COMPUTER GRAPHICS
&
IMAGE PROCESSING

MODULE 5

GRAY LEVEL TRANSFORMATIONS

Namitha Ramachandran

Module - 5 (Image Enhancement in Spatial Domain and Image Segmentation)

Basic gray level transformation functions - Log transformations, Power-Law transformations, Contrast stretching. Histogram equalization. Basics of spatial filtering - Smoothing spatial filter- Linear and nonlinear filters, and Sharpening spatial filters-Gradient and Laplacian.

Fundamentals of Image Segmentation. Thresholding - Basics of Intensity thresholding and Global Thresholding. Region based Approach - Region Growing, Region Splitting and Merging. Edge Detection - Edge Operators- Sobel and Prewitt.

Basic gray level transformation functions

Enhancing an image provides better contrast and a more detailed image as compared to non enhanced image.
The transformation function has been given below

$$S=T(r)$$

where r is the pixels of the input image and s is the pixels of the output image. T is a transformation function that maps each value of r to each value of s.
Image enhancement can be done through gray level transformations

There are three basic gray level transformation.

- **Linear**
- **Log Transformation**
- **Power – law Transformation**

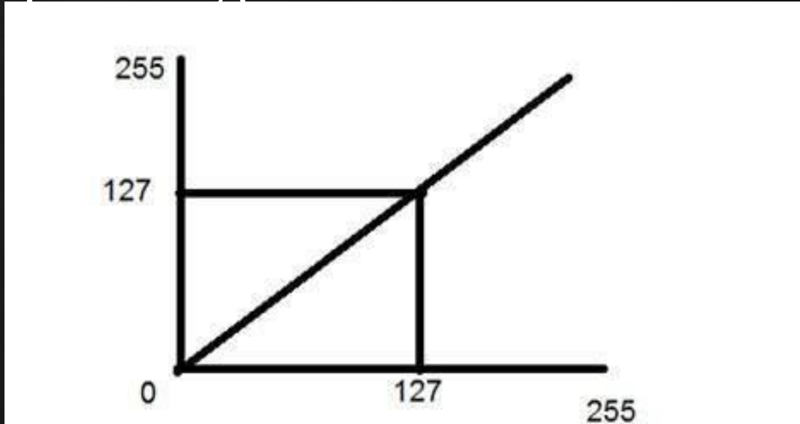
Linear transformation

First, we will look at the linear transformation. The linear transformation includes simple **identity and negative transformation**.

Identity transition

In this transition, each value of the input image is directly mapped to the other value of the output image.

That results in the same input image and output image.

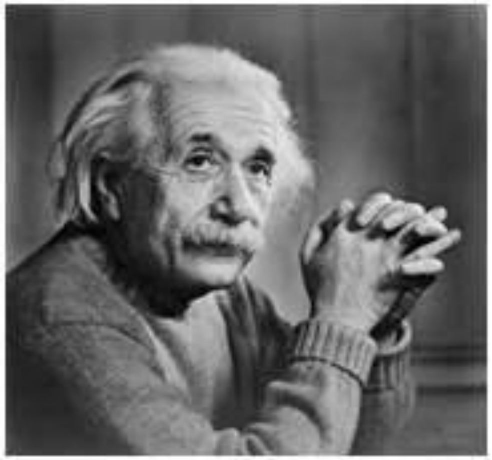


Negative transformation

The second linear transformation is negative transformation, which is an invert of identity transformation.

In negative transformation, **each value of the input image is subtracted from the L-1 and mapped onto the output image.**

Input Image



In this the following transition has been don

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

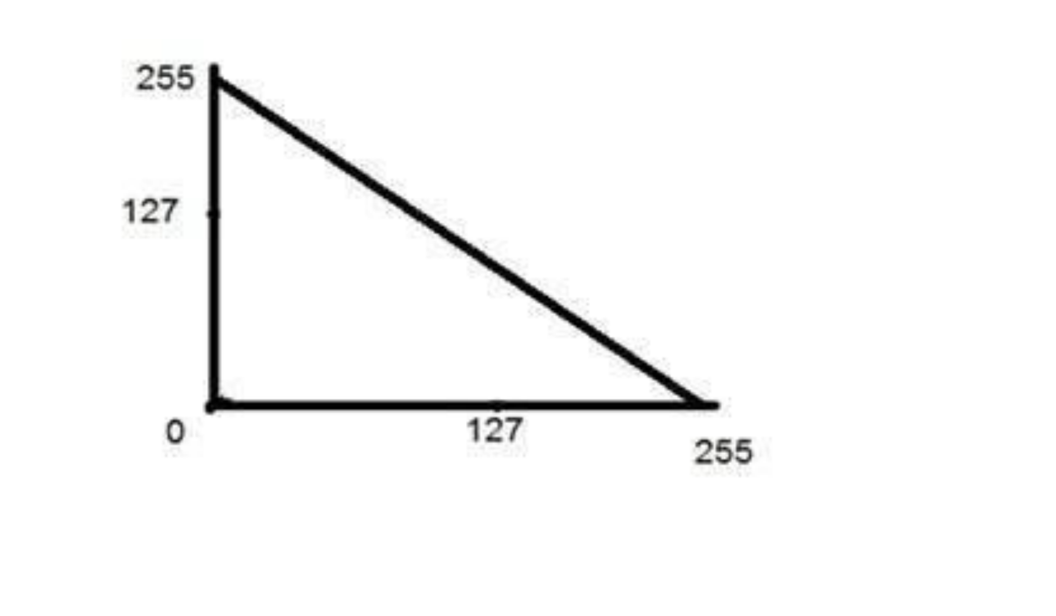
since the input image of Einstein is an 8 bpp image,

so the number of levels in this image are 256.

Putting 256 in the equation,

$$s = 255 - r$$

Namitha Ramachandran



Graph showing image negative of input with 256 different gray shades

Logarithmic transformations

Logarithmic transformation further contains two types of transformation.

Log transformation and inverse log transformation.

The log transformations can be defined by this formula

$$s = c \log(r + 1).$$

Where s and r are the pixel values of the output and the input image and c is a constant.

The value 1 is added to each of the pixel value of the input image because if there is a pixel intensity of 0 in the image, then log (0) is equal to infinity.

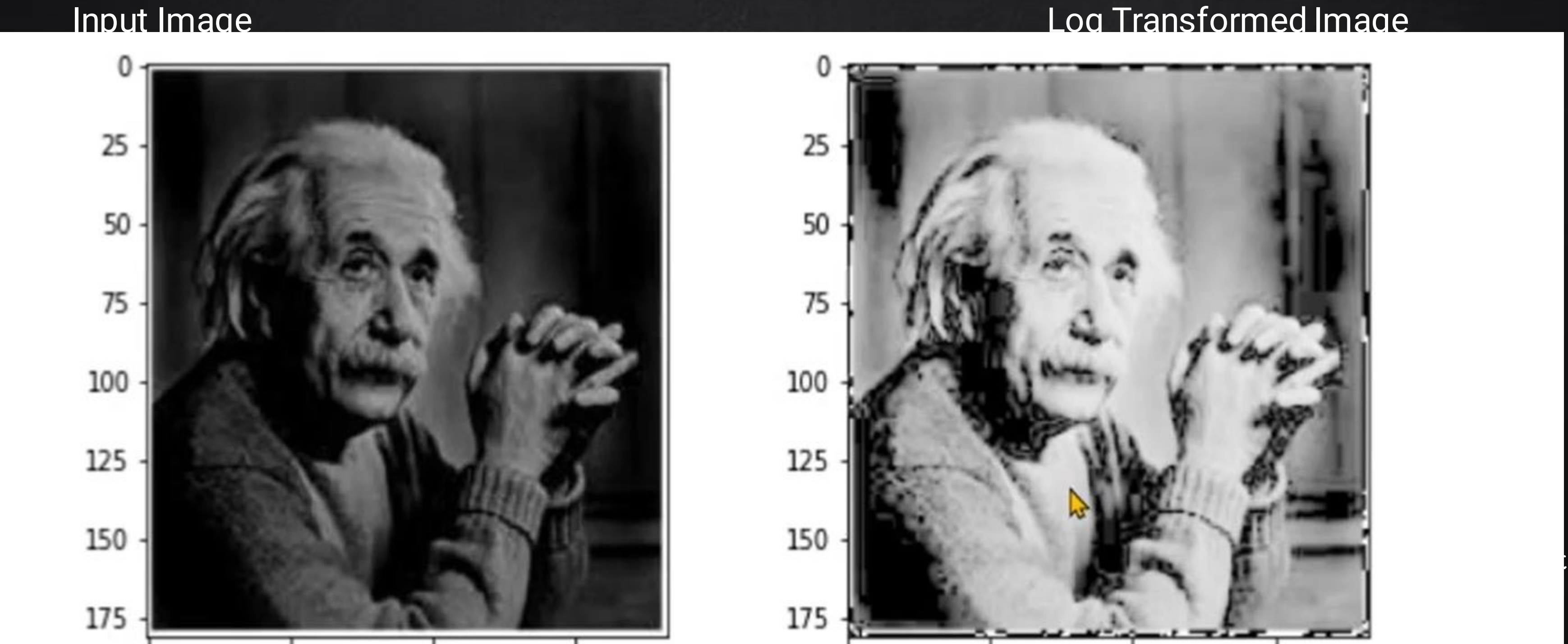
So 1 is added, to make the minimum value at least 1.

Namitha Ramachandran

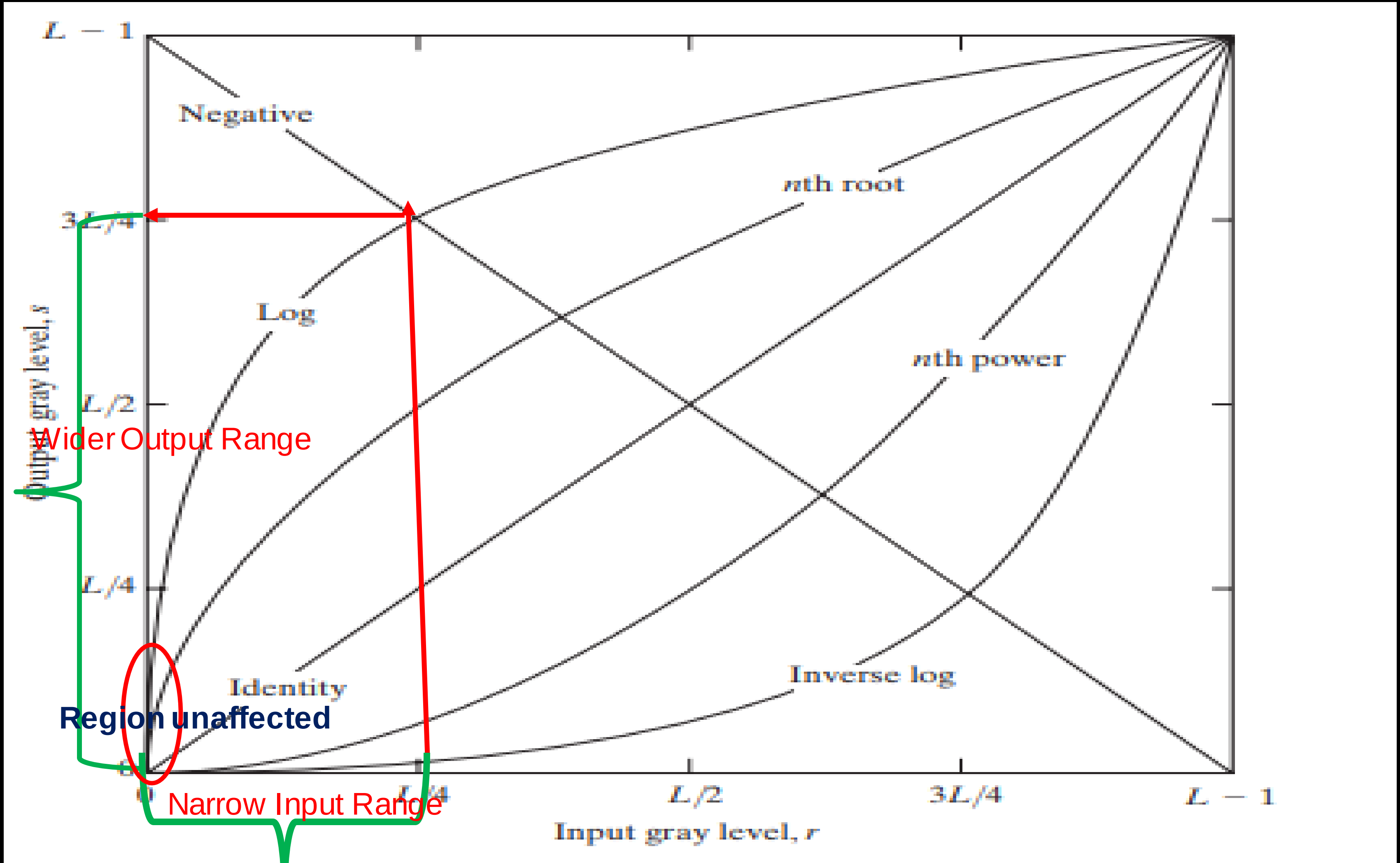
During log transformation, the **dark pixels in an image are expanded as compare to the higher pixel values**.

The higher pixel values are kind of compressed in log transformation.

Log transformation produces high contrast images .



