.

© Ramaiah University of Applied Sciences

Cont..

A. Contrast stretching:

- Low-contrast images can result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture during image acquisition.
- The idea of **contrast stretching** is to increase the dynamic range of the gray levels in the image being processed.
- Figure 3.10(a) shows a transformation used for contrast stretching. The locations of points **(r_1, s_1)** and **(r_2, s_2)** control the shape of the transformation function.

a) $s_1 = r_1$ and $s_2 = r_2$ (*Identity transformation*) ***(no change in the image).***

b) $r_1 = r_2$, $s_1 = 0$, and $s_2 = L-1$ **(***thresholding function)* [*Image **converted** to black* and white i.e. it creates a binary image, as illustrated in Fig. 3.2(b).]

c) Intermediate values of **(r_1, s_1)** and **(r_2, s_2)** produces various degree of spread in the intensity level of the O/P image. Thus affecting its contrast.

© Ramaiah University of Applied Sciences

Cont..

A. Contrast stretching:

- In general, $r_1 \leq r_2$ and $s_1 \leq s_2$, so the function is **single** valued and **monotonically** increasing.
  - This condition preserves the order of gray levels, thus preventing the creation of intensity artifacts in the processed image.
- **Example:** To enhance an 8-bit image with ***low contrast*** using ***contrast stretching,*** and obtained by setting:
  - $(r_1, s_1) = (r_{min}, 0)$ and $(r_2, s_2) = (r_{max}, L-1)$

  where r_{min} and r_{max} denote the *minimum and the maximum intensity level in the* image. [5, 40] => [0, 255]
- So the transformation function stretches the levels linearly from their original range to the full range **[0, L-1].**

© Ramaiah University of Applied Sciences

Contrast Stretching

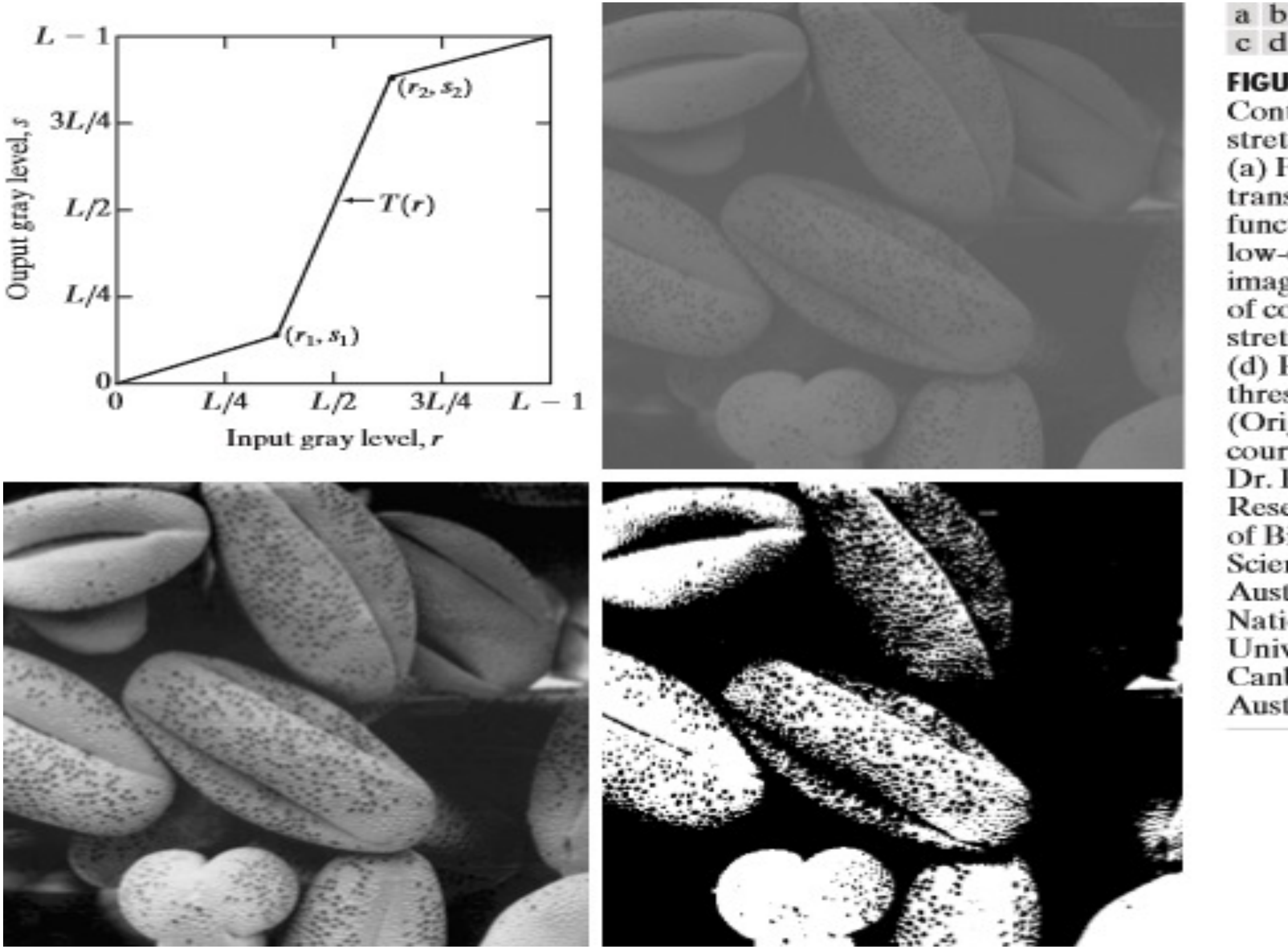


FIGURE 3.10 Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

© Ramaiah University of Applied Sciences

Cont..

- **Example:**
- Figure 3.10(b) shows an 8-bit image with **low contrast**.
- Fig. 3.10(c) shows the result of contrast stretching, obtained by setting
  - $(r_1, s_1) = (r_{min}, 0)$ and $\mathbf{(r_2, s_2) = (r_{max}, L\text{-}1)}$
  - where $\mathbf{r_{min}}$ and $\mathbf{r_{max}}$ denote the *minimum and the maximum intensity level in the* image.
  - So the transformation function stretches the levels linearly from their original range to the full range **[0, L-1].**
- Finally, Fig. 3.10(d) shows the result of using the **thresholding function**, with $\mathbf{r_1 = r_2 = m}$, the mean gray level in the image.
- The original image on which these results are based is a scanning electron microscope image of pollen, magnified approximately 700 times.

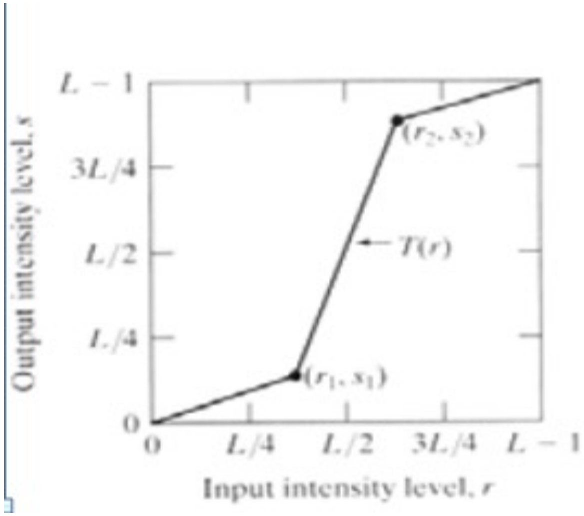
© Ramaiah University of Applied Sciences

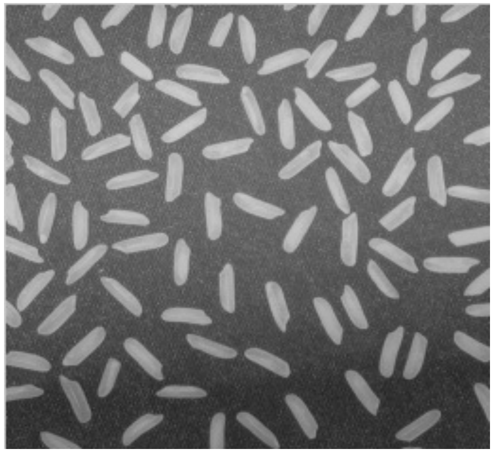
Contrast Stretching

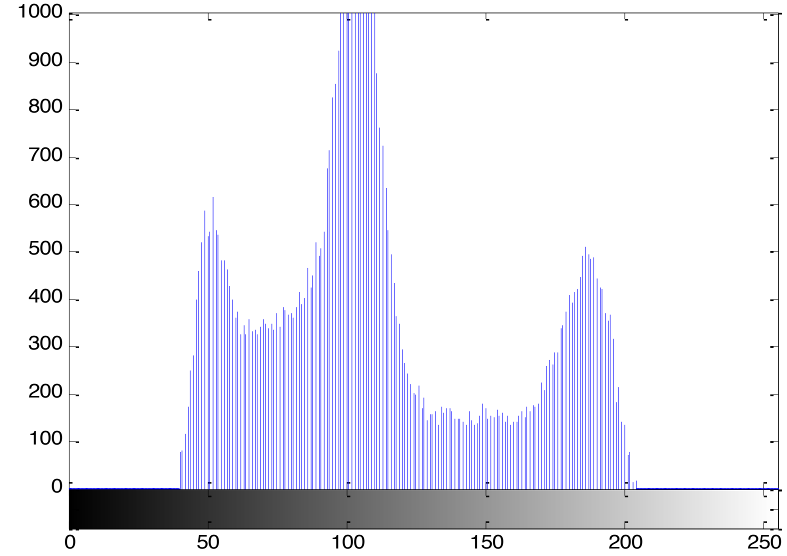
Expands the range of intensity levels in an image to utilize the entire dynamic range of recording media

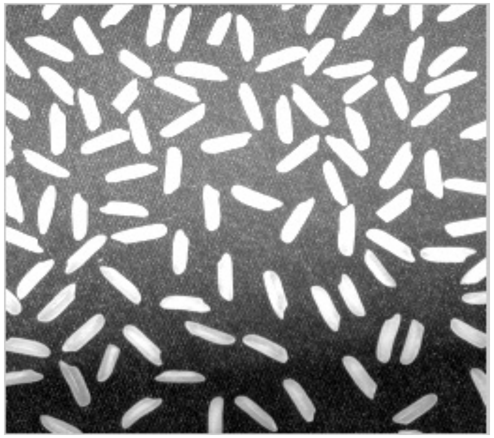
$$(r_1, s_1) = (r_{min}, 0)$$

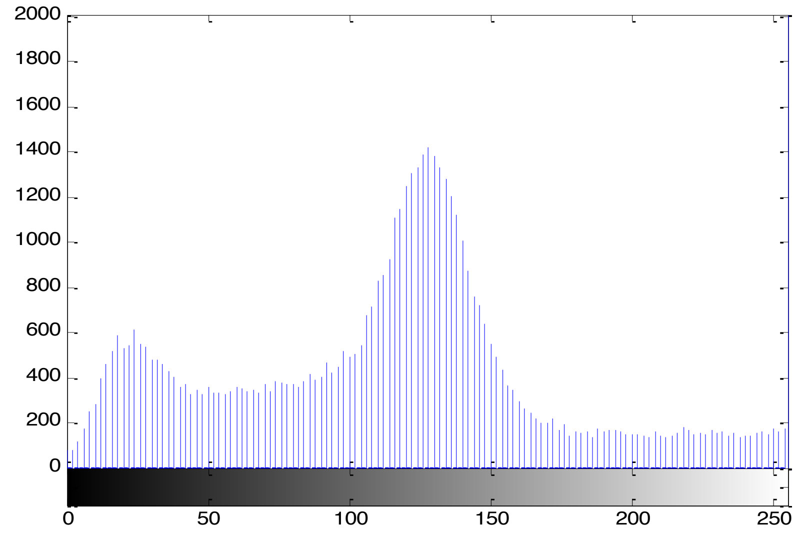
$$(r_2, s_2) = (r_{max}, L\text{-}1)$$











© Ramaiah University of Applied Sciences

Contrast Stretching

$$s = (r - c)\left[\frac{b-a}{d-c}\right] + a$$

Lower and **upper** limits for **a(*s_1*)** and **b(*s_2*)** are **0** and **255**

c (r_{min})and **d(r_{max})** are lower and upper values of input image

r is the input , **s** is the output

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input(M x N)

OUTPUT: Contrast Stretched Gray scale image output(M x N)

STEPS :

1. Read the Input Gray scale Image (M x N)
2. Calculate Contrast Stretching of the input image by applying the equation

$$s = (r - c)\left[\frac{b - a}{d - c}\right] + a$$

Where Lower and upper limits for a and b are 0 and 255; c and d are lower and upper values of input image, r is the input, s is the output

```
I=imread('1.png'); [M,N]=sizeof(I);
for (i=0; i<M; i++)
        for (j=0; j<N; j++)
                output[i][j] =[input[i][j] – c][(b-a)/(d-c)]+a ;
```


© Ramaiah University of Applied Sciences

Cont..

B. **Gray-level slicing (Intensity-level slicing):** This technique is used to *highlight a specific range of gray levels in a given image,* it is called ***intensity-level slicing****, and can be implemented in* several ways.

- **Applications:** enhancing features such as masses of water in satellite imagery and enhancing flaws in X-ray images. There are several ways of doing level slicing, but most of them are variations of two basic themes.

  I. 1st approach is to display a **high value** for all gray levels in the range of interest(ROI) and a **low value** for all other gray levels. This transformation, shown in Fig. 3.11(a), produces a **binary image**.

  II. 2nd approach, **transformation** shown in Fig. 3.11(b), **brightens** the desired range of gray levels but preserves the background and gray-level tonalities in the image.

- Variations of the two transformations shown in Fig. 3.11 are easy to formulate.

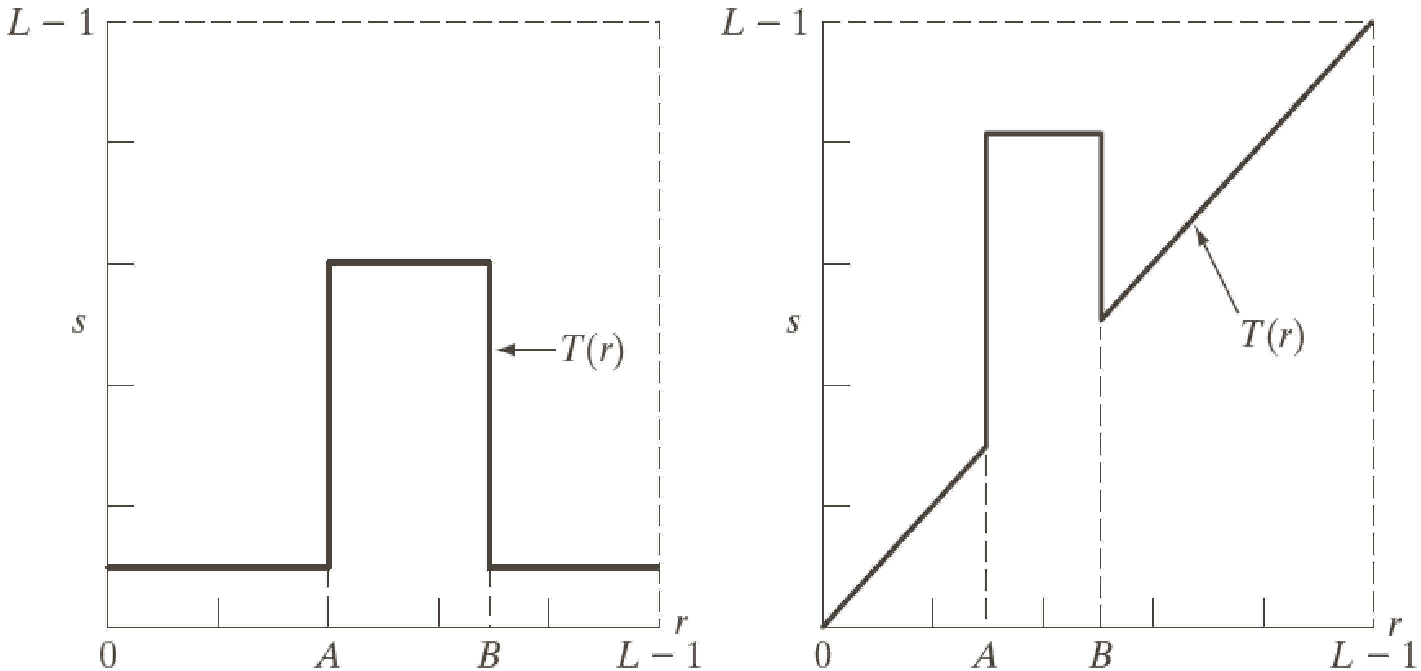
© Ramaiah University of Applied Sciences

Grey-level Slicing

- Highlighting a specific level of intensities (range of gray levels) .
- Used to enhance features in images

a b

FIGURE 3.11 (a) This transformation highlights intensity range [*A*, *B*] and reduces all other intensities to a lower level. (b) This transformation highlights range [*A*, *B*] and preserves all other intensity levels.



Highlights the intensity range [A,B] and reduces all other intensities to a lower level

Highlights the intensity range [A,B] and preserves all other intensity levels

© Ramaiah University of Applied Sciences

Intensity-level slicing

$$s = \begin{cases} 50 & for\ 0 \le r < A \\ 200 & for\ A \le r < B \\ 50 & for\ B \le r \le 255 \end{cases}$$

- Where A & B are the suitable to be identified from the specific input image

$$s = \left\{ \begin{matrix} r & for\ 0 \le r < A \\ 200 + r & for\ A \le r < B \\ r & for\ B \le r \le 255 \end{matrix} \right\}$$

© Ramaiah University of Applied Sciences

Algorithm

INPUT : Gray scale Image input(M x N)

OUTPUT : Intensity-level sliced Gray scale image output(M x N)

STEPS :

1. Read the Input Gray scale Image (M x N)
2. Calculate Intensity-level slicing of the input image by applying the equation

$$s = \begin{cases} 50 & for\ 0 \le r < A \\ 200 & for\ A \le r < B \\ 50 & for\ B \le r \le 255 \end{cases}$$

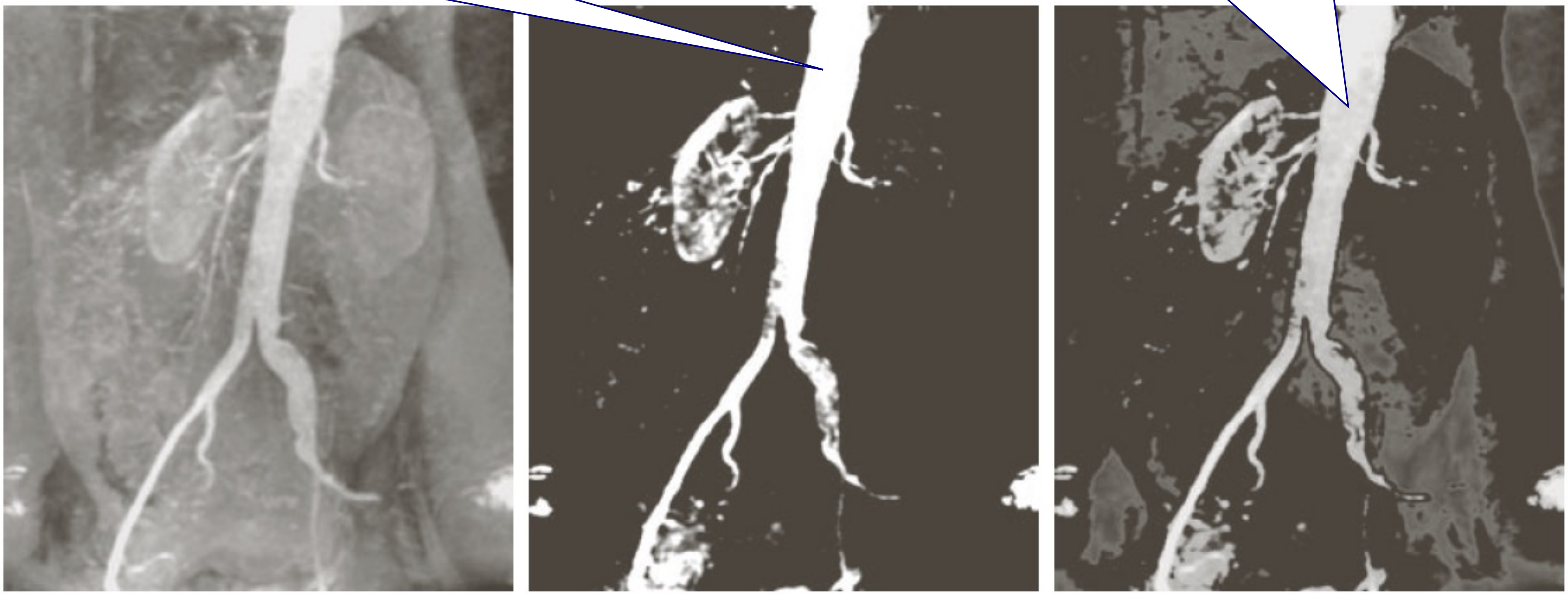
```
for (i=0; i<M; i++)
      for (j=0; j<N; j++)
      {       Apply formula
      }
```


© Ramaiah University of Applied Sciences

Highlight the major blood vessels and study the shape of the flow of the contrast medium (to detect blockages, etc.)

Cont..

Measuring the actual flow of the contrast medium as a function of time in a series of images



a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(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. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

© Ramaiah University of Applied Sciences

Cont..

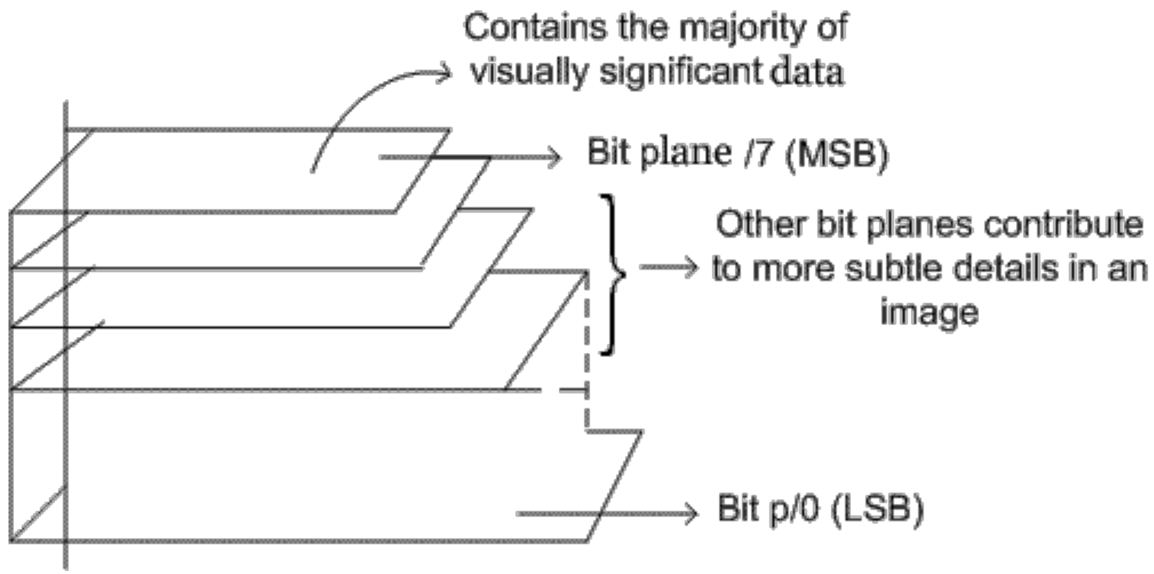
C. Bit- plane slicing: Pixels are digital number composed of bits.

- Instead of highlighting intensity-level range, we could **highlight** the contribution made by each bit, this method is useful and used in image compression.
- Let each pixel in an image is represented by 8 bits Imagine the image is composed of 8, 1-bit planes ranging from bit plane1-0 (LSB)to bit plane 7 (MSB).
- In terms of 8-bits bytes, plane 0 contains all lowest order bits in the bytes comprising the pixels in the image and plane 7 contains all high order bits.
- Figure 3.13 illustrates these ideas, and Fig. 3.14 shows the various bit planes for the image shown in Fig. 3.13.
- **Note**: The higher-order bits (especially the top four) contain the majority of the visually significant data. The other bit planes contribute to more subtle details in the image.

© Ramaiah University of Applied Sciences

Cont..

C. Bit- plane slicing:



- In terms of bit-plane extraction for a 8-bit image, it is seen that binary image for bit plane 7 is obtained by proceeding the I/P image with a thresholding gray-level transformation function that maps all levels between 0 and 127 to one level (e.g 0)and maps all levels from 129 to 253 to another (eg. 255).

© Ramaiah University of Applied Sciences

Bit-plane Slicing

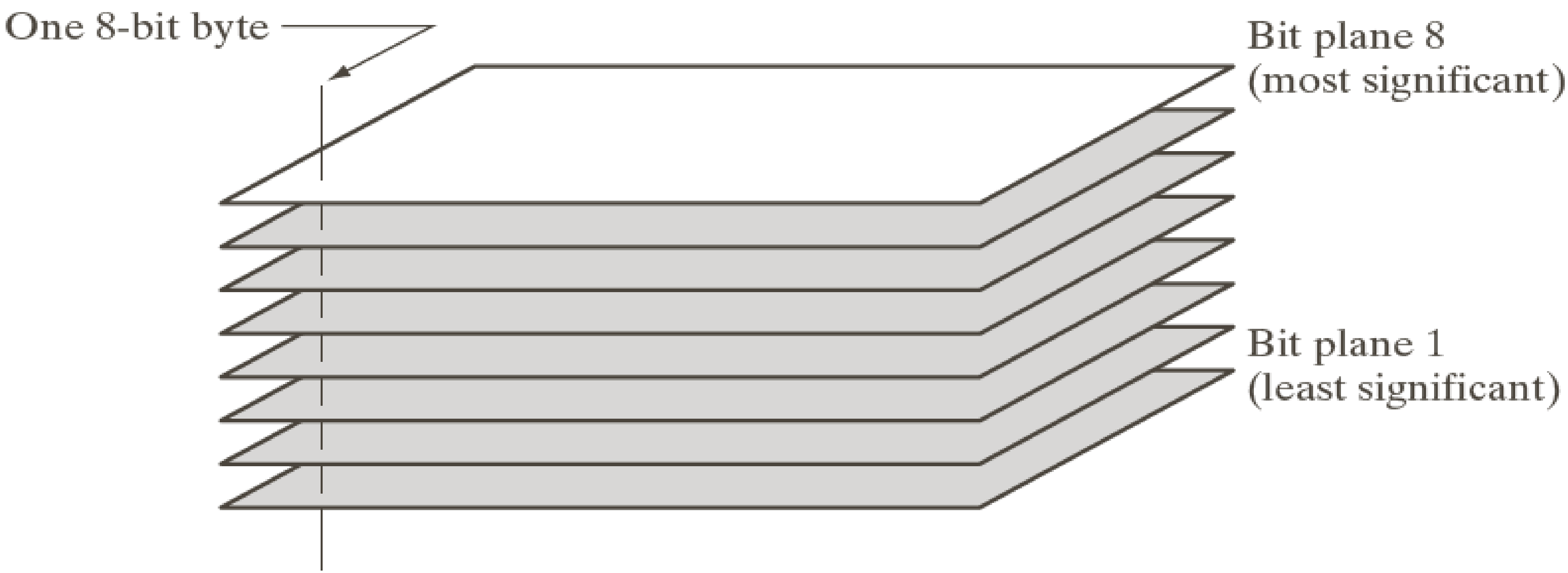


FIGURE 3.13
Bit-plane representation of an 8-bit image.

© Ramaiah University of Applied Sciences

Bit-plane Slicing

a	b	c
d	e	f
g	h	i

FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.

© Ramaiah University of Applied Sciences

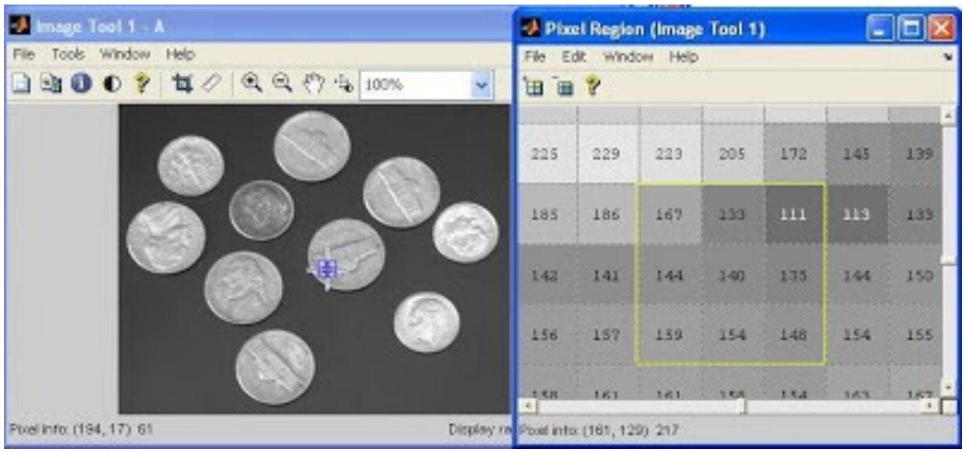
Bit-plane Slicing

FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

© Ramaiah University of Applied Sciences

Cont..

- **Example(Bit- plane slicing:)**
- Digitally, an image is represented in terms of pixels. These pixels can be expressed further in terms of bits.
- Consider the image ‘coins.png’ and the pixel representation of the image.



10100111	10000101	01101111
10010000	10001100	10000111
10011111	10011010	10010100

- Consider the pixels that are bounded within the yellow line. The binary formats for those values are (8-bit representation)

© Ramaiah University of Applied Sciences

Cont..

- The binary format for the pixel value 167 is 10100111
- Similarly, for 144 it is 10010000
- This 8-bit image is composed of eight 1-bit planes.
- Plane 1 contains the lowest order bit of all the pixels in the image.

10100111	10000101	01101111
10010000	10001100	10000111
10011110	10011010	10010100

Fig 1.

10100111	10000101	01101111
10010000	10001100	10000111
10011111	10011010	10010100

Fig 2.

- And plane 8 contains the highest order bit of all the pixels in the image.

© Ramaiah University of Applied Sciences

Cont..

