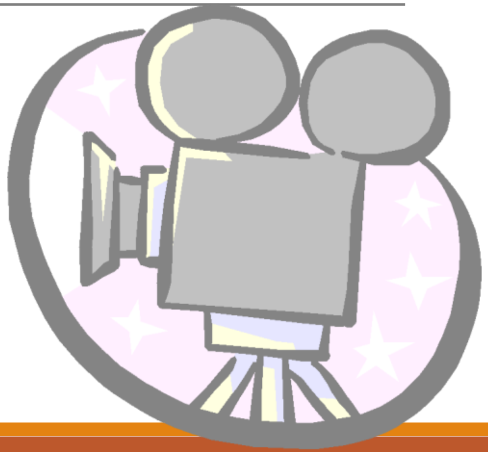
Image Processing

CS-317/CS-341



Outline

- Image Enhancement in spatial domain
- Some basic intensity transformation function
  - Image Negatives
  - Log Transformation
  - Power Law (Gamma) Transformation
- Piecewise Linear Transformation functions

Image Enhancement in the Spatial Domain

Principle Objective of Enhancement

Enhancement refers to accentuation, or sharpening, of image features such as edges, boundaries, or contrast to make graphic display more useful for display and analysis.

Does not increase the inherent information content in the data, only increase the dynamic range of chosen feature.

Principle Objective of Enhancement

Process an image so that the result will be more suitable than the original image for a specific application.

The suitableness is up to each application.

A method which is quite useful for enhancing an image may not necessarily be the best approach for enhancing another images

2 domains

Spatial Domain : (image plane)

- Techniques are based on direct manipulation of pixels in an image

Frequency Domain :

- Techniques are based on modifying the Fourier transform of an image

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

Image quality (Good or Bad?)

For human visual

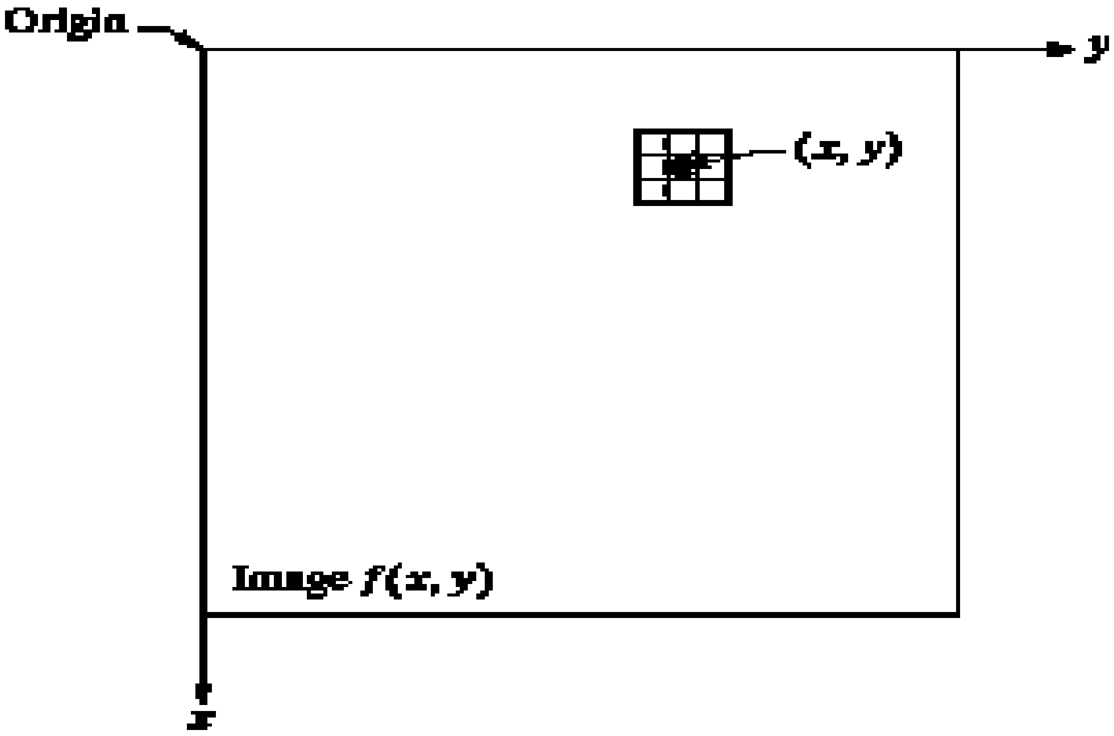
- The visual evaluation of image quality is a highly subjective process.
- It is hard to standardize the definition of a good image.

For machine perception

- The evaluation task is easier.
- A good image is one which gives the best machine recognition results.

A certain amount of trial and error usually is required before a particular image enhancement approach is selected.

Spatial Domain



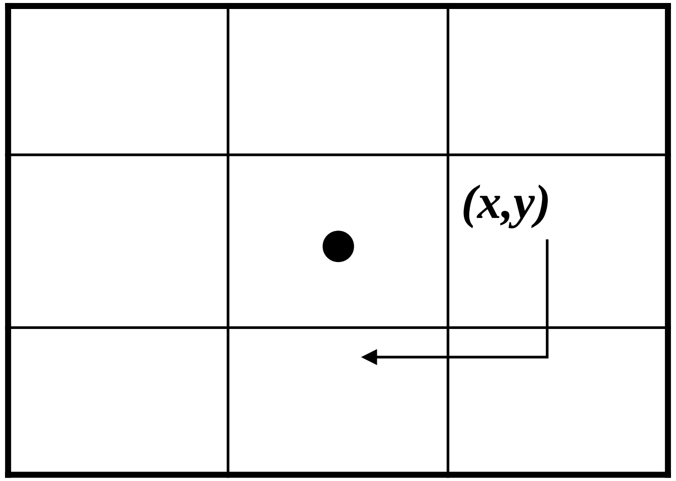
Procedures that operate directly on pixels.

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

where

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

Mask/Filter



Neighborhood of a point (x,y) can be defined by using a square/rectangular (common used) sub image area centered at (x, y).

The center of the sub image is moved from pixel to pixel starting at the top of the corner

Point Processing

Neighborhood = 1x1 pixel

g depends on only the value of f at (x,y)

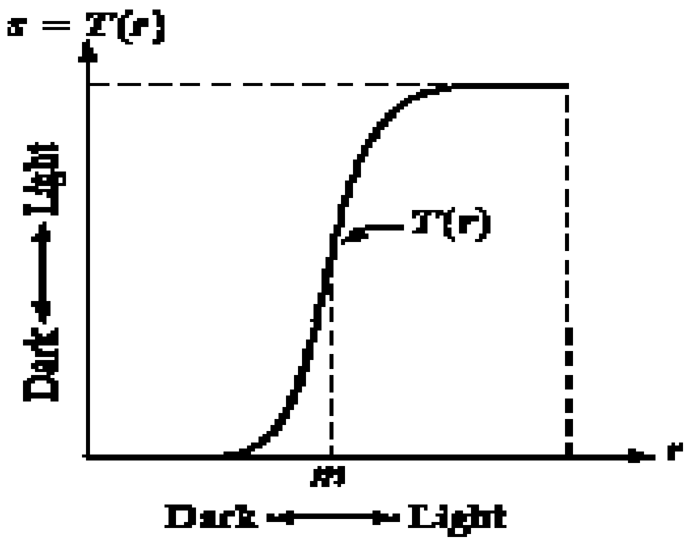
T = gray level (or intensity or mapping) transformation function

$$s = T(r)$$

Where

- r = gray level of $f(x,y)$
- s = gray level of $g(x,y)$

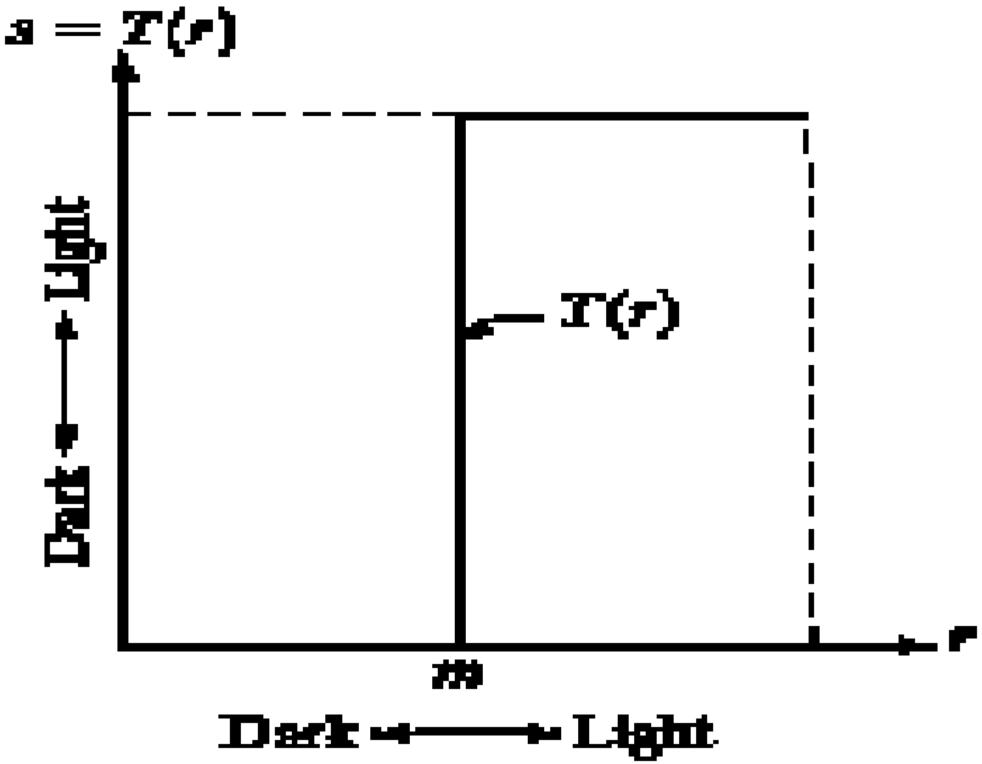
Contrast Stretching



Produce higher contrast than the original by

- darkening the intensity levels below m in the original image
- Brightening the levels above m in the original image