HANDRAN

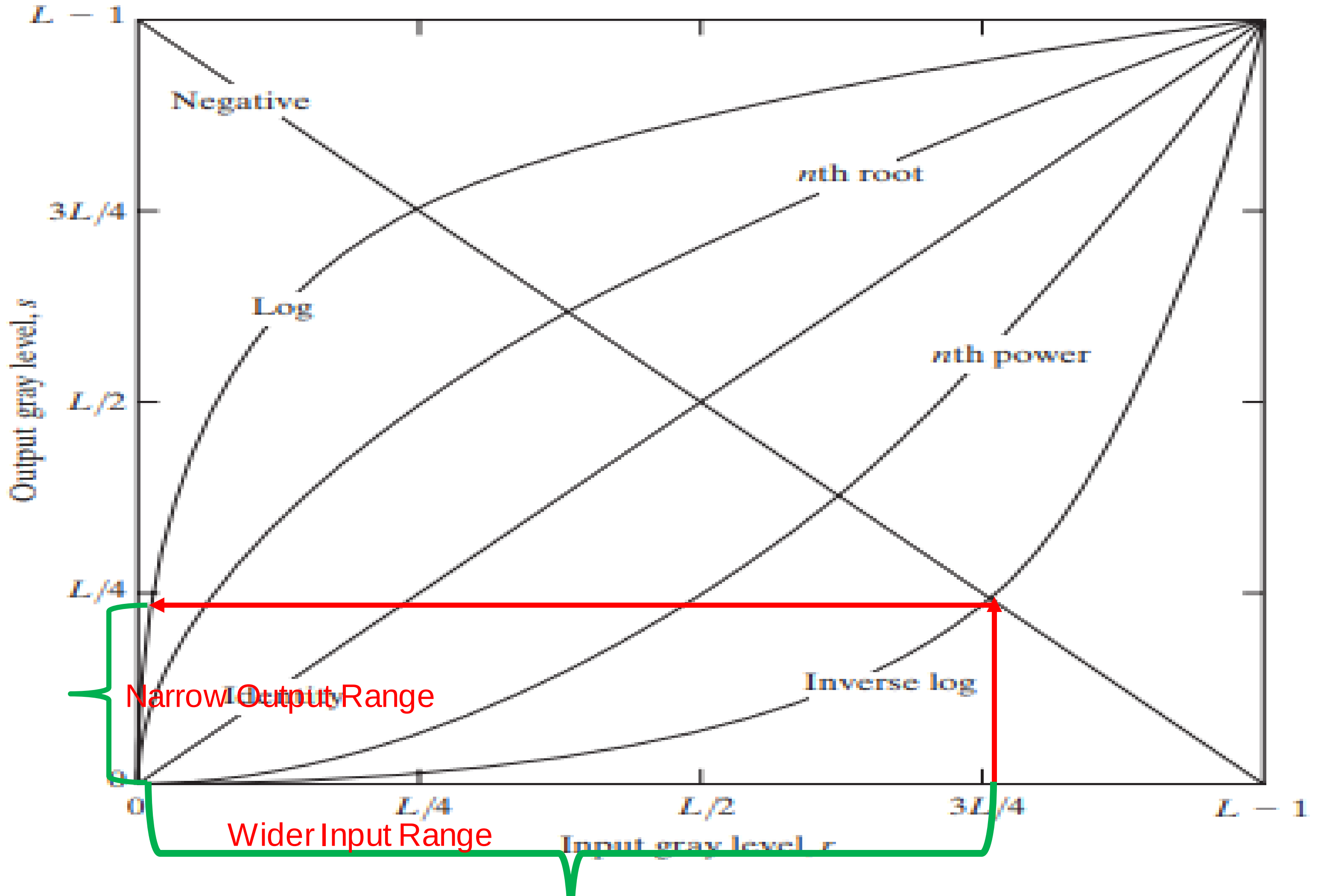


L − 1
3L/4
L/2
L/4
0
Output gray level, s
Negative
Log
nth root
nth power
Identity
Inverse log
L/4
L/2
3L/4
L − 1
Input gray level, r
Wider Output Range
Region unaffected
Narrow Input Range

Inverse log transformation

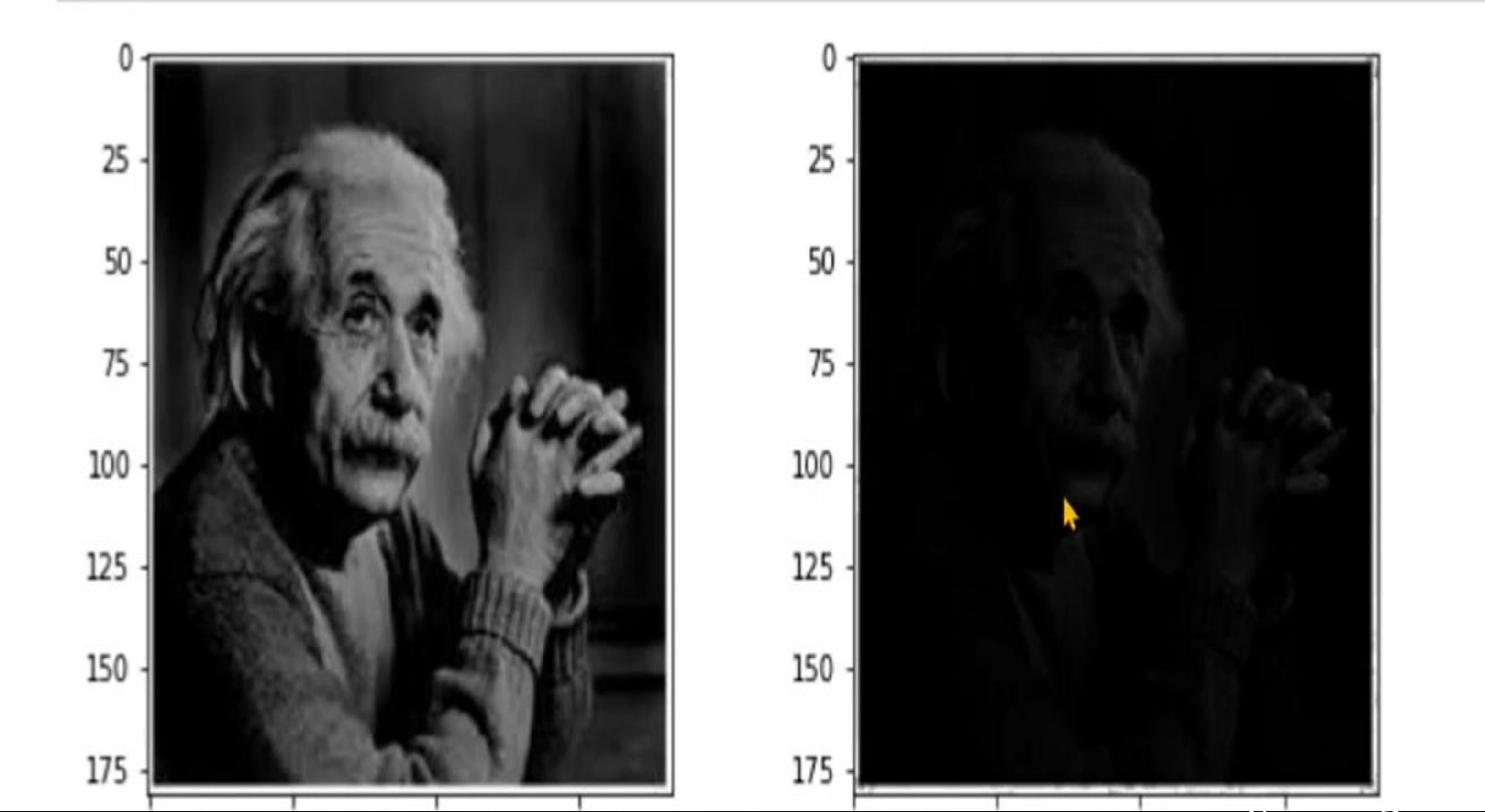
The inverse log transform is **opposite to the log transform**.

Higher input pixel value, lower output pixel value.
Low contrast output image.

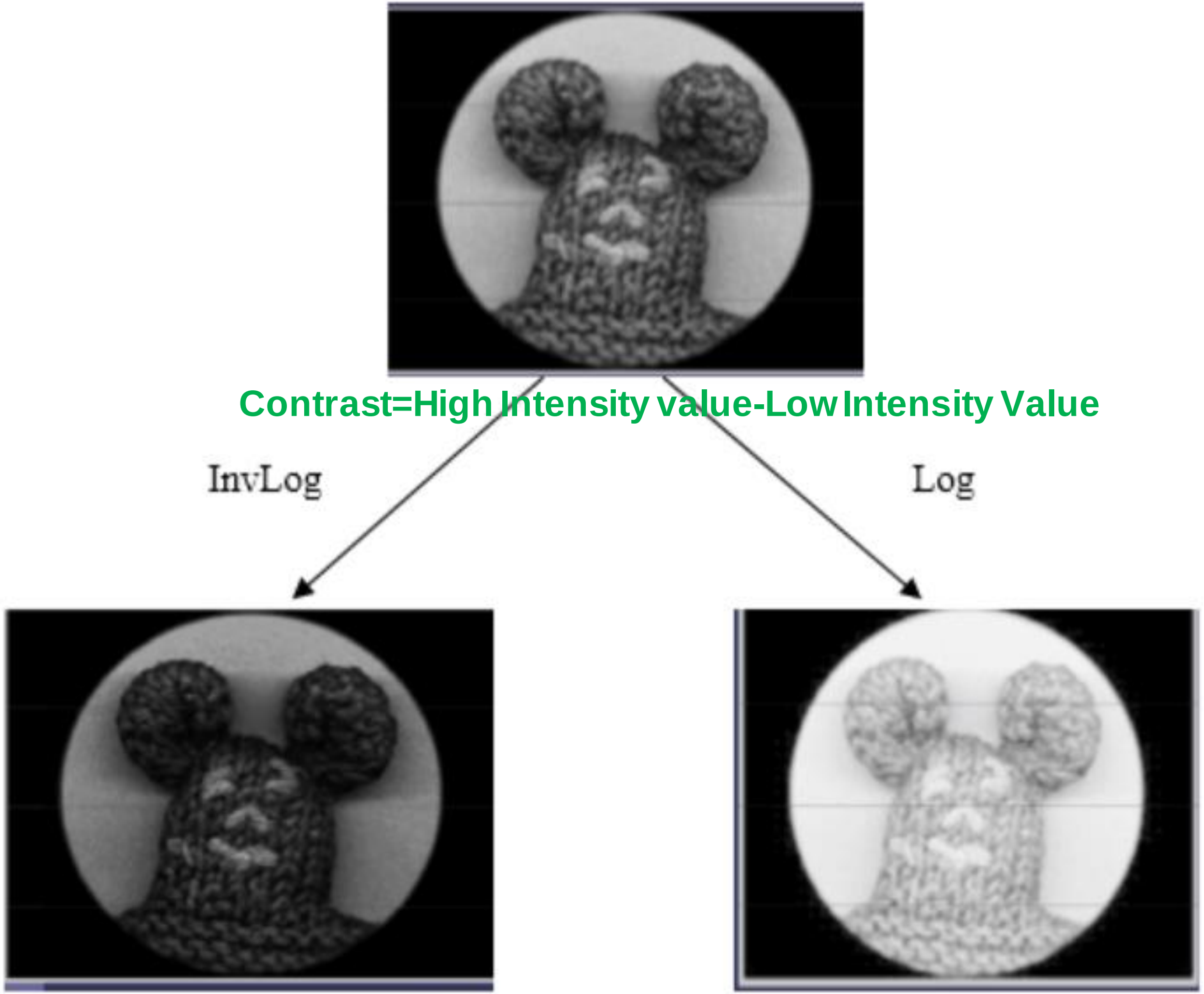


L − 1
3L/4
L/2
L/4
0
Output gray level, s
Negative
nth root
Log
nth power
Identity
Inverse log
Narrow Output Range
Wider Input Range
L/4
L/2
3L/4
L − 1
Input gray level, r

ACHANDRAN


0
25
50
75
100
125
150
175
0
25
50
75
100
125
150
175

Contrast=High Intensity value-Low Intensity Value



Power–Law (Gamma) transformations

Power Law Transformation is of two types of transformation **nth power transformation and nth root transformation**.

$$s=cr^\gamma$$

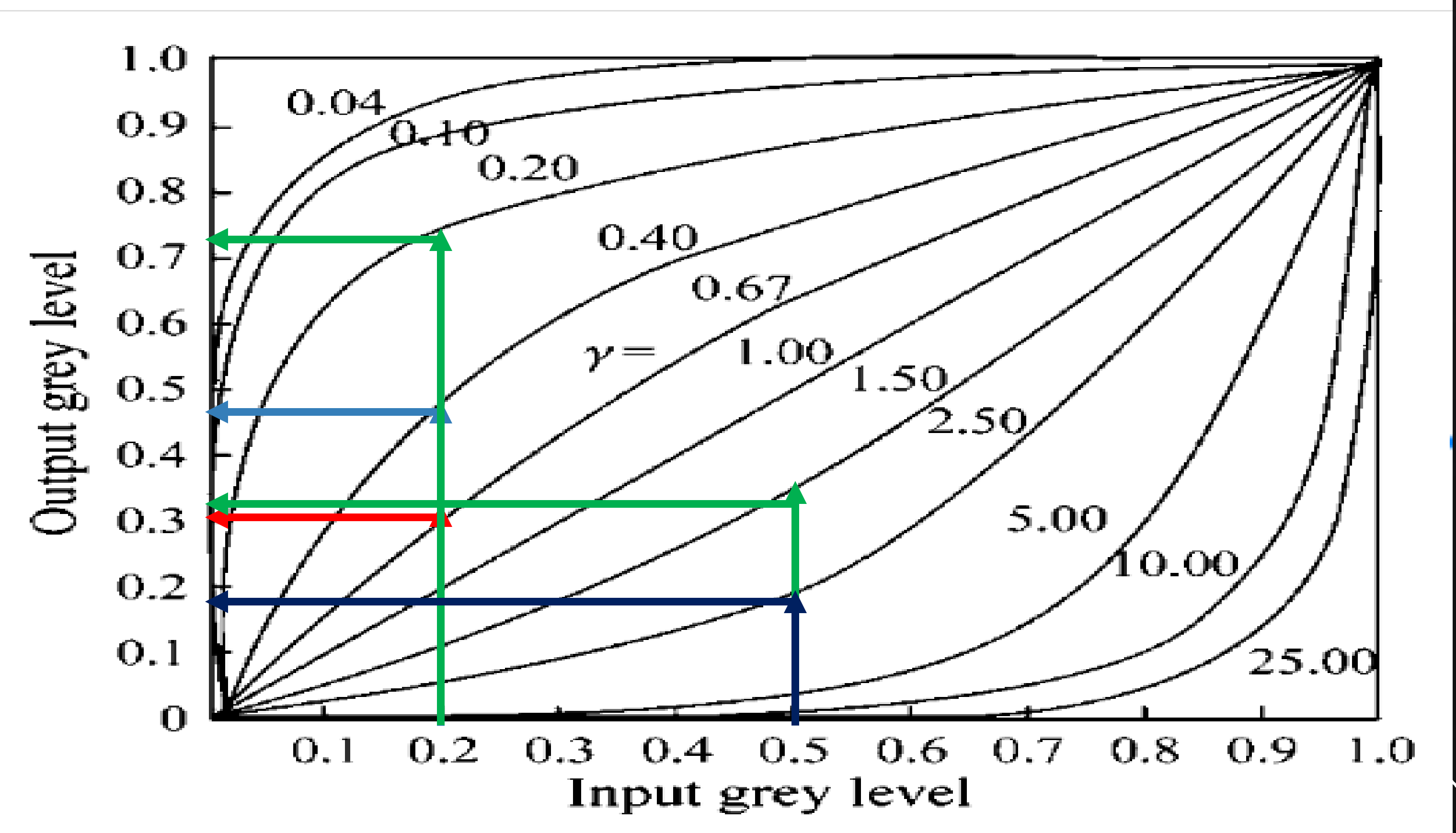
Variation in the value of γ varies the enhancement of the images. C is constant.

Different display devices / monitors have their own gamma correction, that's why they display their image at different intensity.

