# POINT PROCESSING & GRAY LEVEL TRANSFORMATIONS

Presented By,

| | |
|---|---|
| K. Annapushpam | M.Phil (Computer Science) |
| P. Anupriya | M.S.University |
| M. Chithra | Tirunelveli |
| R. Debi Stella | |

# CONTENTS

- Point processing
- Basic gray level transformations
  - Basic gray level transformation graph
  - Linear transformation
    - Negative transformation
    - Identity transformation
  - Log transformation
  - Power law transformation
- Piecewise linear transformation functions
  - Contrast stretching
  - Intensity level slicing
  - Bit plane slicing

# Point processing

- Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further image analysis.
- It has two broad categories:
  - Spatial domain methods
  - Frequency domain methods
- Spatial domain methods are operate directly on the pixels.
- Point processing operation deals with pixel intensity values individually.

# POINT PROCESSING (CONTD)

- The intensity values are altered using particular transformation techniques as per the requirement.
- Enhanced at any point in an image depends only on the gray level at that point techniques are referred as point processing.
- Most spatial domain enhancement operations can be reduced to the form of,

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

- In this case T is referred to as a gray level transformation function or a point processing operation.

# POINT PROCESSING (CONTD)

- where f (x, y) is the input image, g (x, y) is the processed image and T is point operator defined over some neighborhood of (x, y).
- Point processing operations take the form of,

$$s = T ( r )$$

- where s refers to the processed image pixel value and r refers to the original image pixel value.
- Mask is a small matrix useful for blurring, sharpening, edge detection.

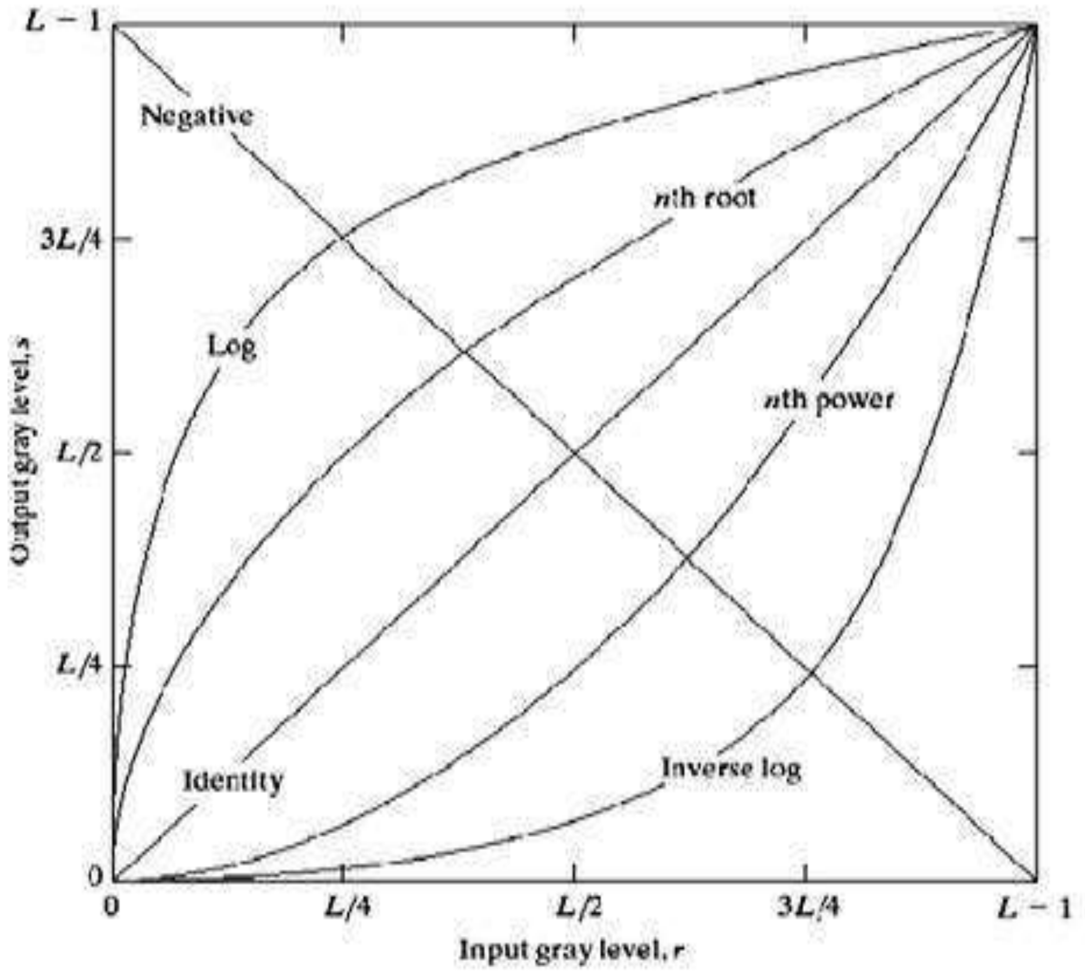
# POINT PROCESSING (CONTD)