Thresholding



Produce a two-level (binary) image

Mask Processing or Filter

Neighborhood is bigger than 1x1 pixel

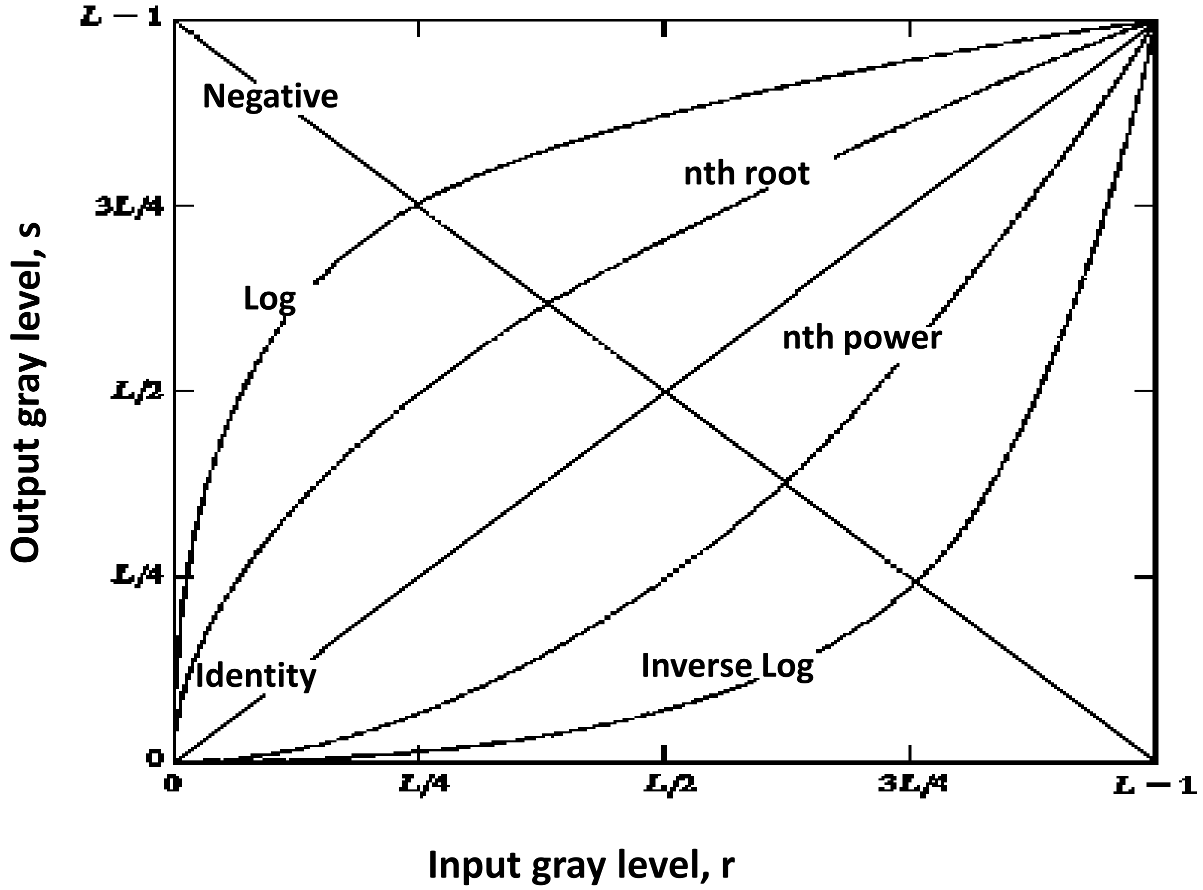
Use a function of the values of f in a predefined neighborhood of (x,y) to determine the value of g at (x,y)

The value of the mask coefficients determine the nature of the process

Used in techniques

- Image Sharpening
- Image Smoothing

3 basic gray-level transformation functions

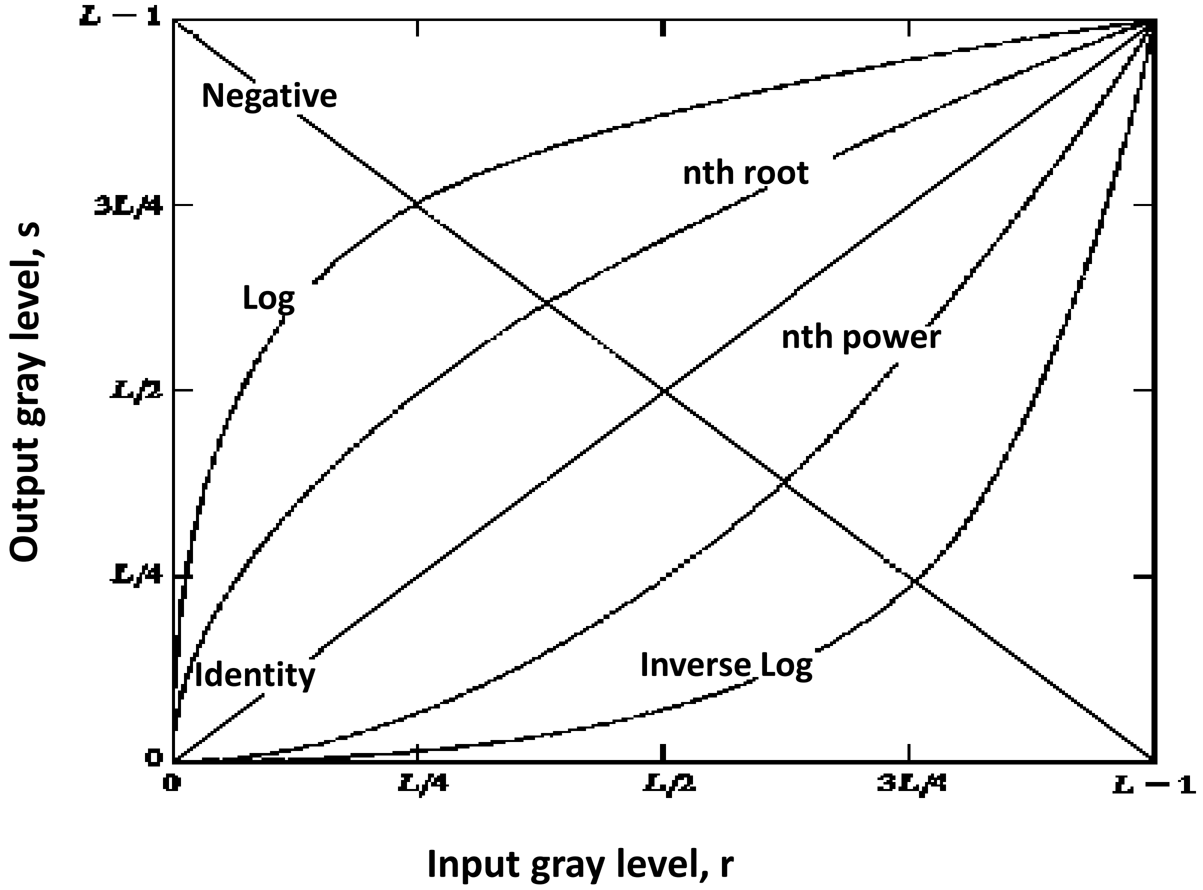


Linear function
- Negative and identity transformations

Logarithm function
- Log and inverse-log transformation

Power-law function
- n^{th} power and n^{th} root transformations

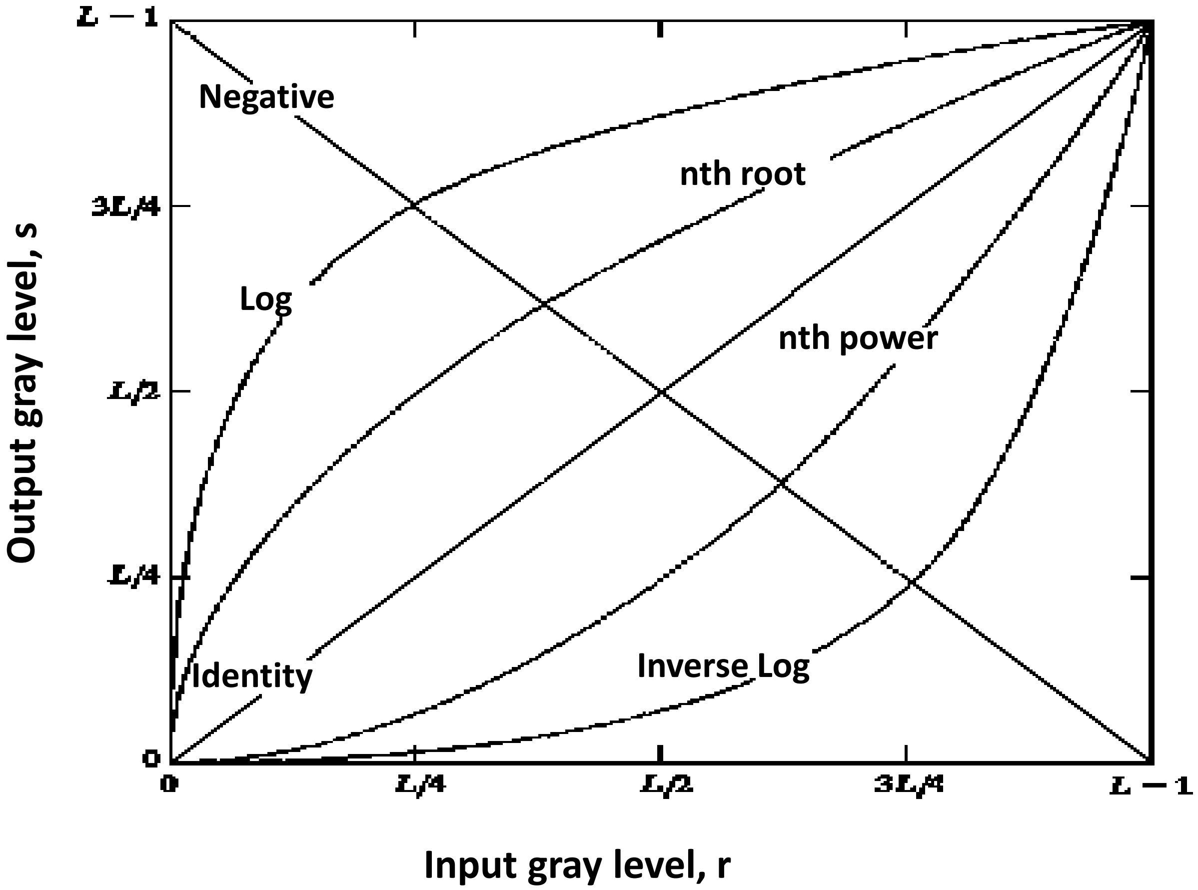
Identity function



Output intensities are identical to input intensities.

