Lecture 5

Image Enhancement in Spatial Domain - I

Background

Intensity Transformation Functions:

- Negative
- Log
- Gamma
- Contrast Stretching
- Intensity-level Slicing
- Bit-plane Slicing

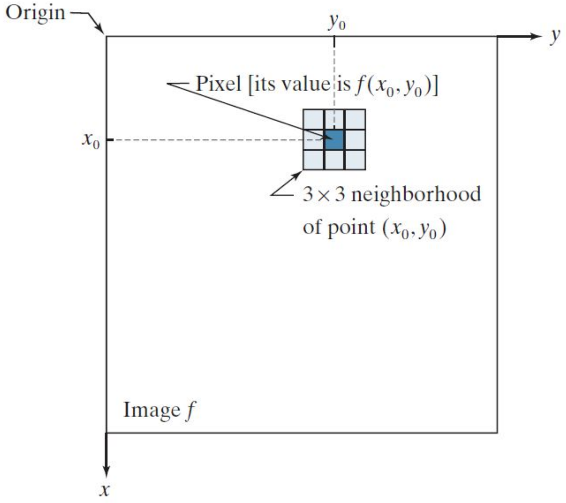
- The term spatial domain refers to the image plane itself.
- Image processing methods in spatial domain are based on direct manipulation of pixels in an image.
- Principal categories of spatial processing:
  - Intensity Transformations
  - Spatial Filtering
- Intensity transformations operate on single pixels of an image for tasks such as contrast manipulation and image thresholding.
- Spatial filtering performs operations on the neighborhood of every pixel in an image. Examples of spatial filtering include image smoothing and sharpening.

The spatial domain processes we discuss in this lecture are based on the expression

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

where $f(x,y)$ is an input image, $g(x,y)$ is the output image, and T is an operator on f defined over a neighborhood of point (x,y).

A 3×3 neighborhood about a point (x_0, y_0) in an image. The neighborhood is moved from pixel to pixel in the image to generate an output image. Recall from Chapter 2 that the value of a pixel at location (x_0, y_0) is $f(x_0, y_0)$, the value of the image at that location.



The smallest possible neighborhood is of size 1×1. In this case, g depends only on the value of f at a single point (x, y)

T becomes an *intensity transformation function* of the form

$$s = T(r)$$

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

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

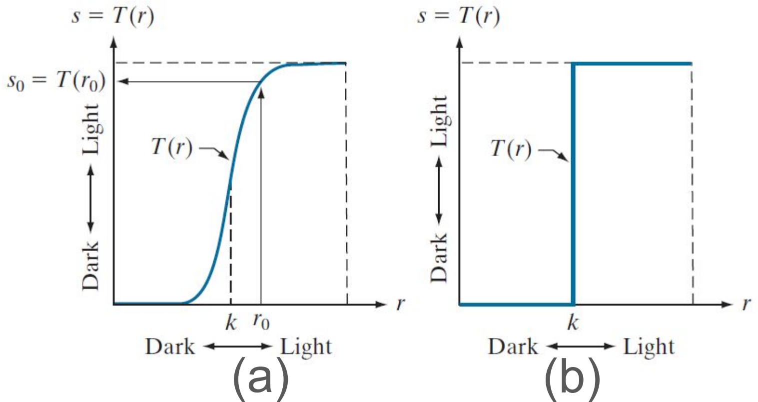


Image enhancement

- Manipulate an image such that the result is more suitable than the original for a specific application.
- A method that is quite useful for enhancing X-ray images may not be the best approach for enhancing infrared images.
- The viewer is the ultimate judge of how well a particular method works, which is difficult to quantify.
- However, in machine perception, enhancement is easier to quantify. In automated character-recognition system, the most appropriate enhancement method would be the one with the best recognition rate.