- New image is generated by multiplying the input image with the mask matrix. Mask can be in any dimension (i.e 3x3, 4x4).
- Contrast stretching expands the range of intensity levels in an image.
- Extreme contrast stretching yields Thresholding.
- Thresholding image has maximum contrast as it has only Black & White gray values.
- Brightness enhancement is shifting of intensity values to higher level.

# Basic gray level transformation

- There are three basic gray level transformation.
  - Linear
  - Logarithmic
  - Power – law

# GRAY LEVEL TRANSFORMATION GRAPH



# LINEAR TRANSFORMATION

- Linear transformation includes following two categories,
  - Negative Transformation
  - Identity Transformation

# NEGATIVE TRANSFORMATION

- Negative images are useful for enhancing white or gray detail embedded in dark regions of an image.
- Negative transform exchanges dark values for light values and vice versa.
- The Negative Transformations can be defined by,

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

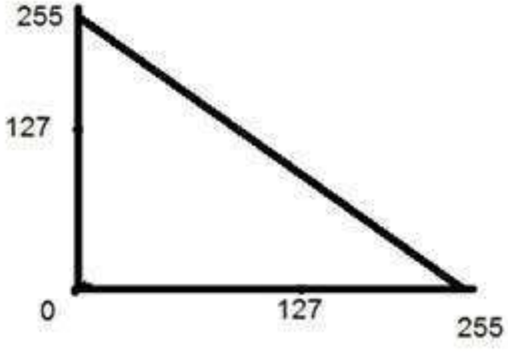
- Negative of an image intensity levels in the range [0,L-1],