Is included in the graph only for completeness.

Image Negatives



An image with gray level in the range $[0,L-1]$ where $L = 2^n$; n = 1, 2…

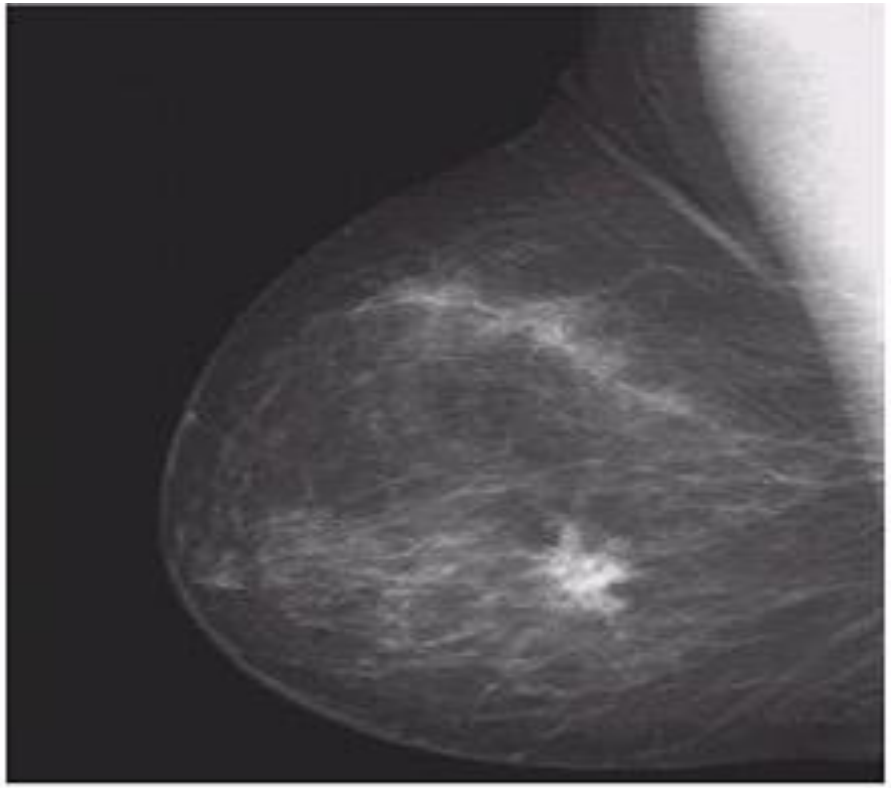
Negative transformation :

$$s = L - 1 - r$$

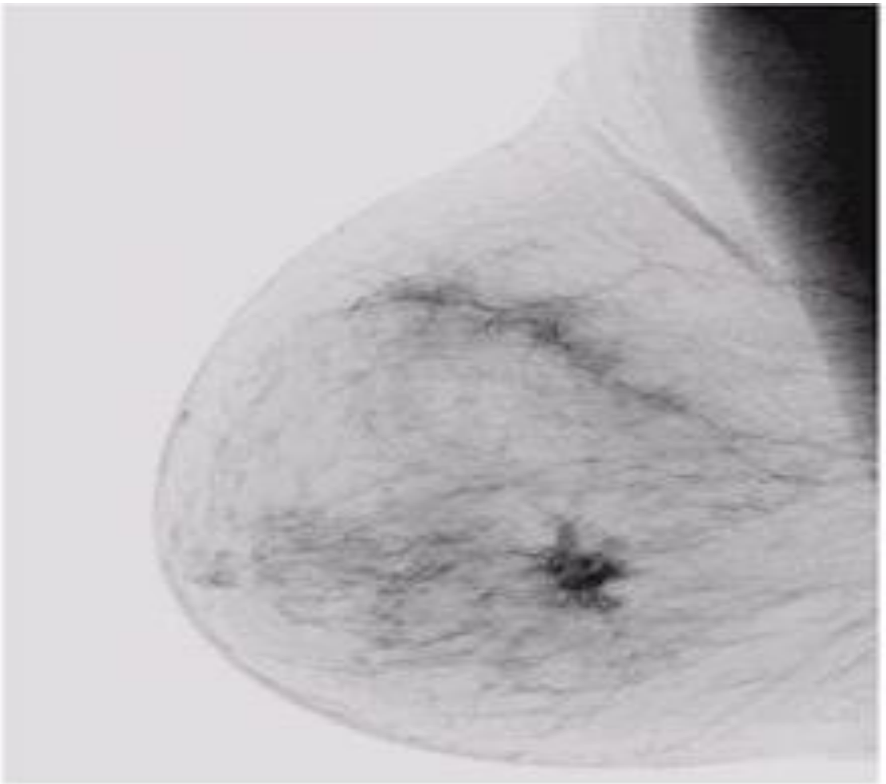
Reversing the intensity levels of an image.

Suitable for enhancing white or gray detail embedded in dark regions of an image, especially when the black area dominant in size.

Example of Negative Image

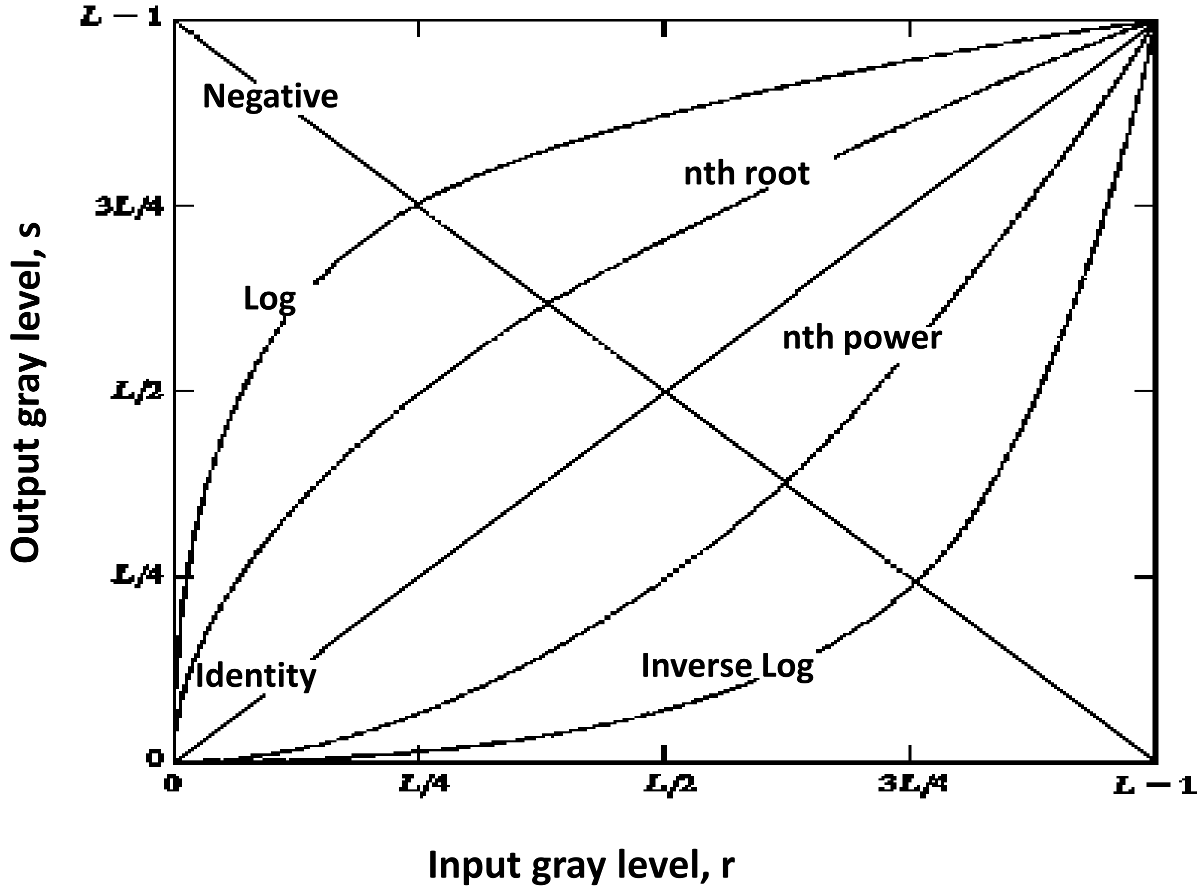


Original mammogram showing a small lesion of a breast



Negative Image : gives a better vision to analyze the image

Log Transformations



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

c is a constant and $r \geq 0$

Log curve maps a narrow range of low gray-level values in the input image into a wider range of output levels.

Used to expand the values of dark pixels in an image while compressing the higher-level values.