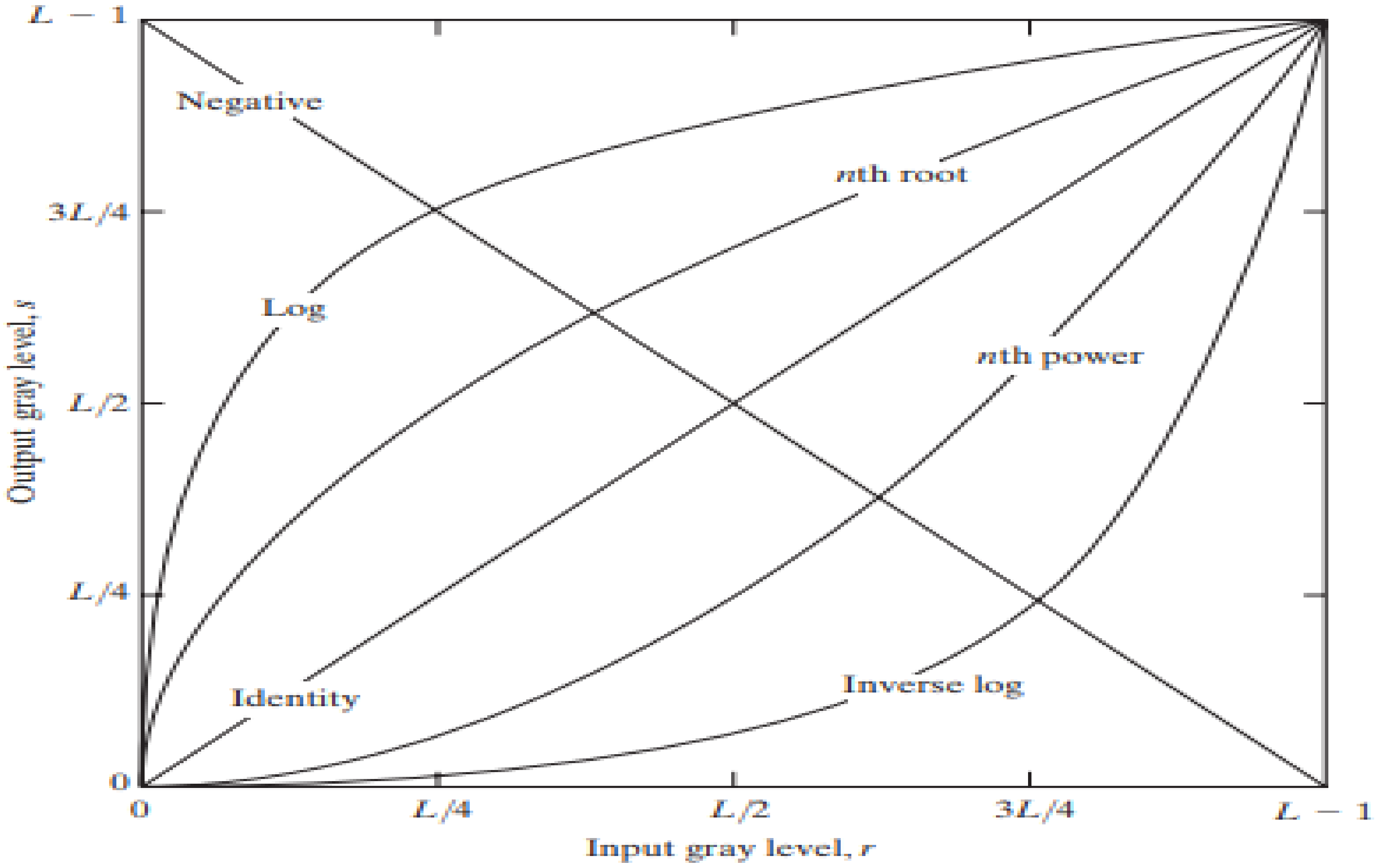
This type of transformation is used for enhancing images for a different types of display devices. **(Gamma Correction)**

The gamma of different display devices is different.

For example, Gamma of CRT lies in between 1.8 to 2.5, Power-law transformation is similar to log transformation, but for different gamma, value output will be different contrast images.



NDRAN



L − 1
3L/4
L/2
L/4
0
Output gray level, s
0
L/4
L/2
3L/4
L − 1
Input gray level, r
Negative
Log
nth root
nth power
Identity
Inverse log

COMPUTER GRAPHICS
&
IMAGE PROCESSING

MODULE 5

CONTRAST STRETCHING

Namitha Ramachandran

Piece-wise Linear Transformation

Piece-wise Linear Transformation is type of gray level transformation that is used for image enhancement.

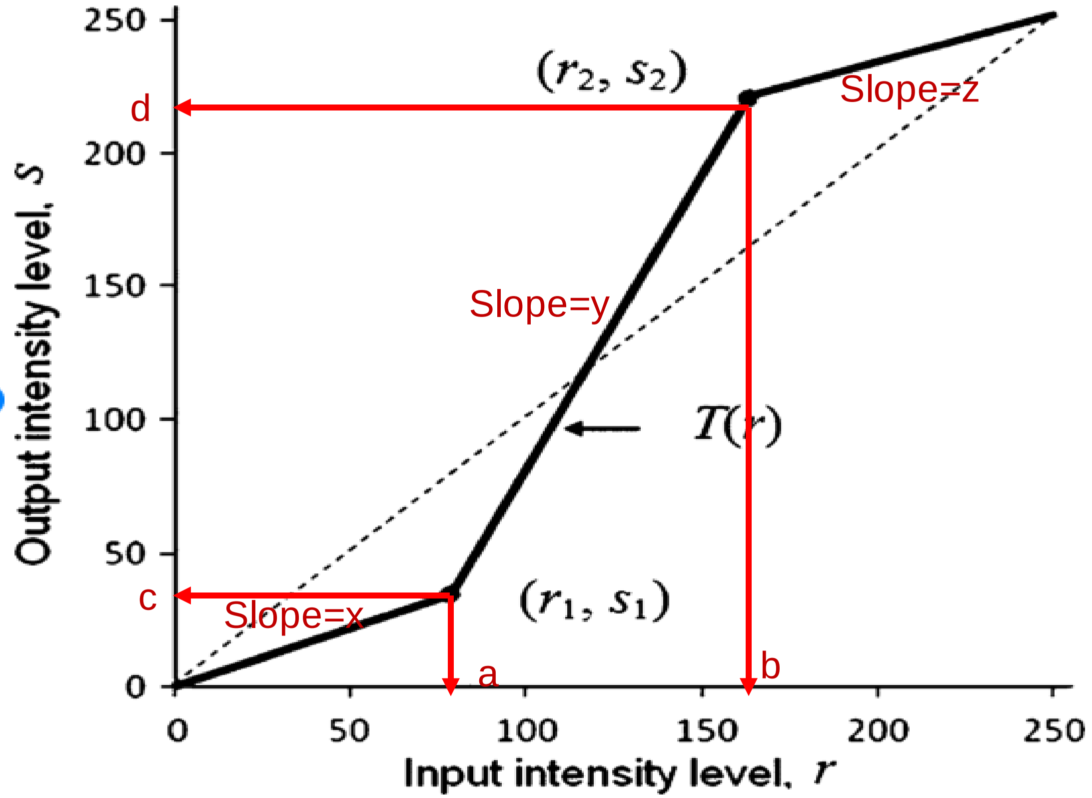
It is a spatial domain method. It is used for the manipulation of an image so that the result is more suitable than the original for a specific application.

Rather than using a well-defined mathematical function, we can use arbitrary user-defined transforms

Contrast Stretching:

Low contrast images occur often due to improper illumination or non-linearly or small dynamic range of an imaging sensor. It increases the dynamic range of grey levels in the image.

Contrast stretching is a process that expands the range of intensity levels in an image so that it spans the full intensity range of the recording medium or display device.



Explanatory illustration of contrast stretching transformation.

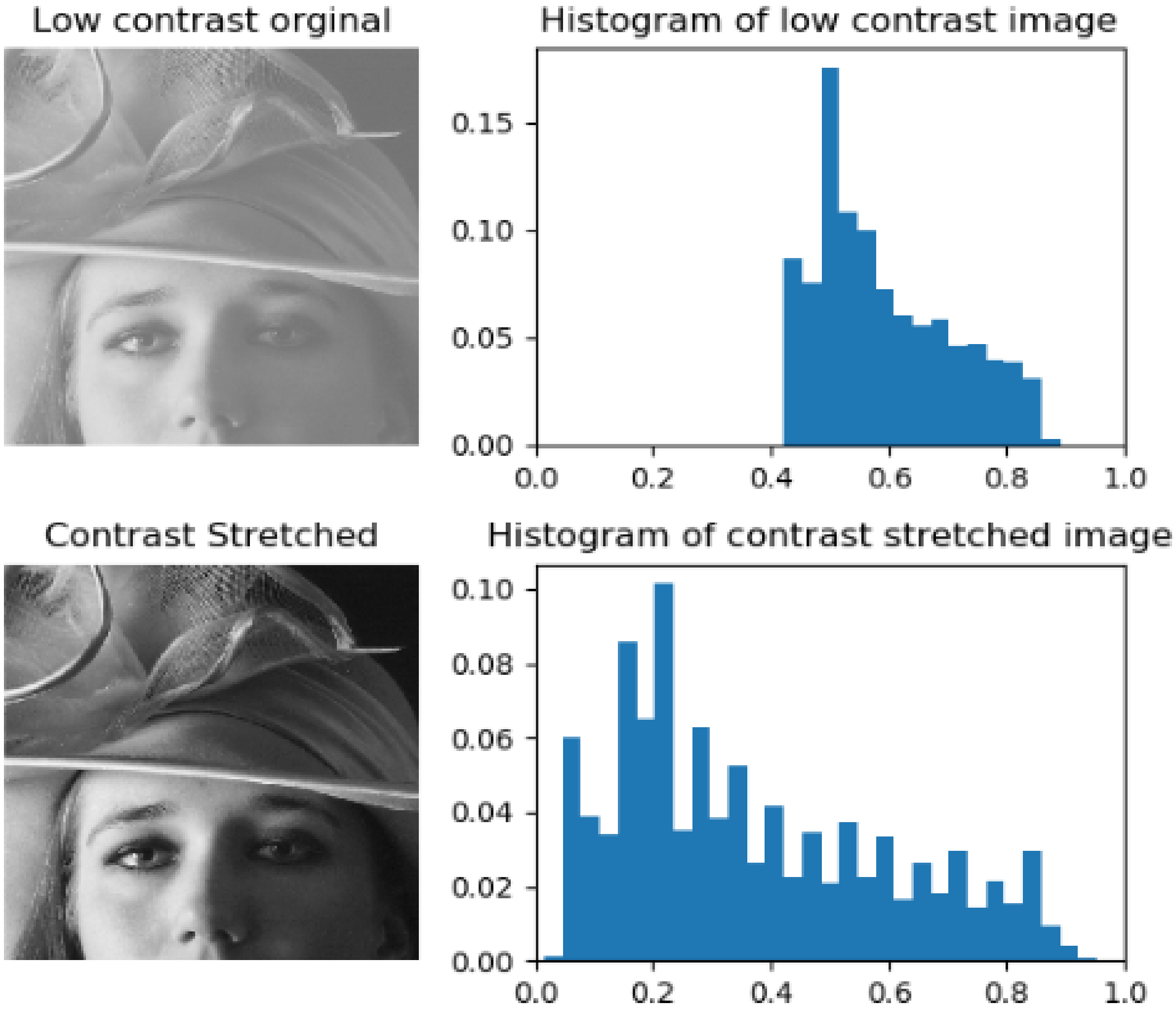
If slope=1 ,output image=input image

If slope>1 , output image is brighter than the input image

If slope<1, output image is darker than input image

$$S = \begin{cases} x \cdot r, & 0 <= r <= a \\ y.(r-a) + c, & a < r <= b \\ z.(r-b) + d, & b < r <= L-1 \end{cases}$$

Namitha Ramachandran


Low contrast orginal
Histogram of low contrast image
0.15
0.10
0.05
0.00
0.0
0.2
0.4
0.6
0.8
1.0
Contrast Stretched
Histogram of contrast stretched image
0.10
0.08
0.06
0.04
0.02
0.00
0.0
0.2
0.4
0.6
0.8
1.0

Intensity Level slicing

This technique is used to highlight a specific range of gray levels in a given image (thresholding).

Other levels can be suppressed or maintained – Useful for highlighting features in an image.

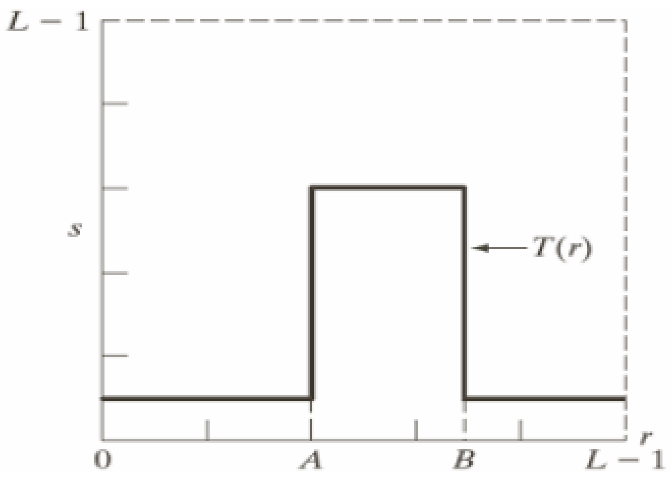
Two basic themes are:

One approach is to display a high value for all gray levels in the range of interest and a low value for all other gray levels.

The second approach, based on the transformation brightens the desired range of gray levels but preserves gray levels unchanged.

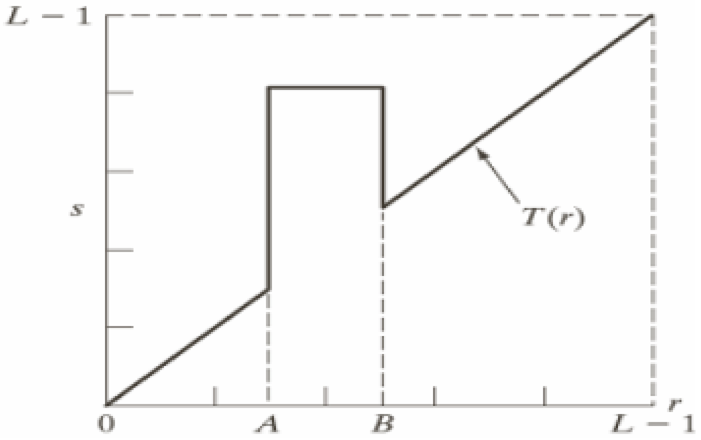
Highlighting a specific range of intensities in an image.