  L-1 = Maximum pixels value

  r = Pixel value of an image

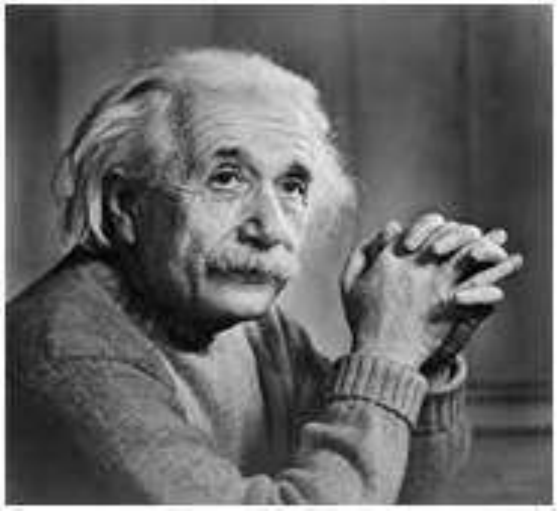
# NEGATIVE TRANSFORMATION EXAMPLE

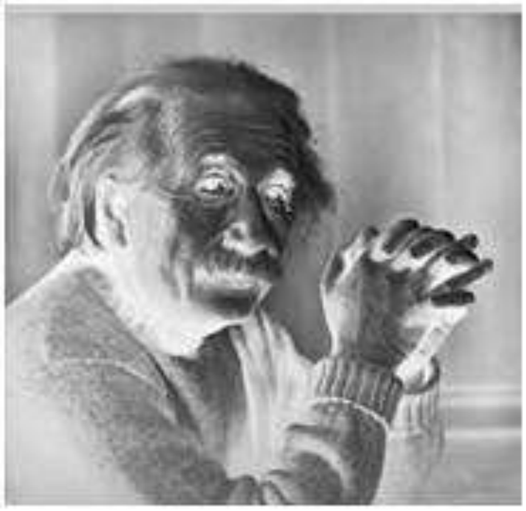
Graph representation



Input image

Output image

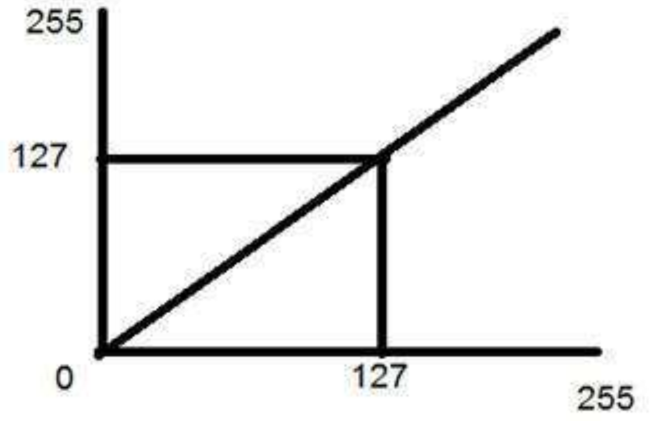




# IDENTITY TRANSFORMATION

- Each value of the input image is directly mapped to each other value of output image.
- That results in the same input image and output image.