Intensity Transformations

IMAGE NEGATIVES

LOG TRANSFORMATIONS

POWER-LAW (GAMMA) TRANSFORMATIONS

PIECEWISE LINEAR TRANSFORMATION FUNCTIONS

Image Negatives

$$s = L - 1 - r$$

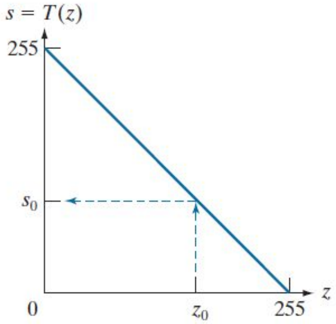
L=number of intensity levels

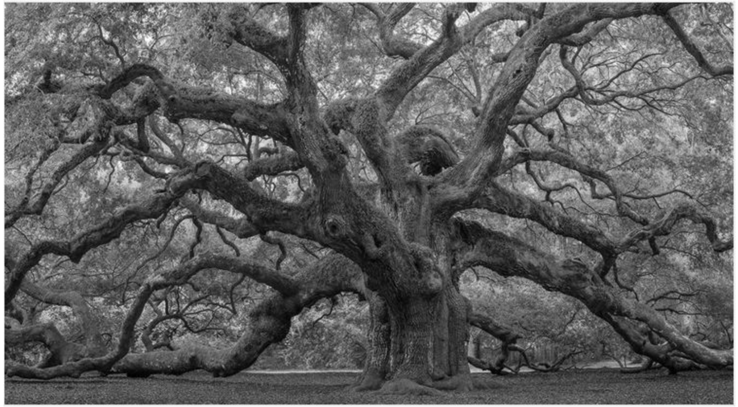
r=pixel intensity level of input image

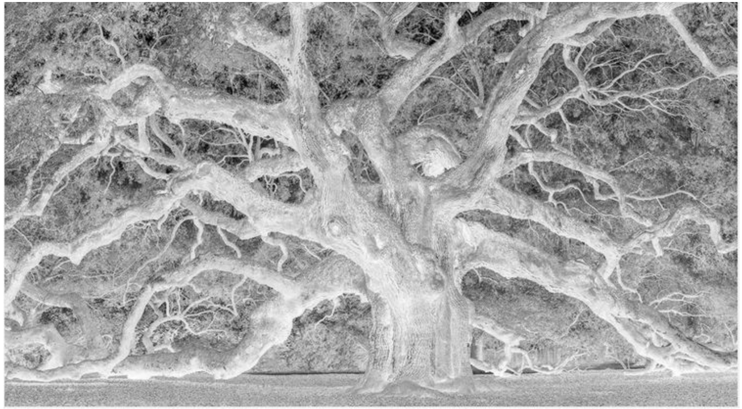
s=output intensity level of output image

Darkest pixels become brightest; brightest pixels become darkest.

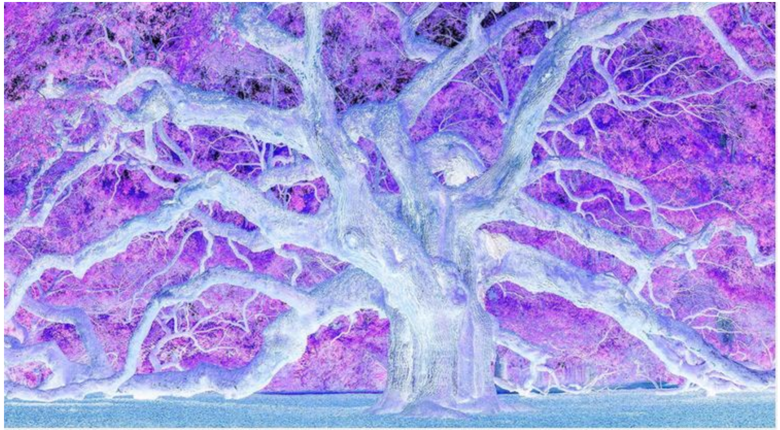
Intensity transformation function used to obtain the digital equivalent of photographic negative of an 8-bit image..