Log Transformations

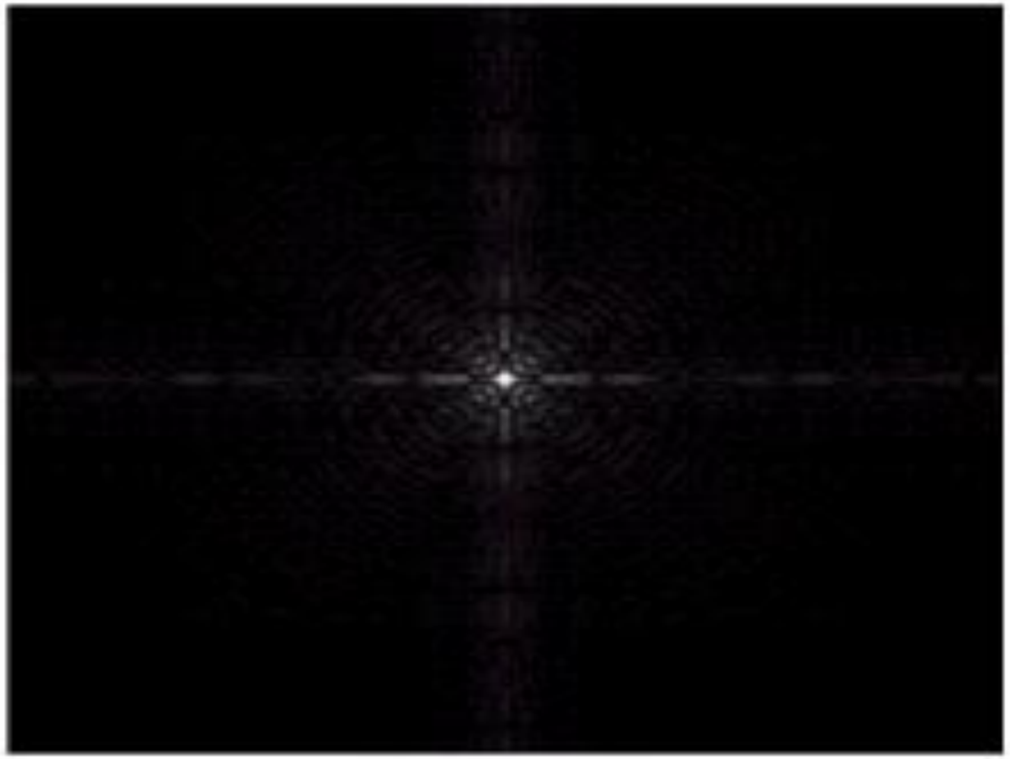
It compresses the dynamic range of images with large variations in pixel values

Example of image with dynamic range: Fourier spectrum image

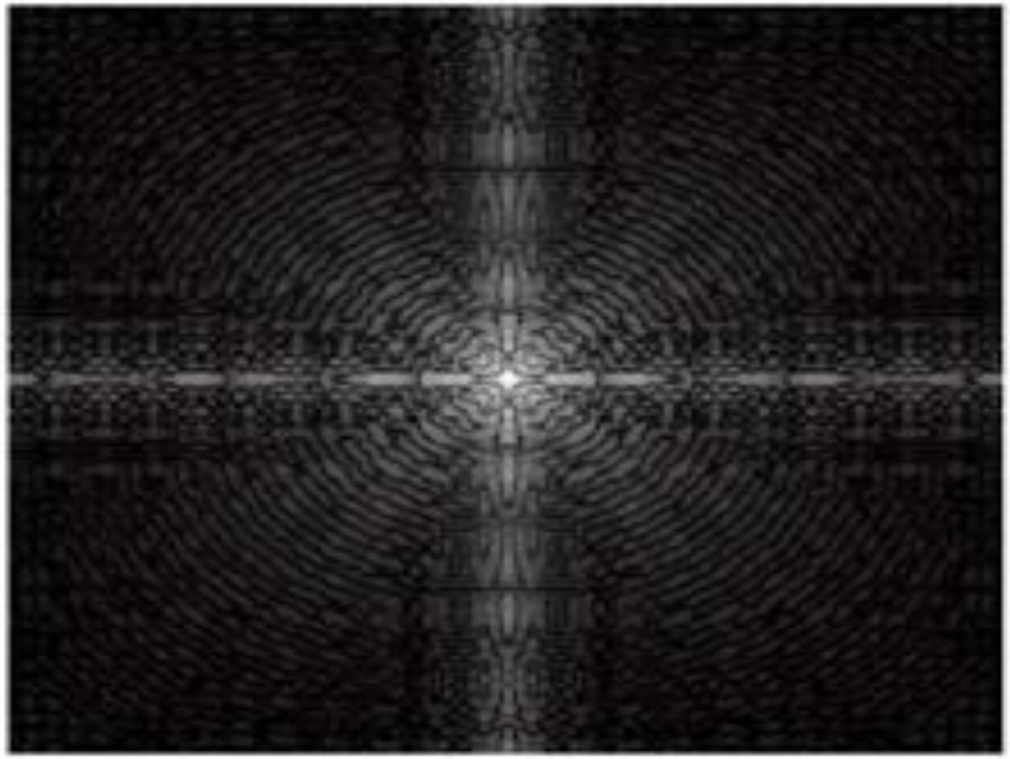
It can have intensity range from 0 to 10^6 or higher.

We can't see the significant degree of detail as it will be lost in the display.

Example of Logarithm Image



Fourier Spectrum with range = 0 to 1.5 x 10^6



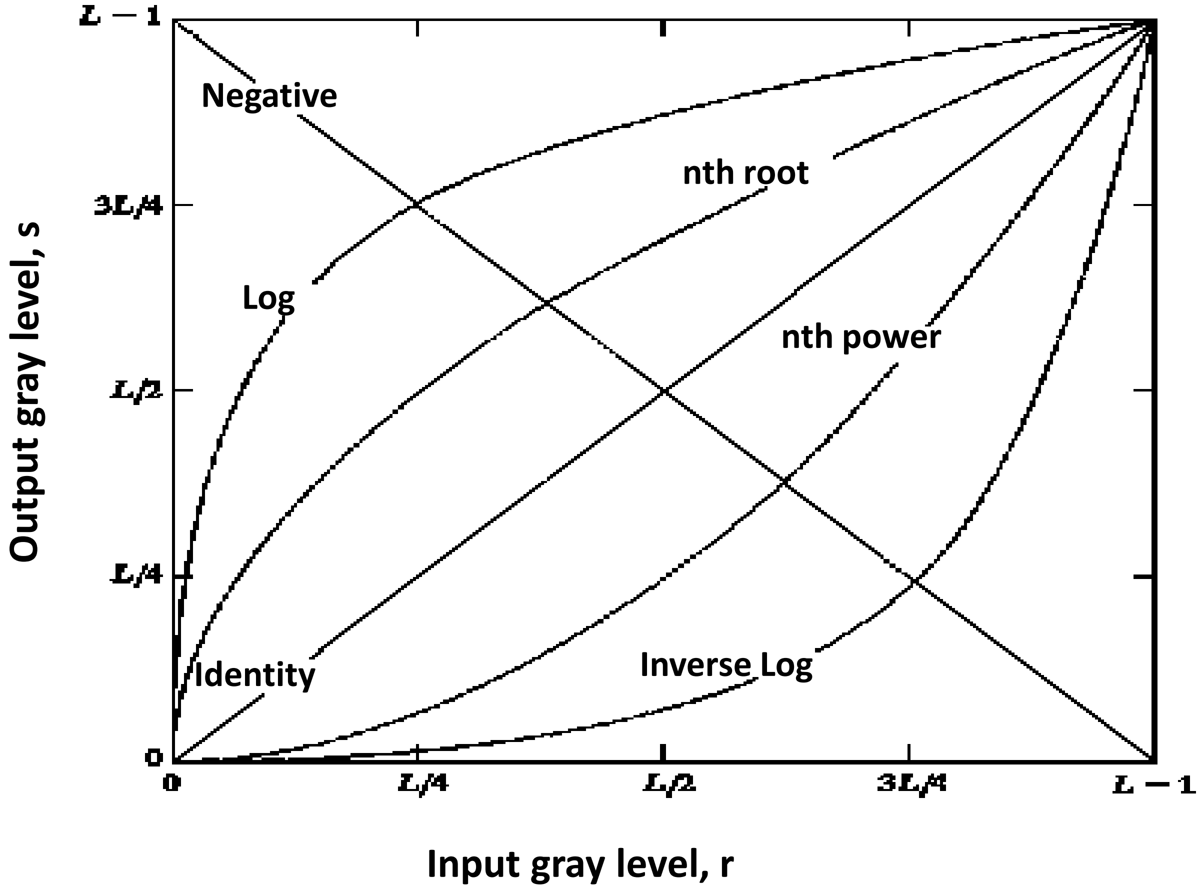
Result after apply the log transformation with c = 1, range = 0 to 6.2

Inverse Logarithm Transformations

Do opposite to the Log Transformations

Used to expand the values of high pixels in an image while compressing the darker-level values.