Graph representation



# LOG TRANSFORMATION

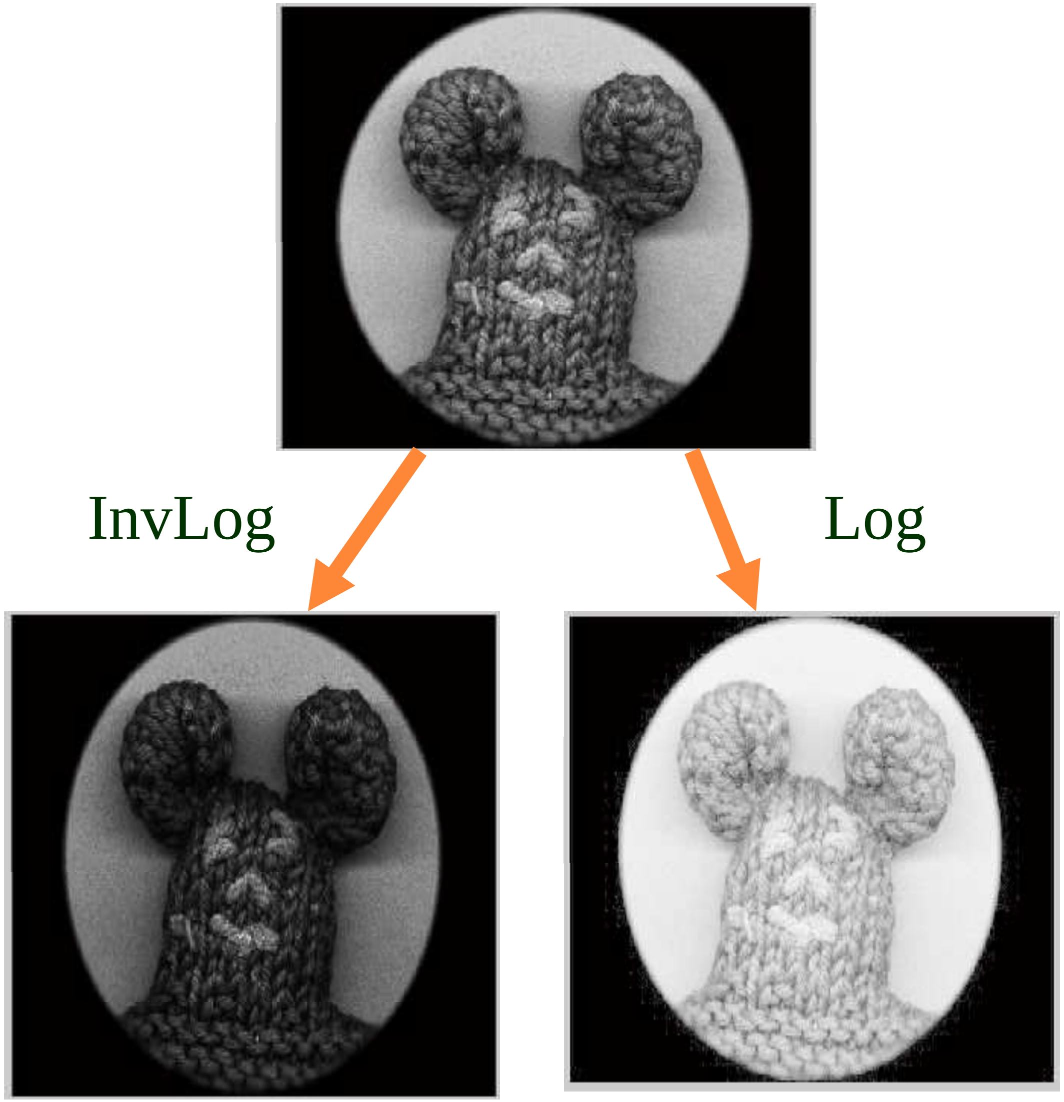
- This transform is used to expand values of dark pixels and compress values of bright pixels.
- It maps a narrow range of low level gray scale intensities into wider range of output values.
- Similarly maps the wide range of high level gray scale intensities into a narrow range of high level output values.
- The log transformations can be defined by this formula