Approach 1

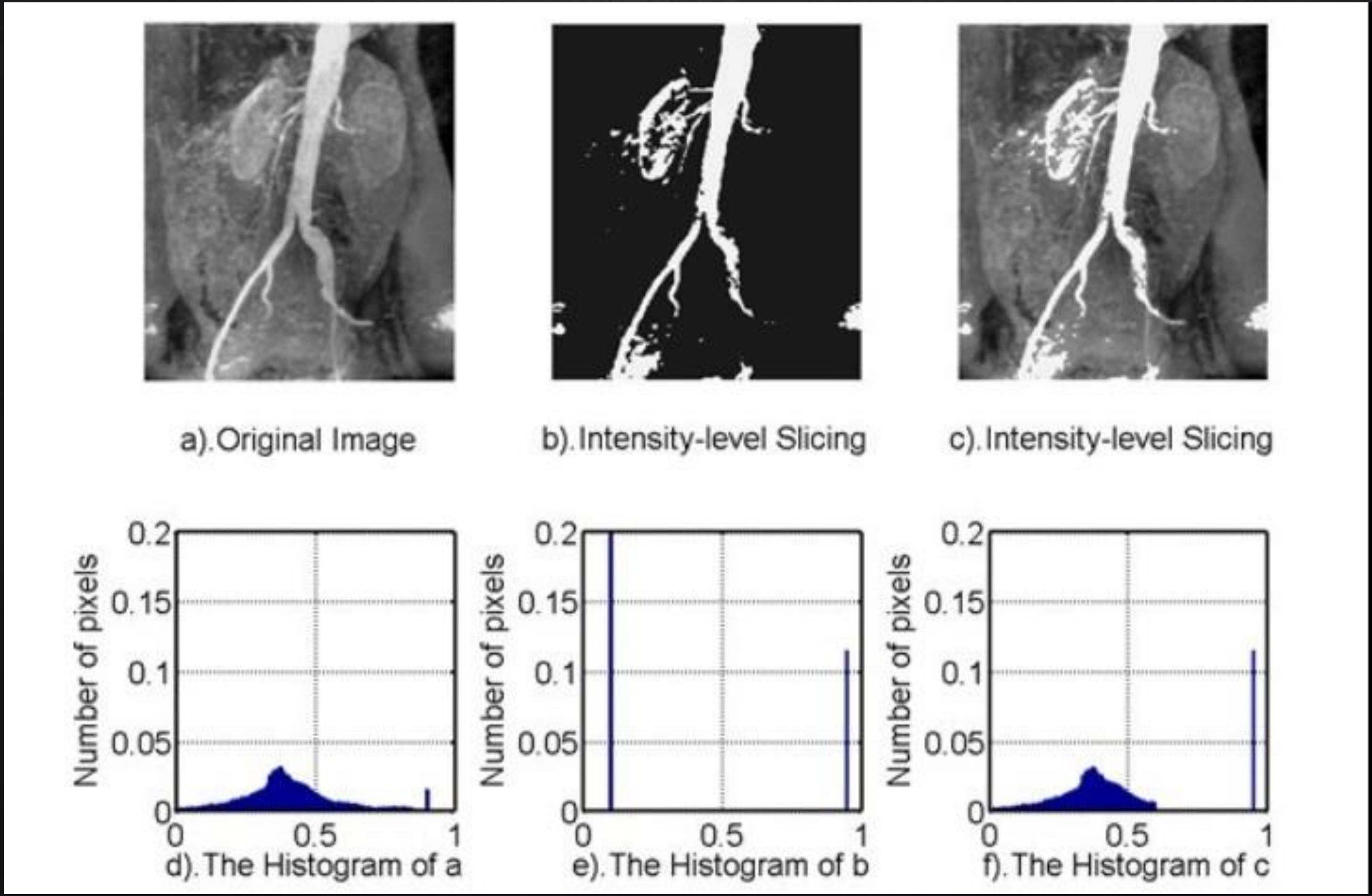


display in one value(e.g white) all the values in the range of interest , and in another (e.g black) all other intensities

Approach 2



Brightens or darkens the desired range of intensities but leaves all other intensity levels in the image unchanged


a).Original Image
b).Intensity-level Slicing
c).Intensity-level Slicing
Number of pixels
0.2
0.15
0.1
0.05
0
0.5
1
d).The Histogram of a
e).The Histogram of b
f).The Histogram of c

COMPUTER GRAPHICS & IMAGE PROCESSING

MODULE 5

Histogram processing

Histogram Equalization

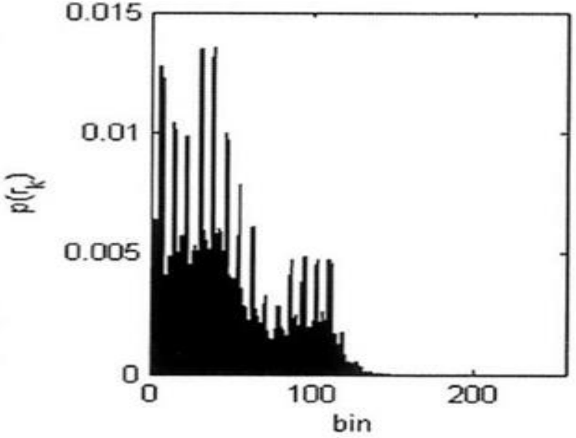
Namitha Ramachandran

Histogram Processing

In digital image processing, the histogram is used for the graphical representation of a digital image.

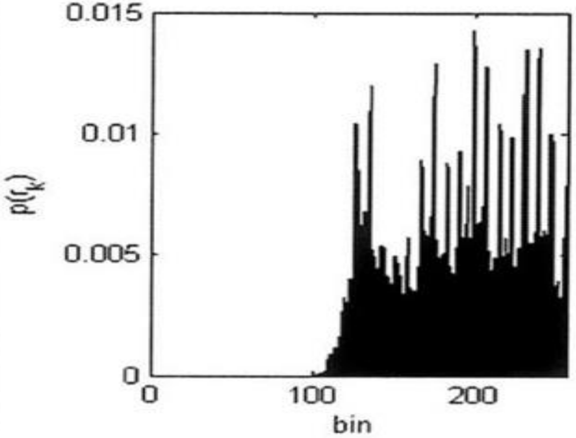
In a graph, the horizontal axis of the graph is used to represent tonal variations whereas the vertical axis is used to represent the number of pixels in that particular pixel.

(a)



(b)

(c)



(d)

ACHANDRAN

Applications of Histograms

It is used to **analyze** an image. Properties of an image can be predicted by the detailed study of the histogram.

The **brightness of the image can be adjusted** by having the details of its histogram.

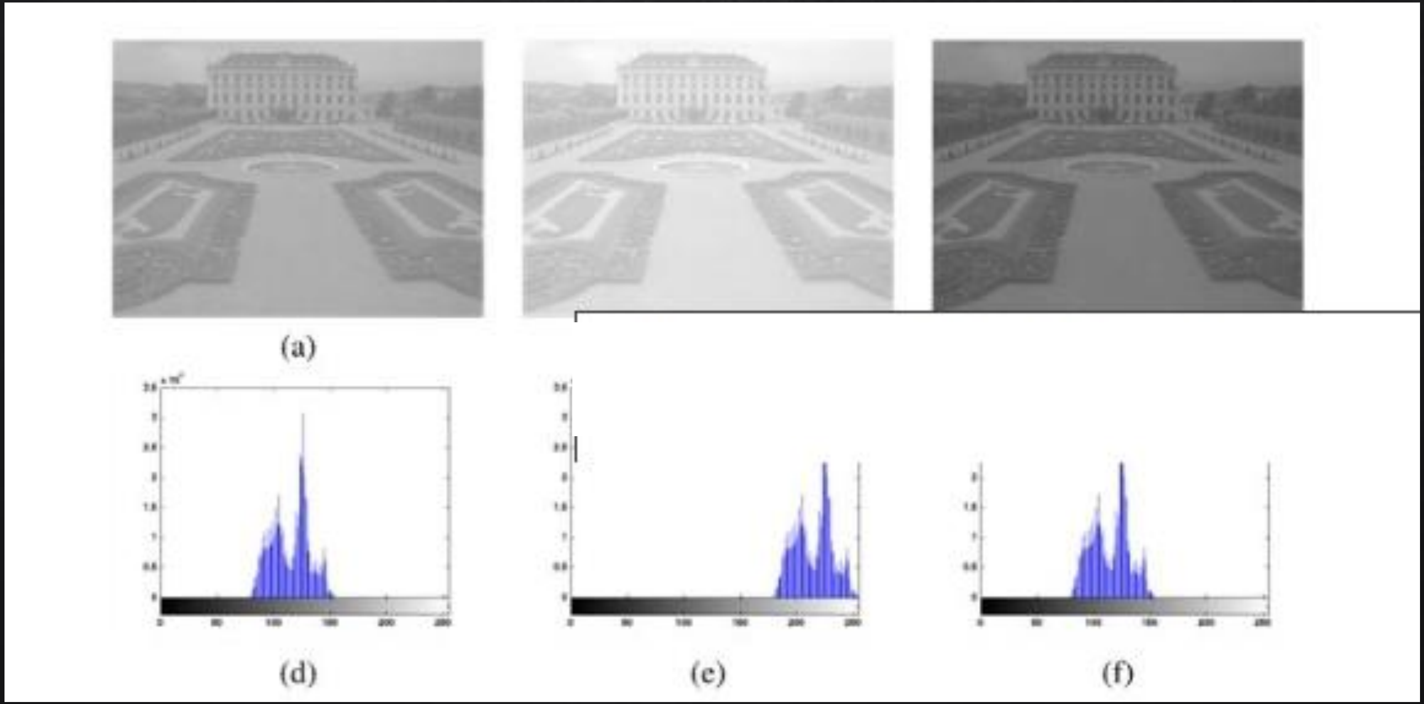
The **contrast of the image can be adjusted** according to the need by having details of the x-axis of a histogram.

It is used for **image equalization**. Gray level intensities are expanded along the x-axis to produce a high contrast image.

Histograms are used in **thresholding** as it improves the appearance of the image.

If we have input and output histogram of an image, we can determine which **type of transformation** is applied in the algorithm.

Histogram Processing Techniques



(a)

(d) (e) (f)

Namitha Ramachandran

Histogram Equalisation

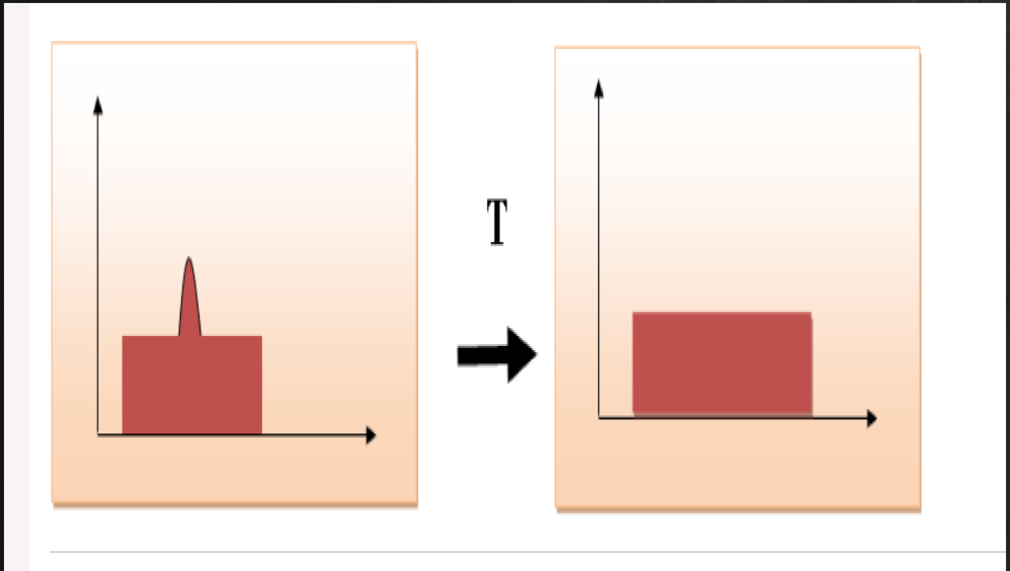
Histogram Equalization is a computer image processing technique used to **improve contrast in images**.

Histogram equalization is used for equalizing all the pixel values of an image, so that a **uniformly flattened histogram is produced**.

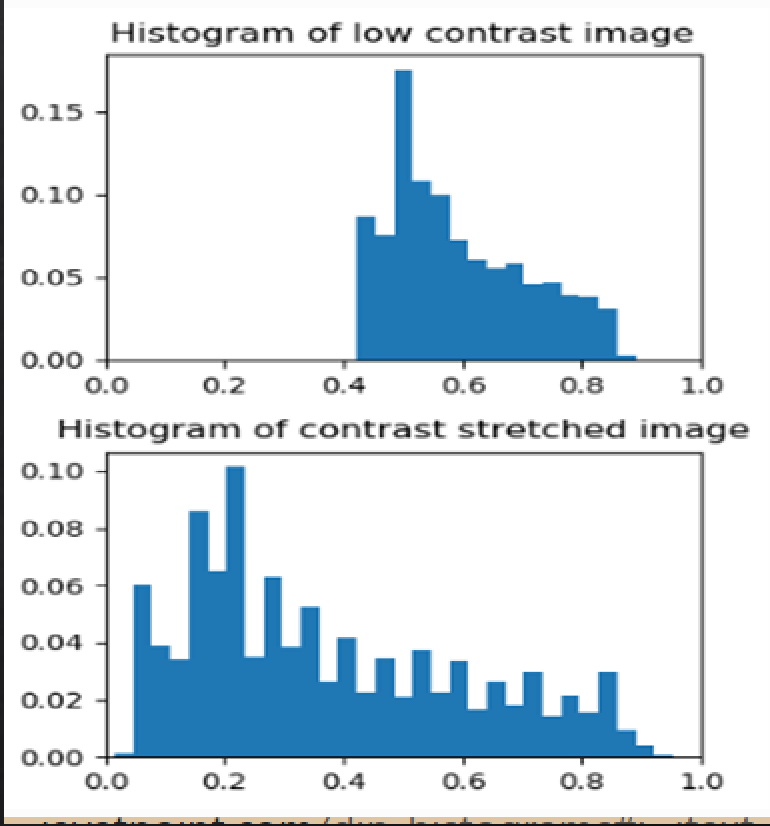
Histogram equalization increases the dynamic range of pixel values and makes an equal count of pixels at each level which produces a flat histogram with high contrast image.

While stretching a histogram, the shape of histogram remains the same whereas in Histogram equalization, the shape of histogram changes and it generates only one image.