- Let’s see how we can do this using MATLAB
- A=[167 133 111;

  144 140 135;

  159 154 148]
- B = bitget(A,1); %LSB of all pixels ‘bitget’ is a MATLAB function used to fetch a bit from the specified position from all the pixels.
- B=[1 1 1

  0 0 1

  1 0 0]
- B = bitget(A,8); %Highest order bit of all pixels
- B=[1 1 0

  1 1 1

  1 1 1]

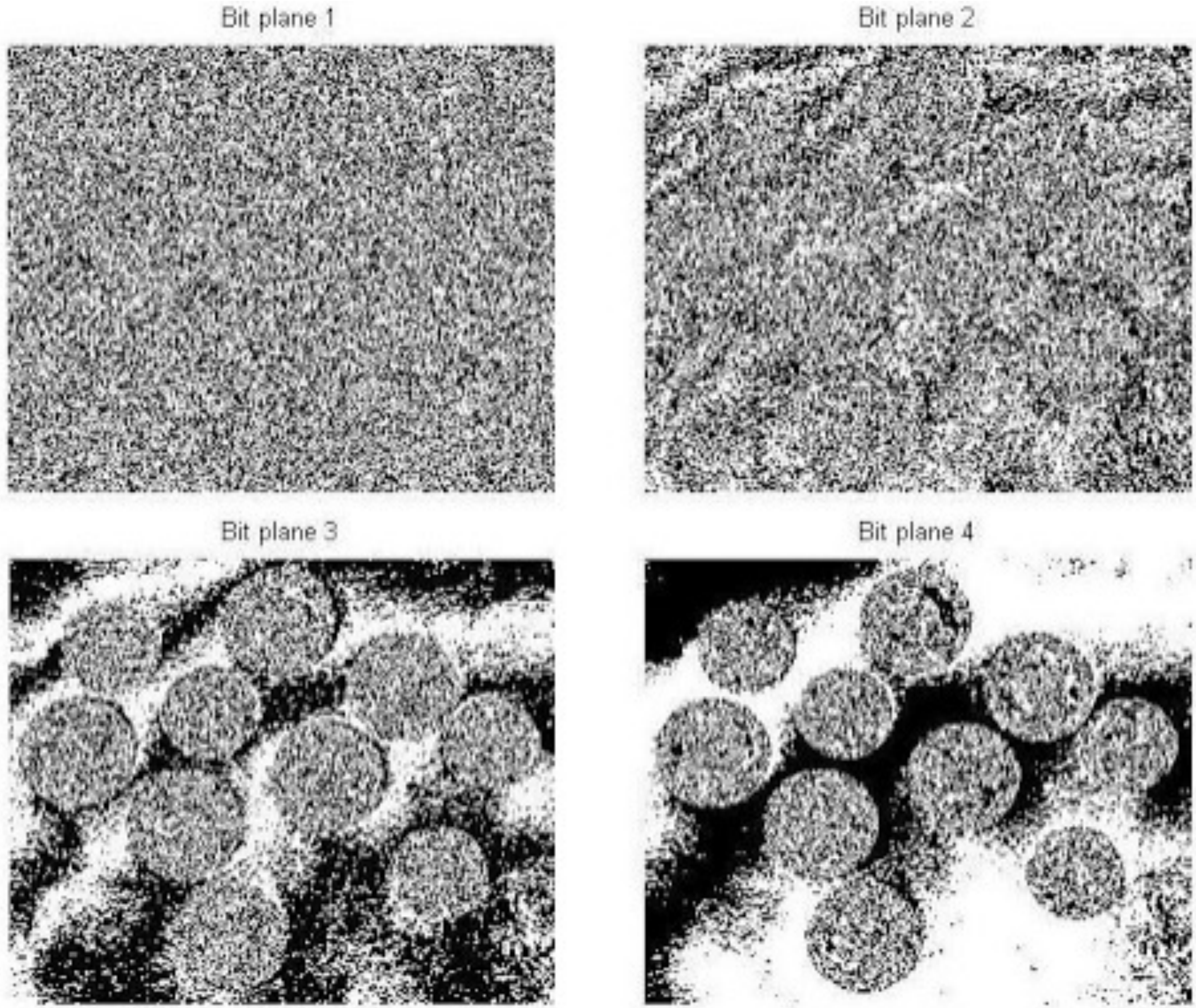
© Ramaiah University of Applied Sciences

Cont..

- **MATLAB CODE:**
- %Bit Planes from 1 to 8. Output Format: Binary
- A = imread('coins.png');
- B = bitget(A,1);
- figure, subplot(2,2,1); imshow(logical(B));title('Bit plane 1');
- B= bitget(A,2);
- subplot(2,2,2); imshow(logical(B));title('Bit plane 2');
- B = bitget(A,3);
- subplot(2,2,3); imshow(logical(B));title('Bit plane 3');
- B = bitget(A,4);
- subplot(2,2,4); imshow(logical(B));title('Bit plane 4');

© Ramaiah University of Applied Sciences

Cont..



© Ramaiah University of Applied Sciences

Cont..

- B = bitget(A,5);
- figure, subplot(2,2,1); imshow(logical(B));title('Bit plane 5');

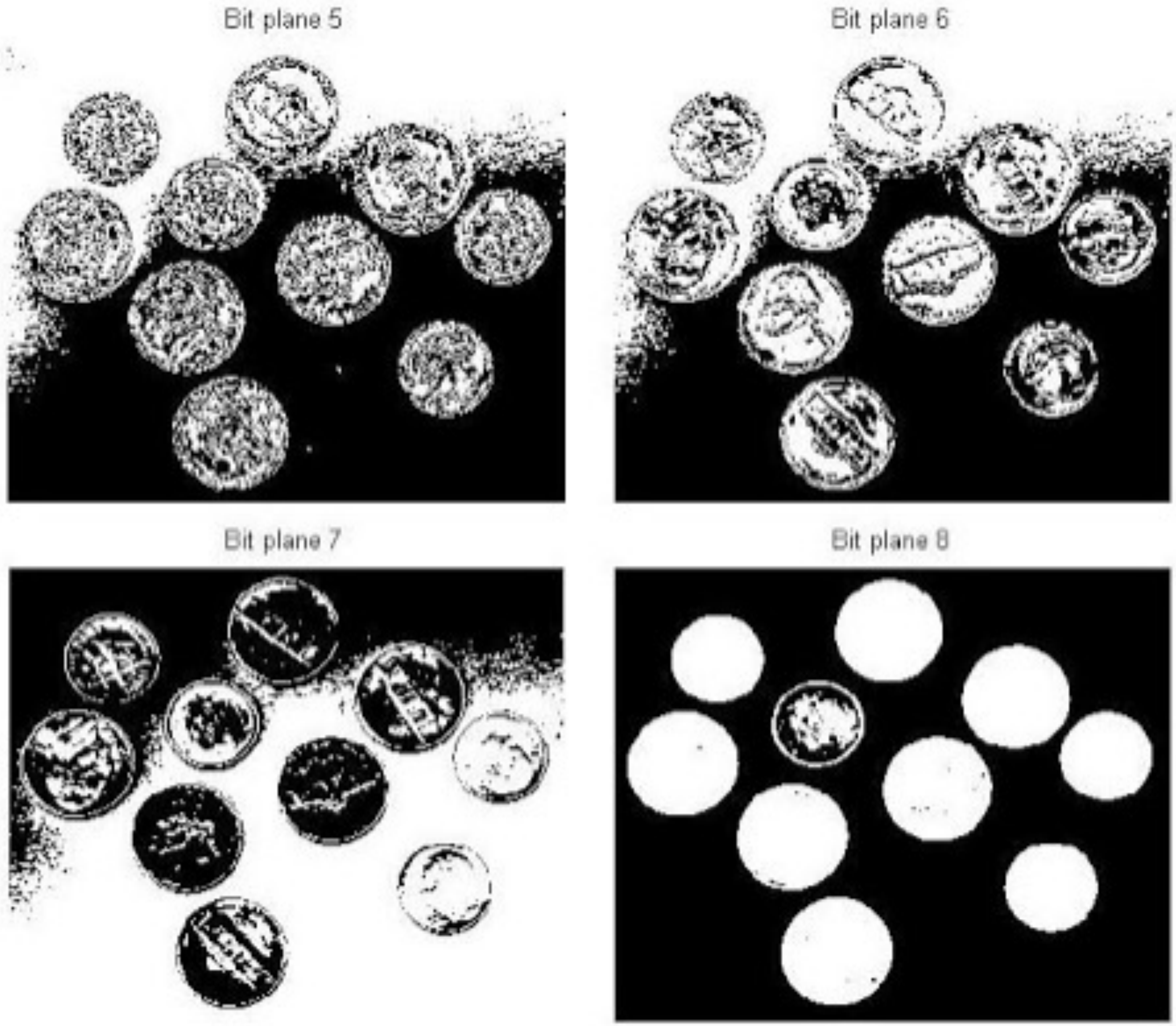
- B = bitget(A,6);
- subplot(2,2,2); imshow(logical(B));title('Bit plane 6');

- B = bitget(A,7);
- subplot(2,2,3); imshow(logical(B));title('Bit plane 7');

- B = bitget(A,8);
- subplot(2,2,4); imshow(logical(B));title('Bit plane 8');

© Ramaiah University of Applied Sciences

Cont..



© Ramaiah University of Applied Sciences

Image reconstruction using n bit planes

1. The nth plane in the pixels are multiplied by the constant $2^{(n-1)}$
2. For instance, consider the matrix
3. A= A=[167 133 111
   144 140 135
   159 154 148] and the respective bit format

10100111	10000101	01101111
10010000	10001100	10000111
10011111	10011010	10010100

© Ramaiah University of Applied Sciences

Cont..

4. Combine the 8 bit plane and 7 bit plane.

   For 10100111, multiply the 8 bit plane with 128 and 7 bit plane with 64.

   (1x128)+(0x64)+(1x0)+(0x0)+(0x0)+(1x0)+(1x0)+(1x0)=128

5. Repeat this process for all the values in the matrix and the final result will be

   [128 128 64

   128 128 128

   128 128 128]

© Ramaiah University of Applied Sciences

Cont..

- **MATLAB CODE:**
- %Image reconstruction by combining 8 bit plane and 7 bit plane
- A=imread('coins.png');
- B=zeros(size(A));
- B=bitset(B,7,bitget(A,7));
- B=bitset(B,8,bitget(A,8));
- B=uint8(B);
- figure, imshow(B);

- **Explanation:**
- ‘bitset’ is used to set a bit at a specified position. Use ‘bitget’ to get the bit at the positions 7 and 8 from all the pixels in matrix A and use ‘bitset’ to set these bit values at the positions 7 and 8 in the matrix B.

© Ramaiah University of Applied Sciences

Cont..

- **MATLAB CODE:**
- %Image reconstruction by combining 8,7,6 and 5 bit planes
- A= imread('coins.png');
- B= zeros(size(A));
- B= bitset(B,8,bitget(A,8));
- B= bitset(B,7,bitget(A,7));
- B= bitset(B,6,bitget(A,6));
- B= bitset(B,5,bitget(A,5));
- B= uint8(B);
- figure, imshow(B);

Image reconstructed using 5,6,7 and 8 bit planes

© Ramaiah University of Applied Sciences

Histogram Processing

- Histogram Equalization
- Histogram Matching
- Local Histogram Processing
- Using Histogram Statistics for Image Enhancement

© Ramaiah University of Applied Sciences

The Histogram of a Grayscale Image

Gray level histogram of an image is the distribution of the gray levels in an image

Examination of the histogram is one of the most useful tools for image enhancement, as it makes easy to see the modifications that may improve an image

The histogram can be modified by a mapping function, which will **stretch**, **shrink** (compress), or **slide** the histogram

Histogram stretching and histogram shrinking are forms of gray scale modification, sometimes referred to as ***histogram scaling***

© Ramaiah University of Applied Sciences

Histogram Processing

- The histogram of a digital image with gray levels in the range [0, L-1] is a discrete function

$$p(r_k) = n_k/n,$$

where,

  - r_k is the k^{th} gray level,
  - n_k is the number of pixels in the image with that gray level,
  - n is the total number of pixels in the image, $k = 0, 1, \ldots, L-1$

© Ramaiah University of Applied Sciences

Example

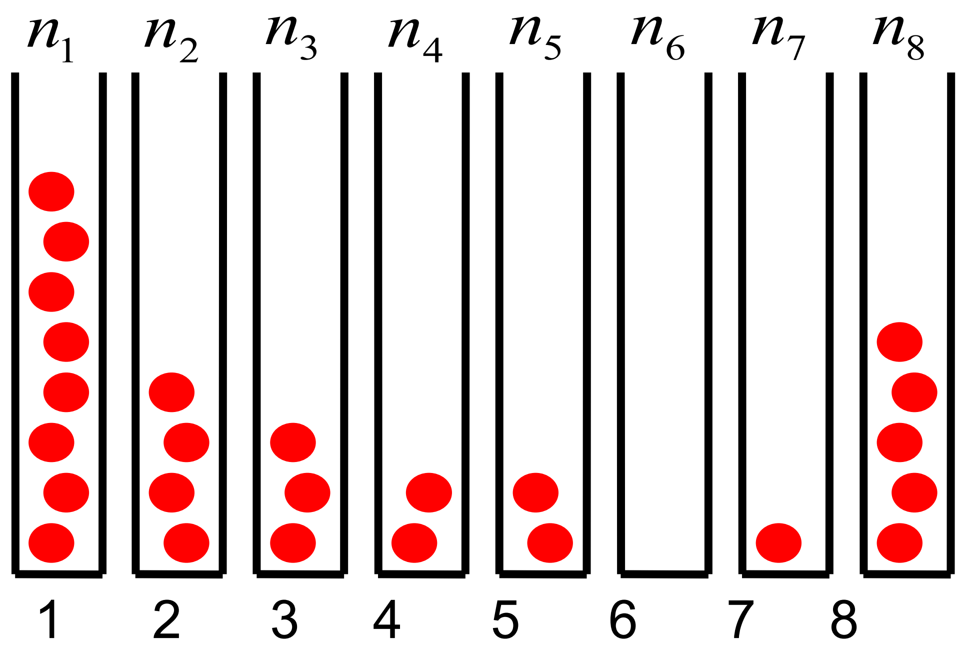
Consider a 5x5 image with integer intensities in the range between one and eight:

1	8	4	3	4
1	1	1	7	8
8	8	3	3	1
2	2	1	5	2
1	1	8	5	2

© Ramaiah University of Applied Sciences

Cont..

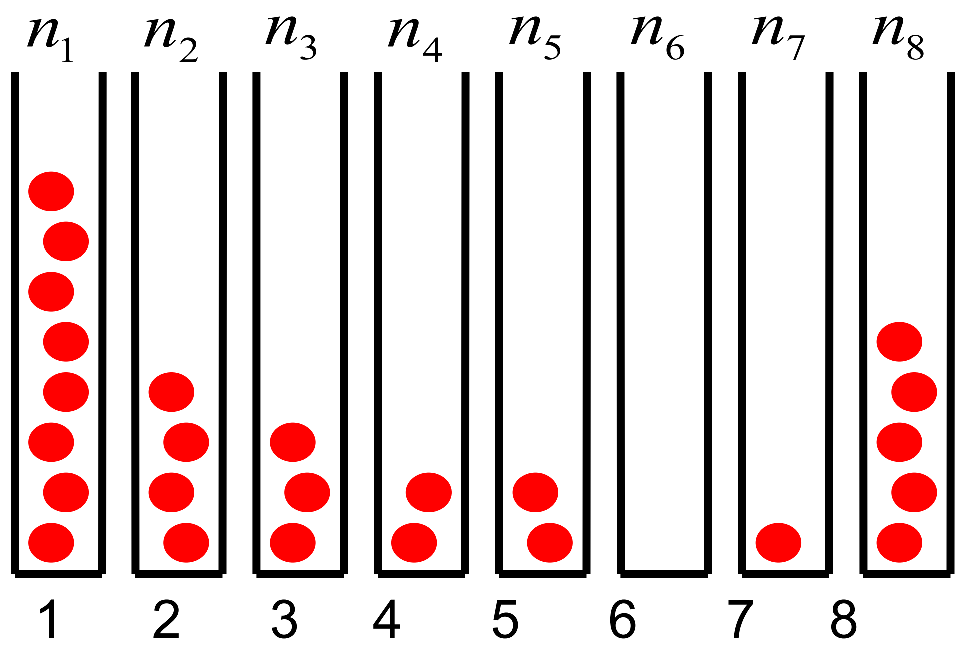
$$h(r_k) = n_k$$



© Ramaiah University of Applied Sciences

Cont..

$$h(r_1) = 8$$
$$h(r_2) = 4$$
$$h(r_3) = 3$$
$$h(r_4) = 3$$
$$h(r_5) = 2$$
$$h(r_6) = 0$$
$$h(r_7) = 1$$
$$h(r_8) = 5$$



© Ramaiah University of Applied Sciences

Normalised Histogram function

The normalised histogram function is the histogram function divided by the total number of the pixels of the image:

$$p(r_k) = \frac{h(r_k)}{n} = \frac{n_k}{n}$$

It gives a measure of how likely is for a pixel to have a certain intensity. That is, it gives the probability of occurrence the intensity.

The sum of the normalised histogram function over the range of all intensities is 1.

© Ramaiah University of Applied Sciences

Normalised Histogram function

$$h(r_1) = 8 \qquad p(r_1) = 8/25 = 0.32$$
$$h(r_2) = 4 \qquad p(r_2) = 4/25 = 0.16$$
$$h(r_3) = 3 \qquad p(r_3) = 3/25 = 0.12$$
$$h(r_4) = 2 \longrightarrow p(r_4) = 3/25 = 0.08$$
$$h(r_5) = 2 \qquad p(r_5) = 2/25 = 0.08$$
$$h(r_6) = 0 \qquad p(r_6) = 0/25 = 0.00$$
$$h(r_7) = 1 \qquad p(r_7) = 1/25 = 0.04$$
$$h(r_8) = 5 \qquad p(r_8) = 5/25 = 0.20$$

© Ramaiah University of Applied Sciences

Histogram Processing

Histogram $h(r_k) = n_k$

r_k is the k^{th} intensity value

n_k is the number of pixels in the image with intensity r_k

Normalized histogram $p(r_k) = \dfrac{n_k}{MN}$

n_k: the number of pixels in the image of size $\mathrm{M} \times \mathrm{N}$ with intensity r_k

© Ramaiah University of Applied Sciences

Histogram Processing

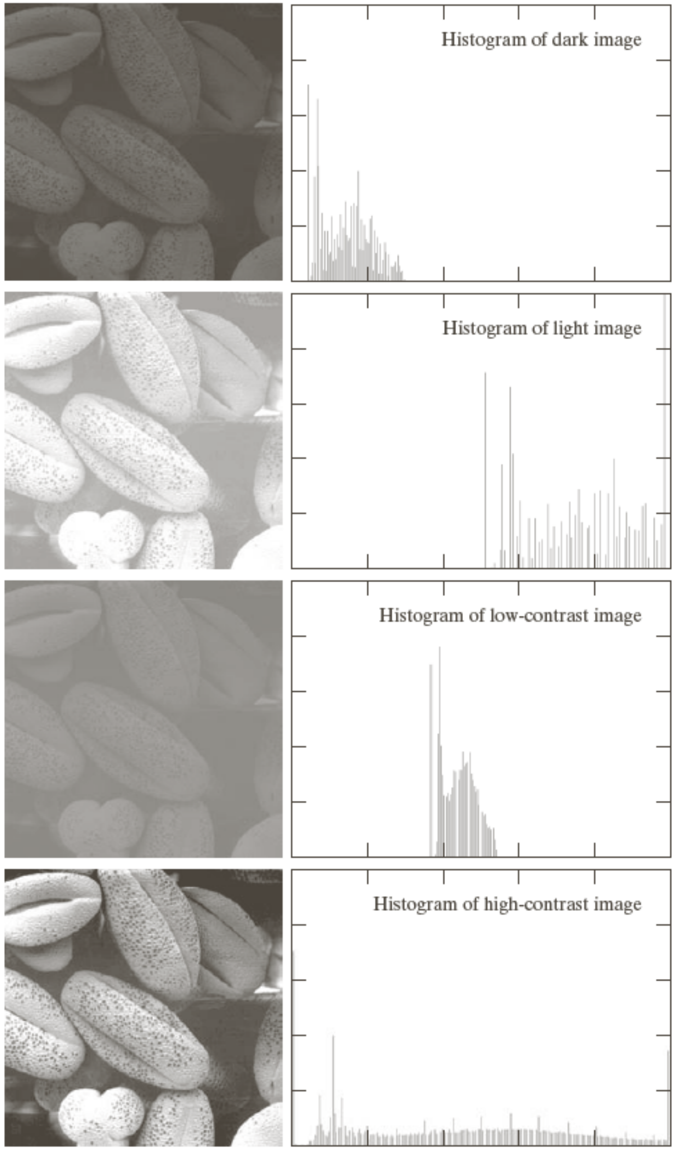


FIGURE 3.16 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms.

© Ramaiah University of Applied Sciences

Histogram Equalization

- The histogram of an image represents the relative frequency of occurrence of the various gray levels in the image.

- It provides a total description of the appearance of an image.

- The type and degree of enhancement obtained depends on the nature of the specified histogram.

- Let the variable **r** represent the grey level of the pixels in the image to be enhanced.

- Assume that the pixel values are normalized to lie in the range $0 <= r <= 1$ with $r = 0$ represents black and $T(r)$ represents white in the gray scale

© Ramaiah University of Applied Sciences

Cont..

- For any r in **[0, 1]**, we consider transformations of the form $s = T(r)$ which produce a level ***S*** for every pixel value **r** in the original image.
- It is assumed that the transformation function satisfies the conditions:
  - The following transformation

    S = T(r) satisfies

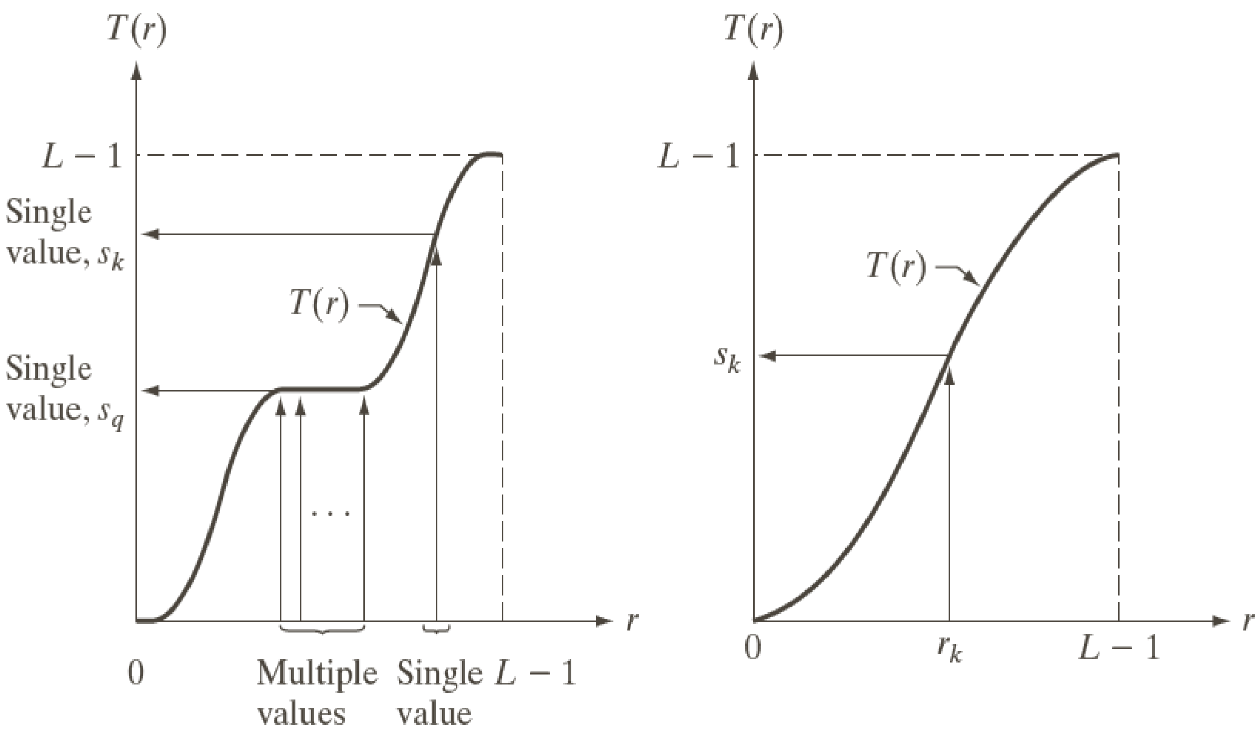
    1. **T(r)** is single-valued and monotonically increasing in the interval **0 <= r <= 1**, and
    2. **0 <= T(r) <= 1** for **0 <= r <= 1**

  - Condition (1) preserve the order from black to white in the grayscale while condition (2) guarantees a mapping that is consistent with the allowed range of the pixel values

© Ramaiah University of Applied Sciences

Cont..

- Gray-level transformation function



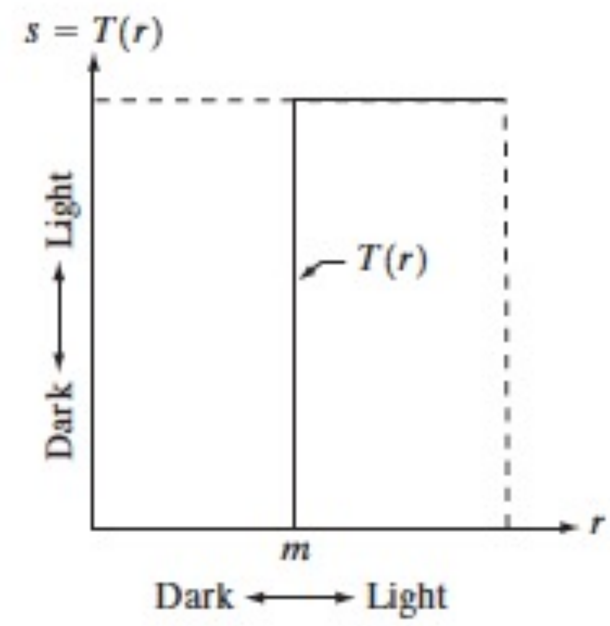
a b

FIGURE 3.17
(a) Monotonically increasing function, showing how multiple values can map to a single value.
(b) Strictly monotonically increasing function. This is a one-to-one mapping, both ways.

© Ramaiah University of Applied Sciences

Cont..

- Example of such a transformation is:



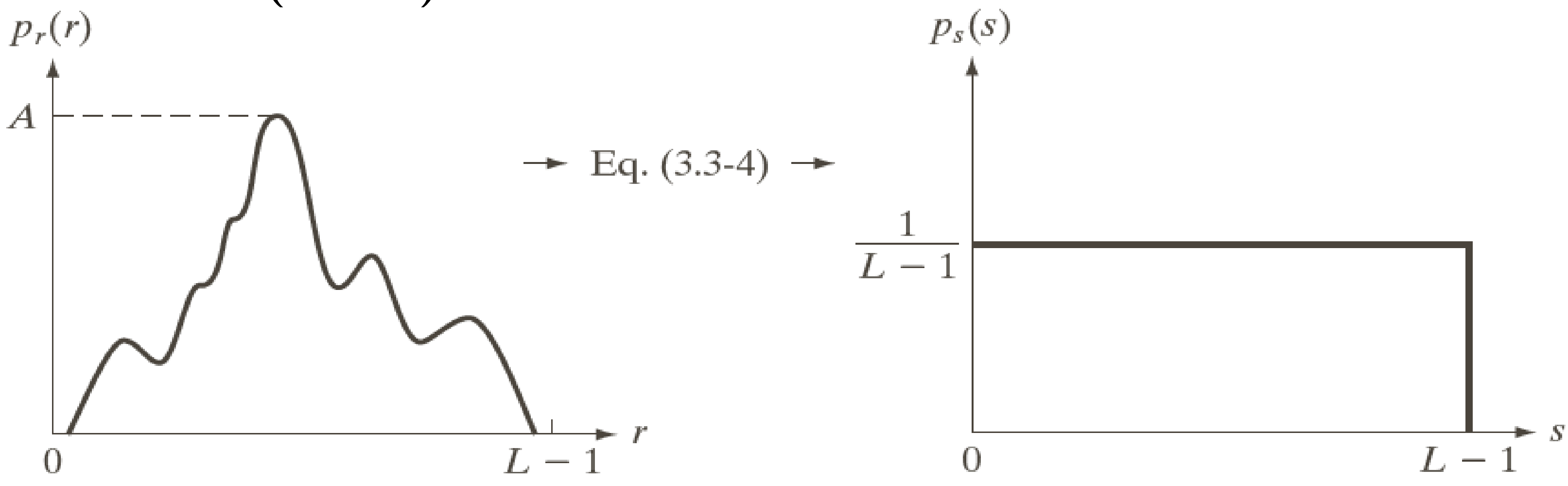
- The inverse transform $r = T^{-1}(s)$ for $0 <= r <= 1$, where it is assumed $T^{-1}(s)$ satisfies conditions (1) or (2) wrt variables. The gray levels in an image are random quantities in the interval [0,1].
- Assuming that they are continuous variables the original and transformed gray levels can be characterized by their probability density functions(PDF) $P_r(r)$ and $P_s(s)$.
- A great deal can be said about the general characteristics of an image from the density functions of its gray levels.

© Ramaiah University of Applied Sciences

Cont..

The intensity levels in an image may be viewed as random variables in the interval [0, L-1].

Let $p_r(r)$ and $p_s(s)$ denote the probability density function (PDF) of random variables r and s.



a b

FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels, r. The resulting intensities, s, have a uniform PDF, independently of the form of the PDF of the r's.