$$\mathbf{s = c \log(r + 1)}$$

# LOG TRANSFORMATION (CONTD)

- 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.
- The inverse log transform is opposite to log transform.

# Log transformation example



InvLog

Log

# Power law transformation

- This type of transformation is used for enhancing images for different type of display devices.
- These transformations can be given by,

$$s=cr^\gamma$$

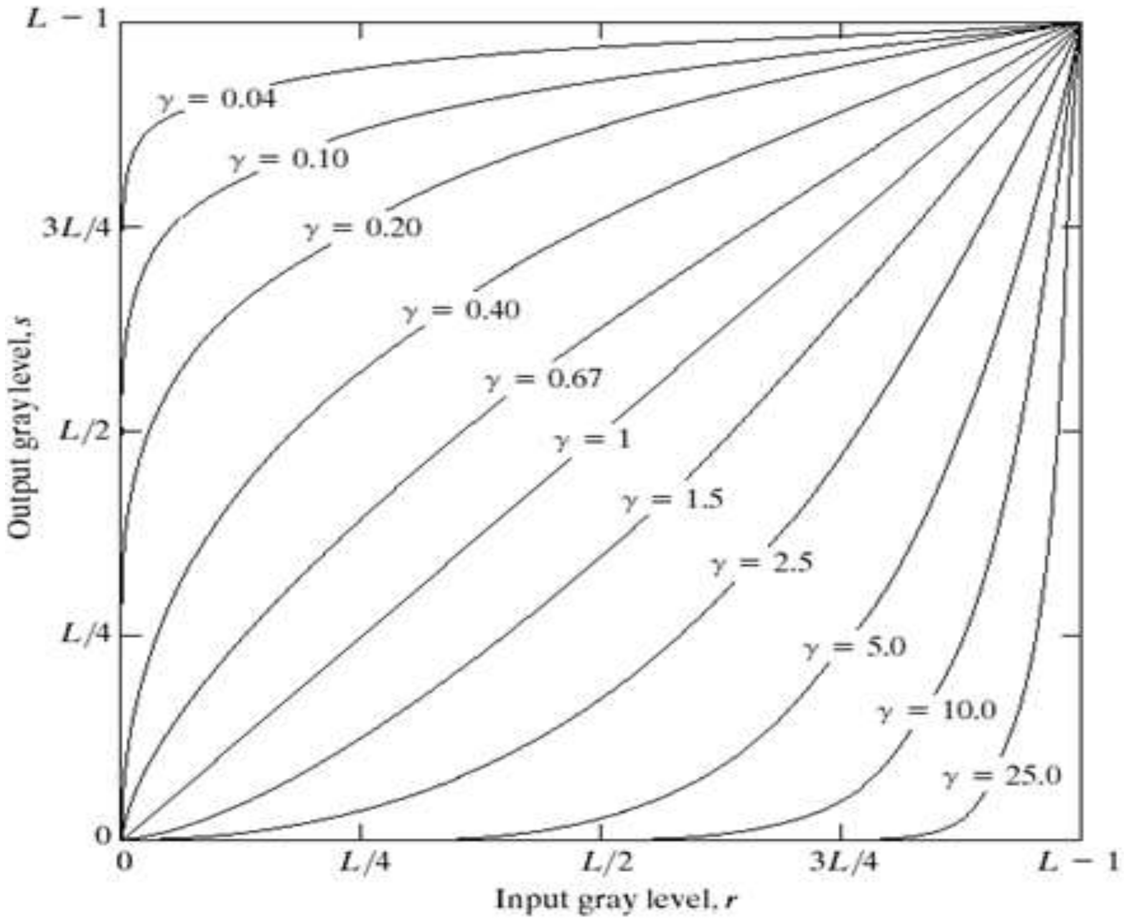
- Here, s is output pixel value, r is the input pixel value, c and $\gamma$ are real numbers.
- Variation in the value of $\gamma$ varies the enhancement of the images. This technique is commonly called as Gamma correction.