Power-Law Transformations



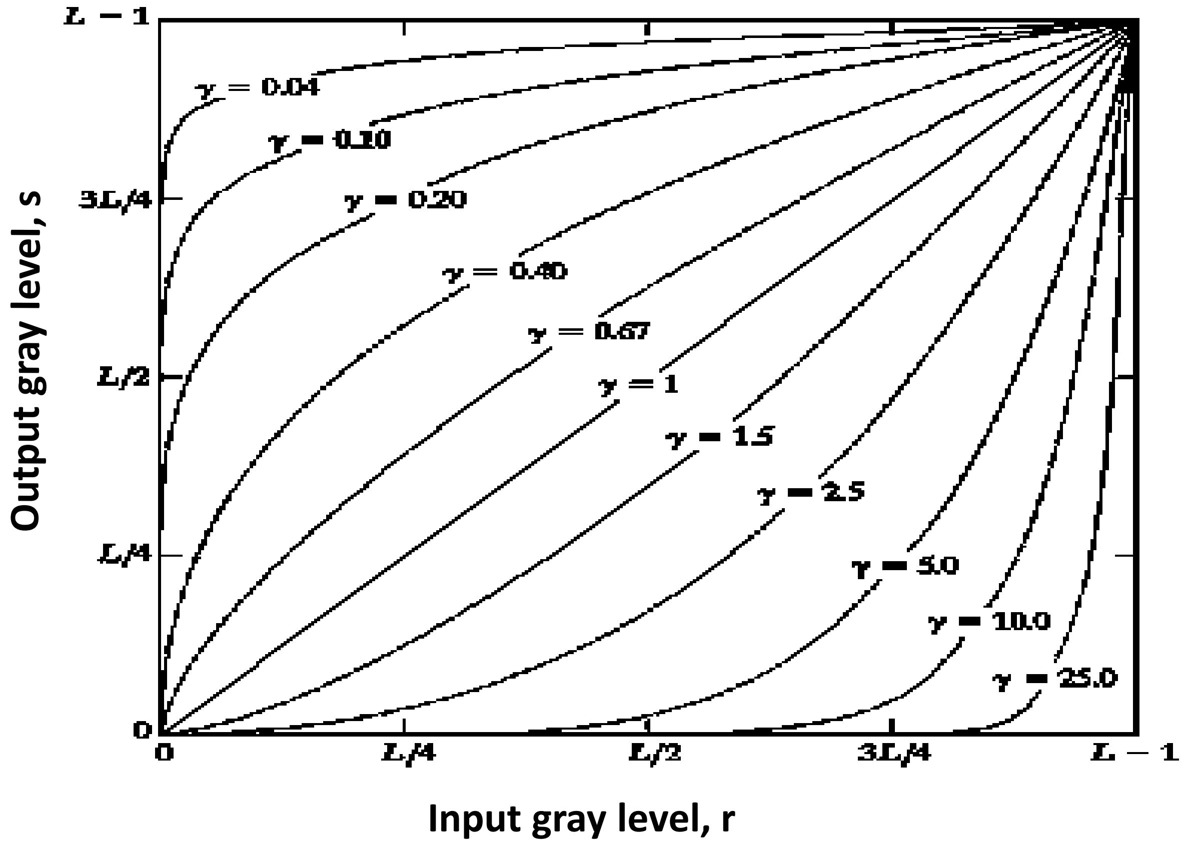
$$s = cr^{\gamma}$$

c and γ are positive constants

$\gamma > 1$ n[th] power transformations

$\gamma < 1$ n[th] root transformations

Power-Law Transformations



Plots of $s = cr^{\gamma}$ for various values of γ

(c = 1 in all cases)

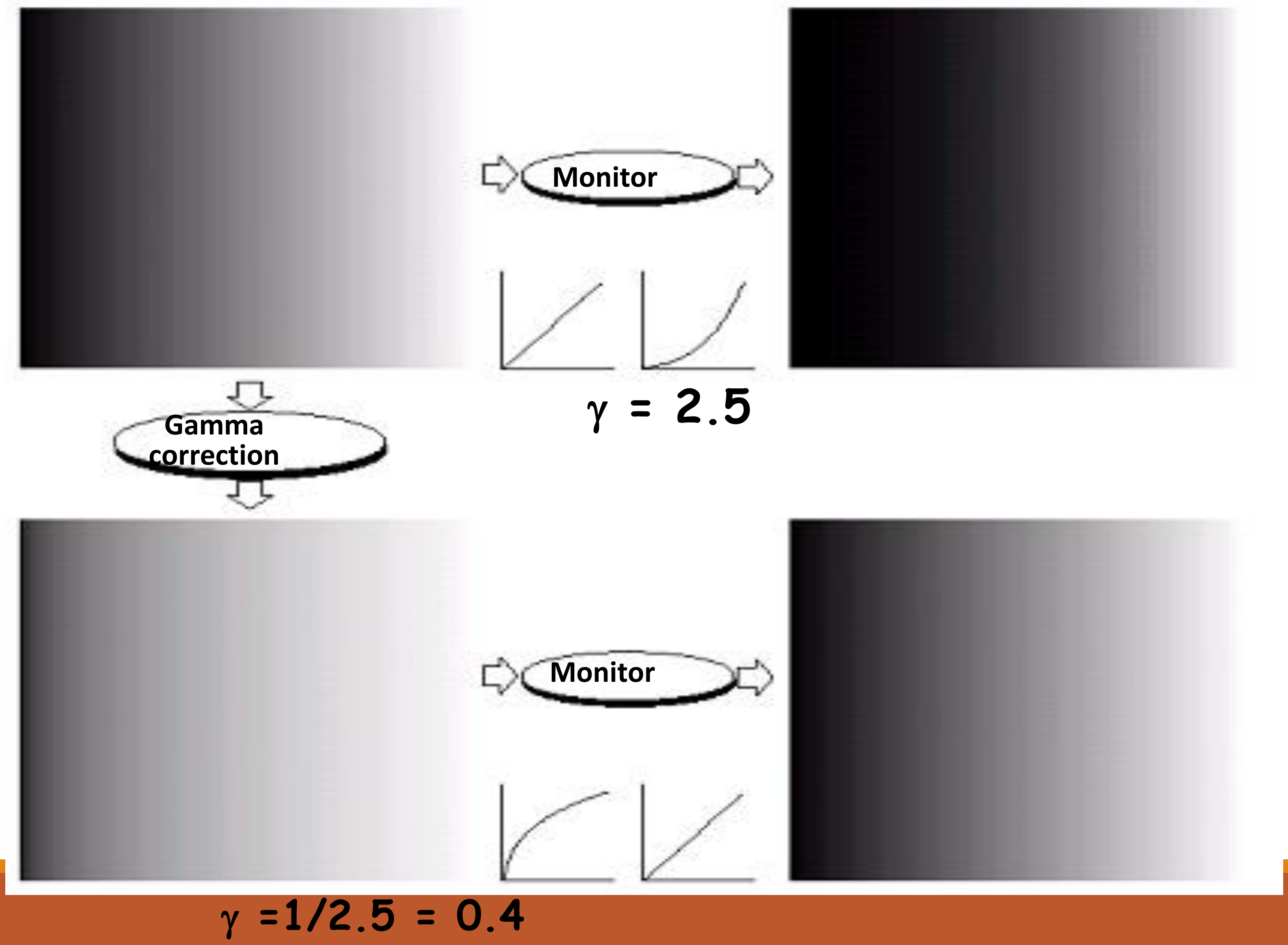
$$s = cr^{\gamma}$$

c and γ are positive constants

Power-law curves with fractional values of γ map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input levels.

$c = \gamma = 1$ ⇨ Identity function

Gamma correction



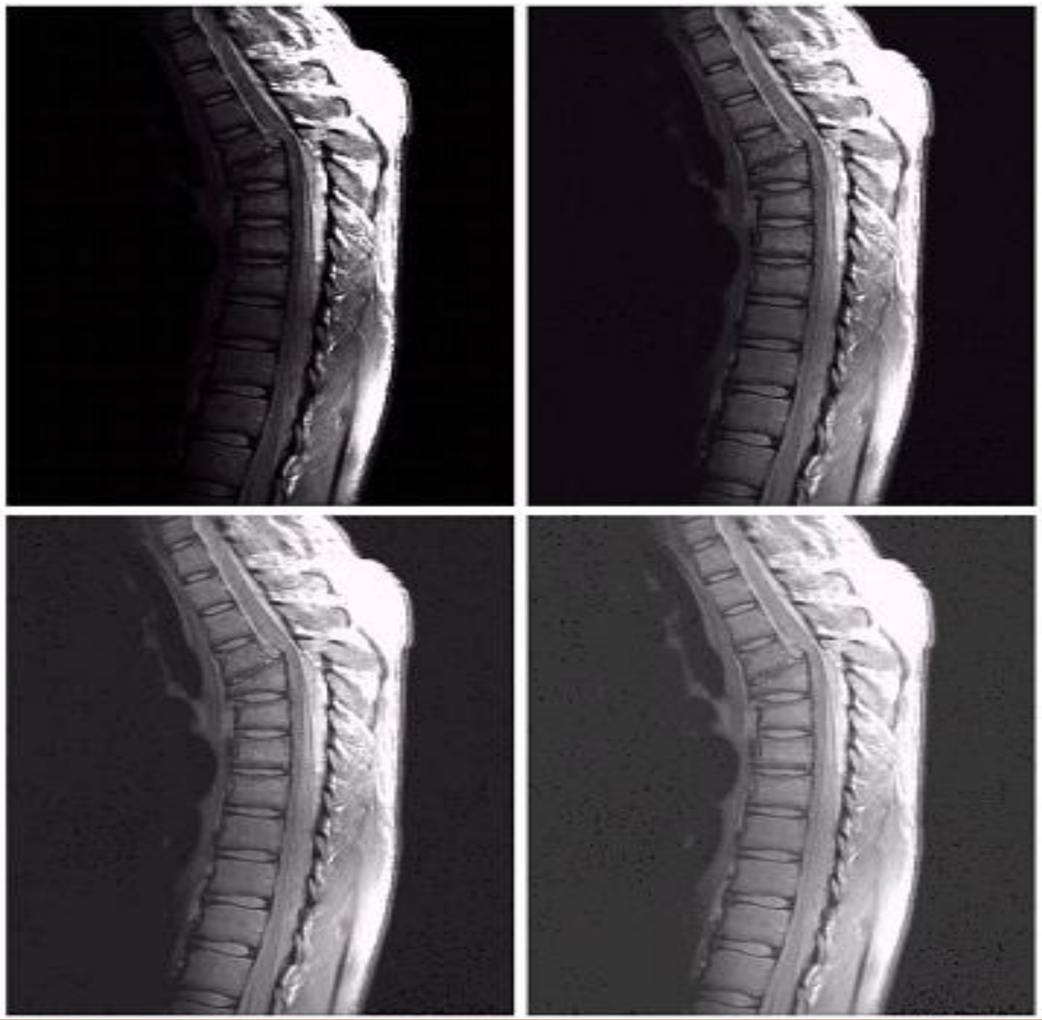
Cathode ray tube (CRT) devices have an intensity-to-voltage response that is a power function, with γ varying from 1.8 to 2.5

The picture will become darker.