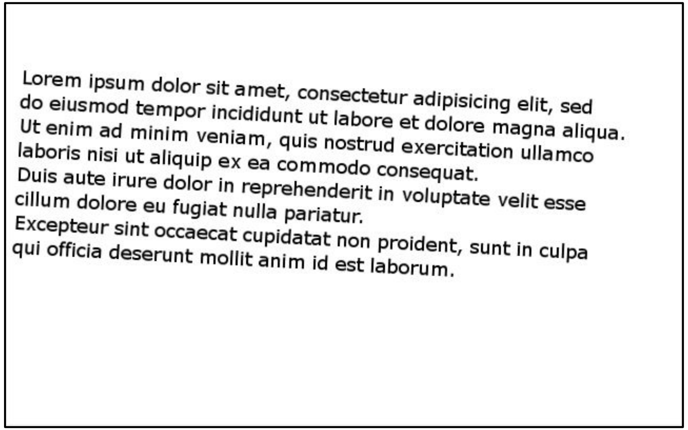


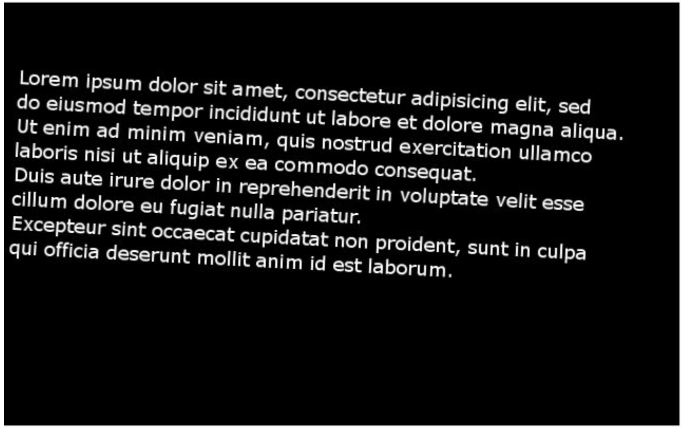
We can do it for color images also by applying it on individual channels



Applications

Text image processing

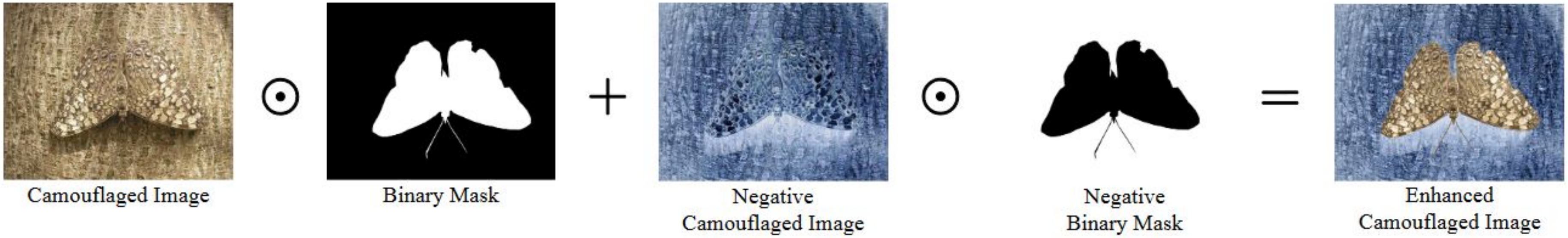




Text becomes the foreground

Another application

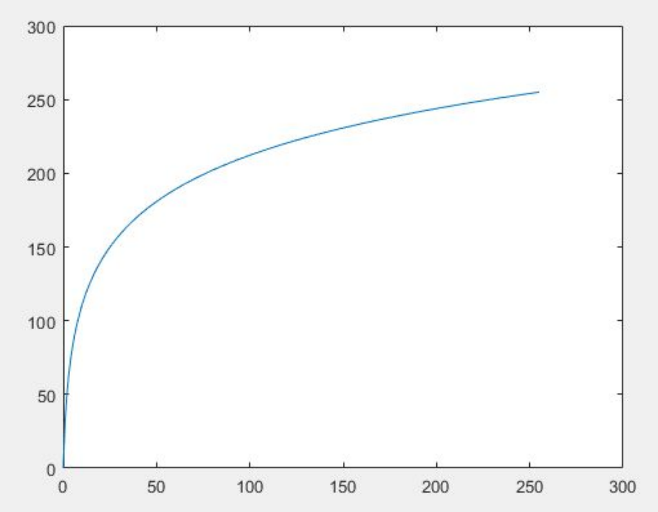
Enhancing camouflaged objects



A. Gupta, K. R. Jerripothula and T. Tillo, "CIRCOD: Co-Saliency Inspired Referring Camouflaged Object Discovery," *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Tucson, AZ, USA, 2025, pp. 8313-8323, doi: 10.1109/WACV61041.2025.00806.