# POWER LAW TRANSFORMATION

- Different display monitors display images at different intensities and clarity  because every monitor has built in gamma correction in it with certain gamma ranges.
- A good monitor automatically corrects all the images displayed on it for the best contrast to give user the best experience.
- The difference between the log transformation function and the power law functions is that using the power law function a group of possible transformation curves can be obtained just by varying $\gamma$.

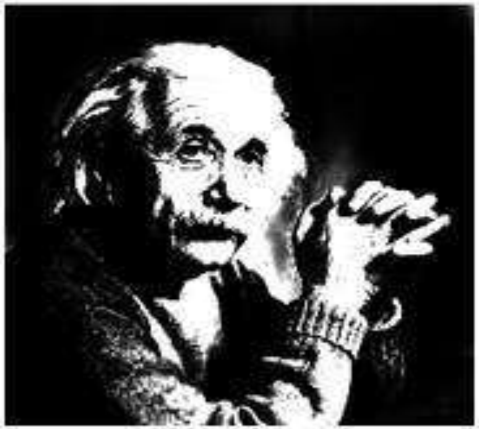
# POWER LAW TRANSFORMATION (CONTD)

Various values for γ

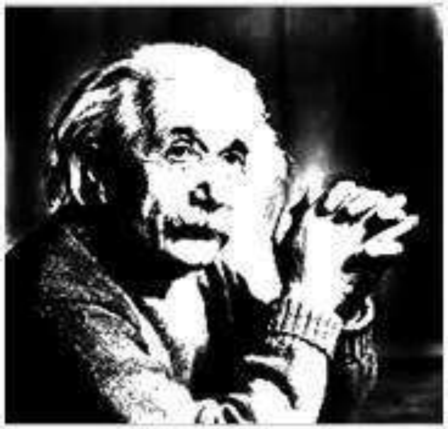


# Power law transformation example

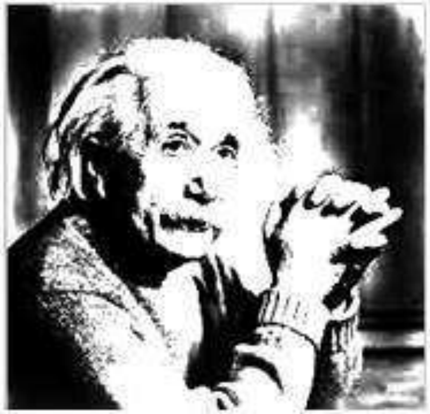
Gamma=10



Gamma=8



Gamma=6



# PIECEWISE LINEAR TRANSFORMATION FUNCTION

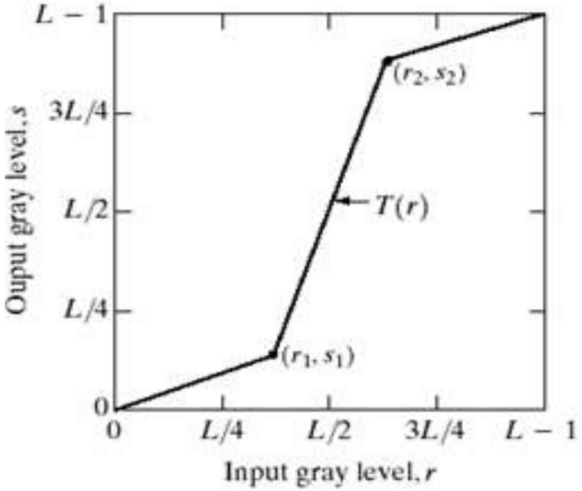
- There are three basic piecewise linear transformation functions.
  - Contrast stretching
  - Intensity level slicing
  - Bit plane slicing

# CONTRAST STRETCHING

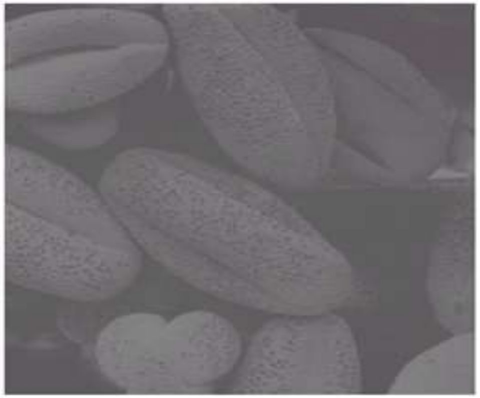
- It enhance the low contrast images.
- Contrast basically the difference between the intensity values of darker and brighter pixels.
- Contrast stretching is done in 3 ways,
  - Multiplying each input pixel intensity value with a constant scalar.
  - Using histogram equivalent.
  - Applying a transform which makes dark portion darker by assigning slope of < 1 and bright portion brighter by assigning slope of > 1.

# CONTRAST STRETCHING EXAMPLE

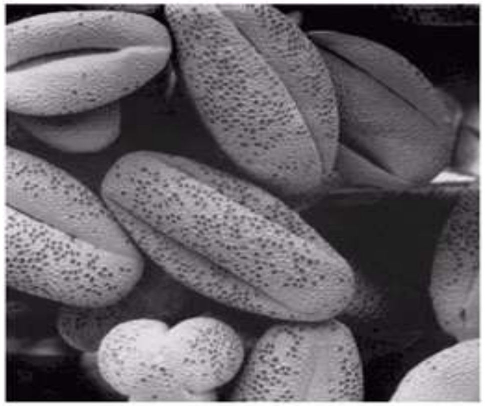
Transformation function



Low contrast image



Contrast stretching image



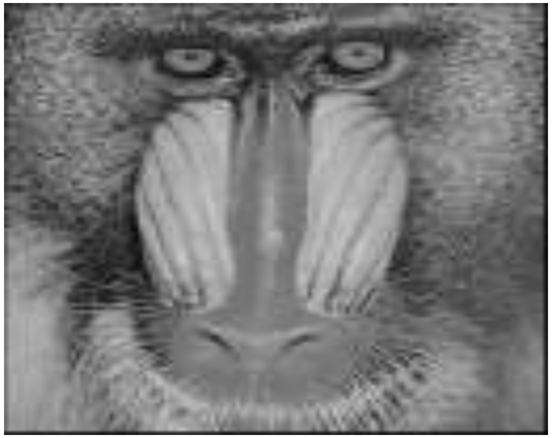
Point processing & Gray level transformations