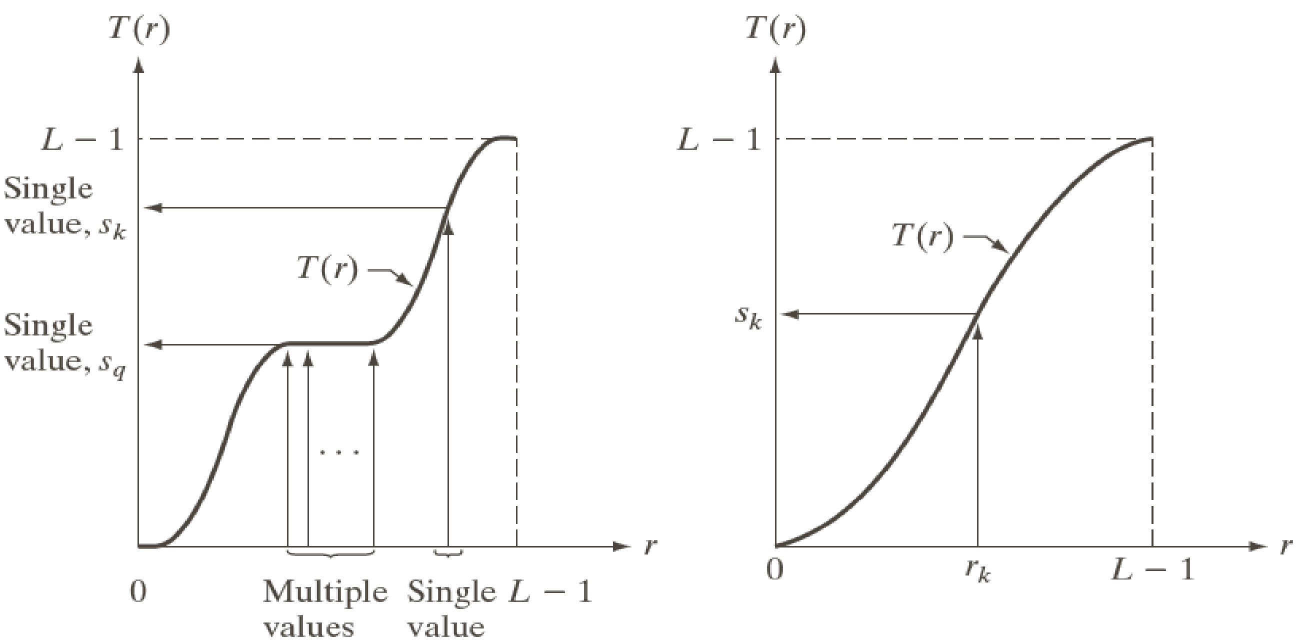
© Ramaiah University of Applied Sciences

Cont..

$$s = T(r) \qquad 0 \le r \le L-1$$

a. T(r) is a strictly monotonically increasing function in the interval $0 \le r \le L-1$;

b. $0 \le T(r) \le L-1$ for $0 \le r \le L-1$.



a b

FIGURE 3.17 (a) Monotonically increasing function, showing how multiple values can map to a single value. (b) Strictly monotonically increasing function. This is a one-to-one mapping, both ways.

© Ramaiah University of Applied Sciences

Histogram Equalization

$$s = T(r) \qquad 0 \le r \le L-1$$

$a.$ T(r) is a strictly monotonically increasing function in the interval $0 \le r \le L\text{-}1$;

$b.$ $0 \le T(r) \le L\text{-}1$ for $0 \le r \le L\text{-}1$.

$T(r)$ is continuous and differentiable.

$$p_s(s)ds = p_r(r)dr$$

© Ramaiah University of Applied Sciences

Histogram Equalization (Understanding)

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = (L-1)\frac{d}{dr}\left[\int_0^r p_r(w)dw\right]$$

$$= (L-1)p_r(r)$$

$$p_s(s) = \frac{p_r(r)dr}{ds} = p_r(r)\Big/\left(\frac{ds}{dr}\right) = p_r(r)\Big/\left((L-1)p_r(r)\right) = \frac{1}{L-1}$$

© Ramaiah University of Applied Sciences

Example

Suppose that the (continuous) intensity values in an image have the PDF

$$p_r(r) = \begin{cases} \dfrac{2r}{(L-1)^2}, & \text{for } 0 \le \text{r} \le \text{L-1} \\ 0, & \text{otherwise} \end{cases}$$

Find the transformation function for equalizing the image histogram.

© Ramaiah University of Applied Sciences

Example

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$

© Ramaiah University of Applied Sciences

Histogram Equalization

Continuous case:

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

Discrete values:

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$= (L-1)\sum_{j=0}^{k} \frac{n_j}{MN} = \frac{L-1}{MN}\sum_{j=0}^{k} n_j \qquad \text{k=0,1,..., L-1}$$

© Ramaiah University of Applied Sciences

Histogram Equalization

The formula for histogram equalisation in the discrete case is given

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j$$

where

- r_k: input intensity
- s_k: processed intensity
- n_j: the frequency of intensity j
- MN: the number of image pixels.

© Ramaiah University of Applied Sciences

Example 1: Histogram Equalization

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$p(r_1) = 8/25 = 0.32$$

$$p(r_2) = 4/25 = 0.16$$

$$p(r_3) = 3/25 = 0.12$$

$$p(r_4) = 3/25 = 0.08$$

$$p(r_5) = 2/25 = 0.08$$

$$p(r_6) = 0/25 = 0.00$$

$$p(r_7) = 1/25 = 0.04$$

$$p(r_8) = 5/25 = 0.20$$

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j)$$

= 7 x0.32=2.24=2

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j)$$

= 7 x(0.32+0.16)=7x0.48=3.36=3

$$s_2 = T(r_2) = 7\sum_{j=0}^{2} p_r(r_j)$$

= 7 x(0.32+0.16+0.12)=7x0.6=4.2=4

© Ramaiah University of Applied Sciences

Example 1: Histogram Equalization (Cont..)

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	New n_k
1	8	0.32	0.32	2.24	2
2	4	0.16	0.48	3.36	3
3	3	0.12	0.6	4.2	4
4	3	0.08	0.68	4.76	5
5	2	0.08	0.76	5.32	5
6	0	0.00	0.76	5.32	5
7	1	0.04	0.8	5.6	6
8	5	0.20	1	7.00	7

Observe that there are 6 distinct intensity levels. r_1 is mapped to s_1=2 and No of pixels are 8.

Equating Gray Levels to No. of Pixels: 1 -> 0; 8 -> 0;

s_2 = 3 -> 4; s_3 = 4 -> 3; r_4&r_5&r_6= s_4 = 5 -> (3+2)5 s_5 = 6 -> 1

s_6 = 7 -> 5; Total : 25

© Ramaiah University of Applied Sciences

Example 2: Histogram Equalization (Cont..)

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	New n_k
0	2	0.11	0.11	0.77	1
1	4	0.11	0.22	1.54	2
2	6	0.33	0.55	3.85	4
3	8	0.11	0.66	4.62	5
4	10	0.11	0.77	5.39	5
5	12	0.11	0.88	6.16	6
6	14	0.11	0.99	6.93	7
7	16	0.01	1	7.00	7

Observe that there are 6 distinct intensity levels. r_0 is mapped to s_0=1 and No of pixels are 2.

Equating Gray Levels to No. of Pixels: 0 -> 0; 3 -> 0;

s_1=2-> 4; s_2 = 4 -> 6; r_3&r_4=s_3 = 5 -> 18; s_4 = 6 -> 12

r_6&r_7=s_5 = 7 -> (14+16)=30 Total : 70

© Ramaiah University of Applied Sciences

Example 3

Perform histogram equalization on the following 3x3, 8-level image :-

1	3	5
4	4	3
5	2	2

Gray Level (r_k)	0	1	2	3	4	5	6	7
No of Pixels (n_k)	0	1	2	2	2	2	0	0

© Ramaiah University of Applied Sciences

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	New n_k
0	0	0	0	0	0
1	1	1/9	1/9	7/9=0.777	1
2	2	2/9	3/9 = 1/3	7/3=2.333	2
3	2	2/9	5/9	35/9=3.888	4
4	2	2/9	7/9	49/9=5.444	5
5	2	2/9	1	7	7
6	0	0	1	7	7
7	0	0	1	7	7

Observe that there are 5 distinct intensity levels. r_1 is mapped to s_1=1 and No of pixels are 1. Last column New n_k shows the final distribution of gray levels pixels, where pixels 2 & 0 corresponds to levels 5 & 6 are map to level 7 in O/P image.

Equating Gray Levels to No. of Pixels: 0 -> 0; 3 -> 0; 6->0

s_2=3-> 2; s_3 = 4 -> 2; s_4 = 5 -> 2; r_5&r_6&r_7 =s_5 = 6 -> (2+0+0)=2

Total : 9

© Ramaiah University of Applied Sciences

Example 3

Image after equalization is as follows :-

1	3	5
4	4	3
5	2	2

old

1	4	7
5	5	4
7	2	2

new

© Ramaiah University of Applied Sciences

Example 4

Consider a 8x8 image with integer intensities in the range between one and eight:

0	1	7	4	3	4	4	5
0	1	1	1	7	5	4	6
0	5	5	1	5	1	4	6
0	2	2	1	5	2	4	6
0	1	1	5	5	2	4	6
0	5	5	4	3	5	4	1
0	2	2	5	5	2	4	2
0	5	5	4	5	2	4	2

Gray Level (r_k)	0	1	2	3	4	5	6	7
No of Pixels (n_k)	8	10	10	2	12	16	4	2

© Ramaiah University of Applied Sciences

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	New n_k
0	8	8/64=0.125	0.125	0.125x7 = 0.875	1
1	10	10/64=0.156	0.281	1.967	2
2	10	10/64=0.156	0.437	3.059	3
3	2	2/64=0.03	0.467	3.269	3
4	12	12/64=0.187	0.654	4.578	5
5	16	16/64=0.25	0.904	6.328	6
6	4	4/64=0.0625	0.97	6.79	7
7	2	2/64=0.03	1	7.00	7

Observe that there are 6 distinct intensity levels. r_0 is mapped to s_0=1 and No of pixels are 8. Last column New n_k shows the final distribution of gray levels pixels, where pixels 10 & 2 corresponds to levels 2&3 are map to level 3 in O/P image.

Equating Gray Levels to No. of Pixels: 0 -> 0; 4 -> 0;

s_1=2-> 10; r_2&r_3=s_2 = 3 -> 12; s_3 = 5 -> 12; s_4 = 6 -> 16

r_6&r_7=s_5 = 7 -> (4+2)=6 Total : 64

© Ramaiah University of Applied Sciences

Example 5: Histogram Equalization

Suppose that a 3-bit image ($L=2^3=8$) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in following table.
Get the histogram equalization transformation function and give the $p_s(s_k)$ for each s_k.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

© Ramaiah University of Applied Sciences

Example 5: Histogram Equalization (Cont..)

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7 \times 0.19 = 1.33 \rightarrow 1$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7 \times (0.19 + 0.25) = 3.08 \rightarrow 3$$

$s_2 = 4.55 \rightarrow 5$ $\quad$ $s_3 = 5.67 \rightarrow 6$

$s_4 = 6.23 \rightarrow 6$ $\quad$ $s_5 = 6.65 \rightarrow 7$

$s_6 = 6.86 \rightarrow 7$ $\quad$ $s_7 = 7.00 \rightarrow 7$

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

© Ramaiah University of Applied Sciences

Cont..

r_k	n_k	$P_r(r_k) = n_k/MN$	$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$
0	790	0.19	1.33 = 1
1	1023	0.25	3.08 = 3
2	850	0.21	4.55 = 5
3	656	0.16	5.67 = 6
4	329	0.08	6.23 = 6
5	245	0.06	6.65 = 7
6	122	0.03	6.86 = 7
7	81	0.02	7.00 = 7

© Ramaiah University of Applied Sciences

Cont..

Gray Levels (r_k)	No. of Pixels n_k	(PDF) Pr(rk) = nk/n	(CDF) Sk =∑ Pk(rk)	(L-1) Sk = 7 x Sk	Rounding off
0	790	0.19	0.19	1.33	1
1	1023	0.25	0.44	3.08	3
2	830	0.21	0.65	4.55	5
3	656	0.16	0.81	5.67	6
4	329	0.08	0.89	6.23	6
5	245	0.06	0.95	6.65	7
6	122	0.03	0.98	6.86	7
7	81	0.02	1	7	7
	n = 4096	1			

These are the values for equalized Histogram. Observe that there are 5 distinct intensity levels. r_0 is mapped to s_0=1 and No of pixels are 790. Equating Gray Levels to No. of Pixels: 0 -> 0; 2 -> 0; 4 -> 0;

s_0 = 1 -> 790; s_1 = 3 -> 1023; s_2 = 5 -> 830 r_3&r_4 = 6 -> (656+329) = 985

r_5& r_6 & r_7=7 -> (245+122+81)=448; Total : 4096 Hence Verified!!

© Ramaiah University of Applied Sciences

Example: Histogram Equalization

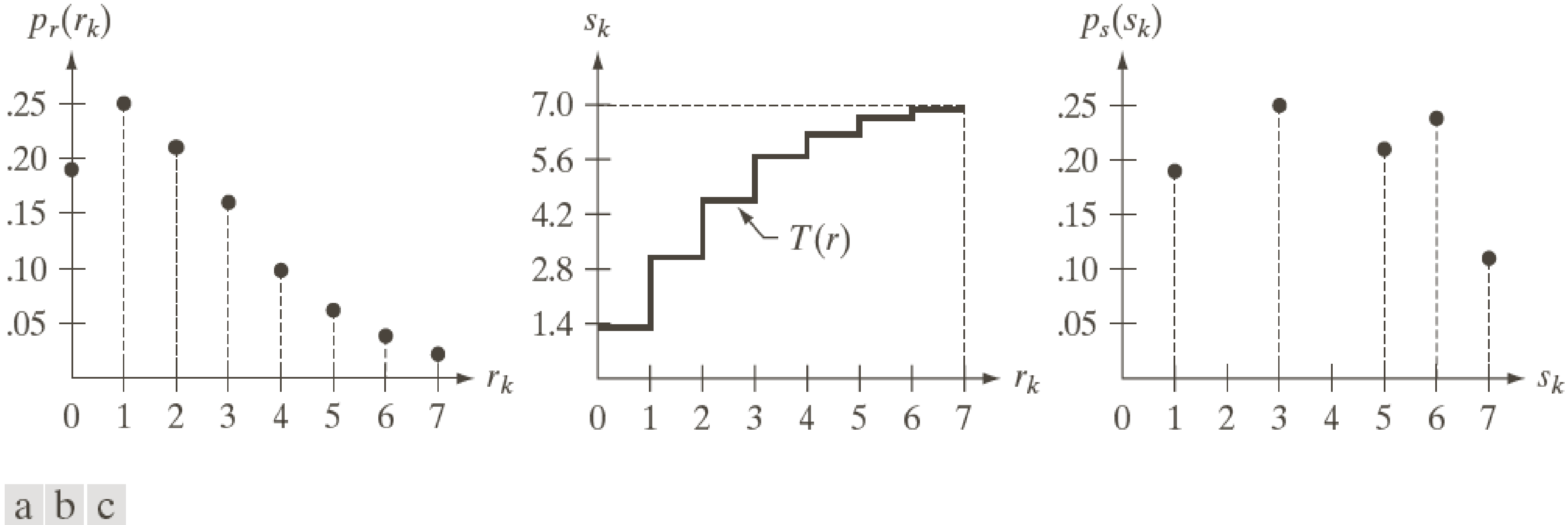


FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

These are the values for equalized Histogram. Observe that there are 5 distinct intensity levels. r_0 is mapped to s_0=1 and No of pixels are 790. Equating Gray Levels to No. of Pixels: 0 -> 0; 4 -> 0; 2 -> 0;

S_0 = 1 -> 790; s_1 = 3 -> 1023; s_2 = 5 -> 850 r_3&r_4 = 6 -> (656+329) = 985

r_5& r_6 & r_7=7 -> (245+122+81)=448; Total : 4096

Now dividing these numbers by MN=4096 gives the equalized Hist. in fig 3.19(c).

© Ramaiah University of Applied Sciences

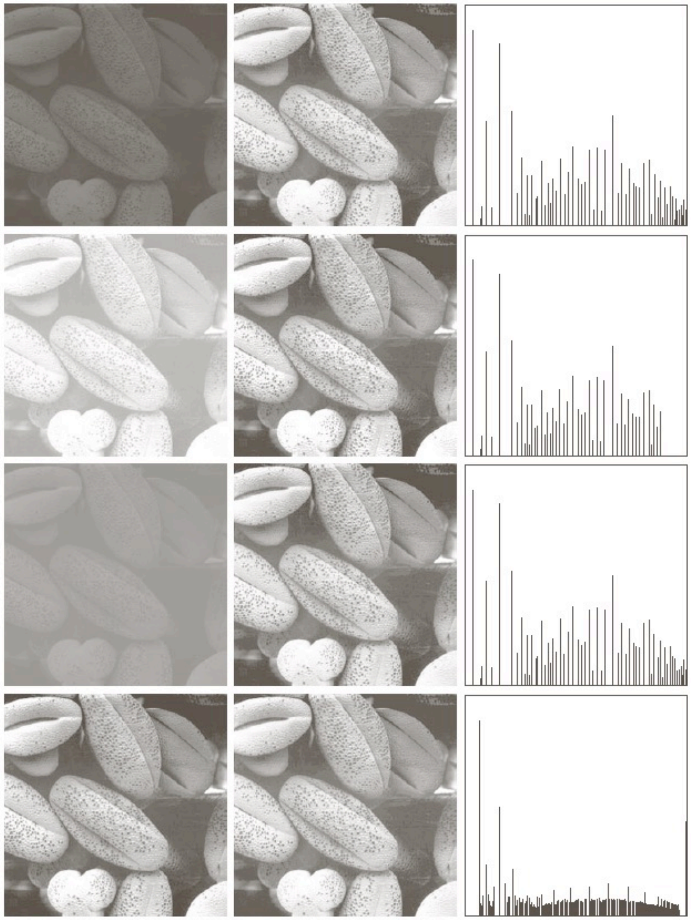
Cont..

- It is clearly seen that
  - Histogram equalization distributes the gray level to reach the maximum gray level (white) because the cumulative distribution function equals 1 when $0 \leq r \leq L-1$
  - If the cumulative numbers of gray levels are slightly different, they will be mapped to little different or same gray levels as we may have to approximate the processed gray level of the output image to integer number
  - Thus, the discrete transformation function can't guarantee the one to one mapping relationship
  - Notice that due to discretization, the resulting histogram will rarely be perfectly flat. However, it will be extended.

© Ramaiah University of Applied Sciences

Histogram Equalization

- Example



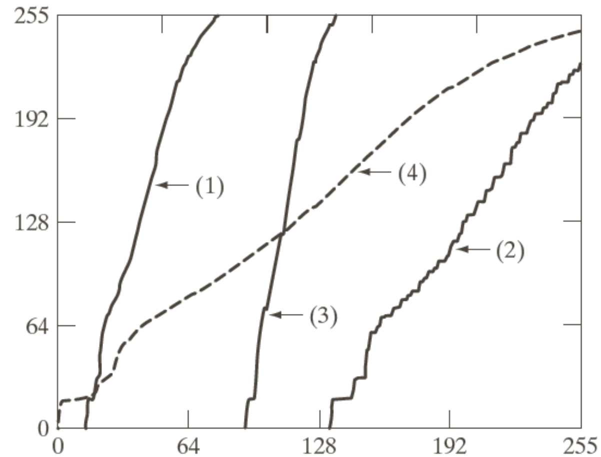


FIGURE 3.21 Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).

FIGURE 3.20 Left column: images from Fig. 3.16. Center column: corresponding histogram-equalized images. Right column: histograms of the images in the center column.

© Ramaiah University of Applied Sciences

Difference

- The goal of HE is to produce an output image that has a **flattened histogram.**
- HM is to take an I/P image and generate an O/P image that is based upon the **shape** say $p_z(z)$ of a specific (or **reference**) **histogram**.
- In HE (also known as histogram flattening), the goal is to **improve contrast** in images that might be either **blurry** or have a **background** and **foreground** that are either/ both **bright** or both **dark**. **Low contrast** images typically have histograms that are concentrated within a tight range of values.
- HE can improve the contrast in these images by spreading out the histogram so that the intensity values are **distributed uniformly** over a larger intensity range. Ideally, the histogram of the output image will be **perfectly flat**.

© Ramaiah University of Applied Sciences

Histogram Specification/Matching

- Equalize the levels of the original image.
- Histogram matching is the transformation of an image.
- The process of Histogram Matching takes in an input image and produces an output image that is based upon a specified histogram.
- The well-known histogram equalization method is a special case in which the specified histogram is uniformly distributed.

© Ramaiah University of Applied Sciences

Histogram Specification or Matching

- Sometimes, this may not be desirable.
- Instead, we may want a transformation that yields an output image with a pre-specified histogram.
- Applying the transformation H to the original image yields an image with histogram.
- Again, the actual histogram of the output image does not exactly but only approximately matches with the specified histogram.
- This is because we are dealing with discrete histograms.

© Ramaiah University of Applied Sciences

Histogram Matching

Histogram matching (histogram specification)

— generate a processed image that has a specified histogram

Let $p_r(r)$ and $p_z(z)$ denote the continous probability density functions of the variables r and z. $p_z(z)$ is the specified probability density function.

Let s be the random variable with the probability

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

Define a random variable z with the probability

$$G(z) = (L-1)\int_0^z p_z(t)dt = s$$

© Ramaiah University of Applied Sciences

Histogram Matching

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$

$$G(z) = (L-1)\int_0^z p_z(t)\,dt = s$$

$$z = G^{-1}(s) = G^{-1}\left[T(r)\right]$$

© Ramaiah University of Applied Sciences

Histogram Matching: Procedure

- Obtain $p_r(r)$ from the input image and then obtain the values of s

$$s = (L-1)\int_0^r p_r(w)dw$$

- Use the specified PDF and obtain the transformation function G(z)

$$G(z) = (L-1)\int_0^z p_z(t)dt = s$$

- Mapping from s to z

$$z = G^{-1}(s)$$

© Ramaiah University of Applied Sciences

Histogram Matching: Example

Assuming continuous intensity values, suppose that an image has the intensity PDF

$$p_r(r) = \begin{cases} \dfrac{2r}{(L-1)^2}, & \text{for } 0 \le r \le L\text{-}1 \\ 0\ , & \text{otherwise} \end{cases}$$

Find the transformation function that will produce an image whose intensity PDF is

$$p_z(z) = \begin{cases} \dfrac{3z^2}{(L-1)^3}, & \text{for } 0 \le z \le (L\text{-}1) \\ 0, & \text{otherwise} \end{cases}$$

© Ramaiah University of Applied Sciences

Histogram Matching: Example

Find the histogram equalization transformation for the input image

$$s = T(r) = (L-1)\int_0^r p_r(w)dw = (L-1)\int_0^r \frac{2w}{(L-1)^2}dw = \frac{r^2}{L-1}$$

Find the histogram equalization transformation for the specified histogram

$$G(z) = (L-1)\int_0^z p_z(t)dt = (L-1)\int_0^z \frac{3t^2}{(L-1)^3}dt = \frac{z^3}{(L-1)^2} = s$$

The transformation function

$$z = \left[(L-1)^2 s\right]^{1/3} = \left[(L-1)^2 \frac{r^2}{L-1}\right]^{1/3} = \left[(L-1)r^2\right]^{1/3}$$

© Ramaiah University of Applied Sciences

Histogram Matching: Discrete Cases

- Obtain $p_r(r_j)$ from the input image and then obtain the values of s_k, round the value to the integer range [0, L-1].

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j$$

- Use the specified PDF and obtain the transformation function $G(z_q)$, round the value to the integer range [0, L-1].

$$G(z_q) = (L-1)\sum_{i=0}^{q} p_z(z_i) = s_k$$

- Mapping from s_k to z_q $\quad z_q = G^{-1}(s_k)$

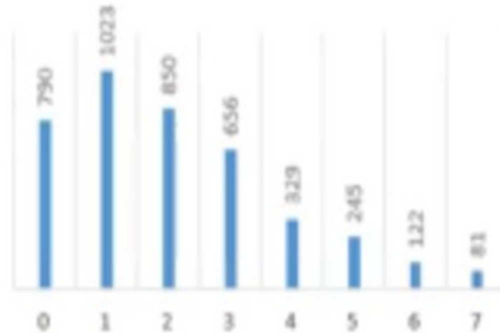
© Ramaiah University of Applied Sciences

Example 1: Histogram Matching

Given histogram (a) & (b), modify histogram (a) as given by histogram (b)

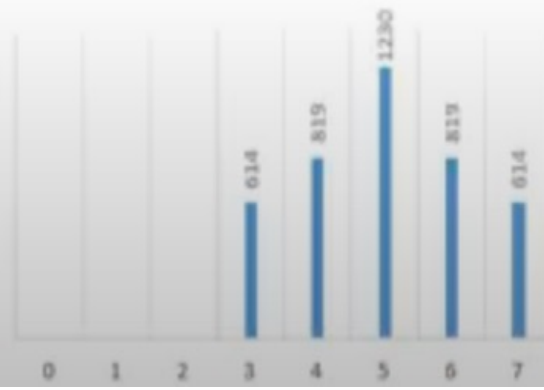
(a)

Gray level.	0	1	2	3	4	5	6	7
No. of pixels	790	1023	850	656	329	245	122	81



(b)

Gray level.	0	1	2	3	4	5	6	7
No. of pixels	0	0	0	614	819	1230	819	614



© Ramaiah University of Applied Sciences

Cont..

Equalize histogram (a)

Gray level	nk	PDF	CDF	Sk x 7	Round off		New nk.
0	790	0.19	0.19	1.33	1		790
1	1023	0.25	0.44	3.08	3		1023
2	850	0.21	0.65	4.55	5		850
3	656	0.16	0.81	5.67	6	656+329	985
4	329	0.08	0.89	6.23	6		
5	245	0.06	0.95	6.65	7	245 +122+81	448
6	122	0.03	0.98	6.86	7		
7	81	0.02	1	7	7		
	N=4096						

© Ramaiah University of Applied Sciences

Cont..

Now Equalize histogram (b)

Gray level	nk	PDF	CDF	Sk x 7	Round off
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	614	0.149	0.149	1.05	1
4	819	0.20	0.35	2.50	3
5	1230	0.30	0.65	4.55	5
6	819	0.20	0.85	5.97	6
7	614	0.15	1	7	7
	N=4096				

© Ramaiah University of Applied Sciences

Cont..

Mapping

First and last columns of histogram (b)

Gray level	Round off
0	0
1	0
2	0
3	1
4	3
5	5
6	6
7	7

Last two columns of histogram (a)

Round off	New nk.
1	790
3	1023
5	850
6	
6	985
7	
7	448
7	

Gray Level	0	1	2	3	4	5	6	7
No of Pixels								

© Ramaiah University of Applied Sciences

Cont..

Mapping

First and last columns of histogram (b)

Gray level	Round off
0	0
1	0
2	0
3	1
4	3
5	5
6	6
7	7

Last two columns of histogram (a)

Round off	New nk.
1	790
3	1023
5	850
6	
6	985
7	
7	448
7	

Gray Level	0	1	2	3	4	5	6	7
No of Pixels				790				

© Ramaiah University of Applied Sciences

Cont..

Mapping

First and last columns of histogram (b)

Gray level	Round off
0	0
1	0
2	0
3	1
4	3
5	5
6	6
7	7

Last two columns of histogram (a)

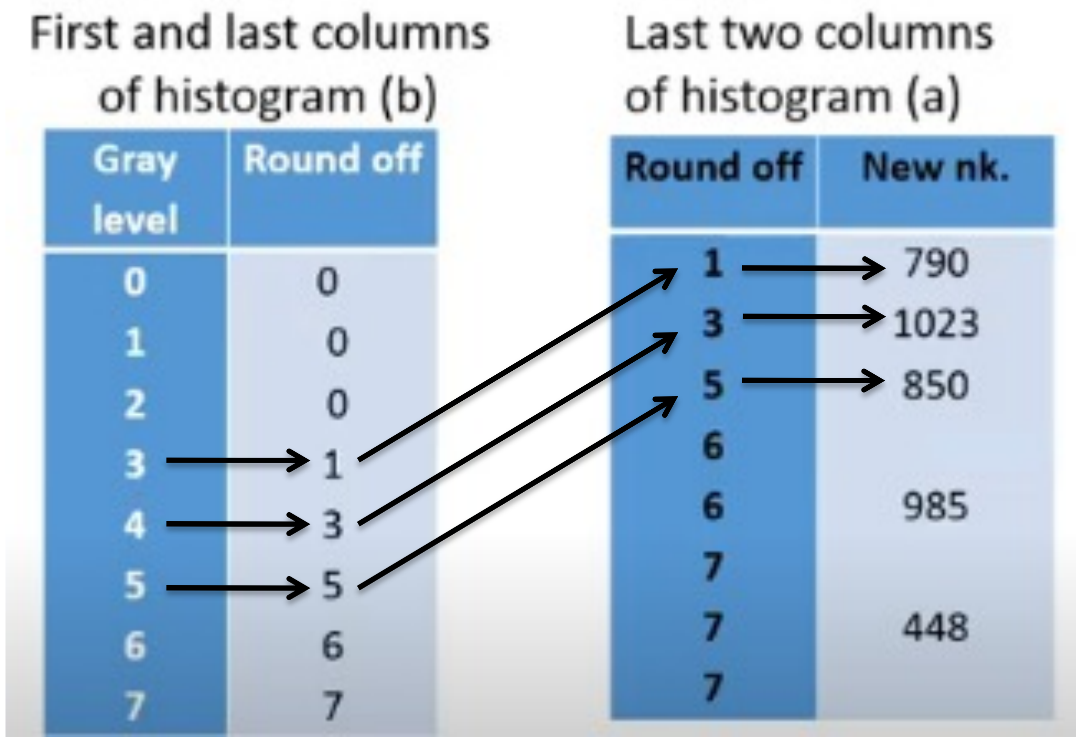
Round off	New nk.
1	790
3	1023
5	850
6	
6	985
7	
7	448
7	

Gray Level	0	1	2	3	4	5	6	7
No of Pixels				790	1023			

© Ramaiah University of Applied Sciences

Cont..

Mapping



First and last columns of histogram (b)

Gray level	Round off
0	0
1	0
2	0
3	1
4	3
5	5
6	6
7	7

Last two columns of histogram (a)

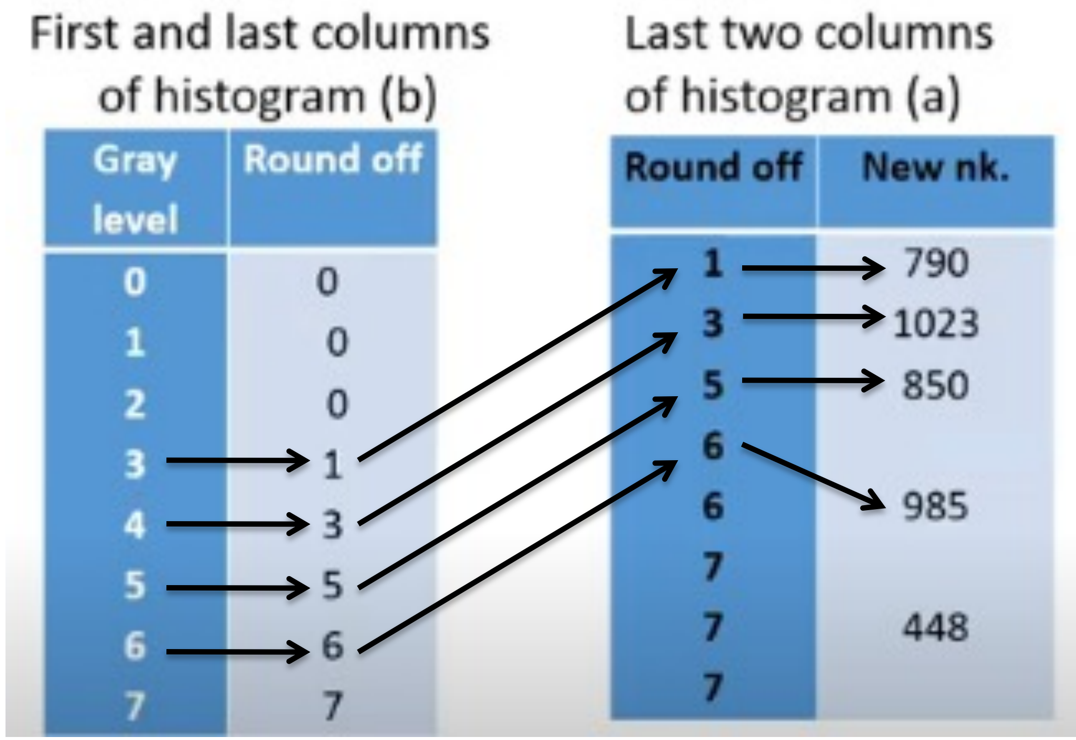
Round off	New nk.
1	790
3	1023
5	850
6	
6	985
7	
7	448
7	

Gray Level	0	1	2	3	4	5	6	7
No of Pixels				790	1023	850		

© Ramaiah University of Applied Sciences

Cont..