Log Transforms

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



c= a constant

r=pixel intensity level of input image

s=output intensity level of output image

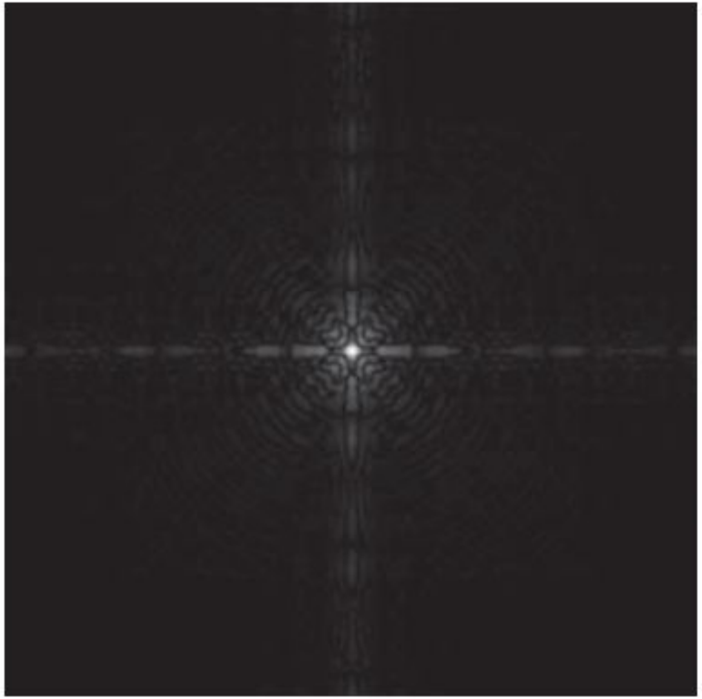
It maps a narrow range of low intensity values in the

input image into a wider range of output levels.

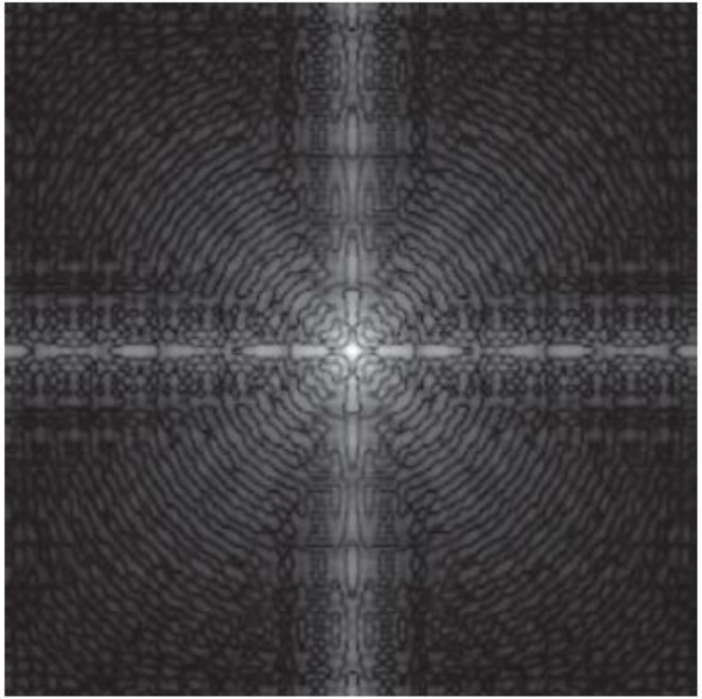
Conversely, higher values of input levels are mapped to a narrower range in the output.

Here,
c=255/log(256)

(a) Fourier spectrum displayed as a grayscale image. (b) Result of applying the log transformation with $c = 1$. Both images are scaled to the range $[0, 255]$.



(a)



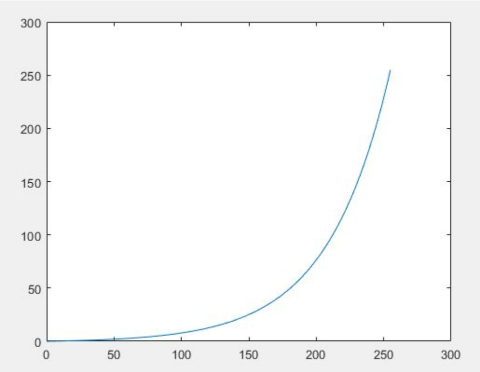
(b)

There is also something called inverse log transformation:

s=exp(r/c)-1

Basically, it’s the inverse function of the log transformation function.