# INTENSITY LEVEL SLICING

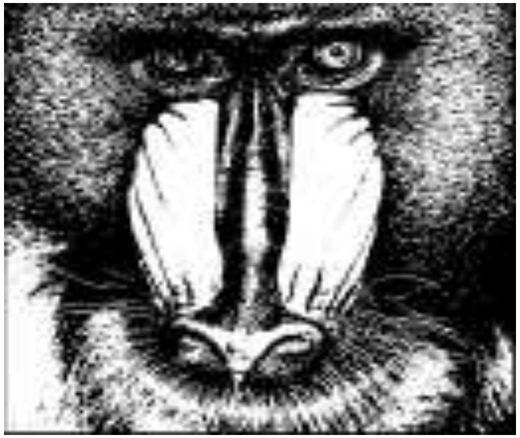
- Highlighting a specific range of gray levels in an image often is desired.
- Gray level slicing function can either emphasize a group intensities and diminish all other or it can emphasize a group of gray levels and leave the rest alone.

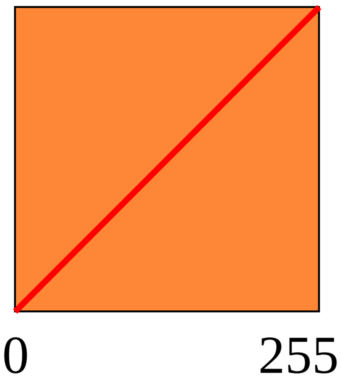
# Intensity level slicing example

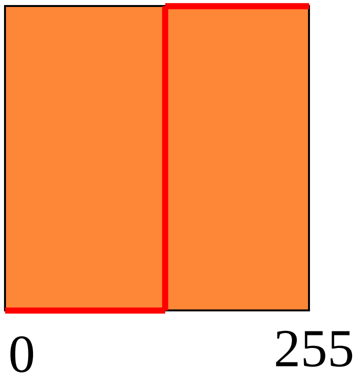
Input image



Output image



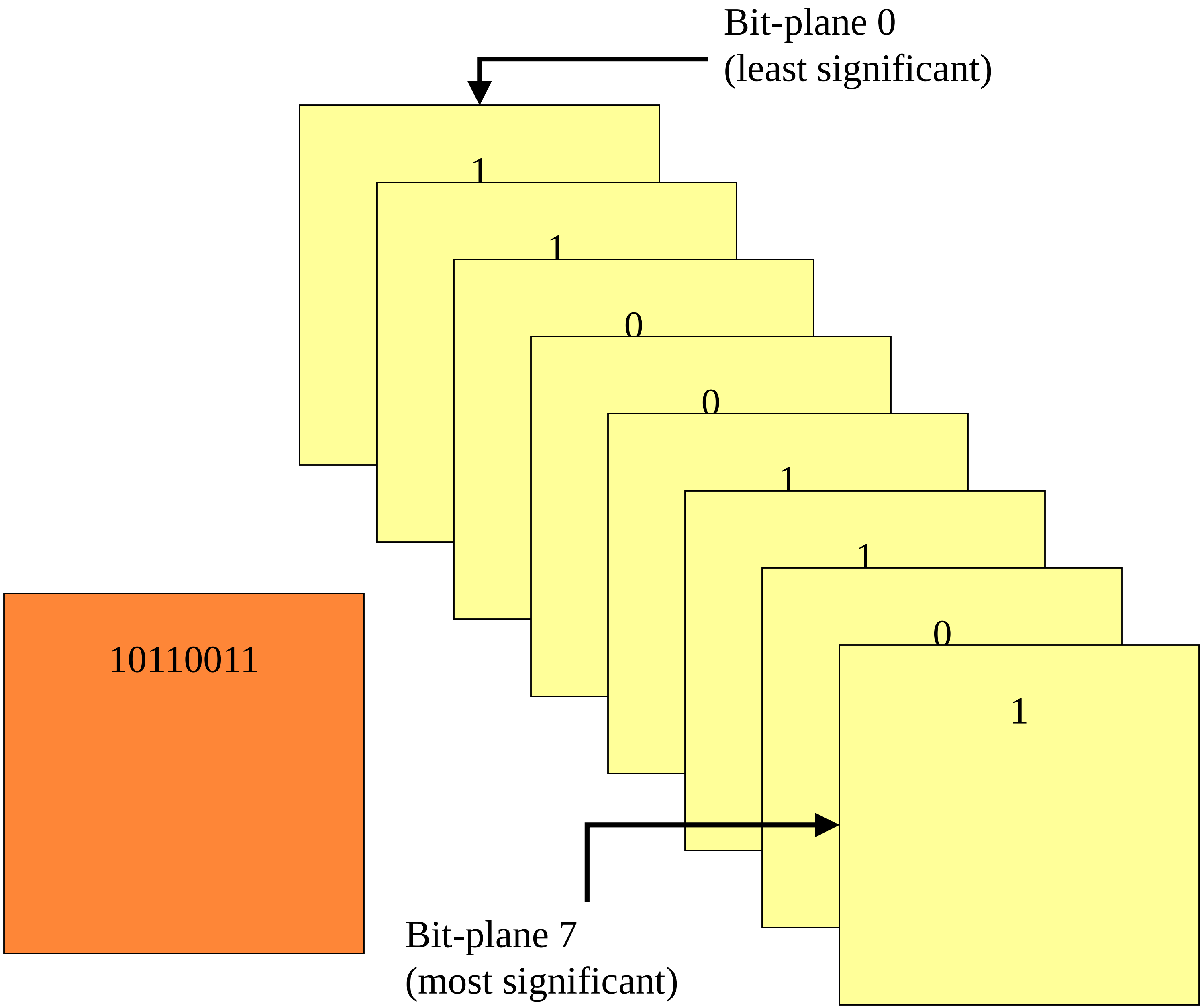




Point processing & Gray level transformations

# BIT PLANE SLICING

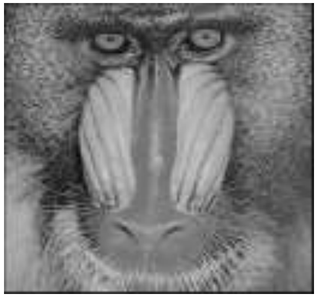
- Bit plane slicing the image is considered to be a stack of binary images.