Mapping



First and last columns of histogram (b)

Gray level	Round off
0	0
1	0
2	0
3	1
4	3
5	5
6	6
7	7

Last two columns of histogram (a)

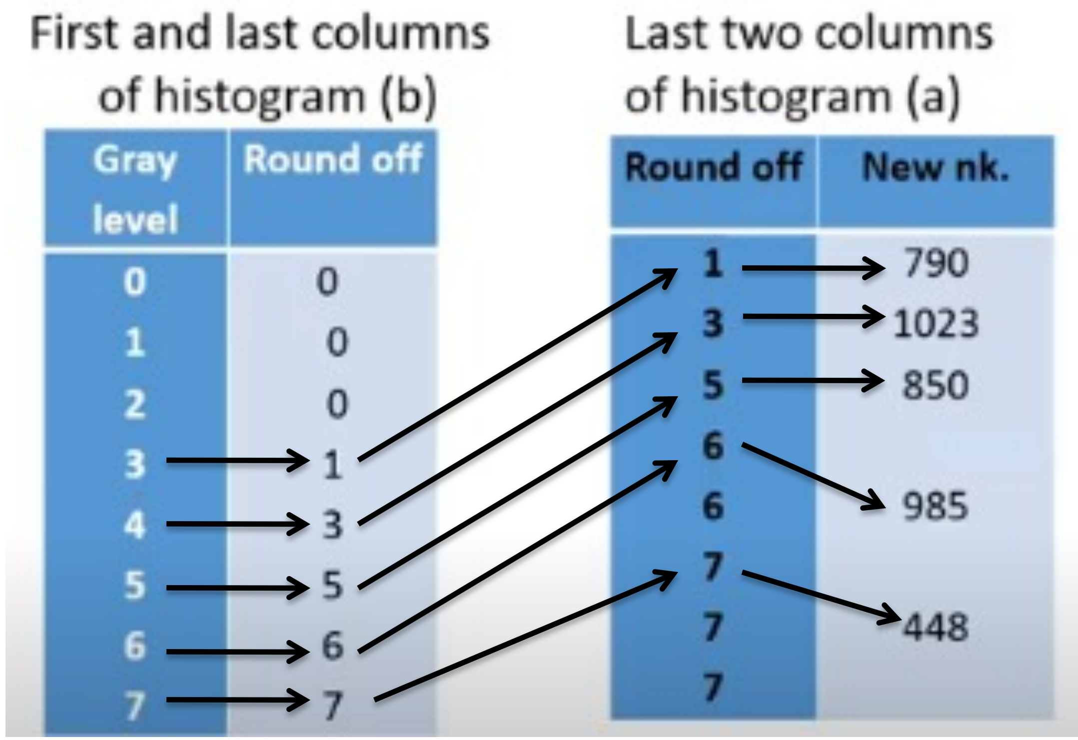
Round off	New nk.
1	790
3	1023
5	850
6	
6	985
7	
7	448
7	

Gray Level	0	1	2	3	4	5	6	7
No of Pixels				790	1023	850	985	

© Ramaiah University of Applied Sciences

Cont..

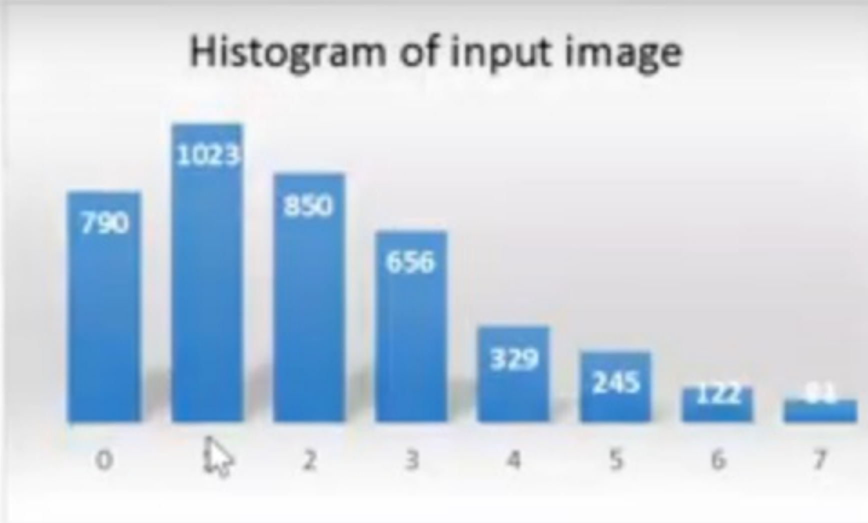
Mapping

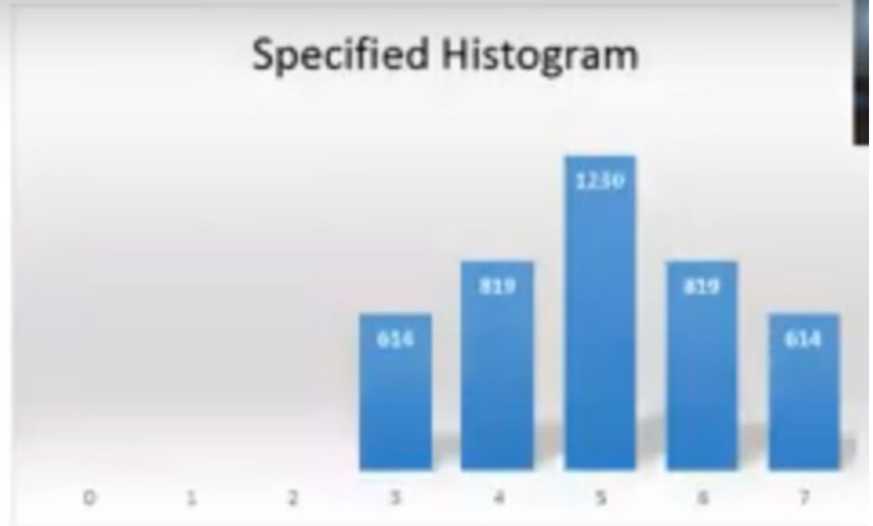


Gray Level	0	1	2	3	4	5	6	7
No of Pixels				790	1023	850	985	448

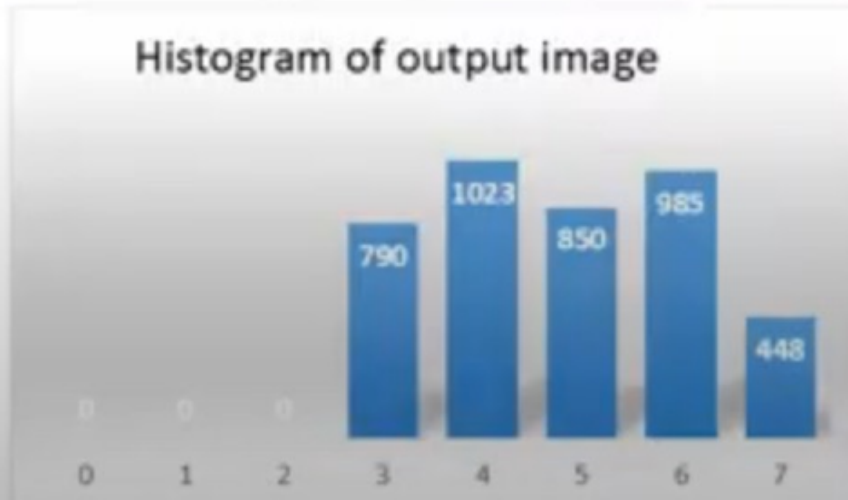
© Ramaiah University of Applied Sciences

Cont..





Plot histogram for modified image.



© Ramaiah University of Applied Sciences

Example : Histogram Matching

Suppose that a 3-bit image (L=8) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in the following table (on the left). Get the histogram transformation function and make the output image with the specified histogram, listed in the table on the right.

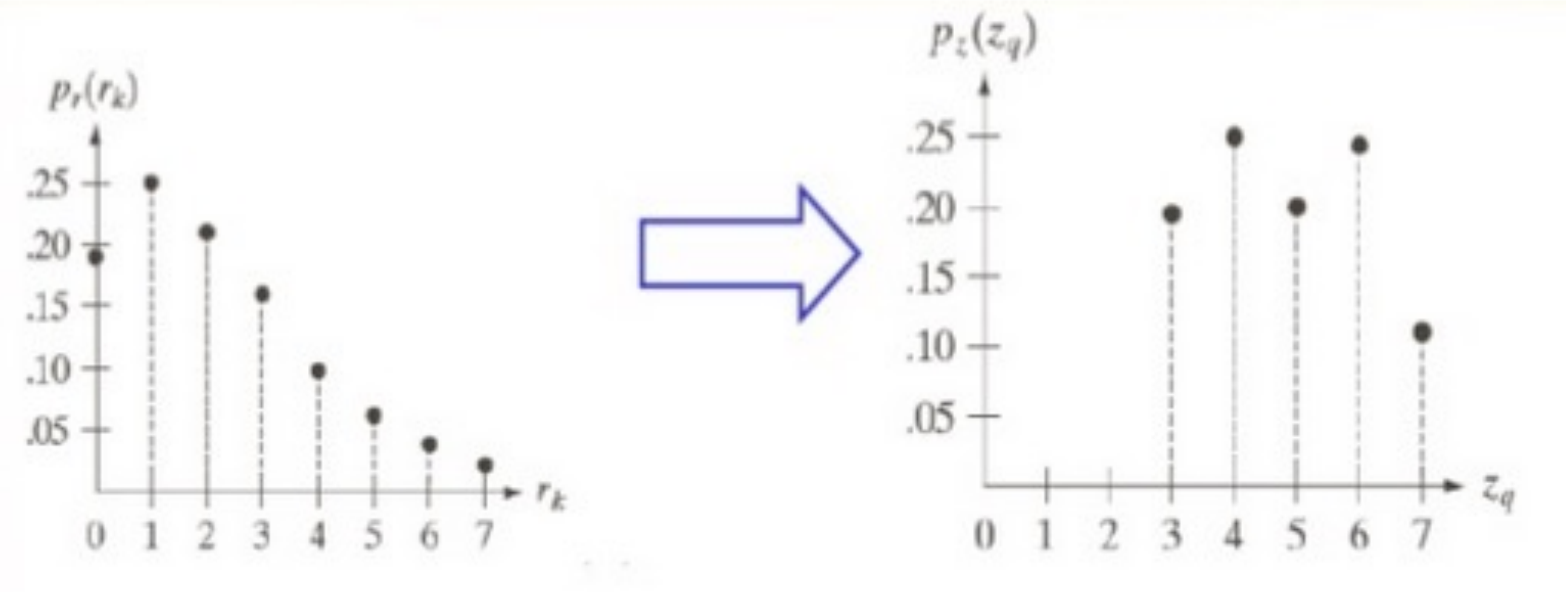
r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

z_q	Specified $p_z(z_q)$
$z_0 = 0$	0.00
$z_1 = 1$	0.00
$z_2 = 2$	0.00
$z_3 = 3$	0.15
$z_4 = 4$	0.20
$z_5 = 5$	0.30
$z_6 = 6$	0.20
$z_7 = 7$	0.15

© Ramaiah University of Applied Sciences

Example: Histogram Matching

Suppose that a 3-bit image (L=8) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in the following table (on the left). Get the histogram transformation function and make the output image with the specified histogram, listed in the table on the right.



© Ramaiah University of Applied Sciences

Example: Histogram Matching

Original (given)			Reference(given)	
r_k	n_k	$p_r(r_k) = n_k/MN$	z_k	$p(z_k)=n_k/MN$
0	790	0.19	0	0
1	1023	0.25	1	0
2	850	0.21	2	0
3	656	0.16	3	0.15
4	329	0.08	4	0.2
5	245	0.06	5	0.3
6	122	0.03	6	0.2
7	81	0.02	7	0.15

© Ramaiah University of Applied Sciences

Example: Histogram Matching(Equalization)

r_k	n_k	PDF $p_r(r_k) = n_k/MN$	CDF	S_k = CDF(L-1)	Round off	New n_k
0	790	0.19	0+0.19=0.19	1.33	1	790
1	1023	0.25	0.19+0.25=0.44	3.08	3	1023
2	850	0.21	0.44+0.21=0.65	4.55	5	850
3	656	0.16	0.65+0.16=0.81	5.67	6	985
4	329	0.08	0.89	6.23	6	
5	245	0.06	0.95	6.65	7	448
6	122	0.03	0.98	6.86	7	
7	81	0.02	1.00	7.00	7	

Original Image (given)

© Ramaiah University of Applied Sciences

Example: Histogram Matching

z_k	n_k	PDF $p_r(z_k) = n_k/MN$	CDF	S_k=CDF(L-1)	Round off
0	790	0	0	0	0
1	1023	0	0	0	0
2	850	0	0	0	0
3	656	0.15	0.15	1.05	1
4	329	0.2	0.35	2.45	2
5	245	0.3	0.65	4.55	5
6	122	0.2	0.85	5.95	6
7	81	0.15	1.00	7	7

Reference Image (given)

© Ramaiah University of Applied Sciences

Cont..

Mapping

1st and last column of HE of Reference Image Table and last 2 columns of source Table

r_k	Round off(S)
0	0
1	0
2	0
3	1
4	2
5	5
6	6
7	7

Round off (H)	New n_k
1	790
3	1023
5	850
6	985
6	
7	448
7	
7	

Map

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	0	0	790				

© Ramaiah University of Applied Sciences

Cont..

Obtain the scaled histogram-equalized values,

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 7,$$
$$s_5 = 7, s_6 = 7, s_7 = 7.$$

Compute all the values of the transformation function G,

$$G(z_0) = 7\sum_{j=0}^{0} p_z(z_j) = 0.00 \rightarrow 0$$

$$G(z_1) = 0.00 \rightarrow 0 \qquad G(z_2) = 0.00 \rightarrow 0$$
$$G(z_3) = 1.05 \rightarrow 1 \qquad G(z_4) = 2.45 \rightarrow 2$$
$$G(z_5) = 4.55 \rightarrow 5 \qquad G(z_6) = 5.95 \rightarrow 6$$
$$G(z_7) = 7.00 \rightarrow 7$$

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11

© Ramaiah University of Applied Sciences

Cont..

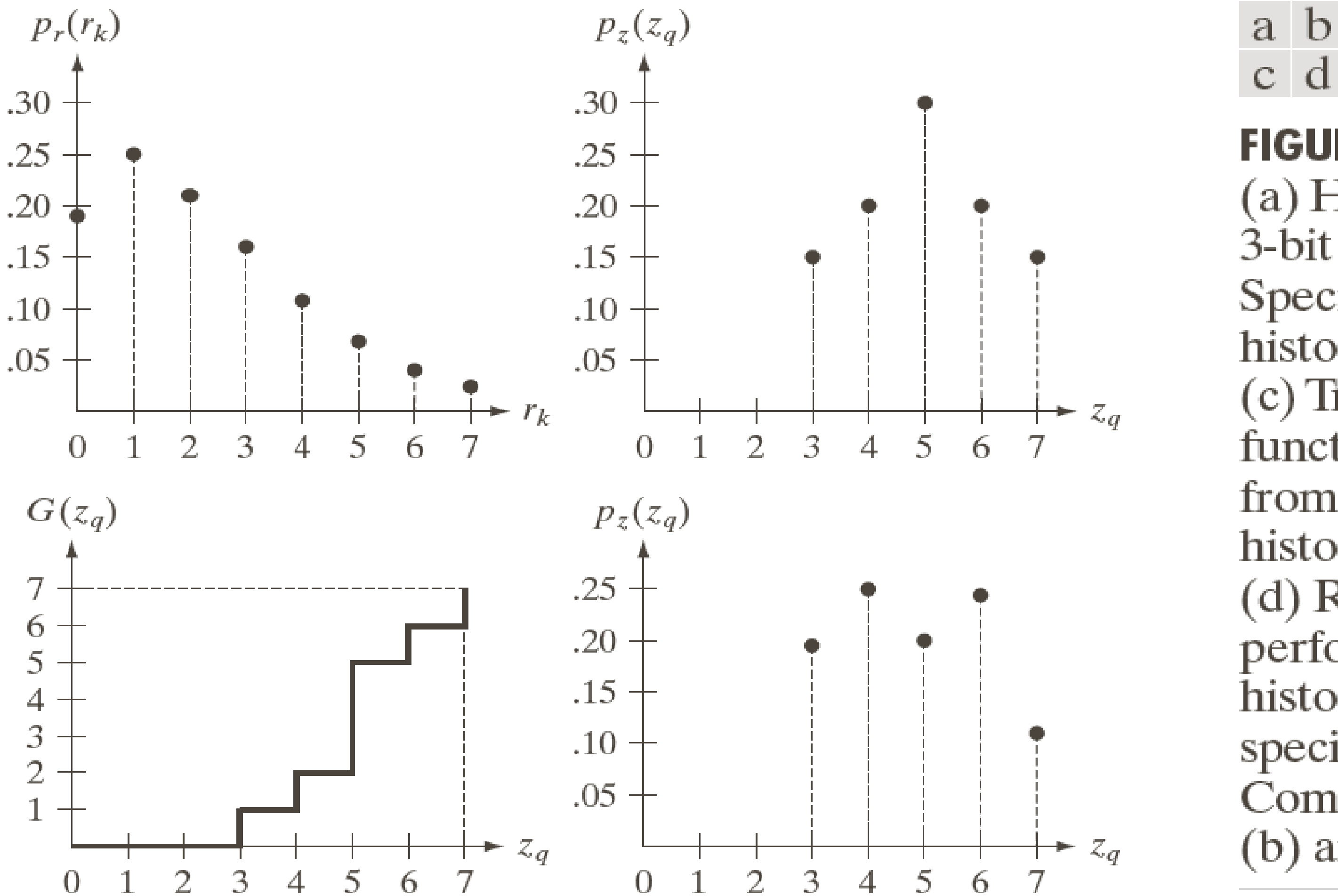


FIGURE 3.22
(a) Histogram of a 3-bit image. (b) Specified histogram. (c) Transformation function obtained from the specified histogram. (d) Result of performing histogram specification. Compare (b) and (d).

© Ramaiah University of Applied Sciences

Example: Histogram Matching

Obtain the scaled histogram-equalized values,

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 7,$$
$$s_5 = 7, s_6 = 7, s_7 = 7.$$

Compute all the values of the transformation function G,

$$G(z_0) = 7\sum_{j=0}^{0} p_z(z_j) = 0.00 \rightarrow 0$$

$G(z_1) = 0.00 \rightarrow 0$ $\quad$ $G(z_2) = 0.00 \rightarrow 0$

$G(z_3) = 1.05 \rightarrow 1$ s_0 $\quad$ $G(z_4) = 2.45 \rightarrow 2$ s_1

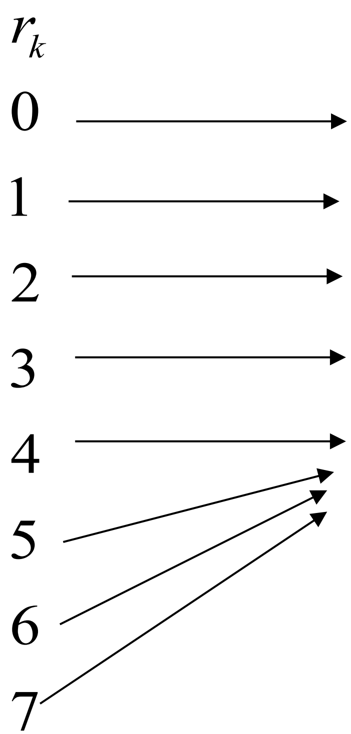
$G(z_5) = 4.55 \rightarrow 5$ s_2 $\quad$ $G(z_6) = 5.95 \rightarrow 6$ s_3

$G(z_7) = 7.00 \rightarrow 7$ s_4 s_5 s_6 s_7

© Ramaiah University of Applied Sciences

Cont..

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 7,$$
$$s_5 = 7, s_6 = 7, s_7 = 7.$$



s_k	$\rightarrow$	z_q
1	$\rightarrow$	3
3	$\rightarrow$	4
5	$\rightarrow$	5
6	$\rightarrow$	6
7	$\rightarrow$	7

© Ramaiah University of Applied Sciences

Cont..

$$r_k \rightarrow z_q$$

$$0 \rightarrow 3$$

$$1 \rightarrow 4$$

$$2 \rightarrow 5$$

$$3 \rightarrow 6$$

$$4 \rightarrow 7$$

$$5 \rightarrow 7$$

$$6 \rightarrow 7$$

$$7 \rightarrow 7$$

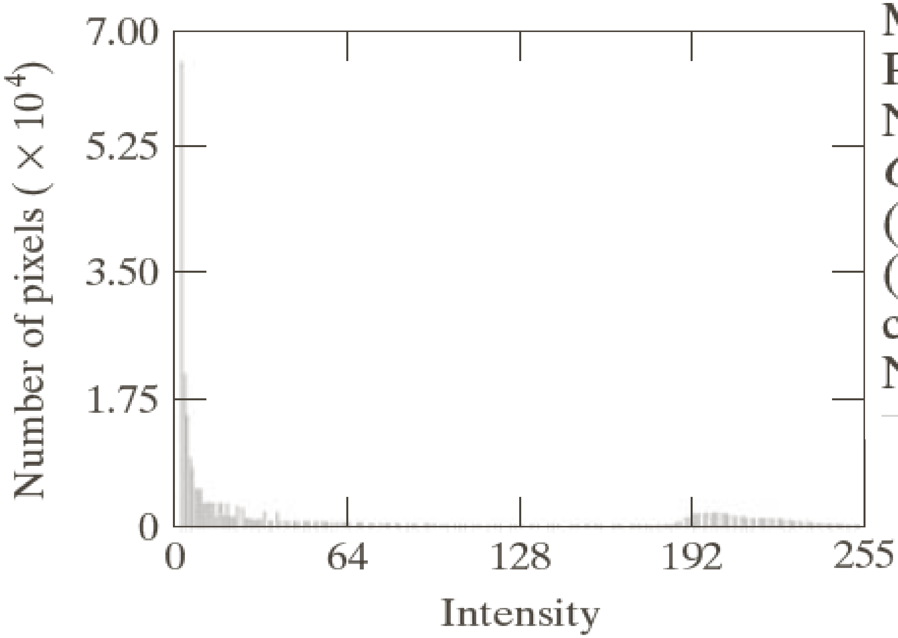
© Ramaiah University of Applied Sciences

Cont..

© Ramaiah University of Applied Sciences

Example: Histogram Matching

Drawbacks/Artefacts



a b

FIGURE 3.23 (a) Image of the Mars moon Phobos taken by NASA's *Mars Global Surveyor.* (b) Histogram. (Original image courtesy of NASA.)

© Ramaiah University of Applied Sciences

Example: Histogram Matching

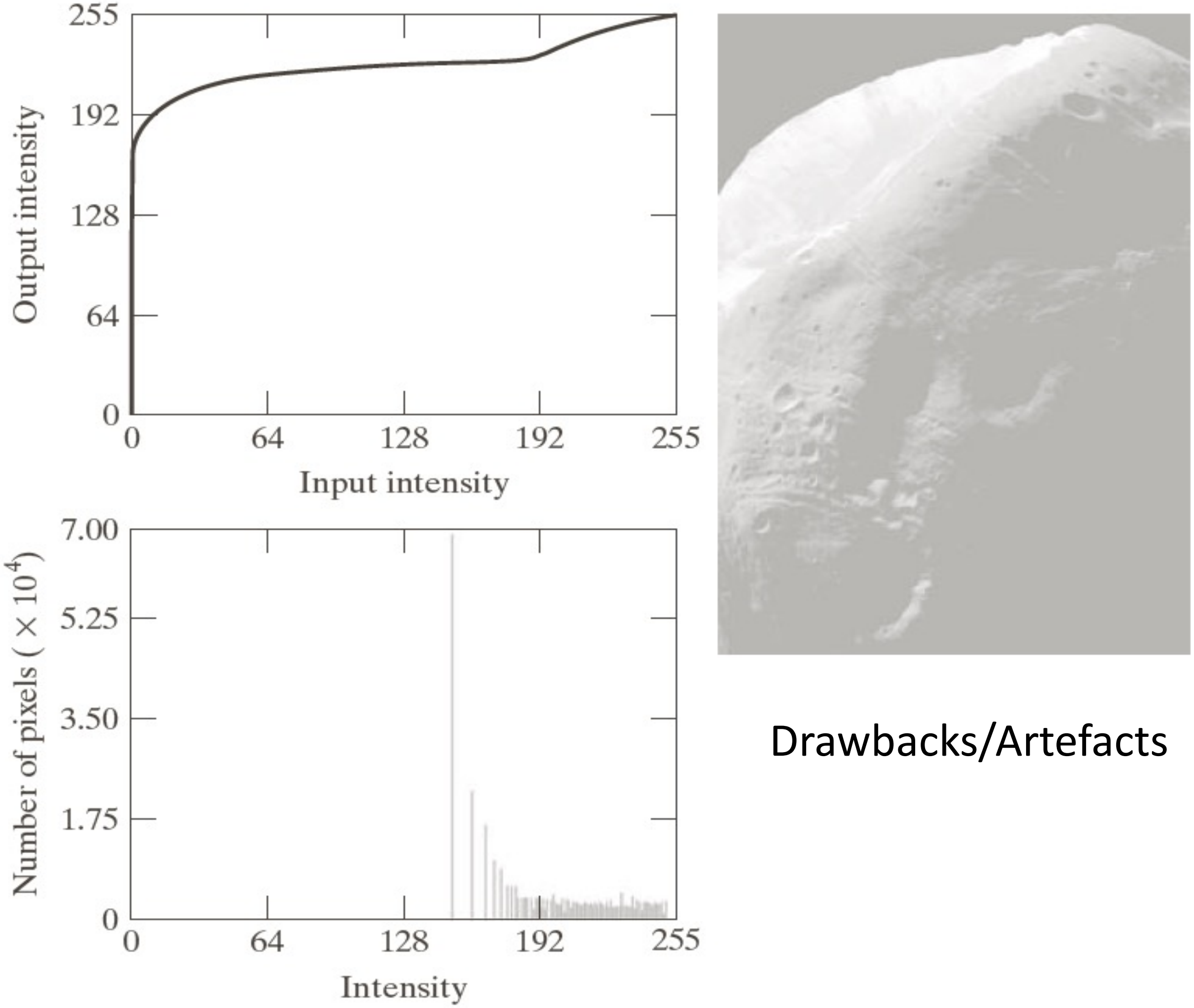
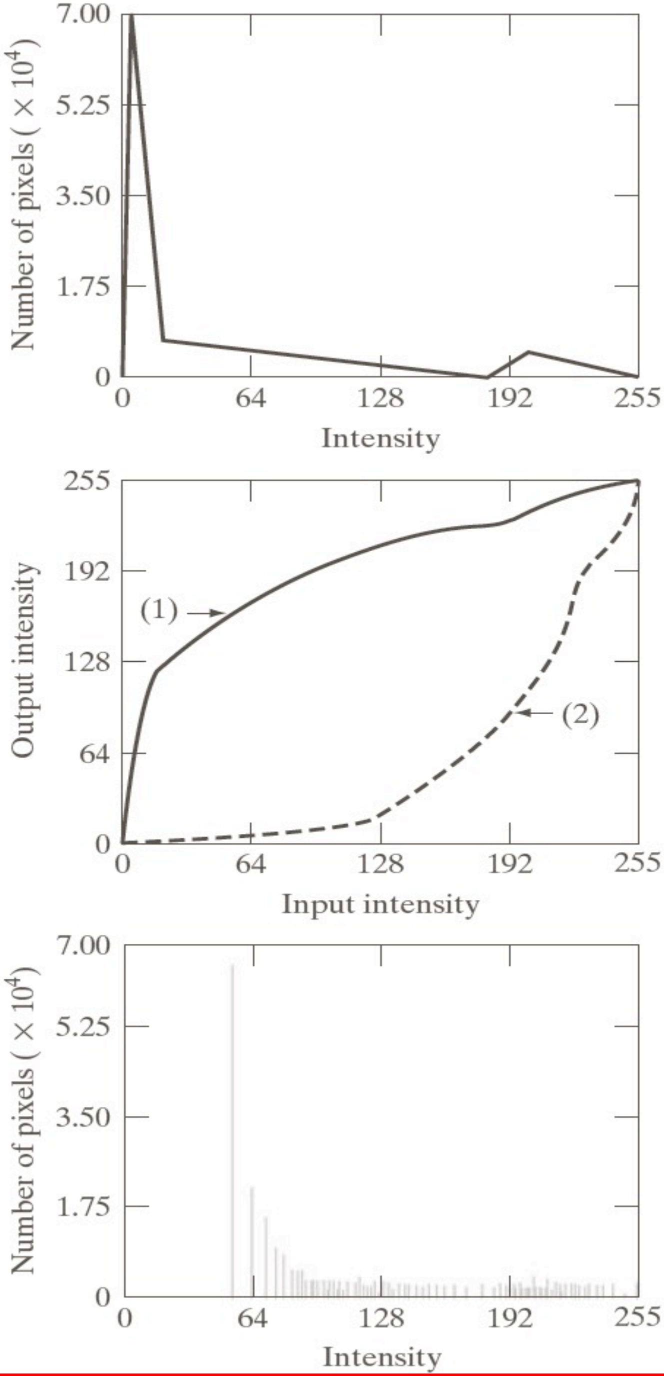


FIGURE 3.24
(a) Transformation function for histogram equalization. (b) Histogram-equalized image (note the washed-out appearance). (c) Histogram of (b).