Namitha Ramachandran



Histogram equalization



Contrast streaching

Histogram equalization of the following image

F(x,y)=

1	2	1	1	1
2	5	3	5	2
2	5	5	5	2
2	5	3	5	2
1	1	1	2	1

Input image

The maximum value of intensity in the image is 5

$2^1=2 \rightarrow (0,1)$

$2^2=4 \rightarrow (0,1,2,3)$

$2^3=8 \rightarrow (0,1,2,3,4,5,6,7)$

Step 1

Gray levels(rk)	0	1	2	3	4	5	6	7
No. of pixels(nk)	0	8	8	2	0	7	0	0

Namitha Ramachandran

Gray levels(rk)	0	1	2	3	4	5	6	7
No. of pixels(nk)	0	8	8	2	0	7	0	0

Step 2: Histogram of input image

Namitha Ramachandran

Step 3

Gray levels (rk)	No.of pixels (nk)	Probability Density Function(PDF)(P(rk)=nk/n)	Sk (cumulative Distribution Function-CDF)	Sk x max.gray level Sk x7	HG Equalized levels
0	0	0	0	0	0
1	8	0.32	0.32	2.24	2
2	8	0.32	0.64	4.48	4
3	2	0.08	0.72	5.04	5
4	0	0	0.72	5.04	5
5	7	0.28	1	7	7
6	0	0	1	7	7
7	0	0	1	7	7

Max.gray level

n=25

NAMITHA RAMACHANDRAN

Gray levels (rk)	No.of pixels (nk)	Probability Distribution Function(PDF)(P(rk)=nk/n	Sk (cumulative Distribution Function-CDF)	Sk x max.gray level Sk x7	HG Equalized levels
0	0	0	0	0	0
1	8	0.32	0.32	2.24	2
2	8	0.32	0.64	4.48	4
3	2	0.08	0.72	5.04	5
4	0	0	0.72	5.04	5
5	7	0.28	1	7	7
6	0	0	1	7	7
7	0	0	1	7	7

Max.gray level

n=25

Namitha Ramachandran

F(x,y)=

1	2	1	1	1
2	5	3	5	2
2	5	5	5	2
2	5	3	5	2
1	1	1	2	1

Input image

F(x,y)=

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

Output image

Namitha Ramachandran

Gray levels (rk)	No.of pixels (nk)	Probability Distribution Function(PDF)(P(rk)=nk/n	Sk (cumulative Distribution Function-CDF)	Sk x max.gray level Sk x7	HG Equalized levels
0	0	0	0	0	0
1	8	0.32	0.32	2.24	2
2	8	0.32	0.64	4.48	4
3	2	0.08	0.72	5.04	5
4	0	0	0.72	5.04	5
5	7	0.28	1	7	7
6	0	0	1	7	7
7	0	0	1	7	7

Max.gray level

n=25

Namitha Ramachandran

Step 4

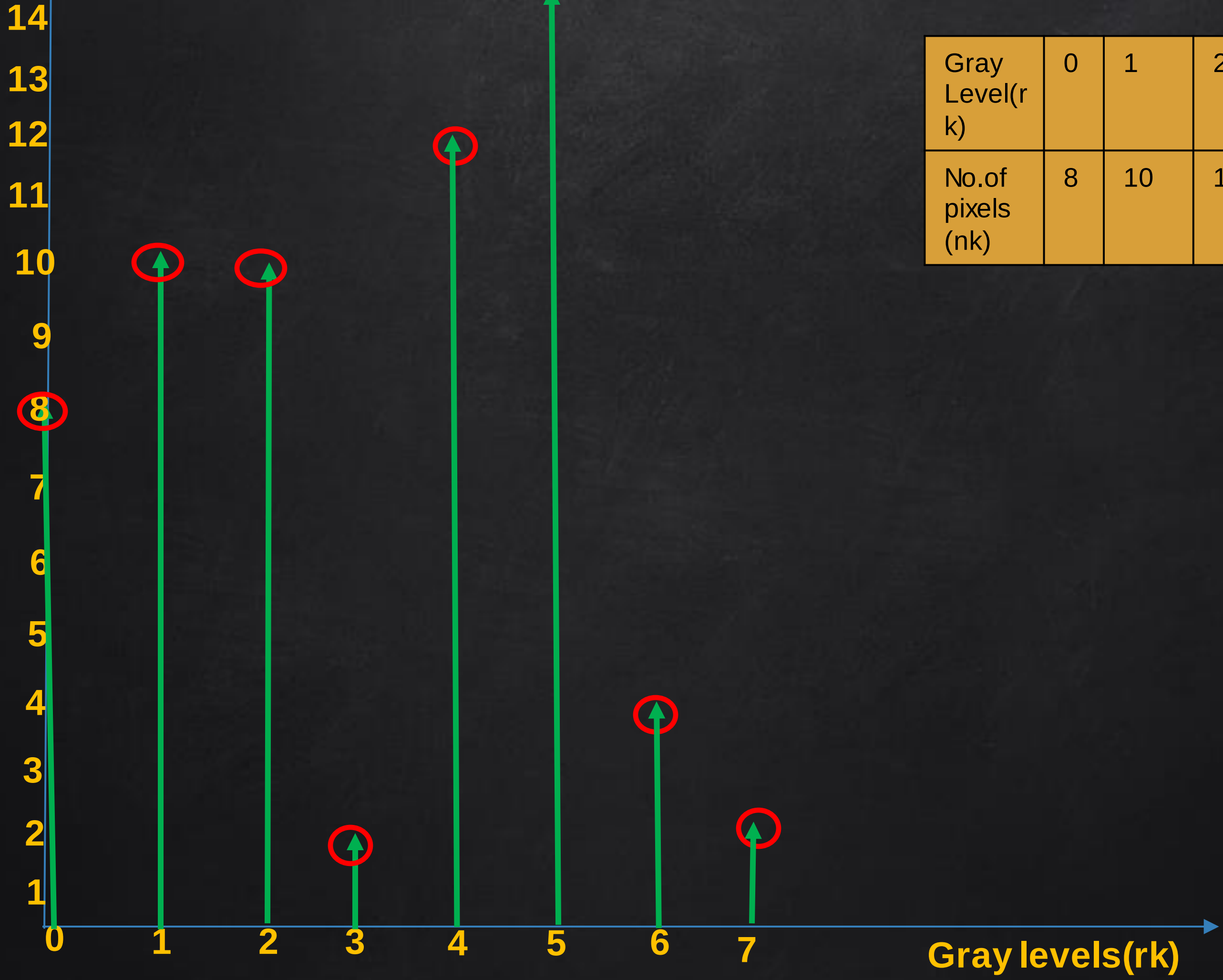
Gray Levels	0	2	4	5	7
No.of pixels	0	8	8	2	7

Step 2: Histogram of output image

Namitha Ramachandran

Perform histogram equalisation on the following image its gray level distribution is given

Gray Level(rk)	0	1	2	3	4	5	6	7
No.of pixels (nk)	8	10	10	2	12	16	4	2



Gray Level(rk)	0	1	2	3	4	5	6	7
No.of pixels (nk)	8	10	10	2	12	16	4	2

Namitha Ramachandran

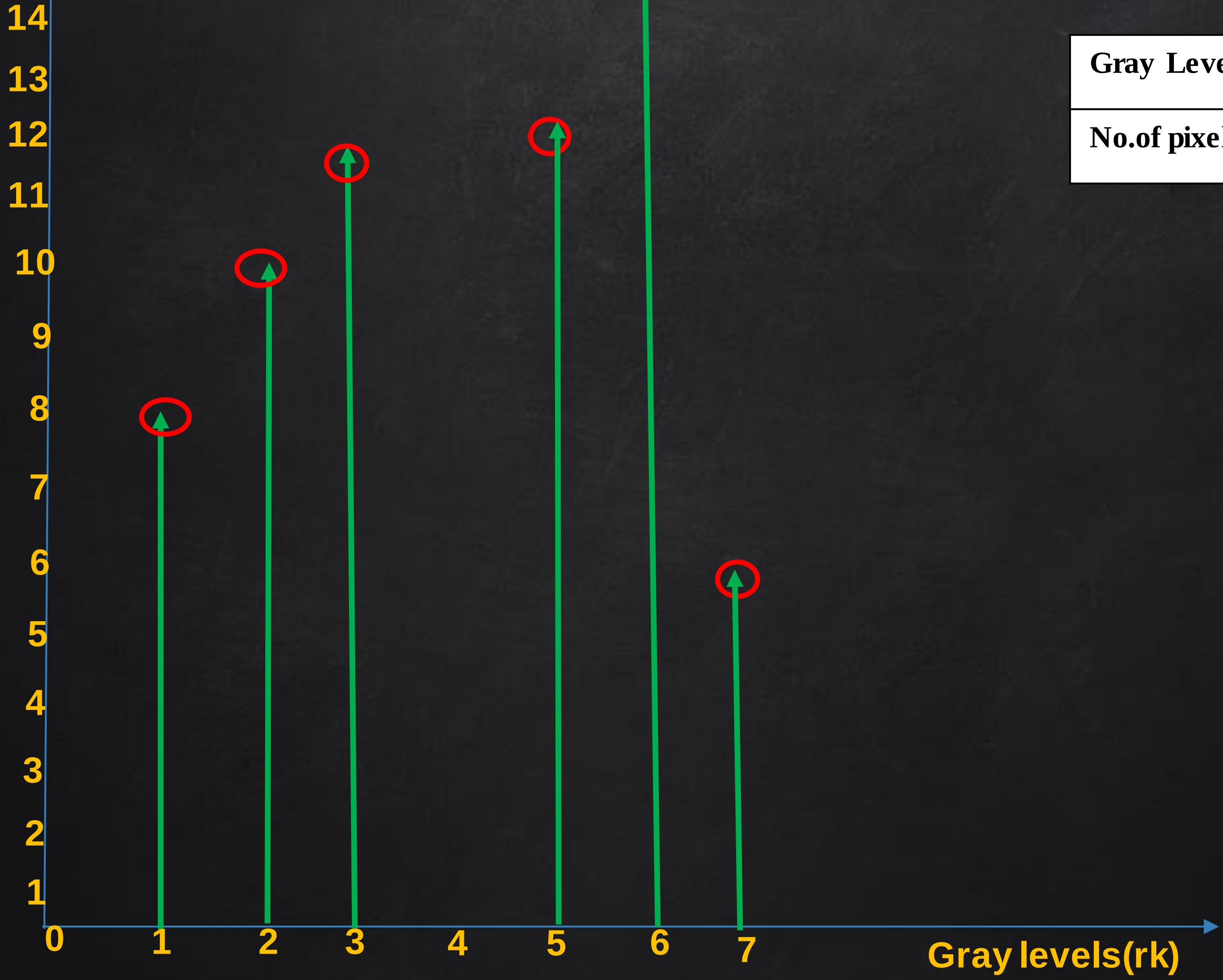
Gray levels (rk)	No.of pixels (nk)	Probability Distribution Function(PDF)(P(rk)=nk/n	Sk (cumulative Distribution Function-CDF)	Sk x max.gray level Sk x7	HG Equalized levels
0	8	0.125	0.125	0.875	1
1	10	0.15625	0.28125	1.96875	2
2	10	0.15625	0.4375	3.0625	3
3	2	0.03125	0.46875	3.28125	3
4	12	0.1875	0.65625	4.59375	5
5	16	0.25	0.90625	6.34375	6
6	4	0.0625	0.96875	6.78125	7
7	2	0.03125	1	7	7

Max.gray level

n=64

Namitha Ramachandran

Gray Levels	1	2	3	5	6	7
No.of pixels	8	10	12	12	16	6



Gray Levels	1	2	3	5	6	7
No.of pixels	8	10	12	12	16	6

Steps Involved in Histogram Equalisation

1. Identify and list gray levels present in your image.

1	2	1	1	1
2	5	3	5	2
2	5	5	5	2
2	5	3	5	2
1	1	1	2	1

For an image segment like this ,maximum intensity used is 5 here,so gray levels ranging from 0 to 7 ,because this range include 5 .

2. Identify the frequency of each gray level.

E.g. in the above image gray level 0 frequency is 0 ,gray level 1 frequency is 8, gray level 2 frequency is 8 and so on.

Namitha Ramachandran

3. Histogram of Input image

Plot the histogram of input image based on gray levels and corresponding frequency values.

4. Evaluate PDF -Probability Density Function (Normalising in the range of 0-1)

Use the formula P(rk)=nk/n

Where nk=frequency of gray level , n= maximum gray level value

5. Evaluate CDF -Cumulative Distribution Function for each PDF value.

$$s(k) = T(rk) = \sum_{j=0}^{k} \Pr(rj), k = 0,1,..L-1$$

6. Multiply each cumulative value using the maximum gray value

in the above example gray levels ranging from 0…7, the maximum gray value is 7 (say n)

n xSK

7. Perform round off operation over each transformed value and get the histogram equalised gray levels .

8. Using the histogram equalized gray levels and the frequency value plot the histogram of the output image.

9. We can identify the output image segment by mapping histogram equalised gray levels with initial gray levels .

Namitha Ramachandran

THANK YOU

Namitha Ramachandran

COMPUTER GRAPHICS & IMAGE PROCESSING

MODULE 5

Basics of spatial filtering

(Smoothing Linear Filters)

Namitha Ramachandran

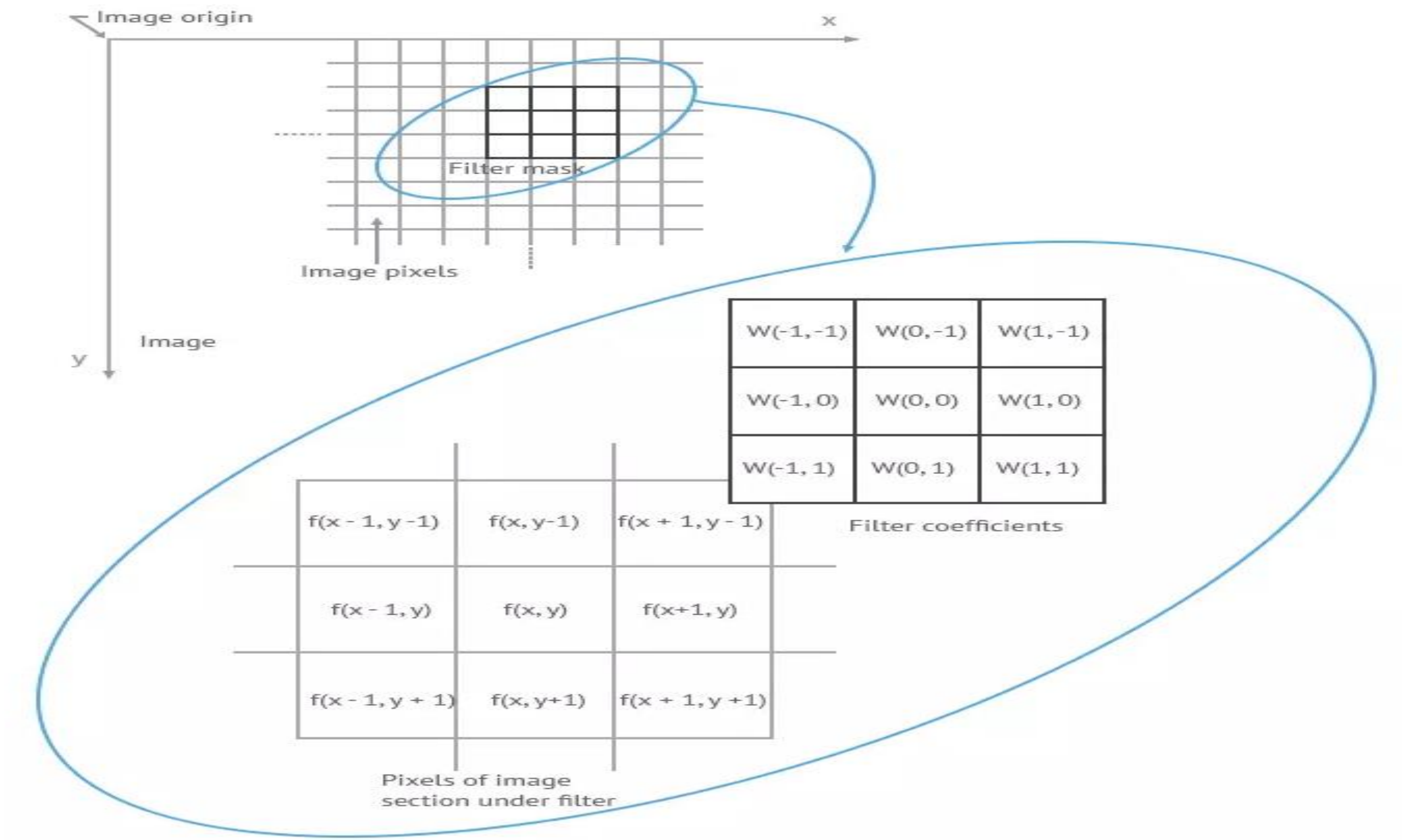
Basics of spatial filtering

The **spatial domain enhancement** is based on pixels in a small range (neighbor).