Gamma correction is done by preprocessing the image before inputting it to the monitor with $s = cr^{1/\gamma}$

Another example : MRI

a	b
c	d



(a) a magnetic resonance image of an upper thoracic human spine with a fracture dislocation and spinal cord impingement

- The picture is predominately dark
- An expansion of gray levels are desirable ⇨ needs $\gamma < 1$

(b) result after power-law transformation with $\gamma = 0.6$, c=1

(c) transformation with $\gamma = 0.4$ (best result)

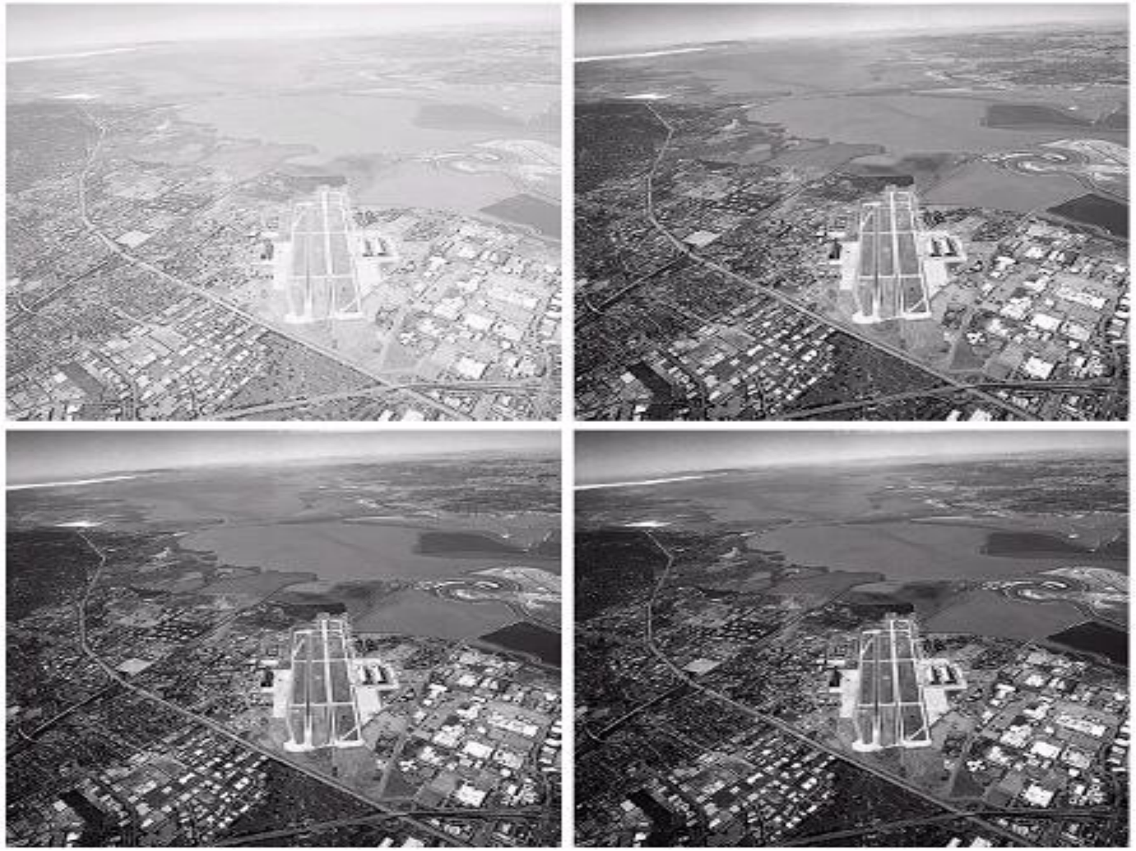
(d) transformation with $\gamma = 0.3$ (under acceptable level)

Effect of decreasing gamma

When the γ is reduced too much, the image begins to reduce contrast to the point where the image started to have very slight “wash-out” look, especially in the background

Another example

a	b
c	d



(a) image has a washed-out appearance, it needs a compression of gray levels ⇨ needs $\gamma > 1$

(b) result after power-law

transformation with $\gamma = 3.0$ (suitable)

(c) transformation with $\gamma = 4.0$

(suitable)

(d) transformation with $\gamma = 5.0$

(high contrast, the image has areas that are too dark, some detail is lost)

Piecewise-Linear Transformation Functions

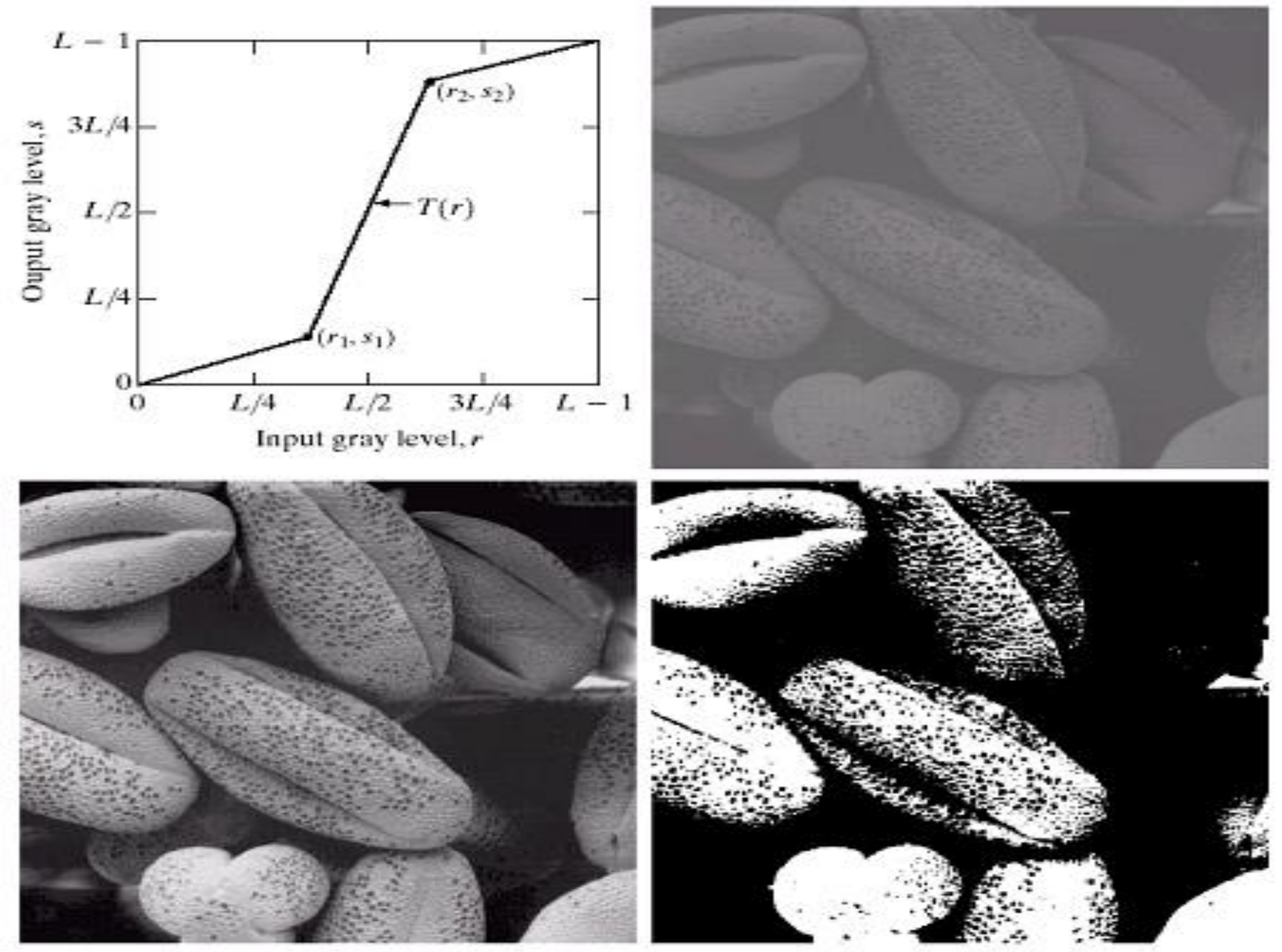
Advantage:

- The form of piecewise functions can be arbitrarily complex

Disadvantage:

- Their specification requires considerably more user input

Contrast Stretching



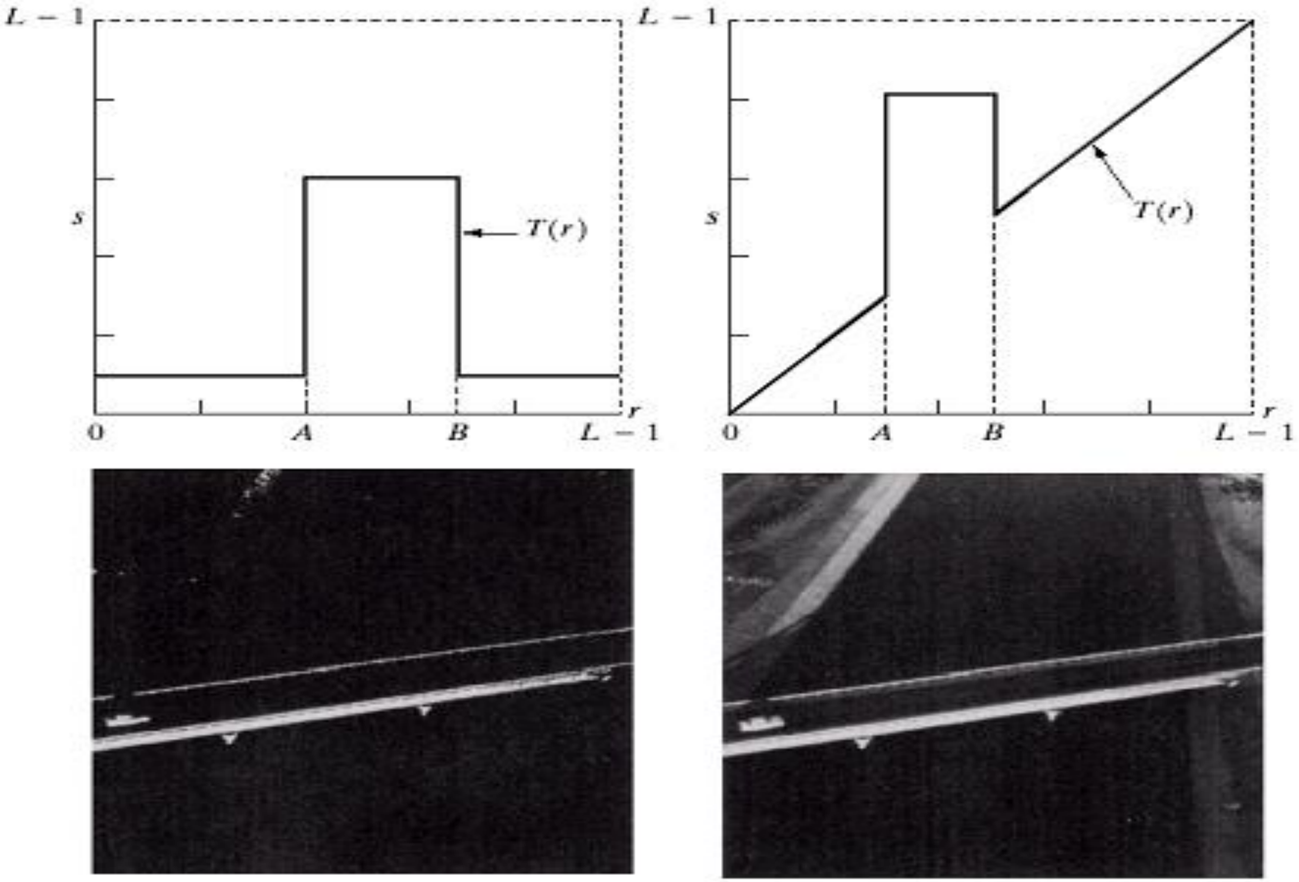
increase the dynamic range of the gray levels in the image

(b) a low-contrast image : result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture of image acquisition

(c) result of contrast stretching: $(r_1,s_1) = (r_{min},0)$ and $(r_2,s_2) = (r_{max},L-1)$

(d) result of thresholding

Gray-level slicing



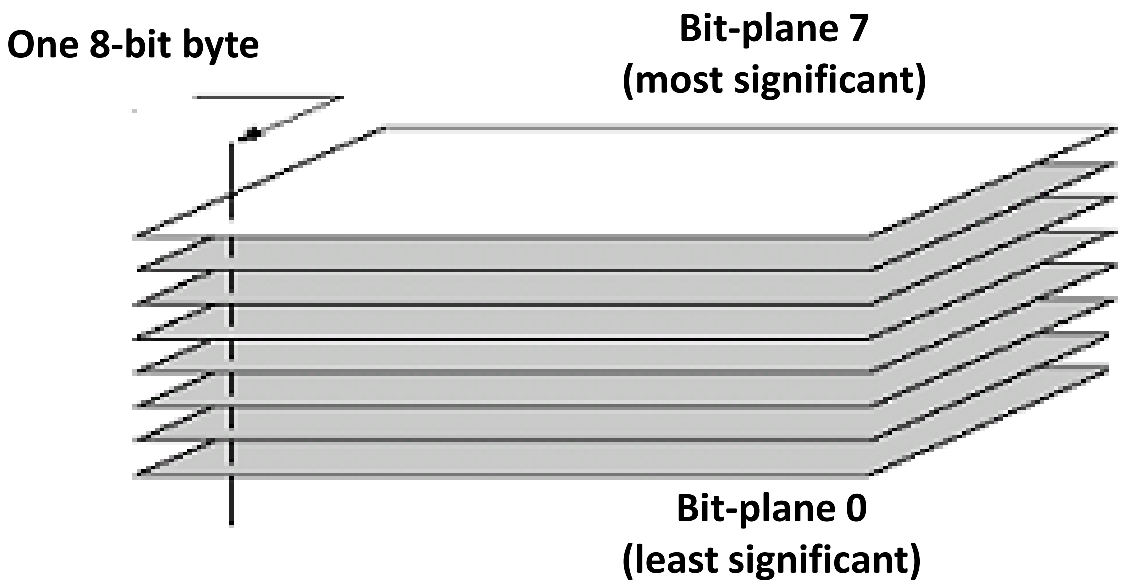
Highlighting a specific range of gray levels in an image

- Display a high value of all gray levels in the range of interest and a low value for all other gray levels

(a) transformation highlights range [A,B] of gray level and reduces all others to a constant level

(b) transformation highlights range [A,B] but preserves all other levels

Bit-plane slicing



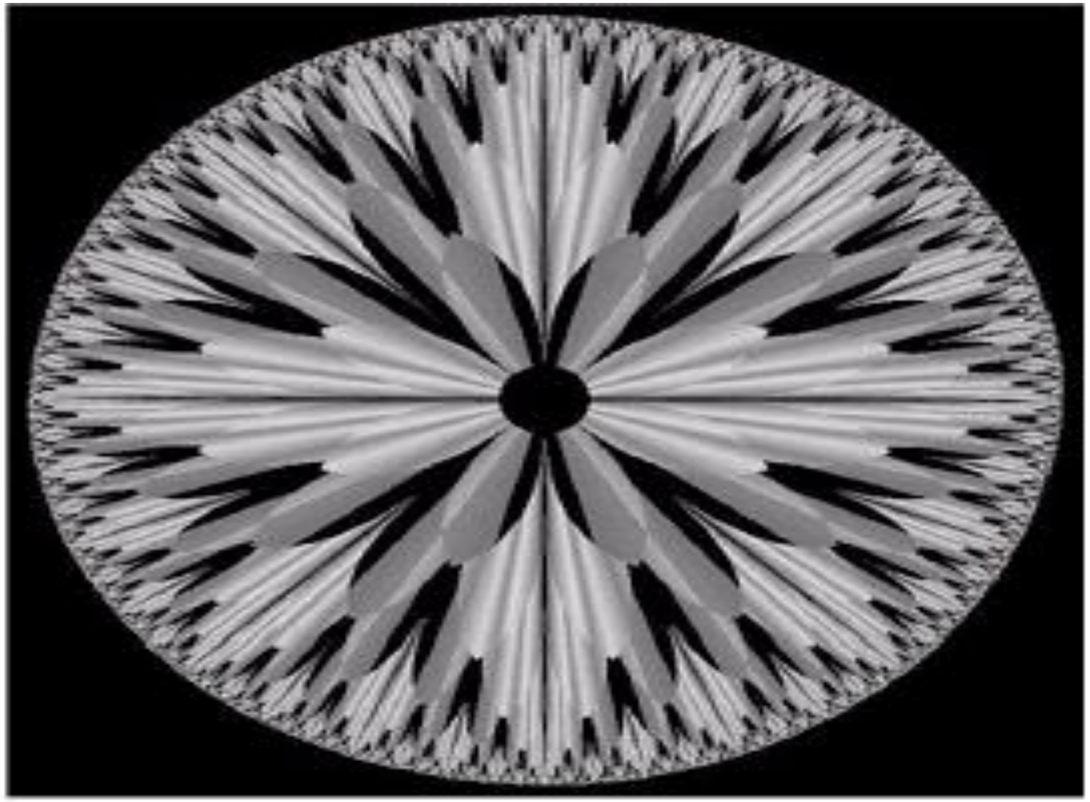
Highlighting the contribution made to total image appearance by specific bits