Drawbacks/Artefacts

© Ramaiah University of Applied Sciences



Transformation function and its inverse

a c
b
d

FIGURE 3.25
(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

© Ramaiah University of Applied Sciences

Cont..

- 3.25(a) shows a *manually specified function of* original histogram, has a smoother transition of levels in the dark region of the gray scale.
- The transformation function G(z) obtained from this histogram using is labeled transformation (1) in Fig. 3.25(b). Similarly, $G^{-1}(s)$ is labeled transformation (2) in Fig. 3.25(b).
- The enhanced image in Fig. 3.25(c) was obtained by applying transformation (2) to the pixels of the histogram-equalized image in Fig. 3.24(b).The improvement of the histogram-specified image over the result obtained by histogram equalization is evident by comparing these two images.
- The histogram of Fig. 3.25(c) is shown in Fig. 3.25(d). The most distinguishing feature of this histogram is how its low end has shifted right toward the lighter region of the gray scale, as desired.

© Ramaiah University of Applied Sciences

Example 2

Perform histogram specification on the 8x8, 8-level image described in the table:

Table 1: Pixel Distribution of the image

Gray Level (r_k)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
No of Pixels (n_k)	8	10	10	2	12	16	4	2

The target histogram is as shown in table:

Table 2: Pixel Distribution of the image

Gray Level (r_k)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
No of Pixels (n_k)	0	0	0	0	20	20	16	8

© Ramaiah University of Applied Sciences

Example 2: Histogram Matching

Given histogram (a) & (b), modify histogram (a) as given by histogram (b)

Original Image

Gray level	0	1	2	3	4	5	6	7
No. of Pixels	8	10	10	2	12	16	4	2

Desired Image

Gray level	0	1	2	3	4	5	6	7
No. of Pixels	0	0	0	0	20	20	16	8

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Input Image (a)

Gray level(r_k)	No. of Pixels(n_k)	PDF(n_k/N) $P_r(r_k)$	CDF	(L-1)*CDF	H_k
0	8	0.13	0.13	0.91	1
1	10	0.16	0.29	2.03	2
2	10	0.16	0.45	3.15	3
3	2	0.03	0.48	3.36	3
4	12	0.18	0.66	4.62	5
5	16	0.25	0.91	6.37	6
6	4	0.06	0.97	6.79	7
7	2	0.03	1.0	7	7
	64	1			

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Input Image (a)

Gray level r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	Round off (H_k)	New n_k
0	8	0.13	0.13	0.91	1	8
1	10	0.16	0.29	2.03	2	10
2	10	0.16	0.45	3.15	3	12
3	2	0.03	0.48	3.36	3	
4	12	0.18	0.66	4.62	5	12
5	16	0.25	0.91	6.37	6	16
6	4	0.06	0.97	6.79	7	6
7	2	0.03	1	7	7	
	64					

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Reference Image (b)

Gray level(r_k)	No. of Pixels(n_k)	PDF(n_k/N) $P_r(r_k)$	CDF	(L-1)*CDF	S_k
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	20	0.31	0.31	2.17	2
5	20	0.31	0.62	4.34	4
6	16	0.25	0.87	6.09	6
7	8	0.13	1.0	7	7
	64	1			

© Ramaiah University of Applied Sciences

Cont..

Mapping

1st and last column of HE of Reference Image Table and last 2 columns of source Table

r_k	Round off(S)
0	0
1	0
2	0
3	0
4	2
5	4
6	6
7	7

Round off (H)	New n_k
1	8
2	10
3	12
3	
5	12
6	16
7	6
7	

Map
4
4
5
5
6
6
7
7

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	0	0	0	18	12	28	6

© Ramaiah University of Applied Sciences

Example 3: Histogram Matching

- Apply the histogram specification to the image given here. Assume the target given in table. Show the resultant final mapping.

1	3	4	5
5	6	6	6
7	7	7	7
5	5	5	5

Gray Level (r_k)	0	1	2	3	4	5	6	7
No of Pixels (n_k)	0	0	0	0	20	20	16	8

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Input Image

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	Round off (H)	New n_k
0	0	0	0	0	0	1
1	1	1/16	1/16	0.4375	0	
2	0	0	1/16	0.4375	0	
3	1	1/16	2/16	0.875	1	2
4	1	1/16	3/16	1.3125	1	
5	6	6/16	9/16	3.94	4	6
6	3	3/16	12/16	5.25	5	3
7	4	4/16	16/16=1	7	7	4

16

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization (HE) of Reference Image Table

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	$S_k = CDF(L-1)$	Round off (S)
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	20	0.3125=0.31	0.31	2.17	2
5	20	0.3125=0.31	0.62	4.34	4
6	16	0.25	0.87	6.09	6
7	8	0.125=0.13	1	7.00	7

64

© Ramaiah University of Applied Sciences

Cont..

Mapping

1st and last column of HE of Reference Image Table and last 2 columns of source Table

r_k	Round off(S)
0	0
1	0
2	0
3	0
4	2
5	4
6	6
7	7

Round off (H)	New n_k
0	1
0	
0	
1	2
1	
4	6
5	3
7	4

Map
1
1
1
4
4
5
6
7

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	1	0	0	2	6	3	4

© Ramaiah University of Applied Sciences

Example 4: Histogram Matching

- Apply the histogram specification to the image given here. Assume the target given in table. Show the resultant final mapping.

1	3	4	5
5	6	6	6
7	7	7	7
5	5	5	5

Gray Level	0	1	2	3	4	5	6	7
Target Mapping(S)	0	0	1	2	2	3	6	7

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Input Image

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	Round off (H)	New n_k
0	0	0	0	0	0	1
1	1	1/16	1/16	0.44	0	
2	0	0	1/16	0.44	0	
3	1	1/16	2/16	0.88	1	2
4	1	1/16	3/16	1.31	1	
5	6	6/16	9/16	3.94	4	6
6	3	3/16	12/16	5.25	5	3
7	4	4/16	16/16=1	7	7	4
	16					

© Ramaiah University of Applied Sciences

Cont..

Mapping

1^{st} and last column of HE of Reference Image Table and last 2 columns of source Table

r_k	Round off(S)
0	0
1	0
2	1
3	2
4	2
5	3
6	6
7	7

Round off (H)	New n_k
0	1
0	
0	
1	2
1	
4	6
5	3
7	4

Map
1
1
1
2
2
5
6
7

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	1	2	0	0	6	3	4

© Ramaiah University of Applied Sciences

Cont..

- By applying these mapping the final image will be:-

1	3	4	5
5	6	6	6
7	7	7	7
5	5	5	5

1	2	2	5
5	6	6	6
7	7	7	7
5	5	5	5

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	1	2	0	0	6	3	4

© Ramaiah University of Applied Sciences

Example 5: Histogram Matching

- Apply the histogram specification to the image given here. Assume the target given in table. Show the resultant final mapping.

1	2	3	4
5	6	7	0
7	7	7	7
5	5	5	5

Gray Level (r_k)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
No of Pixels (n_k)	0	0	0	0	20	20	16	8

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization of Input Image

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	Round off (H)	New n_k
0	1	1/16	1/16	0.44	0	1
1	1	1/16	2/16	0.88	1	2
2	1	1/16	3/16	1.31	1	
3	1	1/16	4/16	1.75	2	2
4	1	1/16	5/16	2.19	2	
5	5	5/16	10/16	4.38	4	5
6	1	1/16	11/16	4.81	5	1
7	5	5/16	16/16=1	7	7	5

16

© Ramaiah University of Applied Sciences

Cont..

Histogram Equalization (HE) of Reference Image Table

r_k	n_k	PDF $P_r(r_k) = n_k/MN$	CDF	S_k= CDF(L-1)	Round off (S)
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	20	0.3125=0.31	0.31	2.17	2
5	20	0.3125=0.31	0.62	4.34	4
6	16	0.25	0.87	6.09	6
7	8	0.125=0.13	1	7.00	7

64

© Ramaiah University of Applied Sciences

Cont..

Mapping

1st and last column of HE of Reference Image Table and last 2 columns of source Table

r_k	Round off(S)
0	0
1	0
2	0
3	0
4	2
5	4
6	6
7	7

Round off (H)	New n_k
0	1
1	2
1	
2	2
2	
4	5
5	1
7	5

Map
1
4
4
4
4
5
6
7

Gray Level	0	1	2	3	4	5	6	7
No of Pixels	0	1	0	0	4	5	1	5

© Ramaiah University of Applied Sciences

Local Histogram Processing

- Define a neighborhood and move its center from pixel to pixel
- At each location, the histogram of the points in the neighborhood is computed. Either histogram equalization or histogram specification transformation function is obtained
- Map the intensity of the pixel centered in the neighborhood
- Move to the next location and repeat the procedure

N_4 of (x,y)-> (x-1,y), (x+1,y), (x,y-1), (x, y+1)

© Ramaiah University of Applied Sciences

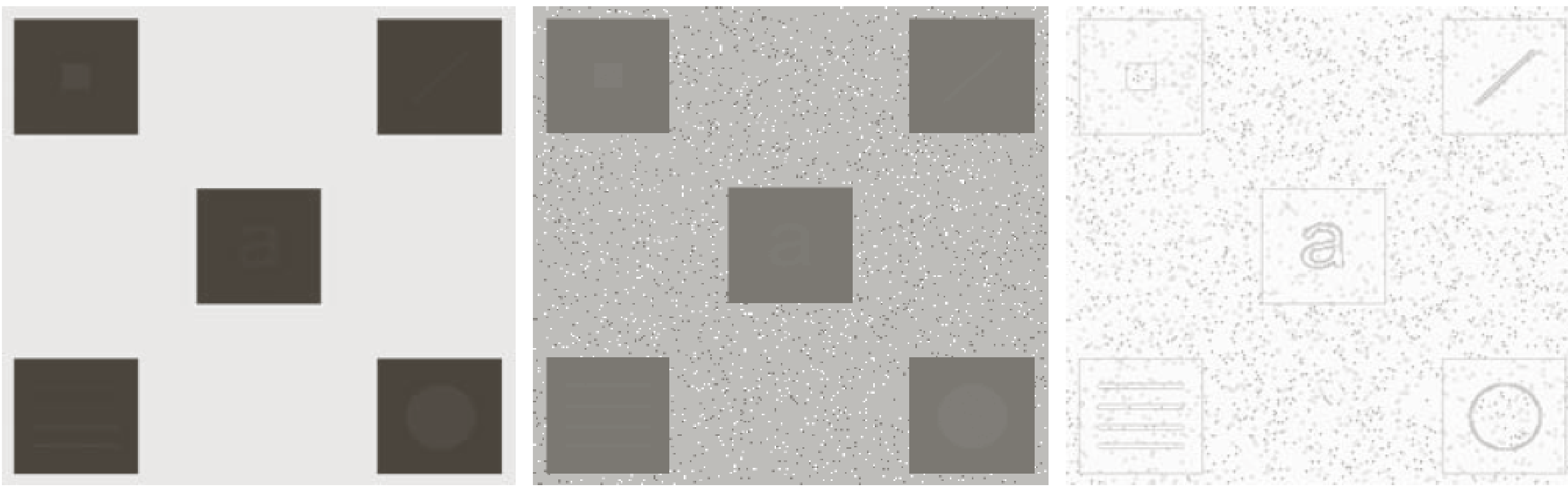
Local Histogram Processing

- Local Histogram Processing
- The two histogram processing methods discussed:
- Global
  - pixels are modified by a transformation function based on the gray-level content of an entire image.
- Global approach is suitable for overall enhancement
  - there are cases in which it is necessary to enhance details over small areas in an image

© Ramaiah University of Applied Sciences

Histogram Processing

- Global vs Local Processing
- The histogram processing methods discussed above are global, in the sense that pixels are modified by a transformation function based on the gray-level content of an entire image.
- However, there are cases in which it is necessary to enhance details over small areas in an image.



original global local

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.

© Ramaiah University of Applied Sciences

Use of Histogram Statistics for Image Enhancement

- Moments can be determined directly from a histogram much faster than they can from the pixels directly.
- Let r denote a discrete random variable representing discrete gray-levels in the range [0,L-1], and $p(r_i)$ denote the normalized histogram component corresponding to the i^{th} value of r, then 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 (average intensity) value of r

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

- For example, the second moment (also the variance of r) is

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

© Ramaiah University of Applied Sciences

Cont..

Average Intensity

$$m = \sum_{i=0}^{L-1} r_i p(r_i) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

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

Variance

$$\sigma^2 = u_2(r) = \sum_{i=0}^{L-1} (r_i - m)^2 p(r_i) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} [f(x,y) - m]^2$$

© Ramaiah University of Applied Sciences

Cont..

- **Global mean** and **variance** are computed on an entire image and useful for gross adjustment in overall intensity and contrast.
- More powerful use is in **local enhancement**, where **local mean** and **variance** are used as the basis for making changes.
- S_{xy} denote a neighborhood of a specified useful (x, y). The mean value of the pixel is given by

Local average intensity

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

Local variance

$$\sigma^2_{s_{xy}} = \sum_{i=0}^{L-1} (r_i - m_{s_{xy}})^2 p_{s_{xy}}(r_i)$$

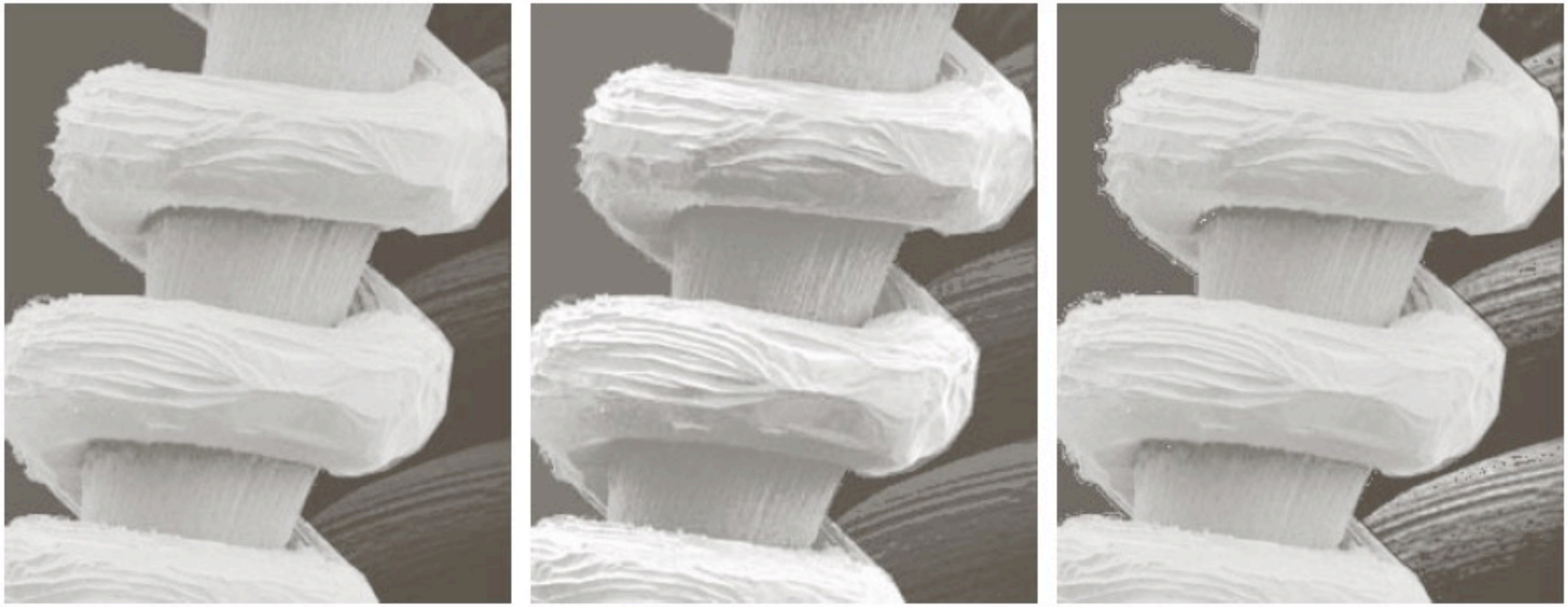
s_{xy} denotes a neighborhood

- Enhance the dark area while leaving the light area unchanged as possible.

© Ramaiah University of Applied Sciences

Histogram Processing

- Global vs Local Processing



a b c

FIGURE 3.27 (a) SEM image of a tungsten filament magnified approximately 130×. (b) Result of global histogram equalization. (c) Image enhanced using local histogram statistics. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

© Ramaiah University of Applied Sciences

Cont..

- Two uses of the mean and variance for enhancement purposes:
  - The global mean and variance (global means for the entire image) are useful for adjusting overall contrast and intensity.
  - The mean and standard deviation for a local region are useful for correcting for large-scale changes in intensity and contrast.

© Ramaiah University of Applied Sciences

Enhancement using Arithmetic/Logic Operations

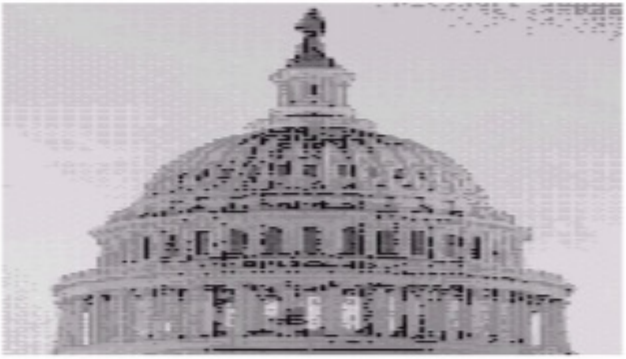
© Ramaiah University of Applied Sciences

Enhancement using Arithmetic/Logic Operations

- Arithmetic/ Logic operations perform on pixel by pixel basis between two or more images
- Except NOT operation which perform only on a single image
- **Logic Operations**
- Logic operation performs on gray-level images, the pixel values are processed as binary numbers
- light represents a binary 1, and dark represents a binary 0
- NOT operation = negative transformation

© Ramaiah University of Applied Sciences

Example of AND Operation



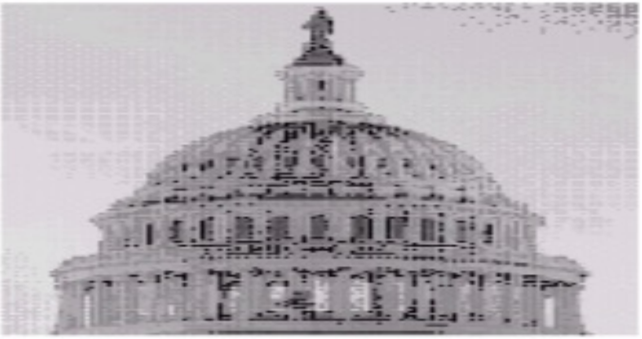
original image

AND image mask

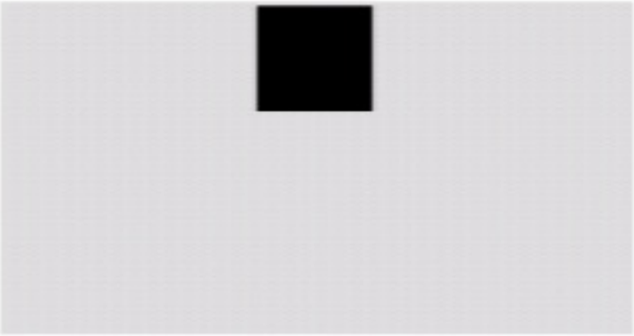
result of AND operation

© Ramaiah University of Applied Sciences

Example of OR Operation



original image



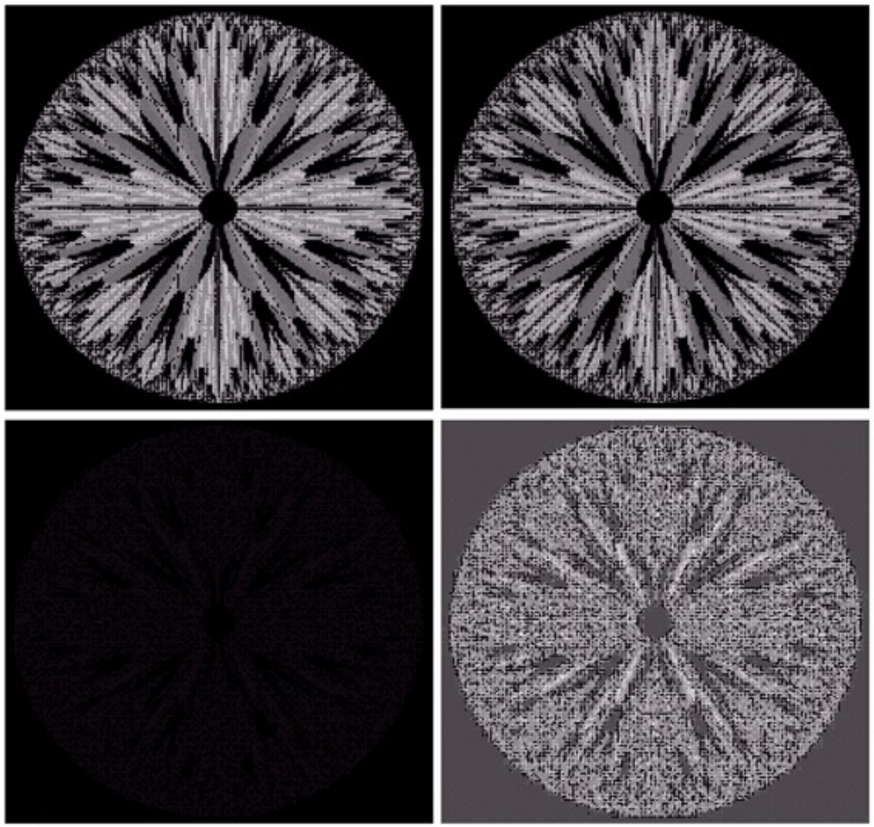
OR image mask

result of OR operation

© Ramaiah University of Applied Sciences

Example of Image Subtraction

$g(x,y) = f(x,y) - h(x,y)$



a	**b**
c	**d**

a) original fractal image

b) result of setting the 4 lower-order bit planes to zero

– refer to the bit-plane slicing

– the higher planes contribute significant detail

– the lower planes contribute more to fine detail

– image b). is nearly identical visually to image a), with a very slightly drop in overall contrast due to less variability of the gray-level values in the image.

c) difference between a) and b) (nearly black)

d) histogram equalization of c). (perform contrast stretching transformation)

© Ramaiah University of Applied Sciences

Spatial Filtering

© Ramaiah University of Applied Sciences

Spatial Filtering

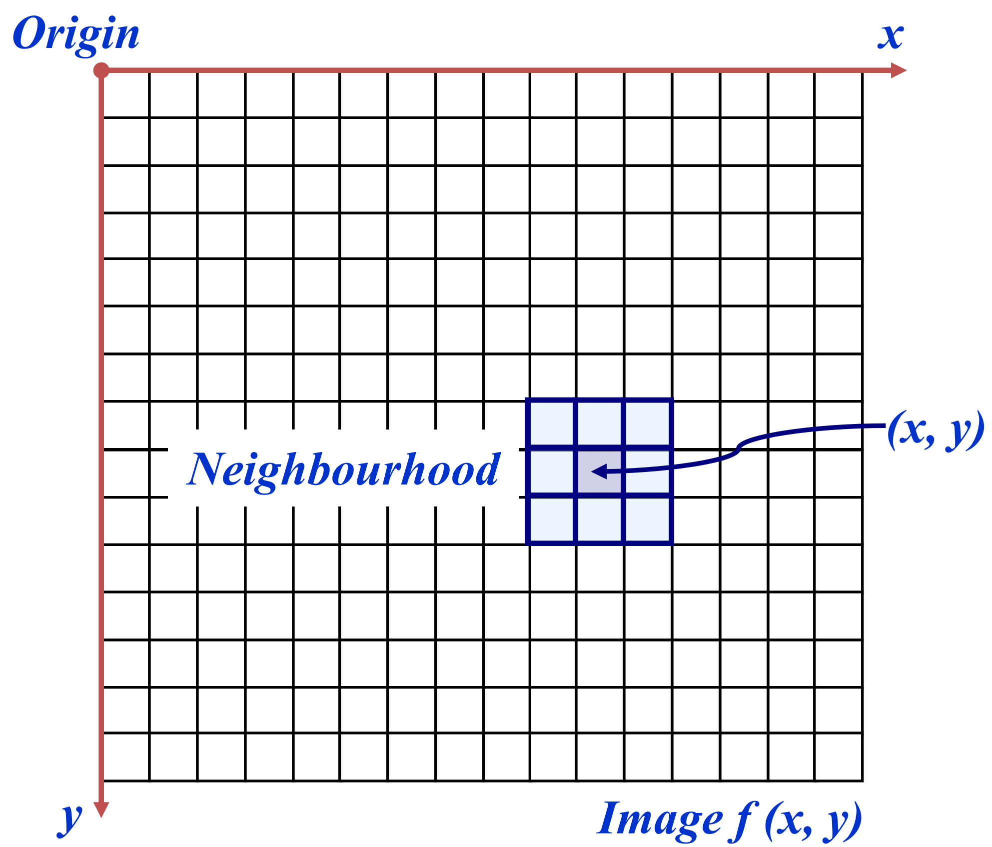
In this lecture we will look at spatial filtering techniques:

- Neighbourhood operations
- What is spatial filtering?
- Smoothing operations
- What happens at the edges?
- Correlation and convolution
- Sharpening filters
- Combining filtering techniques

© Ramaiah University of Applied Sciences

Neighbourhood Operations

- Neighbourhood operations simply operate on a larger neighbourhood of pixels than point operations
- Neighbourhoods are mostly a rectangle around a central pixel
- Any size rectangle and any shape filter are possible



© Ramaiah University of Applied Sciences

Simple Neighbourhood Operations

- 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
  - **Average:** Average of the pixel values
  - **Median:** The median value of a set of numbers is the midpoint value in that set (e.g. from the set [1, 7, 15, 18, 24] 15 is the median). Sometimes the median works better than the average

© Ramaiah University of Applied Sciences

Simple Neighbourhood Operations Example

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

Enhanced Image

© Ramaiah University of Applied Sciences

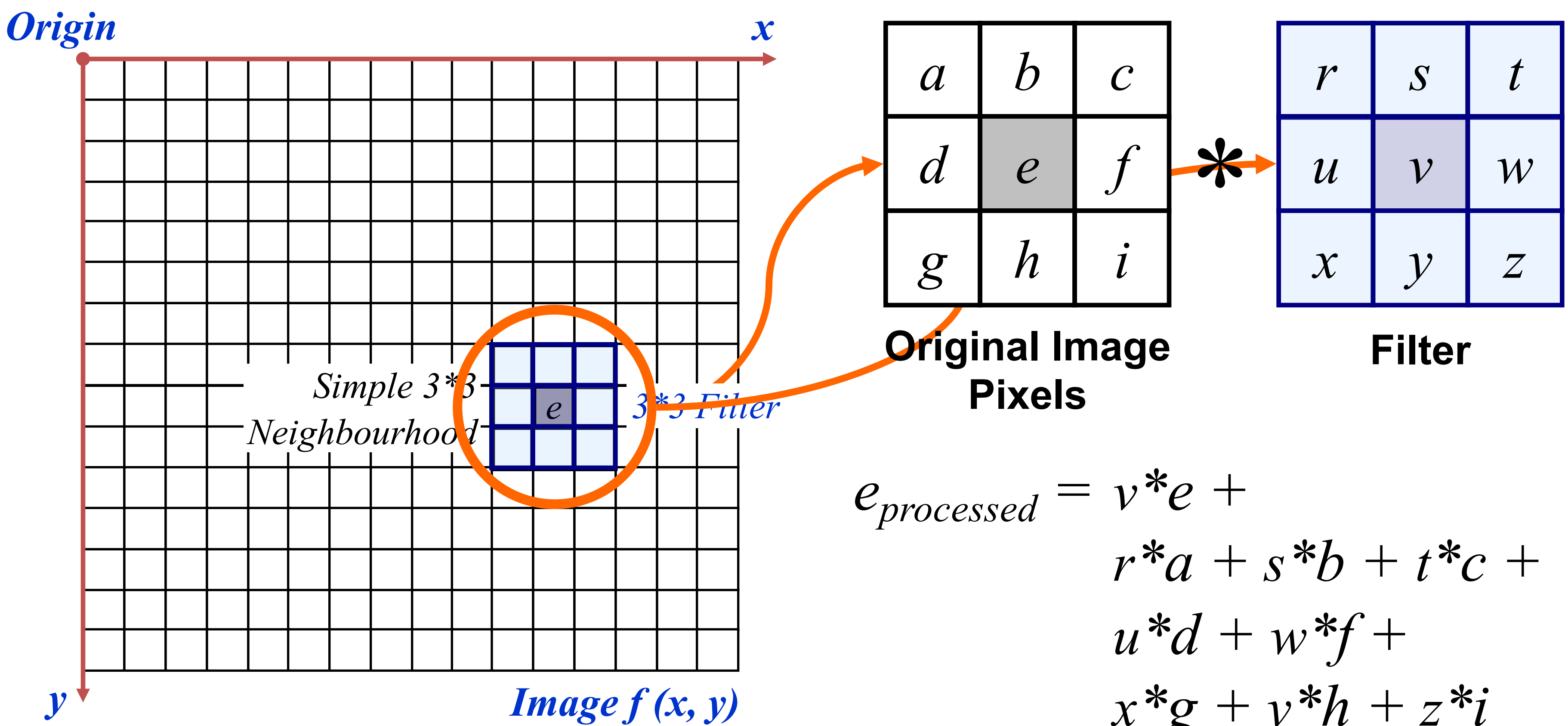
Spatial Filtering

- Fundamentals of spatial filtering:-
  - Filtering is useful for amplifying(passing) or rejecting certain frequency components.
- A spatial filter consists of (a) **a neighborhood**, and (b) **a predefined operation**
- Linear spatial filtering of an image of size MxN with a filter of size mxn is given by the expression