This means the transformed intensity is determined by the gray values of those points within the neighborhood, thus the spatial domain enhancement is also called **neighborhood operation or neighborhood processing.**

If the pixel in the neighborhood is calculated as a linear operation, it is also called 'linear spatial domain filtering', otherwise, it's called 'nonlinear spatial domain filtering.

The process of spatial filtering with a 3 × 3 templates (also known as a filter, kernel, or window).



The response ‘R’ to the template is:

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

For a filter with a size of (2a+1, 2b+1), the output response can be calculated with the following function:

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

Smoothing Linear Filters

Image smoothing is a digital image processing technique that reduces and suppresses image noises, also used for image blurring.

Remove small details from an image prior to object extraction.
Bridge small gaps in lines or curves.

In the spatial domain, neighborhood averaging can generally be used to achieve the purpose of smoothing.

Commonly seen smoothing filters include average smoothing, Gaussian smoothing, and adaptive smoothing.

It includes **average linear filter** and **order statistics nonlinear filters.**

Average Filters

Also referred to as **Lowpass filters**.

The idea is to replace the value of every pixel in an image by the average of the intensity levels in the neighbourhood defined by the filter mask.

The application is **noise reduction**.

Edges are an important part of an image, applying average filters has undesirable side effects of blurring an image.

The figure below shows two 3×3 averaging filters.

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

1	1	1
1	1	1
1	1	1

Standard average filter

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

1	2	1
2	4	2
1	2	1

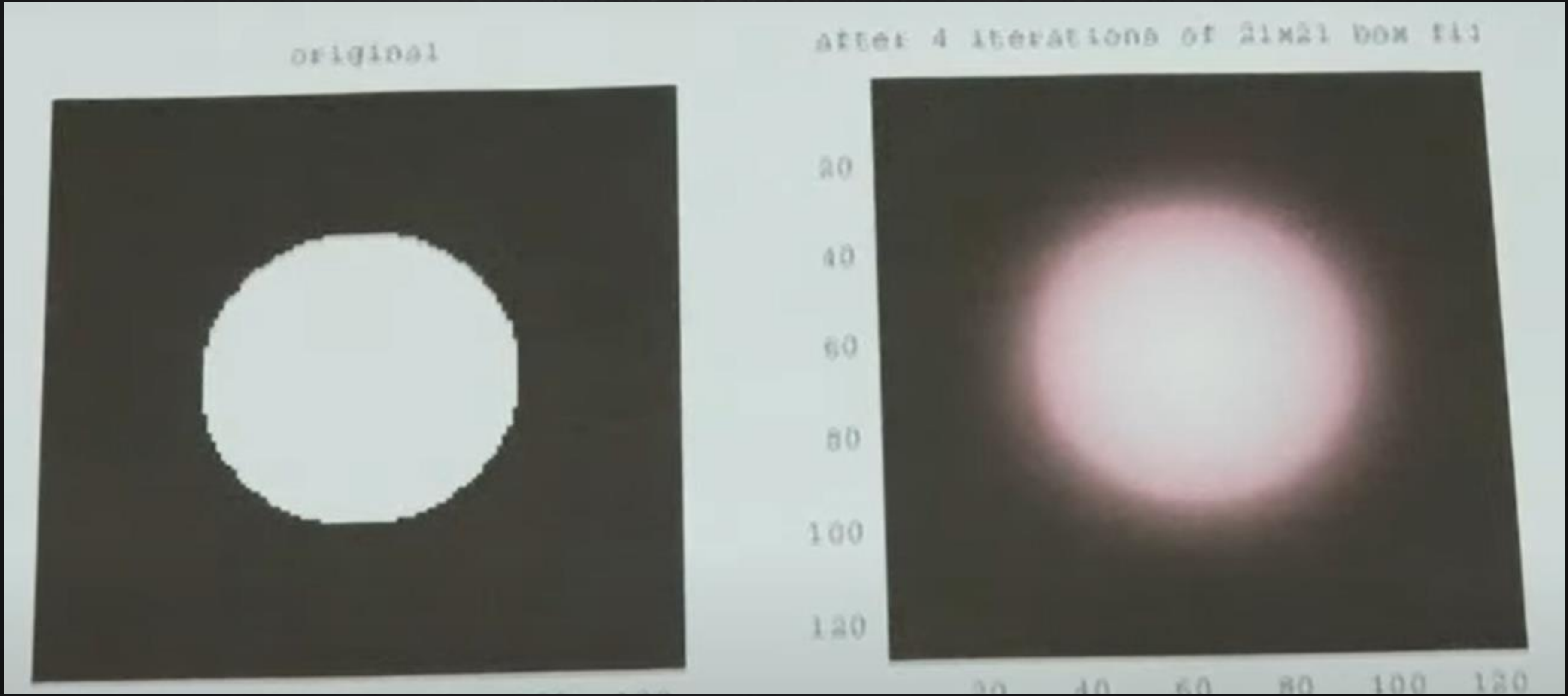
Weighted average filter

$$\frac{1}{9} x \sum_{i=1}^{9} zi$$

Averaging linear filtering of an image f of size $M \times N$ with a filter mask of size $m \times n$ is given by the expression:

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

Result of applying Average Box filters



A spatial average filter in which all the coefficients are equal is known as box filters.

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

Weighted average filter

In this weighted average filter, the middle cell is given the highest weight, which means high priority, this will reduce blurring during the smoothing process.

20	30	50	80	100
30	20	80	100	100
20	30	60	10	110
40	30	70	40	100
50	60	80	30	90

Apply standard average filter for pixel at (3,3)

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

Standard average filter

Pixel at (3,3) is 60

1	1	1
1	1	1
1	1	1

1/9 [20x1+80x1+100x1+30x1+60x1+10x1+30x1+70x1+40x1]

Apply standard average filter for pixel at (1,1)

$$\frac{1}{9}\times\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$$

Standard average filter

0	0	0	0	0	0
0	20	30	50	80	100
0	30	20	80	100	100
0	20	30	60	10	110
0	40	30	70	40	100
0	50	60	80	30	90

Pixel at (1,1) is 20

1	1	1
1	1	1
1	1	1

1/9 [0x1+0x1+0x1+0x1+20x1+30x1+0x1+30x1+20x1]

Apply standard average filter for pixel at (2,3)

20	30	50	80	100
30	20	80	100	100
20	30	60	10	110
40	30	70	40	100
50	60	80	30	90

Pixel at (2,3) is 80

1	2	1
2	4	2
1	2	1

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

Weighted average filter

1/16 [30x1+50x2+80x1+20x2+80x4+100x2+30x1+60x2+10x1]

Namitha Ramachandran

Order static(Nonlinear) smoothing Filters

The response of the filter depends on the ordering or ranking of the pixels in the image. Replacing the center pixel value with the value determined by the ranking result.

1. Median Filter

The filter replaces the value of a pixel in the image by the median of the gray levels encompassed by the filter.

Provide excellent noise reduction compared to linear smoothing filters.

Cause less blurring in the image.

Most effective for removing impulse noise (salt and pepper noise).

Example:

Consider the following 5×5 image:

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Apply a 3×3 median filter on the shaded pixels, and write the filtered image.

Solution

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort:
20, 25, 30, 30, **30**, 70, 80, 80, 255

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort
0, 20, 30, 70, **80**, 80, 100, 100, 255

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort
0, 70, 80, 80, **100**, 100, 110, 120, 130

Filtered Image =

20	30	50	80	100
30	20	80	100	110
25	**30**	**80**	**100**	120
30	30	80	100	130
40	50	90	125	140

Namitha Ramachandran

Max Filter

To find the brightest points in an image.

Finds the maximum value in the area encompassed by the filter.

Reduces the pepper noise as a result of the max operation.

STEPS TO BE FOLLOWED:

1. Consider a matrix A=

$$\begin{bmatrix} 1 & 2 & 2 & 1 \\ 1 & 1 & 0 & 3 \\ 2 & 4 & 1 & 5 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

2. Pad matrix A with zeros modify A=

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 & 3 & 0 \\ 0 & 2 & 4 & 1 & 5 & 0 \\ 0 & 2 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Namitha Ramachandran

3. Consider the elements in the window 3x3

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 & 3 & 0 \\ 0 & 2 & 4 & 1 & 5 & 0 \\ 0 & 2 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ i.e } \begin{bmatrix} 1 & 2 & 2 \\ 1 & 1 & 0 \\ 2 & 4 & 1 \end{bmatrix}$$

4. Find the maximum from the window. Here it is 4.

5. Similarly, find the maximum by sliding the window on the whole matrix.

$$\text{The output matrix B=} \begin{bmatrix} 2 & 2 & 3 & 3 \\ 4 & 4 & 5 & 5 \\ 4 & 4 & 5 & 5 \\ 4 & 4 & 5 & 5 \end{bmatrix}$$

Namitha Ramachandran

MIN Filter

To find the darkest points in an image.
Finds the minimum value in the area encompassed by the filter.
Reduces the salt noise as a result of the min operation.

FORMULA:

$$\hat{f}(x, y) = \min_{(s,t)\in S_{xy}} \{g(s,t)\}$$

COMPUTER GRAPHICS
&
IMAGE PROCESSING

MODULE 5

Basics of spatial filtering

Sharpening spatial filters–Gradient and Laplacian

Namitha Ramachandran

Sharpening spatial filters-Gradient and Laplacian

Sharpening filters **highlight the details of an image**.

Applications range from electronic printing, medical imaging, industrial inspection, and autonomous guidance in the military system.

Blurring vs Sharpening

- Blurring/smooth is done in spatial domain by pixel averaging in a neighbours, it is a process of integration
- Sharpening is an inverse process, to find the difference by the neighborhood, done by spatial differentiation.

Sharpening spatial filters seek to highlight fine detail

- Remove blurring from images
- Highlight edges

Sharpening filters are based on *spatial differentiation*

Derivative operator

- This operator calculates the gradient of the image intensity at each point, giving direction of the largest possible increase from light to dark and rate of change in that direction.
- Image differentiation
  - enhances edges and other discontinuities (noise)
  - deemphasizes area with slowly varying gray-level values.

Foundation of sharpening spatial filters

The basic definition of the first-order derivative of a one dimensional function f(x) is the difference

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

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

The second-order derivative of a one-dimensional function f(x) is the difference

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

$$\frac{\partial 2f}{\partial y2} = f(y+1) + f(y-1) - 2f(y)$$

A Ramachandran

First and Second-order derivative of 2D

Let us consider an image function of two variables, f(x, y), at which time it will be dealed with partial derivatives along the two spatial axes.

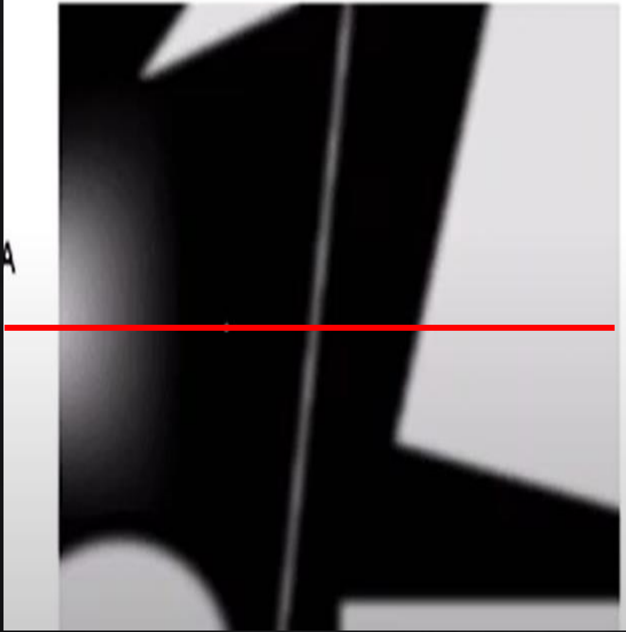
Gradient operator (linear operator)

$$\nabla \mathrm{f} = \frac{\partial f(x,y)}{\partial x \partial y} = \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y}$$