Suppose each pixel is represented by 8 bits

Higher-order bits contain the majority of the visually significant data

Useful for analyzing the relative importance played by each bit of the image

Example

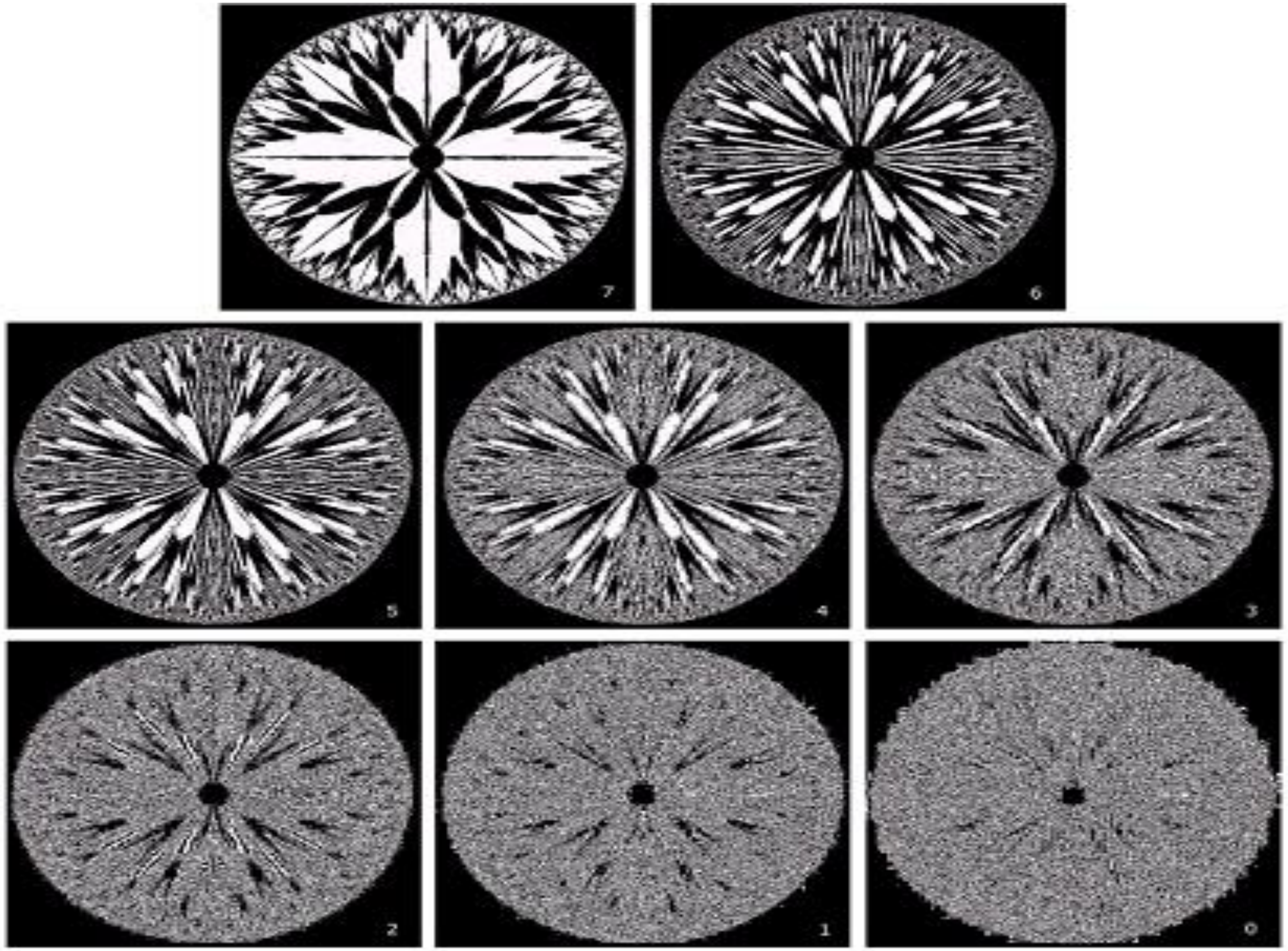


An 8-bit fractal image

The (binary) image for bit-plane 7 can be obtained by processing the input image with a thresholding gray-level transformation.

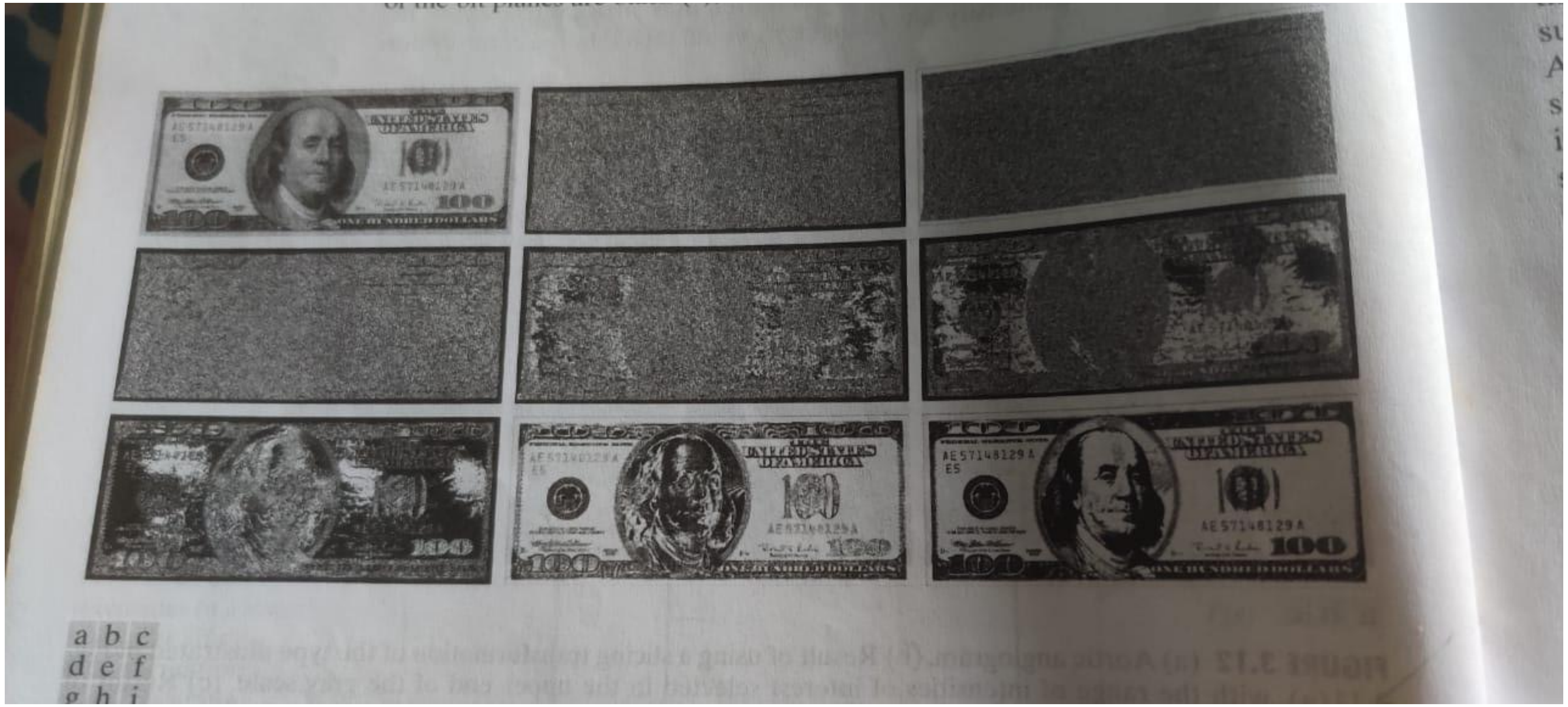
- Map all levels between 0 and 127 to 0
- Map all levels between 128 and 255 to 1

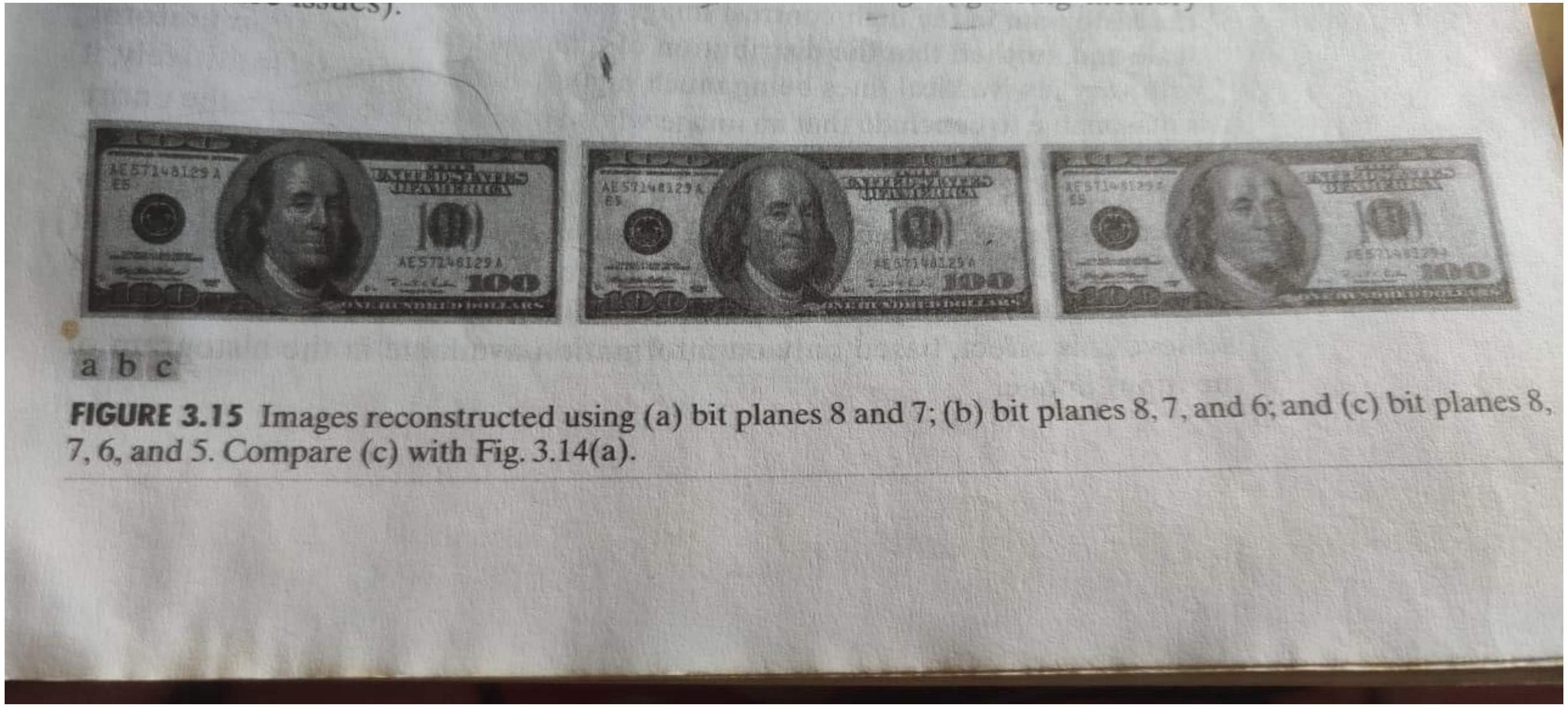
8 bit planes



Bit-plane 7	Bit-plane 6	
Bit-plane 5	Bit-plane 4	Bit-plane 3
Bit-plane 2	Bit-plane 1	Bit-plane 0

8 bit planes





a b c

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).

Suggested Readings

- **Digital Image Processing by Rafel Gonzalez, Richard Woods, Pearson Education India, 2017.**
- **Fundamental of Digital image processing by A. K Jain, Pearson Education India, 2015.**

Thank you