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

© Ramaiah University of Applied Sciences

The Spatial Filtering Process

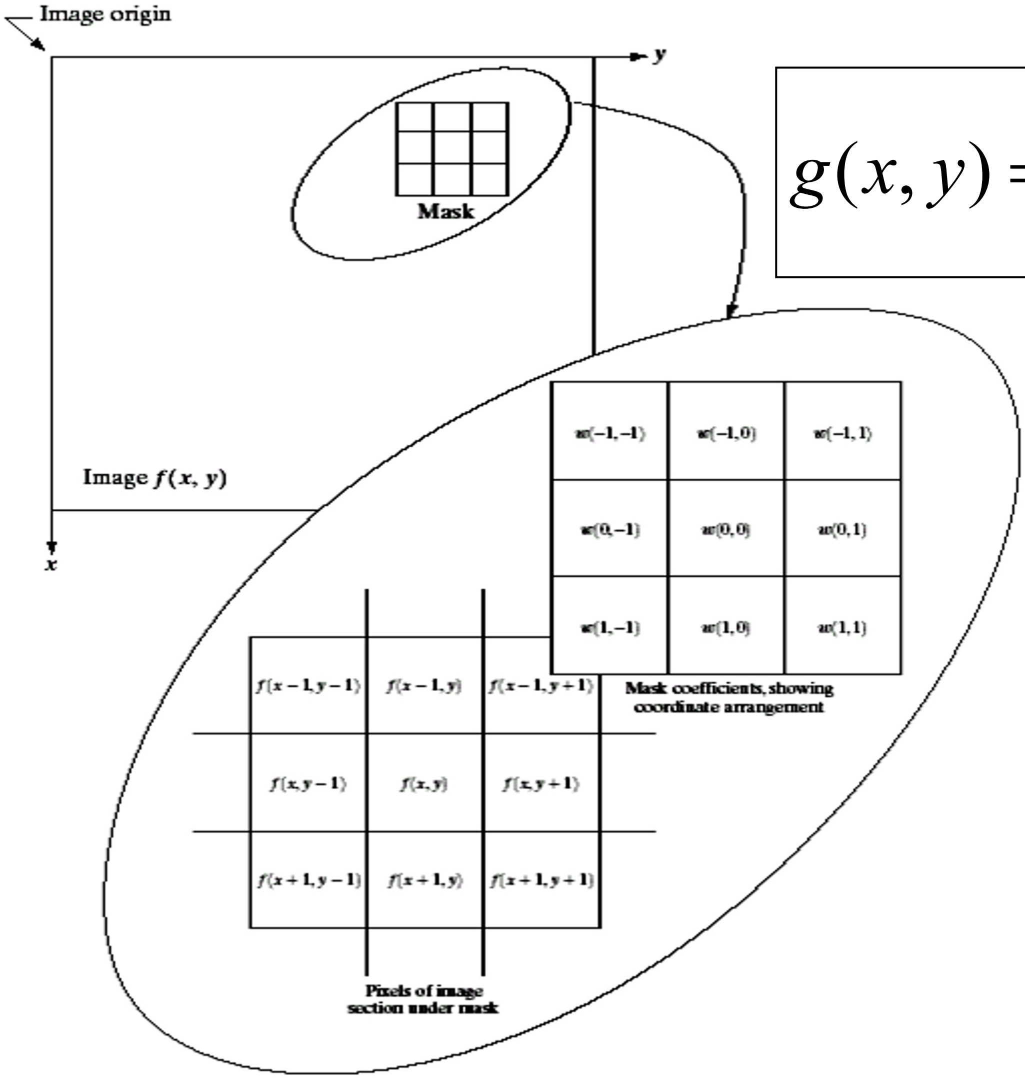


$$e_{processed} = v*e + r*a + s*b + t*c + u*d + w*f + x*g + y*h + z*i$$

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

© Ramaiah University of Applied Sciences

Spatial Filtering: Equation Form



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

- Filtering can be given in equation form as shown above
- Notations are based on the image shown to the left

© Ramaiah University of Applied Sciences

Smoothing Spatial Filters

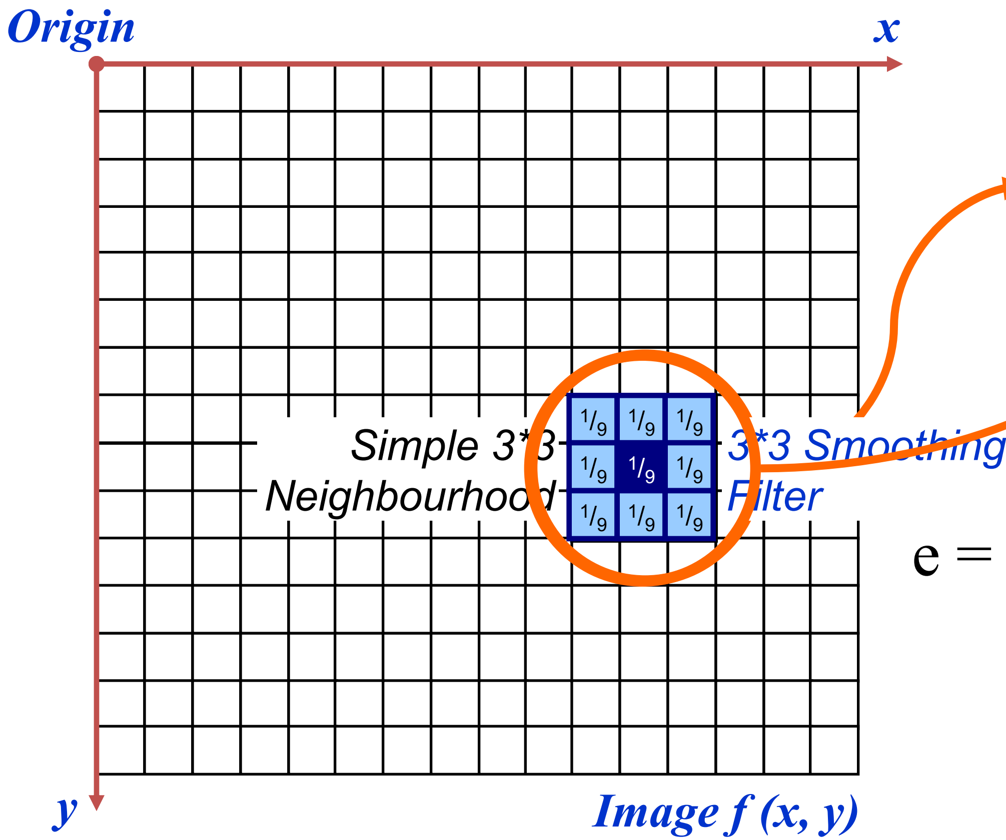
- One of the simplest spatial filtering operations we can perform is a smoothing operation
  - Simply average all of the pixels in a neighbourhood around a central value
  - Especially useful in removing noise from images
  - Also useful for highlighting gross detail

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

Simple averaging filter

© Ramaiah University of Applied Sciences

Smoothing Spatial Filtering



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

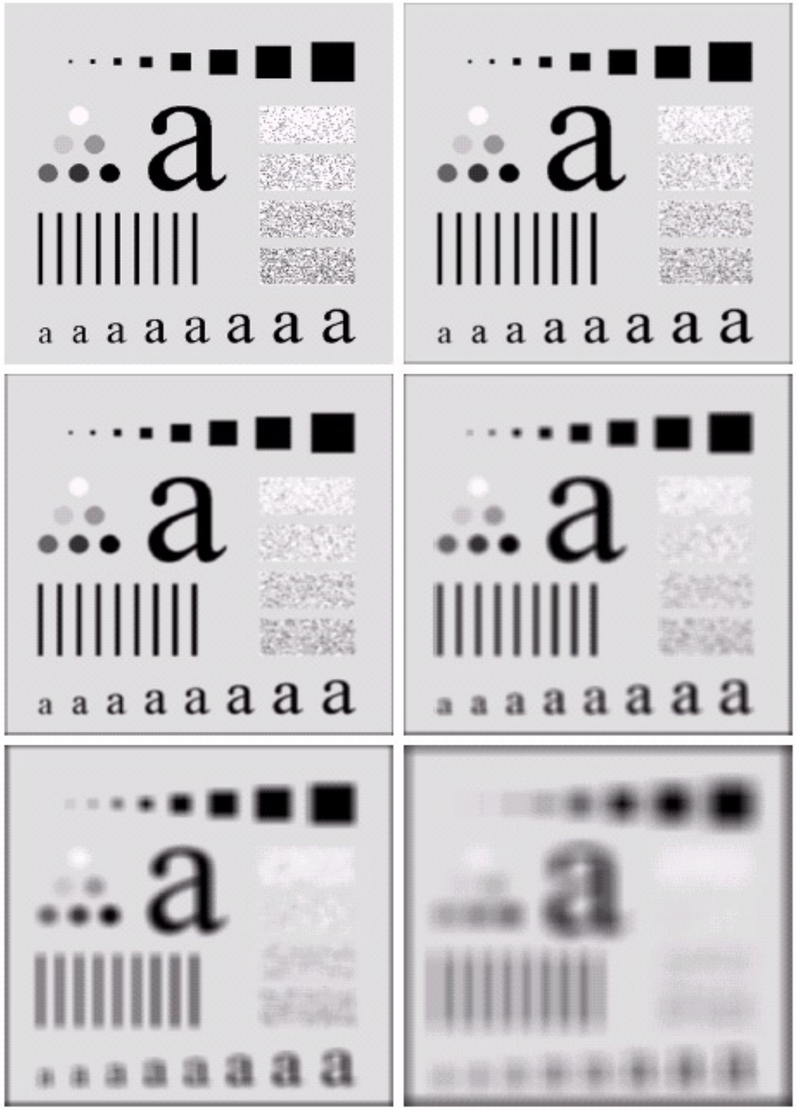
$$
\begin{aligned}
e = {} & {}^1/_9 * 106 + \\
& {}^1/_9 * 104 + {}^1/_9 * 100 + {}^1/_9 * 108 + \\
& {}^1/_9 * 99 + {}^1/_9 * 98 + \\
& {}^1/_9 * 95 + {}^1/_9 * 90 + {}^1/_9 * 85 \\
= {} & 98.3333
\end{aligned}
$$

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

© Ramaiah University of Applied Sciences

Image Smoothing Example

- The image at the top left is an original
  image of size 500*500 pixels
- The subsequent images show the image after filtering with an averaging filter of increasing sizes
  - 3, 5, 9, 15 and 35
- Notice how detail begins to disappear



© Ramaiah University of Applied Sciences

Weighted Smoothing Filters

- More effective smoothing filters can be generated by allowing different pixels in the neighbourhood different weights in the averaging function
  - Pixels closer to the central pixel are more important
  - Often referred to as a *weighted averaging*

$^1/_{16}$	$^2/_{16}$	$^1/_{16}$
$^2/_{16}$	$^4/_{16}$	$^2/_{16}$
$^1/_{16}$	$^2/_{16}$	$^1/_{16}$

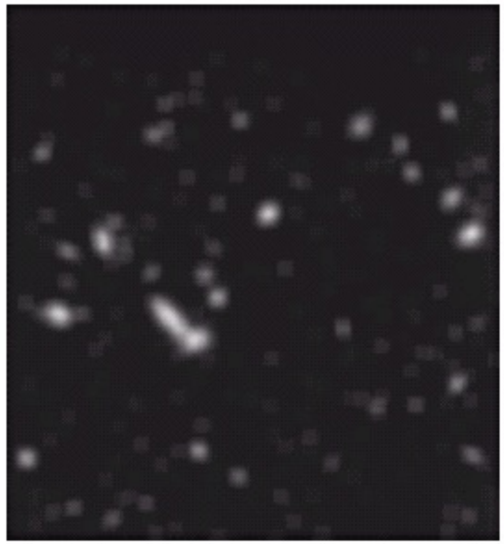
Weighted averaging filter

© Ramaiah University of Applied Sciences

Another Smoothing Example

- By smoothing the original image we get rid of lots of the finer detail which leaves only the gross features for thresholding

Original Image



Smoothed Image

Thresholded Image

© Ramaiah University of Applied Sciences

Order-statistic (Nonlinear) Filters

- **Nonlinear**
- Based on ordering (ranking) the pixels contained in the filter mask
- Replacing the value of the center pixel with the value determined by the ranking result
- E.g., median filter, max filter, min filter

© Ramaiah University of Applied Sciences

Median Filtering

- Median filtering is particularly effective in the presence of impulse noise (salt and pepper noise).

- Unlike average filtering, median filtering does not blur too much image details.

© Ramaiah University of Applied Sciences

Median Filtering

– The median filter is non linear:

$$\text{median}\{x + y\} \neq \text{median}\{x\} + \text{median}\{y\}$$

– It works well for impulse noise (e.g. salt and pepper).

– It requires sorting of the image values.

– It preserves the edges better than an average filter in the case of impulse noise.

– It is robust to impulse noise at 50%.

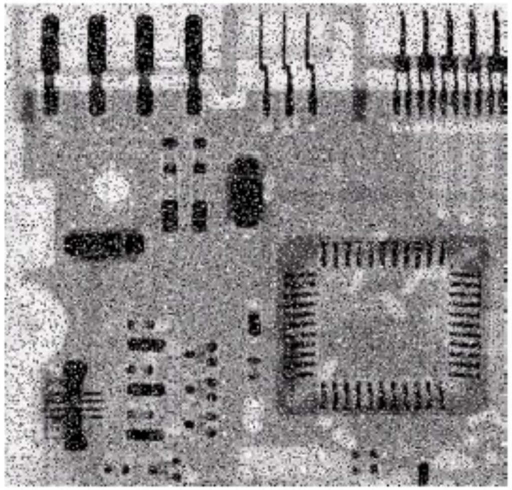
© Ramaiah University of Applied Sciences

Median Filtering

- Advantages:
  - Removes impulsive noise
  - Preserves edges
- Disadvantages:
  - poor performance when # of noise pixels in the window is greater than 1/2 # in the window
  - poor performance with Gaussian noise

© Ramaiah University of Applied Sciences

Averaging Filter Vs. Median Filter Example



Original Image With Noise

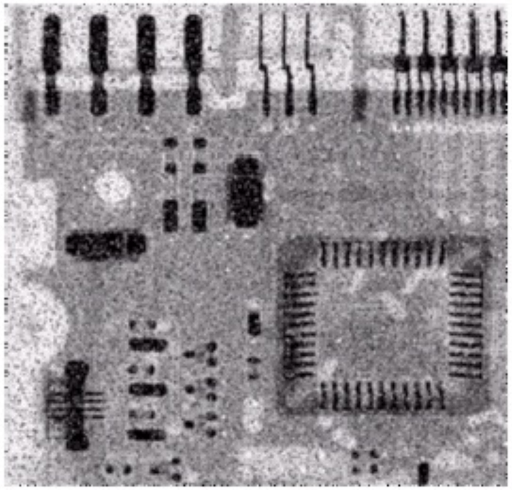


Image After Averaging Filter

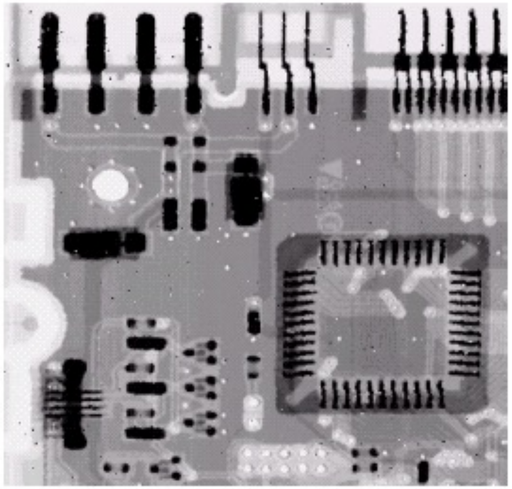


Image After Median Filter

- Filtering is often used to remove noise from images
- Sometimes a median filter works better than an averaging filter

© Ramaiah University of Applied Sciences

Correlation & Convolution

- The filtering we have been talking about so far is referred to as *correlation* with the filter itself referred to as the *correlation kernel*
- *Convolution* is a similar operation, with just one subtle difference

a	b	c
d	e	e
f	g	h

Original Image Pixels

$*$

r	s	t
u	v	w
x	y	z

Filter

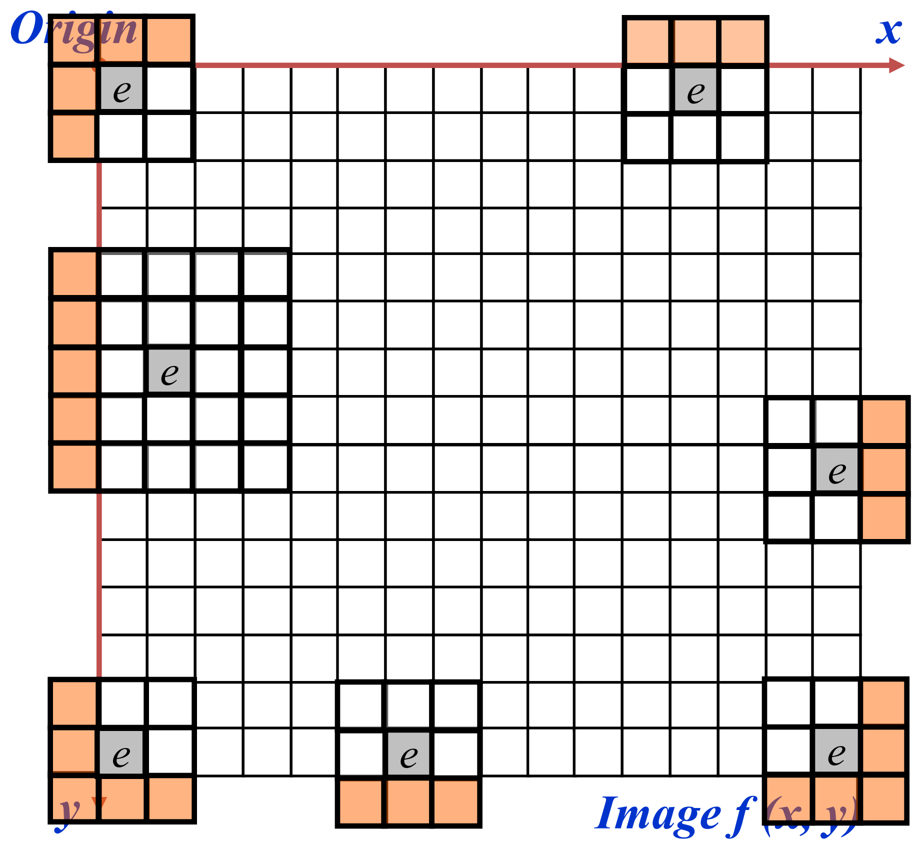
$$
\begin{aligned}
e_{processed} = {} & v*e + \\
& z*a + y*b + x*c + \\
& w*d + u*e + \\
& t*f + s*g + r*h
\end{aligned}
$$

- For symmetric filters it makes no difference.

© Ramaiah University of Applied Sciences

Strange Things Happen At The Edges!

At the edges of an image we are missing pixels to form a neighbourhood



© Ramaiah University of Applied Sciences

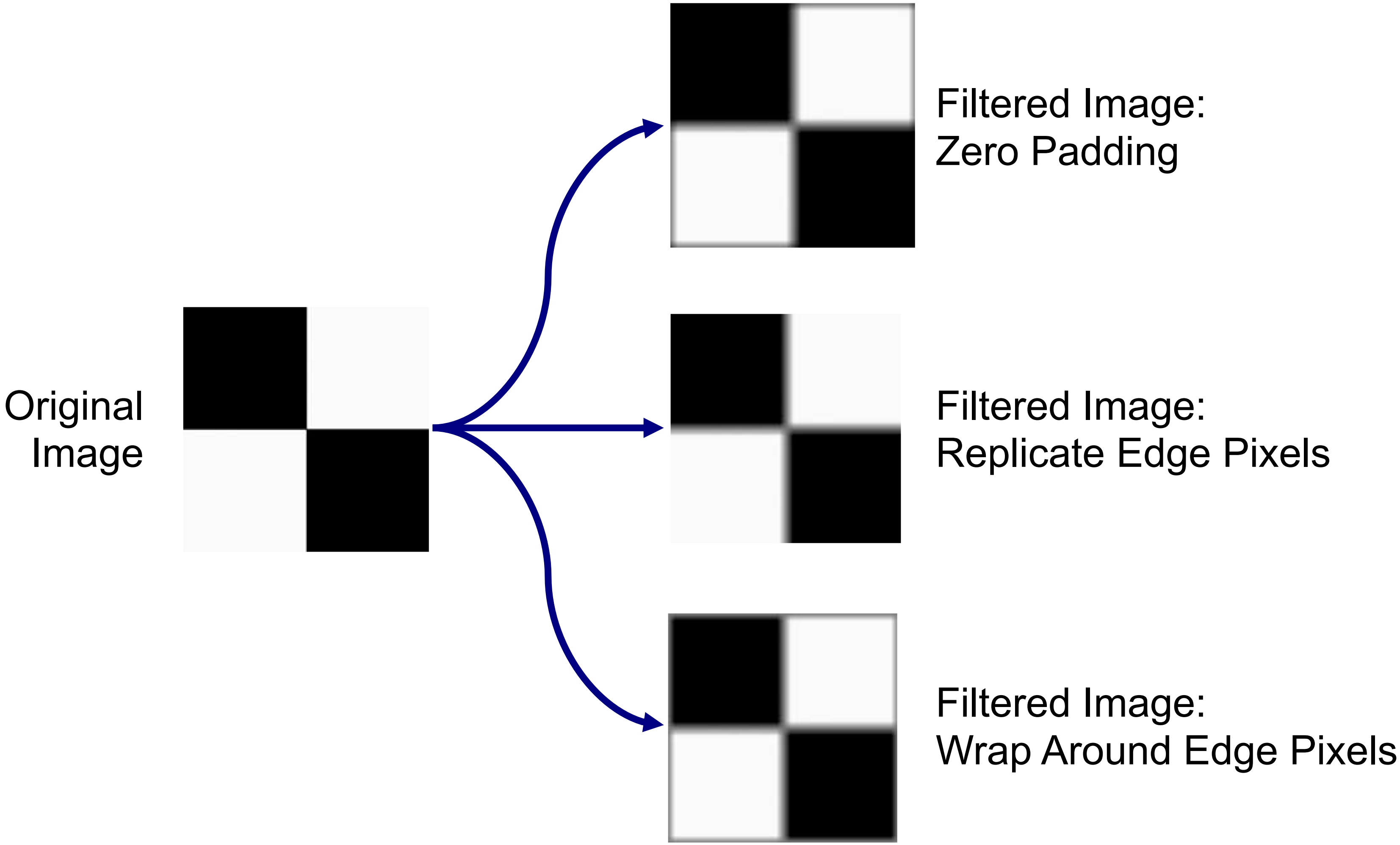
Strange Things Happen At The Edges! (cont…)

There are a few approaches to dealing with missing edge pixels:

- Omit missing pixels
  - Only works with some filters
  - Can add extra code and slow down processing
- Pad the image
  - Typically with either all white or all black pixels
- Replicate border pixels
- Truncate the image
- Allow pixels *wrap around* the image
  - Can cause some strange image artefacts

© Ramaiah University of Applied Sciences

Strange Things Happen At The Edges! (cont…)



© Ramaiah University of Applied Sciences

Fundamentals of Spatial Filtering

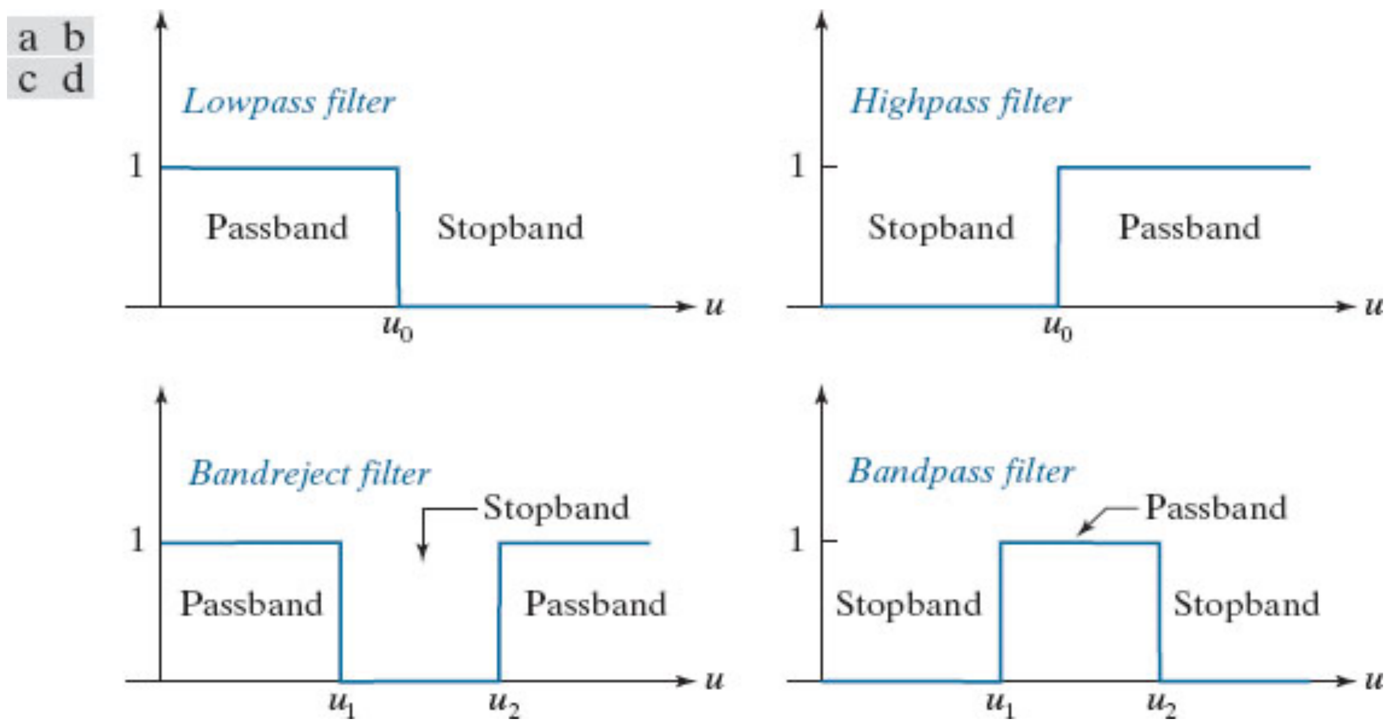
- Filtering refers to accepting(passing) or rejecting certain frequency components. This effectively smoothens or sharpens the image.
- E.g. Low pass filter, high pass filter, etc.
- Such operations can be directly carried out on image in spatial domain also by using spatial filters (kernels, spatial masks, templates, & windows).
- Spatial filters are more versatile as they are used in linear as well as non-linear filtering (Difficult in frequency domain).

© Ramaiah University of Applied Sciences

Cont..

Transfer functions of ideal 1-D filters in the frequency domain (u denotes frequency). (a) Lowpass filter. (b) Highpass filter. (c) Bandreject filter. (d) Bandpass filter. (As before, we show only positive frequencies for simplicity.)

Figure 3.58



© Ramaiah University of Applied Sciences

Cont..

Summary of the four principal spatial filter types expressed in terms of lowpass filters. The centers of the unit impulse and the filter kernels coincide.

Filter type	Spatial kernel in terms of lowpass kernel, *lp*
Lowpass	$lp(x,y)$
Highpass	$hp(x,y) = \delta(x,y) - lp(x,y)$
Bandreject	$br(x,y) = lp_1(x,y) + hp_2(x,y)$ $= lp_1(x,y) + [\delta(x,y) - lp_2(x,y)]$
Bandpass	$bp(x,y) = \delta(x,y) - br(x,y)$ $= \delta(x,y) - [lp_1(x,y) + [\delta(x,y) - lp_2(x,y)]]$

Table 3.7

© Ramaiah University of Applied Sciences

Fundamentals of Spatial Filtering

- Types of Spatial Filtering
    1. Point to point (pixel to pixel) operation (discussed so far)
    2. Mask based (Neighborhood) operations
        i. Operation with 3x3 filter (E.g. Mean, max, min, etc)
        ii. Correlation or Convolution
- Linear vs Non-Linear Filter If the operation performed on the image pixels is linear, then the filter is called a linear spatial filter, otherwise nonlinear.

© Ramaiah University of Applied Sciences

Operation with 3x3 Filter

- At any point (x, y) in the image, the response, g(x, y), of the filter is the sum of products of the filter coefficients and the image pixels encompassed by the filter:

$$g(x, y) = f(x-1,y-1).w1 + f(x-1,y).w2 + f(x-1,y+1).w3 + f(x,y-1).w4 + f(x, y).w5 + f(x,y+1).w6 + f(x+1,y-1).w7 + f(x+1,y).w8 + f(x+1,y+1).w9$$

- For the mask of size m x n, we assume

  m = 2a + 1;

  n = 2b + 1;

  where a & b are positive integers.

X-1,y+1	X,y+1	
x-1,y	x,y	
x-1, y-1	X,y-1	

- 3x3 is the smallest filter.

© Ramaiah University of Applied Sciences

Operation with 3x3 Filter

- Generalized equation:

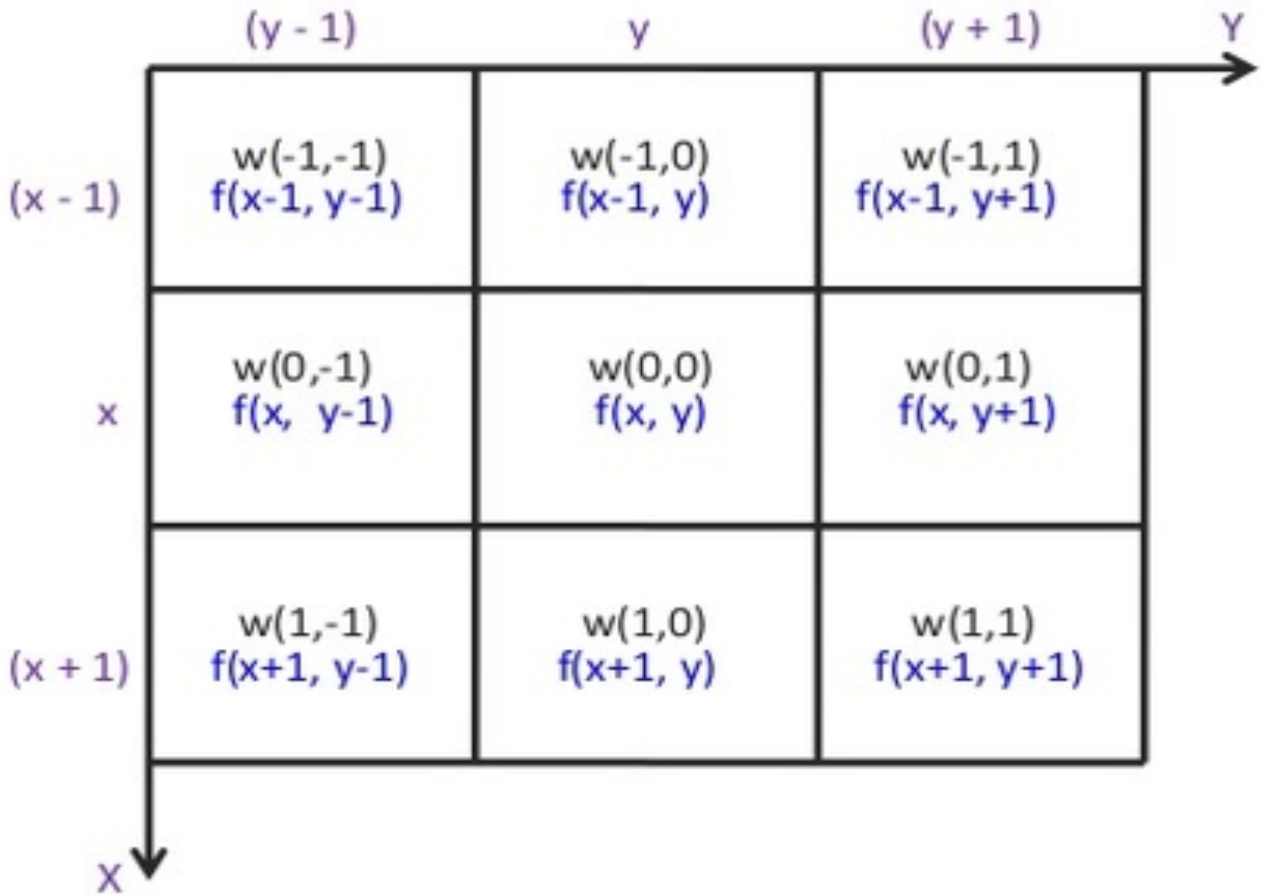
$$g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s.t) f(x+s, y+t)$$

- where, x & y are varied so that each pixel in w visits every pixel in f.