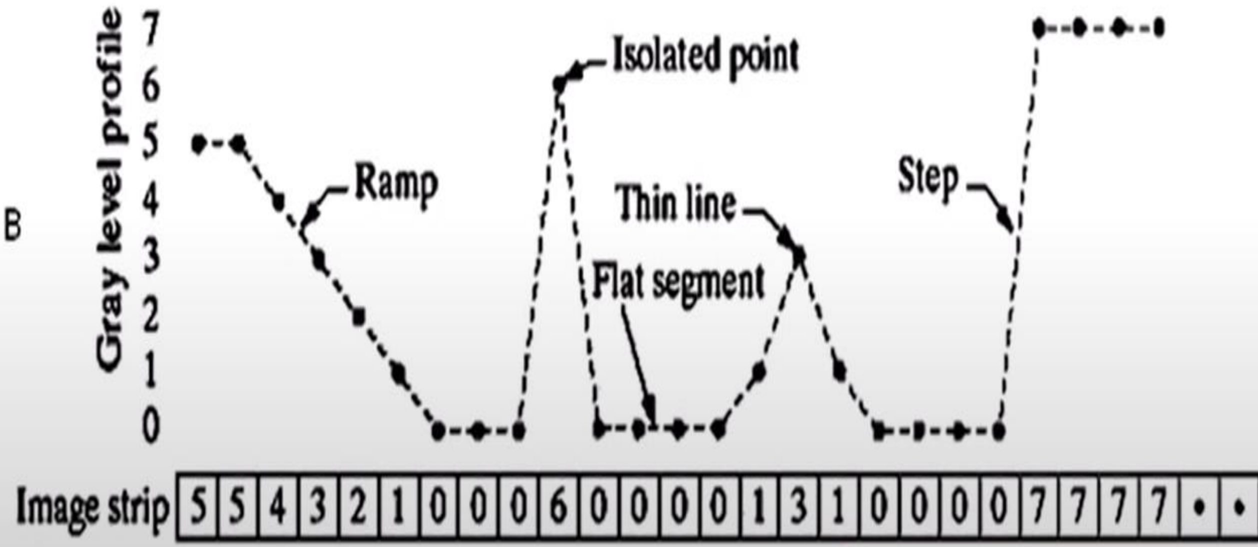
Laplacian operator (non-linear)

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

A



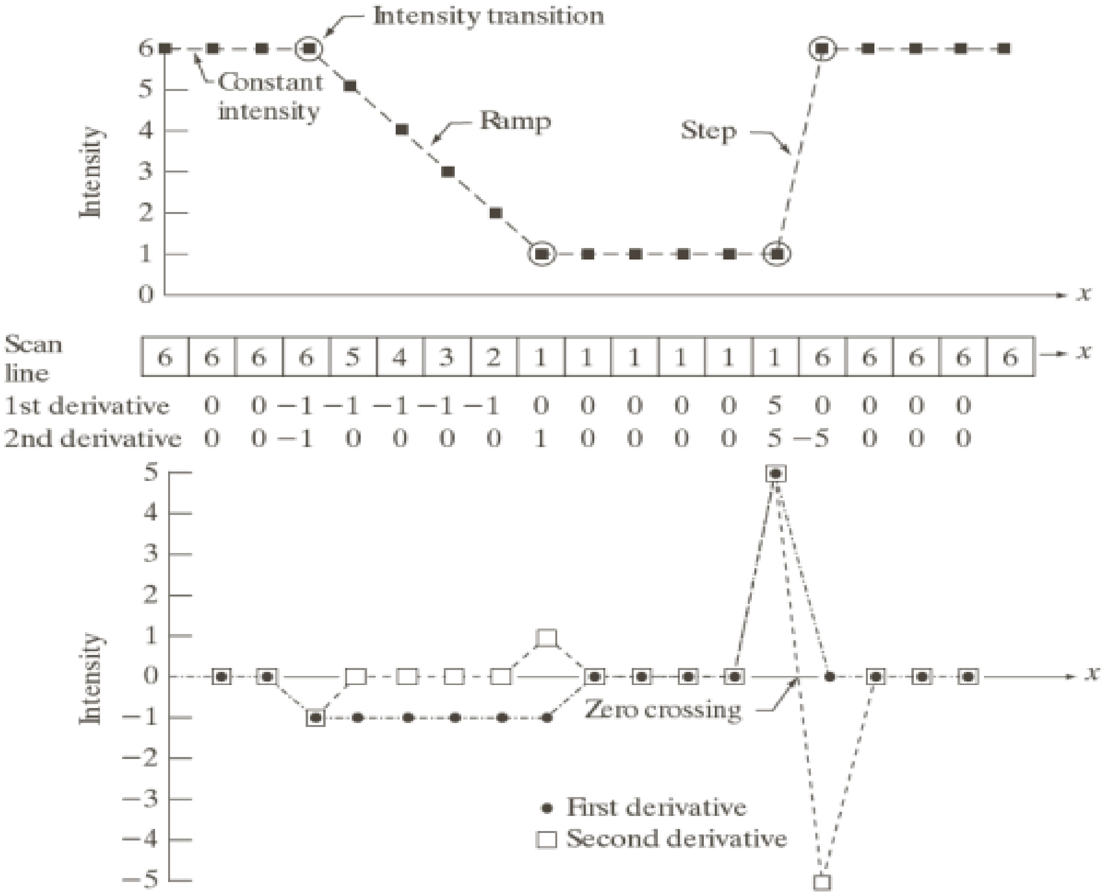
B



Gray level profile
7
6
5
4
3
2
1
0
Isolated point
Ramp
Thin line
Step
Flat segment
Image strip
5 5 4 3 2 1 0 0 0 6 0 0 0 0 1 3 1 0 0 0 0 7 7 7 7

Derivatives of an image: an example

3 sections :
1)Constant intensity
2)Ramp
3)Step



Note: at the step, the second derivative switch the sign (*zero crossing*).

Area of constant intensity: both first-order derivatives and second-order derivatives are zero.
First order derivative nonzero at onset of ramp and step .
The second-order derivative is nonzero at the onset and end of both the ramp and step.
First-order derivative is nonzero and the second is zero along the ramp.

1st derivative detects **thick edges** while the second derivative **detects thin edges**.

2nd derivative has a much stronger response at the gray-level step than 1st derivative.

Second-order derivative enhances fine details(thin edges,lines, including noise.) than that of 1st order derivative.

First Derivative r Filters

For a function $f(x, y)$ the gradient of f at coordinates (x, y) is given as:

$$\nabla f \approx |G_x| + |G_y|$$

Consider following image $f(x, y)$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$\nabla f \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$$

Some other 1st 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

← Gradient in diagonal direction

Roberts

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Edge detection

Prewitt

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Edge detection

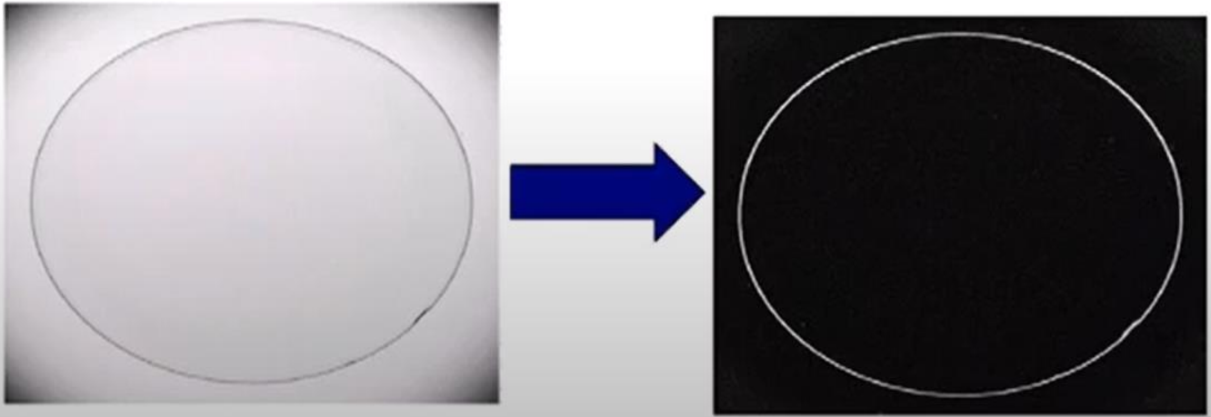
Sobel

Based on the previous equations we can derive the *Sobel Operators*

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

To filter an image it is filtered using both operators and the results of which are added together



Sobel filters are typically used for edge detection

- In a digital image, the second derivatives wrt. x and y are computed as:

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

- Hence, the Laplacian results:

$$
\begin{aligned}
\nabla^2 f(x, y) &= f(x+1, y) + f(x-1, y) + f(x, y+1) \\
&\quad + f(x, y-1) - 4f(x, y)
\end{aligned}
$$

- Also the derivatives along to the diagonals can be considered:

$$
\begin{aligned}
\nabla^2 f(x, y) &+ f(x-1, y-1) + f(x+1, y+1) \\
&+ f(x-1, y+1) + f(x+1, y-1) - 4f(x, y)
\end{aligned}
$$

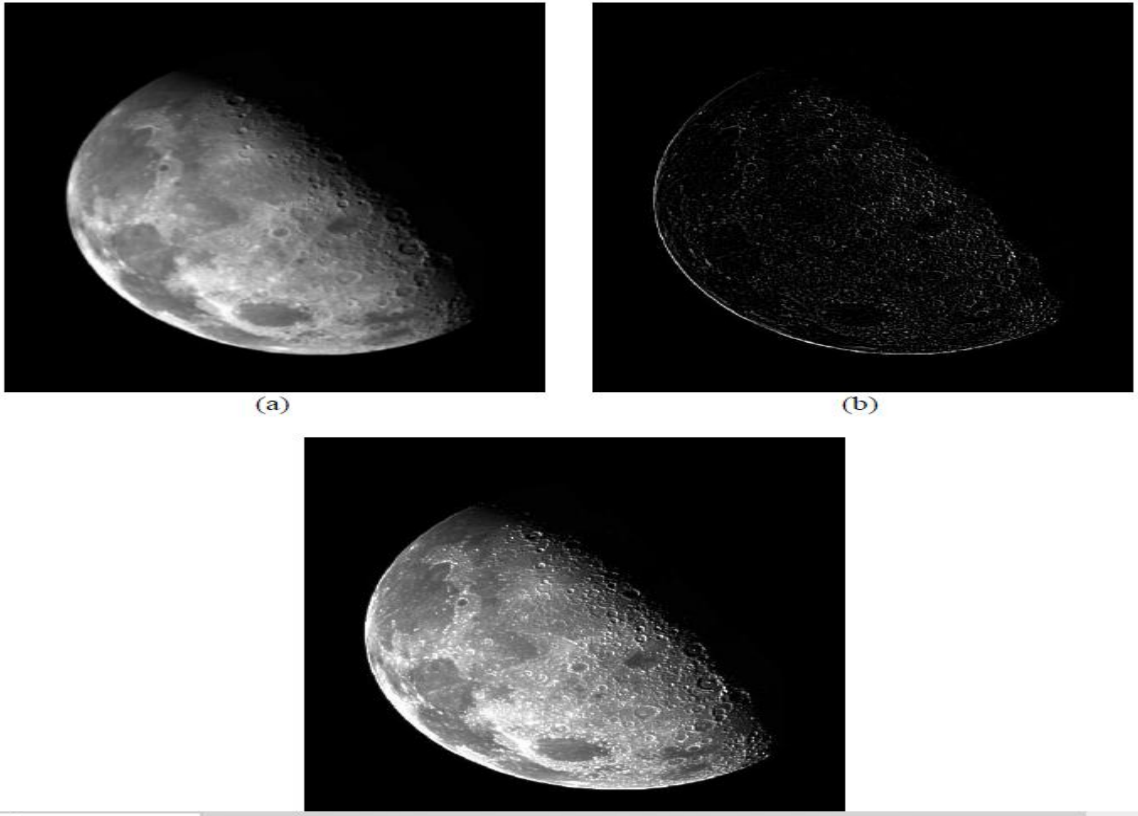
Laplacian filter (2)

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

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

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


(a)
(b)

THANK YOU

Namitha Ramachandran

Fundamentals of Image Segmentation.

Image Segmentation is the process by which a digital image is partitioned into various subgroups (of pixels) called Image Objects, which can reduce the complexity of the image, and thus analyzing the image becomes simpler.

Similarity Detection (Region Approach)

This fundamental approach relies on detecting similar pixels in an image – based on a threshold, region growing, region spreading, and region merging so does classification, which detects similarity based on a pre-defined (known) set of features.

Discontinuity Detection (Boundary Approach)

This is a stark opposite of the similarity detection approach where the algorithm rather searches for discontinuity. Image Segmentation Algorithms like Edge Detection, Point Detection, Line Detection follow this approach – where edges get detected based on various metrics of discontinuity like intensity, etc

Thresholding - Basics of Intensity thresholding and Global Thresholding.

Image thresholding is a technique employed to facilitate easy image segmentation for various image processing tasks.

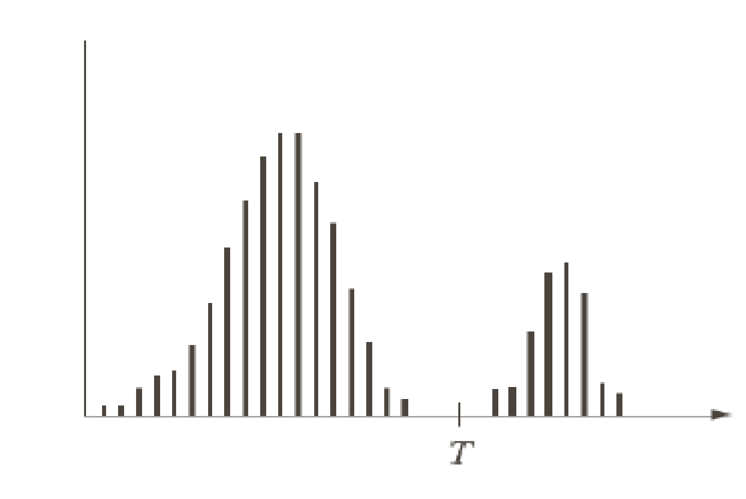
Simple thresholding technique (Binary Thresholding)

In a simple thresholding technique, a standard threshold value is set and each pixel value is compared with the threshold value. If the pixel value is less than the mentioned threshold value then the value is set to 0 or else it is set to the maximum value.