- Instead of highlighting gray-level ranges, highlighting the contribution made to total image appearance by specific bits might be desired.
- The images closes to the bottom are least significant and the on top are most significant.

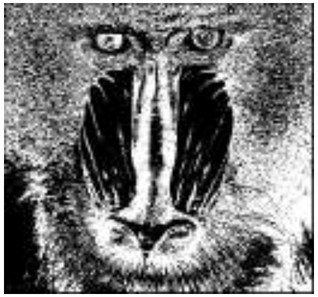


Bit-plane 0
(least significant)
1
1
0
0
1
1
0
1
10110011
Bit-plane 7
(most significant)

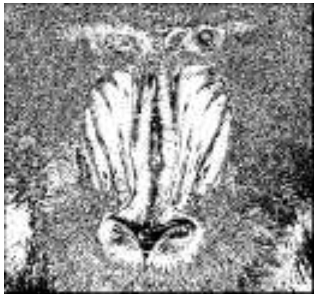
# Bit plane slicing example

Original image

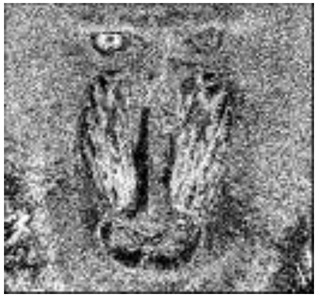




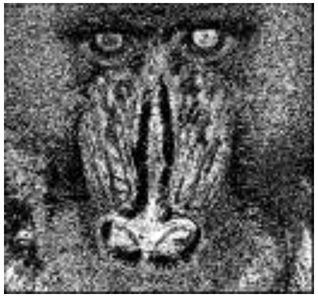
Bit plane 7



Bit plane 6



Bit plane 4



Bit plane 1

Point processing & Gray level transformations

# THANK YOU !