© Ramaiah University of Applied Sciences

Operation with 3x3 Filter

- 3 x 3 Neighborhood / Mask / Window / Template:



© Ramaiah University of Applied Sciences

Correlation & Convolution

- Correlation & Convolution are two closely related concepts used in linear spatial filtering.
- Correlation: It is a process of moving a filter mask over an image & computing the sum of products at each location.
- Convolution: Here, the mechanics are same, except that the filter is first rotated by 180˚.
- Correlation & Convolution are function of displacement. Correlation & Convolution are exactly same if the filter mask is symmetric.
- 1D correlation and convolution of a filter with a discrete unit impulse is shown below.

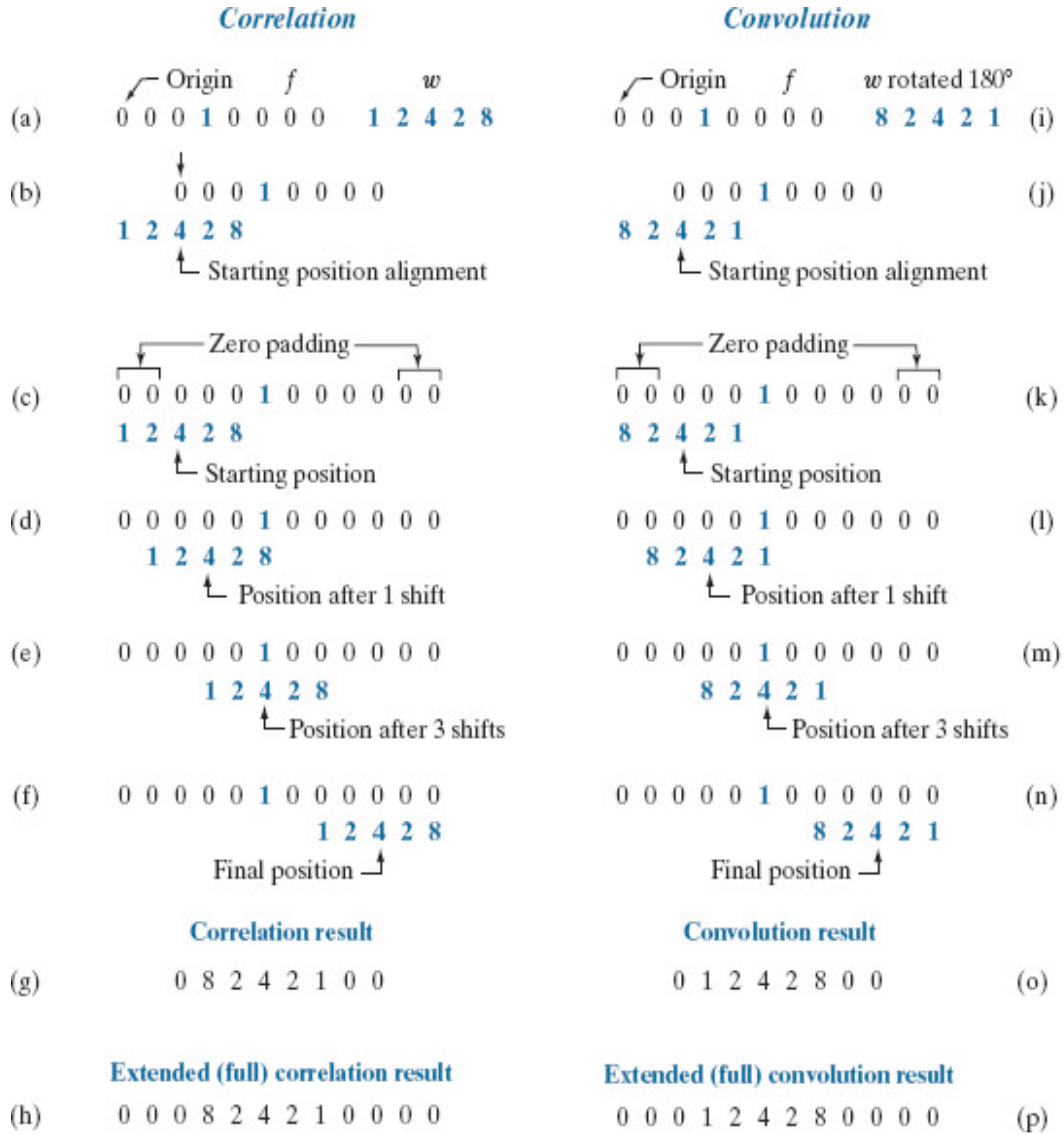
© Ramaiah University of Applied Sciences

Cont..

Illustration of 1-D correlation and convolution of a kernel, w, with a function f consisting of a discrete unit impulse.

Note that correlation and convolution are functions of the variable x, which acts to **displace** one function with respect to the other.

For the extended correlation and convolution results, the starting configuration places the rightmost element of the kernel to be coincident with the origin of f. Additional padding must be used.



© Ramaiah University of Applied Sciences

Correlation & Convolution

- Correlation is a function of displacement of the filter.
- Correlating a filter w with a function that contains all ‘0’ & single ‘1’ yields a 180˚ rotated copy of w.
- Correlating a function with discrete unit impulse yields a rotated (time inverted) version of the function.
- Convolving a function with a unit impulse yields the same function.
- Thus, to perform convolution all we have to do is rotate one function by 180˚ & perform same operation as in correlation.

© Ramaiah University of Applied Sciences

Origin $f(x, y)$

$$
\begin{array}{ccccc}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{array}
\qquad
\begin{array}{ccc}
\multicolumn{3}{c}{w(x, y)} \\
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{array}
$$

(a)

FIGURE 3.30 Correlation (middle row) and convolution (last row) of a 2-D filter with a 2-D discrete, unit impulse. The 0s are shown in gray to simplify visual analysis.

Correlation & Convolution

- Summarizing in equation form, we have that

- The **Correlation** of a filter w(x, y) of size mxn with an image f(x, y)is given by:

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

- The **Convolution** of w(x, y) and f(x, y) is given by:

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

© Ramaiah University of Applied Sciences

Vector representation of Linear Filtering

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

FIGURE 3.31 Another representation of a general 3×3 filter mask.

- $R = w_1 z_1 + w_2 z_2 + \ldots.+ w_9 z_9$

  $= \sum_{k=1}^{9} w_k z_k$

  $= w^T z$

- Where, w & z are 9-dimensional vectors formed from coefficients of the mask & image intensities encompassed by the mask, resp.

© Ramaiah University of Applied Sciences

Generating Spatial Filter Masks

1. Average Mean Filter

- The average value at any location (x, y) in the image is the sum of the nine intensity values in the 3 x 3 neighborhood centered on (x, y) divided by 9.
- If z_i, i = 1, 2, …, 9 denote these intensities, then the average is:
- $R = \frac{1}{9}\sum_{i=1}^{9} z_i$

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

a b

FIGURE 3.32 Two 3 × 3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

© Ramaiah University of Applied Sciences

Generating Spatial Filter Masks

- General implementation for filtering an M x N image with a weighted average filter of size m x n is given by:

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

2. Exponential Filter

- Some applications have a continuous function of 2 variables.
- E.g. Gaussian function Spatial filter mask has the basic form:

$$h(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- where, σ is standard deviation

© Ramaiah University of Applied Sciences

Smoothing Spatial Filters

Smoothing filters are used for blurring and for noise reduction

Blurring is used in removal of small details and bridging of small gaps in lines or curves

Smoothing spatial filters include linear filters and nonlinear filters.

© Ramaiah University of Applied Sciences

Spatial smoothing and image approximation

- Spatial smoothing may be viewed as a process for estimating the value of a pixel from its neighbours.
- What is the value that “best” approximates the intensity of a given pixel given the intensities of its neighbours?
- We have to define “best” by establishing a criterion.

© Ramaiah University of Applied Sciences

Generating Spatial Filter Masks

- In the resultant image the Low frequency region has remained unchanged.
- Sharp transition between 10 & 50 has changed from 10 to 23.33 to 36.66 and finally to 50.
- Thus, Sharp edges has become blurred.
- Best result when used over image corrupted by Gaussian noise.
- Other types of low pass averaging mask are:

$$\frac{1}{6}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Order-Statistic Filters

- These are non-linear spatial filters whose response is based on ordering (increasing / decreasing) the pixels contained in the image area encompassed by the filter.
- Then replacing the value of the center with the middle value determined by ranking result.
- E.g. Median filter, Max filter, Min Filter

i. Median Filter:

- Popular with certain random noise and impulse noise (Salt & Pepper noise).
- They provide excellent noise reduction
- Comparatively less blurring than linear smoothing filter of same size.

© Ramaiah University of Applied Sciences

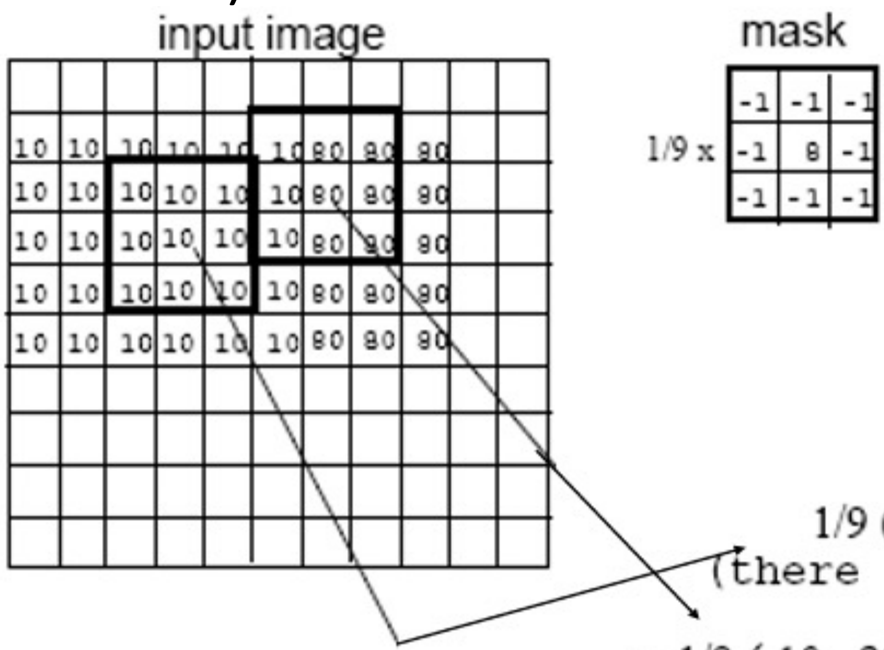
Spatial smoothing and image approximation

- **Background:** To highlight fine detail in an image or to enhance blurred detail
- **Applications:** electronic printing, medical imaging, industrial inspection, autonomous target detection (smart weapons)etc.,
- **Foundation (Blurring vs Sharpening):**
- Blurring/smoothing is performed by spatial averaging (equivalent to integration)
- Sharpening is performed by noting only the gray level changes in the image that is the differentiation

© Ramaiah University of Applied Sciences

Cont..

- Operation of Image Differentiation
  - Enhance edges and discontinuities (magnitude of output gray level >>0)
  - De-emphasize areas with slowly varying gray-level values (output gray level: 0)



$1/9\,(-10 - 10 - 10 - 10 + 80 - 10 - 10 - 10 - 10) = 0$
(there is no variation in the gray-levels)

$1/9\,(-10 - 80 - 80 - 10 + 640 - 80 - 10 - 80 - 80) = 210/9 > 0$
(there is variation in the gray-levels)

© Ramaiah University of Applied Sciences

Cont..

- Common Sharpening Filters
  - **Unsharp masking**
  - **High Boost filter**
  - **Derivative**

© Ramaiah University of Applied Sciences

Cont..

- Unsharp Masking
  - Obtain a sharp image by subtracting a lowpass filtered (i.e., smoothed) image from the original image:

Highpass = Original -Lowpass

(with some contrast enhancement)

© Ramaiah University of Applied Sciences

Cont..

- High Boost
  - Image sharpening emphasizes edges but details are lost.
  - Idea: amplify input image, then subtract a lowpass image.

Highboost = A Original – Lowpass

=(A-1) Original + Original – Lowpass

=(A-1) Original + Highpass

(A-1)

© Ramaiah University of Applied Sciences

Cont..

- High Boost
  - If A=1, the result is unsharp masking.
  - If A>1, part of the original image is added back to the high pass filtered image.

$$\text{Highboost} = (A-1)\ \text{Original} + \text{Highpass}$$

  - One way to implement high boost filtering is using these masks:

A>=1
w = 9A-1

-1	-1	-1
-1	w	-1
-1	-1	-1

A=2
w = 17

-1	-1	-1
-1	17	-1
-1	-1	-1

© Ramaiah University of Applied Sciences

Sharpening Spatial Filter

- The principal objective of sharpening is to highlight transitions in intensity.
- Applications: Electronics Printing, Medical Imaging, Industrial Inspection, Autonomous guidance in military systems, etc.
- Derivatives of a digital function are defined in terms of differences:
- **First order derivative** of 1D function f(x) is difference:

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

- **Second order derivative** of f(x) as the difference:

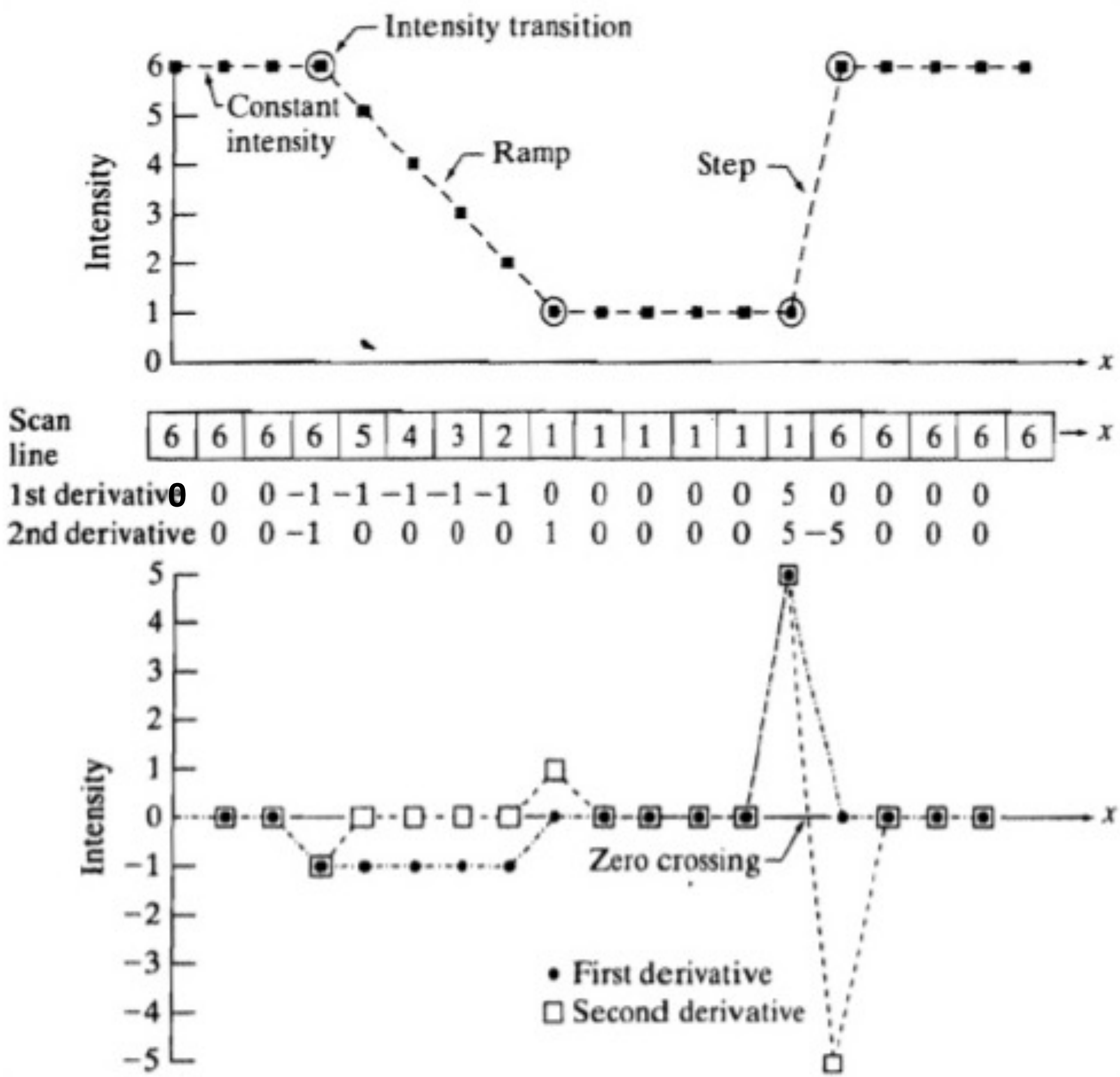
$$\frac{\delta^2 f}{\delta x^2} = f(x+1) + f(x-1) - 2f(x)$$

© Ramaiah University of Applied Sciences

Sharpening Spatial Filter

First Derivative:	Second derivative:
must be zero in areas of constant intensity.	must be zero in constant areas
must be nonzero at the onset of an intensity step / ramp.	Must be nonzero at the onset and end of an intensity step / ramp
Must be nonzero along ramps.	Must be zero along ramps of constant slope.

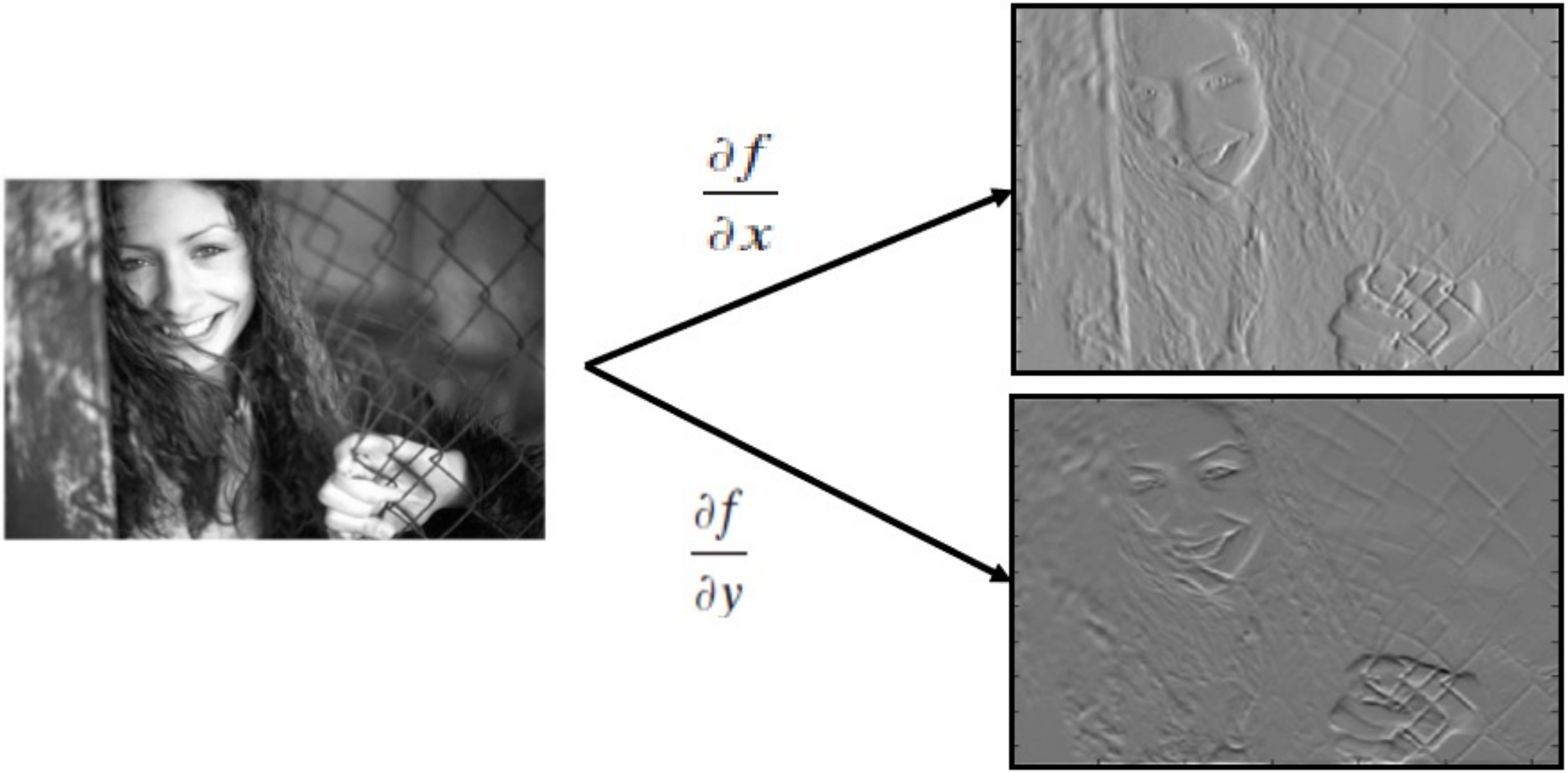
© Ramaiah University of Applied Sciences



© Ramaiah University of Applied Sciences

Example: visualize partial derivatives

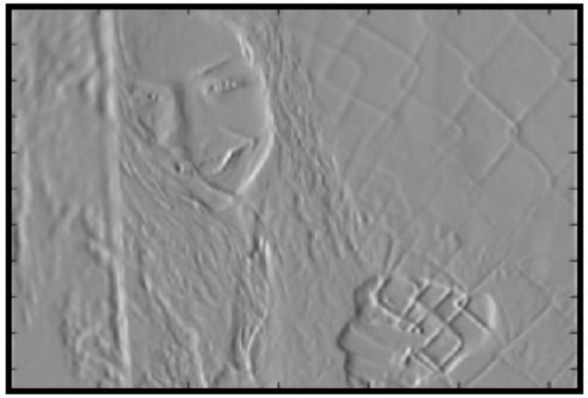
- The gradient can be visualized by mapping the values to [0, 255]



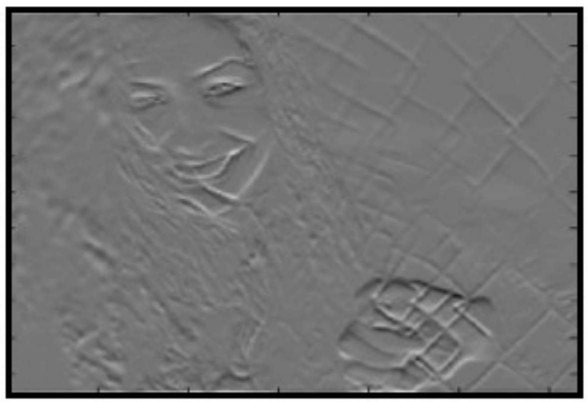
© Ramaiah University of Applied Sciences

Example: visualize partial derivatives

$$\frac{\partial f}{\partial x}$$



$$\frac{\partial f}{\partial y}$$



Gradient Magnitude

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

(isotropic, i.e.,
edges in all directions)

© Ramaiah University of Applied Sciences

Implement Gradient Using Masks

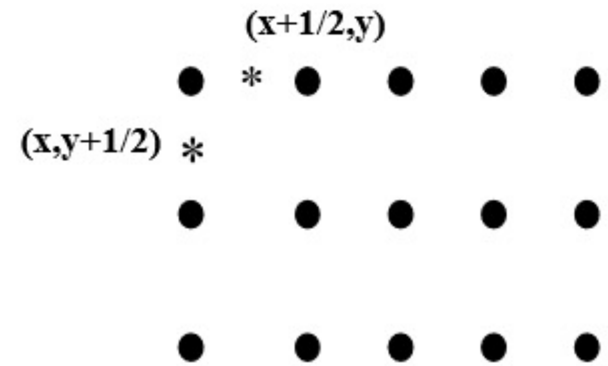
- We can implement $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ using masks:

-1	1

1
-1

$\frac{\partial f}{\partial x}$ good approximation at (x+1/2,y)

$\frac{\partial f}{\partial y}$ good approximation at (x,y+1/2)



$$\frac{\partial f}{\partial x} \quad f(x+1,y)-f(x,y)$$

$$\frac{\partial f}{\partial y} \quad f(x,y)-f(x,y+1)$$

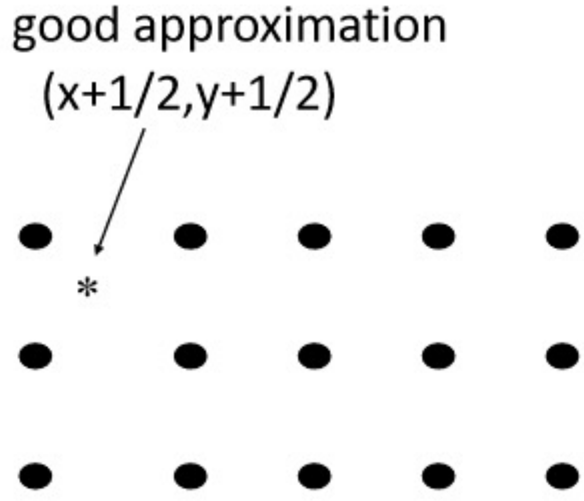
© Ramaiah University of Applied Sciences

Implement Gradient Using Masks

- A different approximation of the gradient:

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

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



© Ramaiah University of Applied Sciences

1st Derivative Filtering

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Roberts Cross Gradient

$$G_x = (z_9 - z_5) \quad G_y = (z_8 - z_6)$$

−1	0
0	1

0	−1
1	0

Sobel Gradient

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

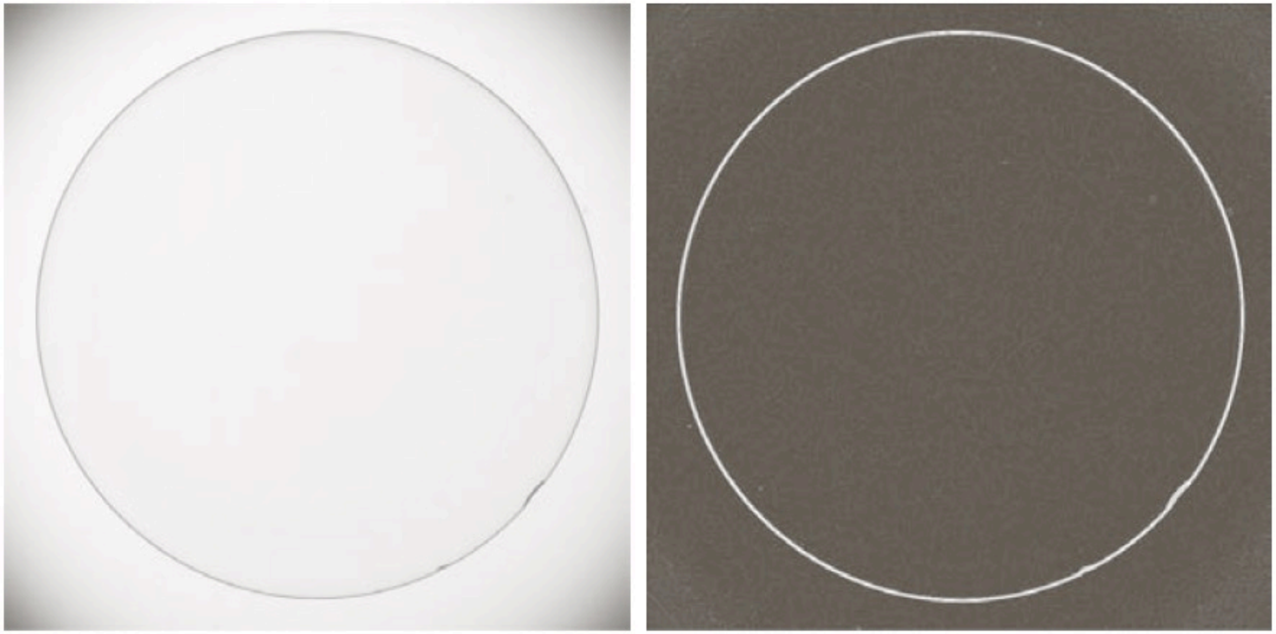
$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

© Ramaiah University of Applied Sciences

Example



a b

FIGURE 3.42 (a) Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Pete Sites, Perceptics Corporation.)

© Ramaiah University of Applied Sciences

Second Order Derivative

© Ramaiah University of Applied Sciences

The Laplacian

- To express the equation in discrete form,
- In x-direction,

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

- In y-direction,

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

- Thus, discrete Laplacian of 2 variables is,

$$\Delta^2 f(x,y) = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2}$$

$$= f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y) \quad (3\text{-}53)$$

© Ramaiah University of Applied Sciences

The Laplacian

- **Isotropic Filter:** They are rotation invariant.

- **Laplacian** is simplest Isotropic derivative operator. It is defined as:

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

- Laplacian is a **linear operator**.

© Ramaiah University of Applied Sciences

Cont..