A thresholded image $g(x, y)$ is defined as

$$g(x,y) = \begin{cases} 1, & \textbf{if } f(x,y) > T \\ 0, & \textbf{if } f(x,y) \leq T \end{cases},$$

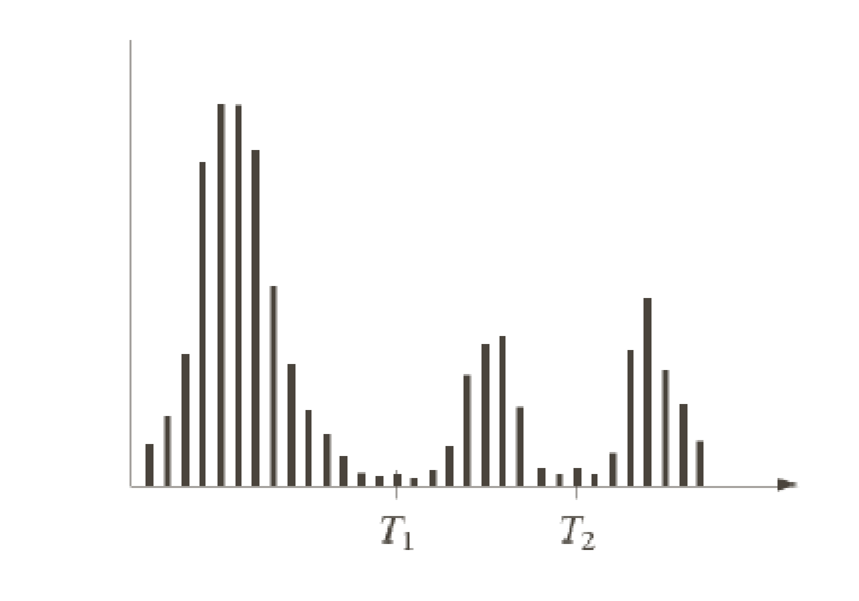
where 1 is object and 0 is background



Namitha Ramachandran

Multiple thresholding:

$$g(x,y) = \begin{cases} a, & \textbf{if } f(x,y) > T_2 \\ b, & \textbf{if } T_1 < f(x,y) \le T_2 \\ c, & \textbf{if } f(x,y) \le T_1 \end{cases},$$



Namitha Ramachandran

Global thresholding: T **is constant applicable over whole image**

Variable (local/regional) thresholding: T **changes over an image**

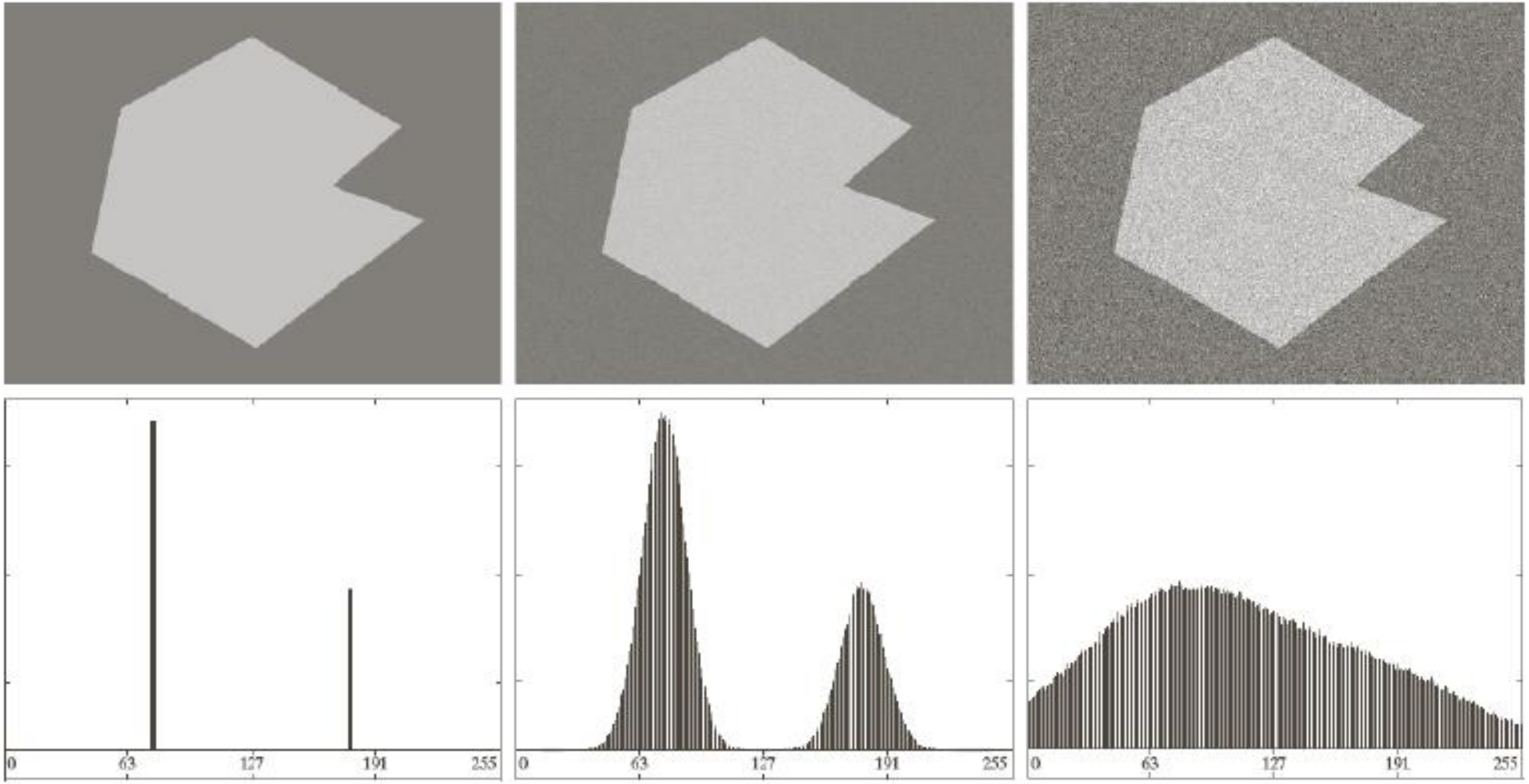
Dynamic (adaptive) thresholding: T **depends on spatial coordinates** (x, y)

Multiple thresholding:

$$g(x,y) = \begin{cases} a, & \textbf{if } f(x,y) > T_2 \\ b, & \textbf{if } T_1 < f(x,y) \leq T_2 \\ c, & \textbf{if } f(x,y) \leq T_1 \end{cases},$$

- When the threshold T changes over the image, it is called as variable threshold.
- This variable threshold value at a point (x, y) can be selected either based on neighbourhood of (x, y) or based on spatial coordinates (x, y) itself.
- When T at any point (x, y) is a function of neighbourhood of (x, y), it is called as local threshold and when T at any point (x, y) is a function of spatial coordinates (x, y) itself, it is called as adaptive or dynamic threshold.

The role of noise in image thresholding



Basics of Global Thresholding

When the intensity distributions of objects and background pixels are sufficiently distinct, it is possible to use a single global threshold applicable over the entire image.

We have algorithms for estimating automatically the threshold value for each image.

Iterative algorithm

Otsu’s method

Iterative algorithm for <u>automatic</u> estimation of threshold T:

(1) Select an initial estimate for T

(2) Segment image using $T \longrightarrow$ Group G_1 (values $> T$)
Group G_2 (values $\leq T$)

(3) Compute average intensity values for G_1, $G_2 \longrightarrow m_1$, m_2

(4) Compute a new threshold value $T = \frac{1}{2}(m_1 + m_2)$

(5) Repeat (2) through (4) until the difference in T in successive iterations is smaller than ΔT

Average intensity is good initial estimate for T

Summary of Otsu's algorithm

(1) Compute normalized histogram of the image, $p_i = \dfrac{n_i}{MN}, \ i = 0, \ldots, L-1$

(2) Compute cumulative sums, $P_1(k) = \displaystyle\sum_{i=0}^{k} p_i, \ k = 0, \ldots, L-1$

(3) Compute cumulative means, $m(k) = \displaystyle\sum_{i=0}^{k} i\, p_i, \ k = 0, \ldots, L-1$

(4) Compute global intensity mean, $m_G = \displaystyle\sum_{i=0}^{L-1} i\, p_i$

(5) Compute between-class variance, $\sigma_B^2(k) = \dfrac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}, \ k = 0, .., L-1$

(6) Obtain the Otsu threshold, k^***, that is the value of** k **for which** $\sigma_B^2(k^*)$ **is a maximum – if this maximum is not unique, obtain** k^* **by avaraging the values of** k **that correspond to the various maxima detected**

(7) Obtain the separability measure $\eta(k^*) = \dfrac{\sigma_B^2(k^*)}{\sigma_G^2}$

Region-based Segmentation Approach - Region Growing, Region Splitting and Merging

A region can be classified as a group of connected pixels exhibiting similar properties. The similarity between pixels can be in terms of intensity, color, etc. In this type of segmentation, some predefined rules must be obeyed by a pixel to be classified into similar pixel regions. Region-based segmentation methods are preferred over edge-based segmentation methods in case of a noisy image. Region-Based techniques are further classified into 2 types based on the approaches they follow.

Region growing method

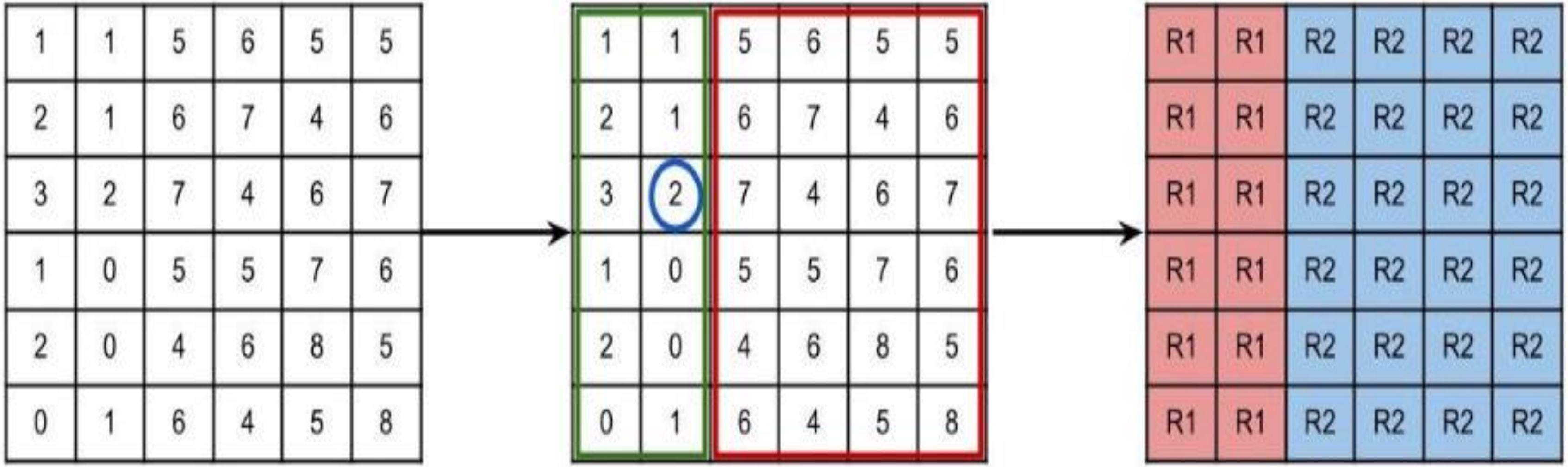
Region splitting and merging method

Region Growing Technique

In the case of the Region growing method, we start with some pixel as the seed pixel and then check the adjacent pixels.
If the adjacent pixels abide by the predefined rules, then that pixel is added to the region of the seed pixel and the following process continues till there is no similarity left. This method follows the bottom-up approach.

In case of a region growing, the preferred rule can be set as **a threshold.**

For example: Consider a seed pixel of 2 in the given image and a threshold value of 3, if a pixel has a value greater than 3 then it will be considered inside the seed pixel region. Otherwise, it will be considered in another region.
Hence 2 regions are formed in the following image based on a threshold value of 3.



Original Image

Region growing process with 2 as the seed pixel.

Splitting image into two regions based on a threshold.

Region Splitting and Merging Technique

In Region splitting, the whole image is first taken as a single region. If the region does not follow the predefined rules, then it is further divided into multiple regions (usually 4 quadrants) and then the predefined rules are carried out on those regions in order to decide whether to further subdivide or to classify that as a region. The following process continues till there is no further division of regions required i.e every region follows the predefined rules.

In Region merging technique, we consider every pixel as an individual region. We select a region as the seed region to check if adjacent regions are similarly based on predefined rules. If they are similar, we merge them into a single region and move ahead in order to build the segmented regions of the whole image.

Usually, first region splitting is done on an image so as to split an image into maximum regions, and then these regions are merged in order to form a good segmented image of the original .

Apply region splitting on the following image. Assume the threshold value be<=4.

R1

R2

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

R4

R3

Higher value-Lower value>4 then split

7-0=7>4 ,split into 4 quadrants

NAMITHA RAMACHANDRAN

Higher value-Lower value>4 then split

R1

7-4=3<=4 ,NO split

R1

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

R2

R4

R3

Namitha Ramachandran

Higher value-Lower value>4 then split

R2

7-2=5>4 ,split

R21 R22

R1 R2

R24 R23

R4 R3

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

R21
7-5=2<=4 ,NO split

R22
7-4=3<=4 ,NO split

R23
6-4=2<=4 ,NO split

R24
3-2=1<=4 ,NO split

Namitha Ramachandran

Higher value-Lower value>4 then split

R3

7-0=7>4 ,split

R31
3-2=1<=4 ,NO split

R32
6-4=3<=4 ,NO split

R33
5-4=1<=4 ,NO split

R34
3-0=3<=4 ,NO split

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

R1 R21 R22 R2 R24 R23 R31 R32 R4 R33 R34

Namitha Ramachandran

Higher value-Lower value>4 then split

R4

3-0=3<=4 ,No split

R21 R22

R1 R2

R24 R23

R31 R32

R4 R33

R34

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

Namitha Ramachandran

Merging

Compare Higher value-Lower value>4 then split in both direction

R21 R22

R1 R2 R24 R23 R31 R32 R4 R33 R34

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

Higher value-Lower value>4 then split

R1 and R21 (MAX-7,MIN-5)
7-5=2<=4,merge
R21 and R1 (MAX-7,MIN-4)

7-4=3<=4 ,MERGE

R1- R21 and R22 (MAX-7,MIN-4)

7-4=3<=4 ,MERGE

R1- R21 and R22 (MAX-7,MIN-4)
7-4=3<=4 ,MERGE

Namitha Ramachandran

Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

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

R21 R22 R1 R2 R24 R23 R31 R32 R4 R33 R34

R22 and R23 (MAX-7,MIN-4)
7-4=3<=4,merge
R23 and R22 (MAX-6,MIN-4)

6-4=2<=4 ,MERGE

R23 and R32 (MAX-6,MIN-4)

6-4=3<=4 ,MERGE

R32- R23 and R22 (MAX-7,MIN-4)
7-4=3<=4 ,MERGE

Namitha Ramachandran

Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

R21 R22

R1 R2

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

R24 R23

R31 R32

R4 R33

R34

R32 and R33 (MAX-7,MIN-4)
7-4=3<=4,merge
R33 and R32 (MAX-5,MIN-4)

5-4=2<=4 ,MERGE

R33 and R34 (MAX-5,MIN-0)

5-0=5>4 ,NO MERGE

Namitha Ramachandran

Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

R21 R22

R1

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

R2 R24 R23 R31 R32 R4 R33 R34

R32 and R33 (MAX-7,MIN-4)
7-4=3<=4,merge
R33 and R32 (MAX-5,MIN-4)

5-4=2<=4 ,MERGE

R23 and R32 (MAX-6,MIN-4)

6-4=3<=4 ,MERGE

R32- R23 and R22 (MAX-7,MIN-4)
7-4=3<=4 ,MERGE

Namitha Ramachandran

Merging

Higher value-Lower value>4 then split

Repeat the same with all other regions and finally we get 2 regions

R21 R22

R1 R2

R24 R23

R31 R32

R4

R33

R34

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

Namitha Ramachandran