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \; 90^{\circ} \; isotropic \qquad \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \; 45^{\circ} \; isotropic$$

© Ramaiah University of Applied Sciences

The Laplacian

(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 3.45 (a) Laplacian kernel used to implement Eq. (3-53). (b) Kernel used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other Laplacian kernels.

© Ramaiah University of Applied Sciences

The Laplacian

- If background features need to be recovered by still preserving the sharpening effect of the Laplacian, then
- The basic way to use the Laplacian for image sharpening is:
- $g(x, y) = f(x, y) + c[\Delta^2 f(x, y)]$

  Where,

  - f(x, y) – input image
  - g(x, y) – sharpened image
  - c – (-1) or (+1) for filters with negative & positive center resp.

© Ramaiah University of Applied Sciences

Unsharp Masking & Highboost Filtering

- The process of subtracting an unsharp (smoothed) version of an image from the original image is called unsharp masking.

- It consists of three steps:

  i. Blur the original image. $\bar{f}(x, y)$

  ii. Subtract the blurred image from the original (results in mask).

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y)$$

  iii. Add the mask to the original. where if, k = 1, unsharp masking, k > 1, highboost filtering, k < 1, de-emphasizes the contribution of unsharp mask.

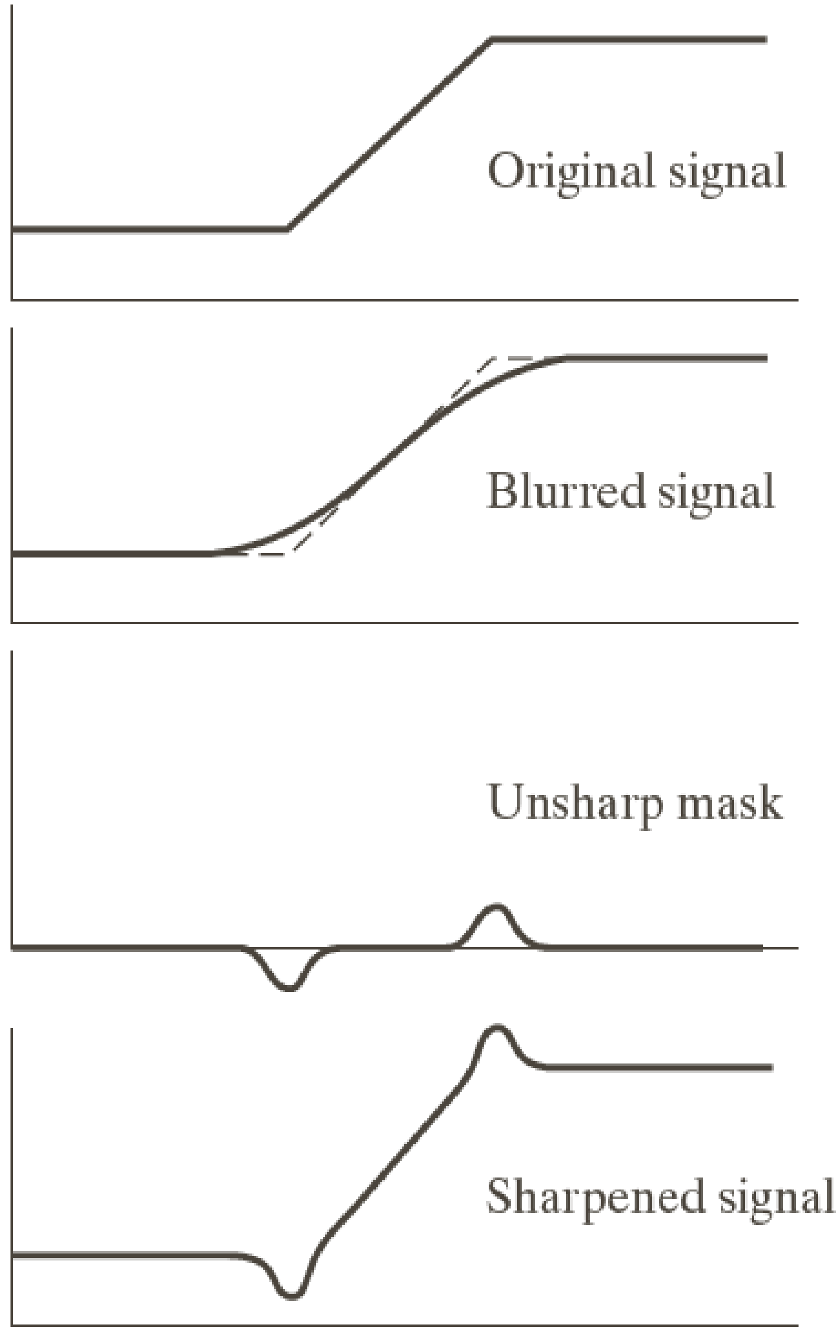
$$g(x, y) = f(x, y) + k * g_{mask}(x, y)$$

© Ramaiah University of Applied Sciences

Cont..

Mechanics of

Unsharp masking



a
b
c
d

FIGURE 3.39 1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

© Ramaiah University of Applied Sciences

Cont..

- Unsharp masking

a
b
c
d
e

FIGURE 3.40
(a) Original image.
(b) Result of blurring with a Gaussian filter.
(c) Unsharp mask. (d) Result of using unsharp masking.
(e) Result of using highboost filtering.

© Ramaiah University of Applied Sciences

Using First-Order Derivative (Gradient)

- First derivative are implemented using the magnitude of the gradient.
- For image f(x, y), the gradient of f at (x, y) is given by:

$$\nabla f = grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\delta f}{\delta x} \\ \dfrac{\delta f}{\delta y} \end{bmatrix}$$

- This vector points in the direction of the greatest rate of change of f at (x, y).
- Its magnitude is given by: $M(x, y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$
- Sometimes $M(x, y) \approx |g_x| + |g_y|$

© Ramaiah University of Applied Sciences

Cont..

- More Derivative Filters

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 3.41 A 3×3 region of an image (the zs are intensity values). (b)–(c) Roberts cross gradient operators. (d)–(e) Sobel operators. All the mask coefficients sum to zero, as expected of a derivative operator.

© Ramaiah University of Applied Sciences

Miscellaneous content Low Pass Median Filtering.

- Averaging Filter removes the noise by blurring till it is no longer seen.
- It blurs the edges too.
- Bigger the averaging mass more the blurring.
- Sometimes the image contains ‘salt & pepper noise’.
- If averaging filter is used then it will remove the noise at the cost of ruined edges.
- Thus a nonlinear filter Median filter is required.
- They are also called as order statistics filter since their response is based on ordering or ranking of pixels contained within the mask.

© Ramaiah University of Applied Sciences

Cont..

- Here we are going to discuss another method of dealing with images.
- This other method is known as convolution. Usually the black box(system) used for image processing is an LTI system or linear time invariant system.

- By linear we mean that such a system where output is always linear, neither log nor exponent or any other. And by time invariant we means that a system which remains same during time.
- So now we are going to use this third method.
  - It can be mathematically represented as two ways

$$\mathbf{g(x,y) = h(x,y) * f(x,y)}$$

© Ramaiah University of Applied Sciences

Cont..

- It can be explained as the “mask convolved with an image”.
- Or $\mathbf{g(x,y) = f(x,y) * h(x,y)}$

- It can be explained as “image convolved with mask”.
- There are two ways to represent this because the convolution operator(*) is commutative. The h(x,y) is the mask or filter.
- **What is mask?**
- Mask is also a signal. It can be represented by a two dimensional matrix. The mask is usually of the order of 1x1, 3x3, 5x5, 7x7 . A mask should always be in odd number, other wise you cannot find the mid of the mask. Why do we need to find the mid of the mask. The answer lies below, in topic of, how to perform convolution?

© Ramaiah University of Applied Sciences

Cont..

- **How to perform convolution?**
  - In order to perform convolution on an image, following steps should be taken.
  - Flip the mask (horizontally and vertically) only once
  - Slide the mask onto the image.
  - Multiply the corresponding elements and then add them
  - Repeat this procedure until all values of the image has been calculated.
- **Example of convolution**
  - Let's perform some convolution. Step 1 is to flip the mask.
- **Mask**
  - Let's take our mask to be this.

1	2	3
4	5	6
7	8	9

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Flipping the mask horizontally

3	2	1
6	5	4
9	8	7

1	2	3
4	5	6
7	8	9

Original Mask

  - Flipping the mask vertically

9	8	7
6	5	4
3	2	1

- Image
  - Let's consider an image to be like this

2	4	6
8	10	12
14	16	18

© Ramaiah University of Applied Sciences

Convolution

- Convolving mask over image. It is done in this way. Place the center of the mask at each element of an image. Multiply the corresponding elements and then add them and paste the result onto the element of the image on which you place the center of mask.

© Ramaiah University of Applied Sciences

Cont..

- The box in red color is the mask, and the values in the orange are the values of the mask. The black color box and values belong to the image. Now for the first pixel of the image, the value will be calculated as
- First pixel = (5*2) + (4*4) + (2*8) + (1*10)

  = 10 + 16 + 16 + 10 = 52
- Place 52 in the original image at the first index and repeat this procedure for each pixel of the image.

- **Why Convolution**
  - Convolution can achieve something, that the previous two methods of manipulating images can't achieve. Those include the blurring, sharpening, edge detection, noise reduction etc.

© Ramaiah University of Applied Sciences

Cont..

- **Example 1:** What is the effect of applying Laplacian filter on an image.
- **Mask**
  – Let's take Laplacian mask to be this.

1	1	1
1	- 8	1
1	1	1

- **Image**
  – Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let's take Laplacian mask to be this.

1	1	1
1	- 8	1
1	1	1

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Padded with Boundary elements

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 1: 1st Block :-

8x50 — 8x50

= 400—400

= 0

0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 2: 2nd Block :-
8x50 — 8x50
= 400—400
= 0

0	0				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 3: 3rd Block :-

8x50 — 8x50 = 0

0	0	0			

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

0	0	0	0		

Case 4: 4^{th} Block :-

8x50 — 8x50 = 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

0	0	0	0	0	

Case 5: 5th Block :-

8x50 — 8x50 = 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 6: 6th Block :-

8x50 — 8x50 = 0

0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 7: 7^{th} Block :-

8x50 — 8x50 = 0

0	0	0	0	0	0
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Similarly we filled for block 8th to 12th

Case 13: 13th Block :-

5x50 — 8x50 +3x100

= 250—400 +300 = 550 —400

= 150

0	0	0	0	0	0
0	0	0	0	0	0
150					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

0	0	0	0	0	0
0	0	0	0	0	0
150	150				

Case 14: 14th Block :-

5x50 — 8x50 +3x100

= 250—400 +300 = 550 —400

= 150

© Ramaiah University of Applied Sciences

Cont..

Image

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Mask

1	1	1
1	- 8	1
1	1	1

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150					

Similarly we filled for block 13th to 18th

Case 19: 19th Block :-

3x50 — 8x100 + 5x100

= 150 — 800 + 500 = 650 —800

= -150

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Case 20: 20th Block :-

3x50 — 8x100 + 5x100= 650 —400

= 150 — 800 + 500 = 650 —800

= -150

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Similarly we filled for block 20th to 24th

Case 25: 25th Block :-

8x100 — 8x100

= 800 —800

= 0

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0					

Similarly we filled for block 25^{th} to 30^{th}

Case 31: 31^{st} Block :-

8x100 — 8x100

= 800 —800

= 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Similarly we filled for block 32th to 36th

Case 36: 36th Block :-

8x100 — 8x100

= 800 —800

= 0

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

- **Example 2:** What is the effect of applying Laplacian filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

- 1	- 1	- 1
- 1	8	- 1
-1	-1	- 1

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let’s take Laplacian mask to be this.

- 1	- 1	- 1
- 1	8	- 1
- 1	- 1	- 1

- **Image**
  - Let’s consider an image to be like this

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Padded with Boundary elements

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	8	- 1
-1	- 1	- 1

Mask

Case 1: 1st Block :-

— 8x50 + 8x50

= — 400 + 400

= 0

0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	8	- 1
-1	- 1	- 1

Mask

Case 2: 2^{nd} Block :-

— 8x50 + 8x50

= — 400 + 400

= 0

0	0				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
-1	8	-1
- 1	- 1	- 1

Mask

Case 7: 7^{th} Block :-

— 8x50 — 8x50 = 0

0	0	0	0	0	0
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	-1	- 1
- 1	8	- 1
- 1	-1	- 1

Mask

Similarly we filled for block 8th to 12th

Case 13: 13th Block :-

$$- 5 \times 50 + 8 \times 50 - 3 \times 100$$
$$= - 250 + 400 - 300$$
$$= - 550 + 400$$
$$= - 150$$

0	0	0	0	0	0
0	0	0	0	0	0
- 150					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	8	- 1
- 1	- 1	- 1

Mask

Case 14: 14th Block :-

— 5x50 + 8x50 — 3x100

= — 250 + 400 — 300

= — 550 + 400

= — 150

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	8	- 1
- 1	- 1	- 1

Mask

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150	- 150	- 150	- 150	- 150
150					

Similarly we filled for block 13th to 18th

Case 19: 19th Block :-

— 3x50 + 8x100 — 5x100

= — 150 + 800 — 500 = — 650 + 800

= 150

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	8	- 1
- 1	- 1	- 1

Mask

Case 20: 20th Block :-

$-3\times50 + 8\times100 - 5\times100$

$= -150 + 800 - 500 = -650 + 800$

$= 150$

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150	- 150	- 150	- 150	- 150
150	150				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	- 8	- 1
- 1	- 1	- 1

Mask

Similarly we filled for block 20th to 24th

Case 25: 25th Block :-

— 8x100 + 8x100

= — 800 + 800

= 0

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150	- 150	- 150	- 150	- 150
150	150	150	150	150	150
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

- 1	- 1	- 1
- 1	- 8	- 1
- 1	- 1	- 1

Mask

Similarly we filled for block 25th to 30th

Case 31: 31st Block :-

— 8x100 + 8x100

= — 800 + 800

= 0

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

1	1	1
1	- 8	1
1	1	1

Mask

Similarly we filled for block 32th to 36th

Case 36: 36th Block :-

— 8x100 + 8x100

= — 800 + 800

= 0

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

- **Example 3:** What is the effect of applying Laplacian filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

0	1	0
1	- 4	1
0	1	0

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let’s take Laplacian mask to be this.

0	1	0
1	- 4	1
0	1	0

- **Image**
  - Let’s consider an image to be like this

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Padded with Boundary elements

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

0					

Case 1: 1^{st} Block :-

4x50 — 4x50

= 200 — 200

= 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

0	0				

Case 2: 2^{nd} Block :-

4x50 — 4x50

= 200 — 200

= 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

0	0	0	0	0	0
0					

Case 7: 7th Block :-

4x50 — 4x50 = 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Similarly we filled for block 8th to 12th

Case 13: 13th Block :-

3x50 — 4x50 + 1x100

= 150 — 200 + 100

= 250 — 200

= 50

0	0	0	0	0	0
0	0	0	0	0	0
50					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Case 14: 14th Block :-

3x50 — 4x50 + 1x100

= 150 — 200 + 100

= 250 — 200

= 50

0	0	0	0	0	0
0	0	0	0	0	0
50	50				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Similarly we filled for block 13th to 18th

Case 19: 19th Block :-

1x50 — 4x100 + 3x100

= 50 — 400 + 300 = 350 — 400

= — 50

0	0	0	0	0	0
0	0	0	0	0	0
50	50	50	50	50	50
- 50					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Case 20: 20th Block :-

1x50 — 4x100 + 3x100

= 50 — 400 + 300 = 350 — 400

= — 50

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150	- 150	- 150	- 150	- 150
150	150				

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Similarly we filled for block 20th to 24th

Case 25: 25th Block :-

4x100 — 4x100

= 400 — 400

= 0

0	0	0	0	0	0
0	0	0	0	0	0
- 150	- 150	- 150	- 150	- 150	- 150
150	150	150	150	150	150
0					

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0					

Similarly we filled for block 25th to 30th

Case 31: 31st Block :-

4x100 — 4x100

= 400 — 400

= 0

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

0	1	0
1	- 4	1
0	1	0

Mask

Similarly we filled for block 32th to 36th

Case 36: 36th Block :-

4x100 — 4x100

= 400 — 400

= 0

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

- **Example 4:** What is the effect of applying Laplacian filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

0	- 1	0
- 1	4	- 1
0	- 1	0

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

i. Padded with Boundary elements
ii. Padded with zero

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let’s take Laplacian mask to be this.

0	- 1	0
- 1	4	- 1
0	- 1	0

- **Image**
  - Let’s consider an image to be like this

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Padded with Boundary elements

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let's take Laplacian mask to be this.

0	- 1	0
- 1	4	- 1
0	- 1	0

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with zero

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 1: 1st Block :-

4x50 — 2x50

= 200—100

= 100

100					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

100	50				

Case 2: 2nd Block :-

4x50 — 3x50

= 200—150

= 50

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 3: 3rd Block :-

4x50 — 3x50

= 200—150

= 50

100	50	50			

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 4: 4th Block :-

4x50 — 3x50

= 200—150

= 50

100	50	50	50		

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 5: 5th Block :-

4x50 — 3x50

= 200—150

= 50

100	50	50	50	50	

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 6: 6th Block :-

4x50 — 2x50

= 200—100

= 100

100	50	50	50	50	100

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 7: 7th Block :-

4x50 — 3x50

= 200—150

= 50

100	50	50	50	50	100
50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 8: 8th Block :-

4x50 — 4x50

= 200—200

= 0

100	50	50	50	50	100
50	0				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Similarly 9th to 11th block values are same

Case 12: 12th Block :-

4x50 — 3x50

= 200—150

= 50

100	50	50	50	50	100
50	0	0	0	0	50

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 13: 13th Block :-

4x50 — 2x50 — 1x100

= 200—200

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 14: 14th Block :-

4x50 — 3x50 — 1x100

= 200—250

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0	-50				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 15: 15th Block :-

4x50 — 3x50 — 1x100

= 200—250

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50			

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 16: 16th Block :-

4x50 — 3x50 — 1x100

= 200—250

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50		

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 18: 18th Block :-

4x50 — 2x50 — 1x100

= 200—200

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 19: 19th Block :-

4x100 — 1x50 — 2x100

= 400—250

= 150

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 20: 20th Block :-

4x100 — 1x50 — 3x100

= 400—350

= 50

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 24: 24th Block :-

4x100 — 1x50 — 2x100

= 400—250

= 150

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 25: 25th Block :-

4x100 — 3x100

= 400—300

= 100

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 26: 26th Block :-

4x100 — 4x100

= 400—400

= 0

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100	0				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100	0	0	0	0	100

Similarly block 27 to 29 will have zero

Case 30: 30th Block :-

4x100 — 3x100

= 400—300

= 100

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 31: 31st Block :-

4x100 — 2x100

= 400—200

= 200

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100	0	0	0	0	100
200					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 32: 32th Block :-

4x100 — 3x100

= 400—300

= 100

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100	0	0	0	0	100
200	100				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

0	- 1	0
- 1	4	-1
0	- 1	0

Mask

Case 36: 36th Block :-

4x100 — 2x100

= 400—200

= 200

100	50	50	50	50	100
50	0	0	0	0	50
0	-50	-50	-50	-50	0
150	50	50	50	50	150
100	0	0	0	0	100
200	100	100	100	100	200

© Ramaiah University of Applied Sciences

Cont..

- **Example 5:** What is the effect of applying 3x3 Median filter on an image.

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Padded with Boundary elements

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

Case 1: 1st Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50					

Case 1: 1st Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50					

Case 2: 2nd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50				

Case 2: 2nd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50			

Case 3: 3rd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50		

Case 4: 4th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50

Case 5-6: 5th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50

Case 7: 7th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50					

Case 7: 7th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50					

Case 8: 8th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50				

Case 8: 8th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50				

Case 9: 9^{th} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50			

Case 9: 9th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50			

Case 10-12: 10th - 12th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50

Case 10-12: 10th - 12th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50

Case 13: 13th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50					

Case 13: 13th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50				

Case 14: 14th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50

Case 15-18: 15th - 18th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50

Case 19: 19th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100					

Case 19: 19th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100					

Case 25: 25th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100					

Case 31: 31^{st} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Image

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

Case 36: 36th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

Case 1: 1st Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	50	50	50	50	50	0
50	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

Case 1: 1st Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

0					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0					

Case 2: 2nd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50				

Case 2: 2nd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50			

Case 3: 3rd Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50		

Case 4: 4th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	

Case 5: 5^{th} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	

Case 6: 6th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	50	50	50	0	0	50
0	50	50	50	50	50	50	50
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0

Case 6: 6th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0

Case 7: 7th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	0	0	50	50	50	0	0
50	50	50	50	50	50	50	0
50	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0
50					

Case 7: 7^{th} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0
50					

Case 8: 8th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0
50	50				

Case 8: 8th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

0	50	50	50	50	0
50	50	50	50	50	

Case 12: 12^{th} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

Padded with
Zero elements

Image

0	0	0	0	0	0	0	0
0	50	50	50	50	0	0	0
0	50	50	50	50	50	50	50
0	50	50	50	50	50	50	50
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

0	50	50	50	50	0
50	50	50	50	50	50

Case 12: 12th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

Case 13: 13th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

0	50	50	50	50	0
50	50	50	50	50	50
50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with
Zero elements

Image

Case 19: 19th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

0	50	50	50	50	0
50	50	50	50	50	50
50	50	50	50	50	50
50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with Zero elements

Image

0	50	50	50	50	0
50	50	50	50	50	50
50	50	50	50	50	50
50	100				

Case 20: 20th Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Cont..

- **Example 6:** What is the effect of applying Sobel filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

- 1	- 2	- 1
0	0	0
1	2	1

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Example 7:** What is the effect of applying Sobel filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

- 1	0	1
- 2	0	2
- 1	0	1

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Example 8:** What is the effect of applying Robert Cross Gradient filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

- 1	0
0	1

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Mask**
  - Let’s take Laplacian mask to be this.

- 1	0
0	1

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Padded with zero

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 1: 1st Block :-

1x50 = 50

50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 2: 2^{nd} Block :-

1x50 = 50

50	50				

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 2: 2nd Block :-
1x50 = 50

50	50	50			

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

50	50	50	50	50	0

Case 6: 6th Block :-

0

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 7: 7^{th} Block :-
1x 50=50

50	50	50	50	50	0
50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 8: 8th Block :-

-1x50 + 1x 50= 0

50	50	50	50	50	0
50	0	0	0	0	

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 12: 12th Block :-

-1x50 = - 50

50	50	50	50	50	0
50	0	0	0	0	-50

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 13: 13th Block :-

1x50 = 50

50	50	50	50	50	0
50	0	0	0	0	-50
50					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 14: 14th Block :-

1x50 – 1x50= 0

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 19: 19th Block :-
1x100

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100					

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 20: 20^{th} Block :-

– 1x50 + 1x100= 50

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50				

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 24: 24th Block :-

– 1x50 = -50

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 25: 25th Block :-

1x100 = -50

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50
100					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 26: 26th Block :-

1x100 – 1x100 = 0

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50
100	0	0	0	0	-100

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 31: 31^{th} Block :-

= 0

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50
100	0	0	0	0	-100
0					

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 32: 32th Block :-

-1x100= -100

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50
100	0	0	0	0	-100
0	-100	-100	-100	-100	

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	50	50	50	50	50	50	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	100	100	100	100	100	100	0
0	0	0	0	0	0	0	0

Image

- 1	0
0	1

Mask

Case 36: 36th Block :-

-1x100= -100

50	50	50	50	50	0
50	0	0	0	0	-50
50	0	0	0	0	-50
100	50	50	50	50	-50
100	0	0	0	0	-100
0	-100	-100	-100	-100	-100

© Ramaiah University of Applied Sciences

Cont..

- **Example 9:** What is the effect of applying Robert Cross Gradient filter on an image.
- **Mask**
  - Let's take Laplacian mask to be this.

0	- 1
1	0

- **Image**
  - Let's consider an image to be like this

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

© Ramaiah University of Applied Sciences

Cont..

- **Example 10:** What is the effect of applying 3x3 Mean Filter filter on an image.

- **Image**
  - Let’s consider an image to be like this

50	25	30	40
15	50	25	20
35	15	50	20
10	30	20	40

© Ramaiah University of Applied Sciences

Cont..

- **Example 11:** What is the effect of applying 3x3 Median filter on an image.

- **Image**
  - Let's consider an image to be like this

5	*5*	*6*	*6*	*6*	*7*
5	*5*	*2*	*2*	*2*	*2*
5	*1*	*2*	*1*	*2*	*1*
1	*1*	*1*	*1*	*2*	*2*

© Ramaiah University of Applied Sciences

Cont..

- **Example 11:** What is the effect of applying 3x3 Median filter on an image.

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0					

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5				

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2			

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2		

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5					

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5	5				

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5	5	2			

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5	5	2	2		

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5	5	2	2	2	

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	0
5	5	2	2	2	2

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1					

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2				

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1			

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2		

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0					

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0	1				

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let’s consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0	1	1			

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0	1	1	1		

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	5	5	6	6	6	7	0
0	5	5	2	2	2	2	0
0	5	1	2	1	2	1	0
0	1	1	1	1	2	2	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0	1	1	1	1	

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0	0	0
0	*5*	*5*	*6*	*6*	*6*	*7*	0
0	*5*	*5*	*2*	*2*	*2*	*2*	0
0	*5*	*1*	*2*	*1*	*2*	*1*	0
0	*1*	*1*	*1*	*1*	*2*	*2*	0
0	0	0	0	0	0	0	0

0	5	2	2	2	2
5	5	2	2	2	2
1	2	1	2	2	2
0	1	1	1	1	0

© Ramaiah University of Applied Sciences

Cont..

- **Example 11:** What is the effect of applying 3x3 Median filter on an image.

- **Image**
  - Let’s consider an image to be like this

50	25	30	40
15	50	25	20
35	15	50	20
10	30	20	40

© Ramaiah University of Applied Sciences

Cont..

- **Image**
  - Let's consider an image to be like this

0	0	0	0	0	0
0	50	25	30	40	0
0	15	50	25	20	0
0	35	15	50	20	0
0	10	30	20	40	0
0	0	0	0	0	0

Padded with Boundary elements

Cont..

0	0	0	0	0	0
0	50	25	30	40	0
0	15	50	25	20	0
0	35	15	50	20	0
0	10	30	20	40	0
0	0	0	0	0	0

Image

Case 1: 1^{st} Block :-

- first sort all the pixel values from the surrounding neighborhood into numerical order
- Replacing the pixel being considered with the middle pixel value.

© Ramaiah University of Applied Sciences

Generating Spatial Filter Masks

1. Average Mean Filter

- The average value at any location (x, y) in the image is the sum of the nine intensity values in the 3 x 3 neighborhood centered on (x, y) divided by 9.
- If z_i, i = 1, 2, ..., 9 denote these intensities, then the average is:
- $R = \frac{1}{9}\sum_{i=1}^{9} z_i$

$$\frac{1}{9} \times \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

a b

FIGURE 3.32 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

© Ramaiah University of Applied Sciences

Cont..

- General implementation for filtering an M x N image with a weighted average filter of size m x n is given by:

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

© Ramaiah University of Applied Sciences

Cont..

- **Example 12** 8x8 Pseudo image with a single edge (High Frequency) of 10 & 50. Remove using a 3x3 size averaging mask.

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

- 8x8 Image

© Ramaiah University of Applied Sciences

Cont..

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\1 & 1 & 1\\1 & 1 & 1\end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0						
0	10	10	10	10	10	10	10	10
0	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0						
0	4.44	10	10	10	10	10	10	10
0	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50

$\frac{1}{9}$

0	0	0
0	10	10
0	10	10

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0					
0	4.44	6.66	10	10	10	10	10	10
0	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{bmatrix} 0 & 0 & 0 \\ 10 & 10 & 10 \\ 10 & 10 & 10 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0				
0	4.44	6.66	6.66	10	10	10	10	10
0	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline\end{array}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0			
0	4.44	6.66	6.66	6.66	10	10	10	10
0	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50
	50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline \end{array}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	10	0
	10	10	10	10	10	10	10	10	
	10	10	10	10	10	10	10	10	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline \end{array}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	10	0
	10	10	10	10	10	10	10	10	
	10	10	10	10	10	10	10	10	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline \end{array}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	6.66	0
0	6.66	10	10	10	10	10	10	6.66	0
0	15.55	10	10	10	10	10	10	10	0
0	50	50	50	50	50	50	50	50	0
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 0 & 10 & 10 \\ \hline 0 & 10 & 10 \\ \hline 0 & 50 & 50 \\ \hline\end{array}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	6.66	0
0	6.66	10	10	10	10	10	10	6.66	0
0	15.55	23.33	10	10	10	10	10	10	0
0	50	50	50	50	50	50	50	50	0
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	

$$\frac{1}{9}\begin{bmatrix} 10 & 10 & 10 \\ 10 & 10 & 10 \\ 50 & 50 & 50 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	6.66	0
0	6.66	10	10	10	10	10	10	6.66	0
0	15.55	23.33	23.33	23.33	23.33	23.33	23.33	15.55	0
0	24.44	36.66	36.66	36.66	36.66	36.66	36.66	24.44	0
0	33.33	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	
	50	50	50	50	50	50	50	50	

$$\frac{1}{9}\begin{bmatrix} 50 & 50 & 50 \\ 50 & 50 & 50 \\ 50 & 50 & 50 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

0	0	0	0	0	0	0	0	0	
0	4.44	6.66	6.66	6.66	6.66	6.66	6.66	4.44	0
0	6.66	10	10	10	10	10	10	6.66	0
0	6.66	10	10	10	10	10	10	6.66	0
0	15.55	23.33	23.33	23.33	23.33	23.33	23.33	15.55	0
0	24.44	36.66	36.66	36.66	36.66	36.66	36.66	24.44	0
0	33.33	50	50	50	50	50	50	33.33	0
0	33.33	50	50	50	50	50	50	33.33	0
0	22.22	33.33	33.33	33.33	33.33	33.33	33.33	22.22	0
	0	0	0	0	0	0	0	0	0

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 50 & 50 & 0 \\ \hline 50 & 50 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

© Ramaiah University of Applied Sciences

Cont..

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	23.33	23.33	23.33	23.33	23.33	23.33	10
50	36.66	36.66	36.66	36.66	36.66	36.66	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Cont..

- In the resultant image the Low frequency region has remained unchanged.
- Sharp transition between 10 & 50 has changed from 10 to 23.33 to 36.66 and finally to 50.
- Thus, Sharp edges has become blurred.
- Best result when used over image corrupted by Gaussian noise.
- Other types of low pass averaging mask are:

$$\frac{1}{6}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

© Ramaiah University of Applied Sciences

Thank You

© Ramaiah University of Applied Sciences