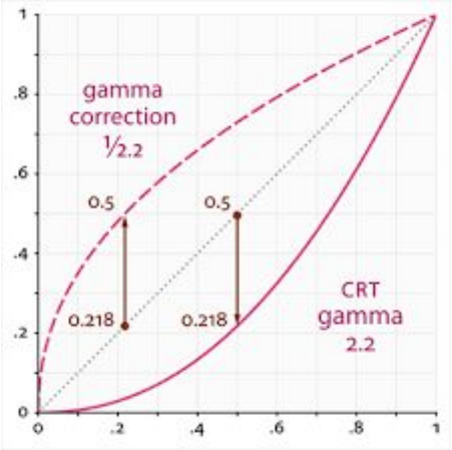
It gives opposite results: compresses lower values; expands higher values



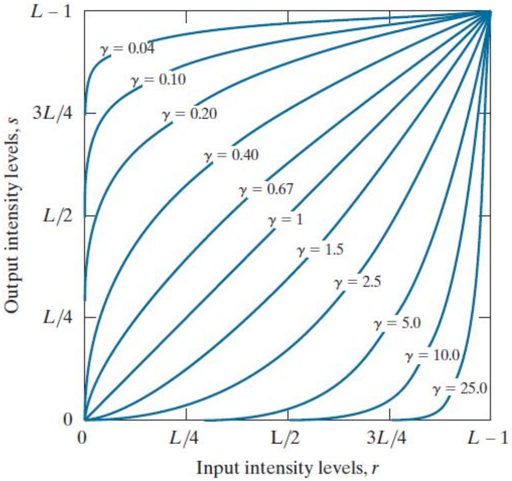
Here,
c=255/log(256)

POWER-LAW (GAMMA) Transformation

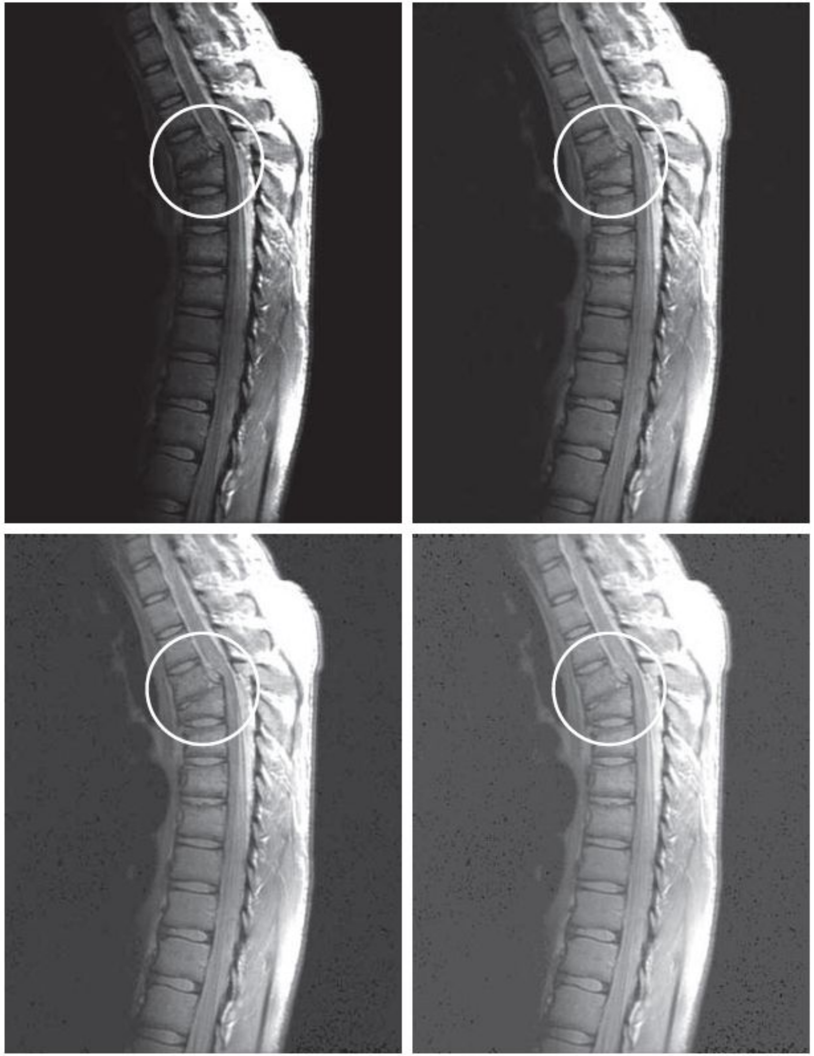
$$s = cr^{\gamma}$$



A family of curves gets generated



The response of many devices used for image capture, printing, and display obey a power law. By convention, the exponent in a power-law equation is referred to as gamma



a b
c d

(a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) 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.)

a b
c d

(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3-5) with $\gamma = 3.0, 4.0$, and 5.0, respectively. ($c = 1$ in all cases.) (Original image courtesy of NASA.)

What should be the value of gamma?

If the ROI/Image has more bright values, use gamma>1

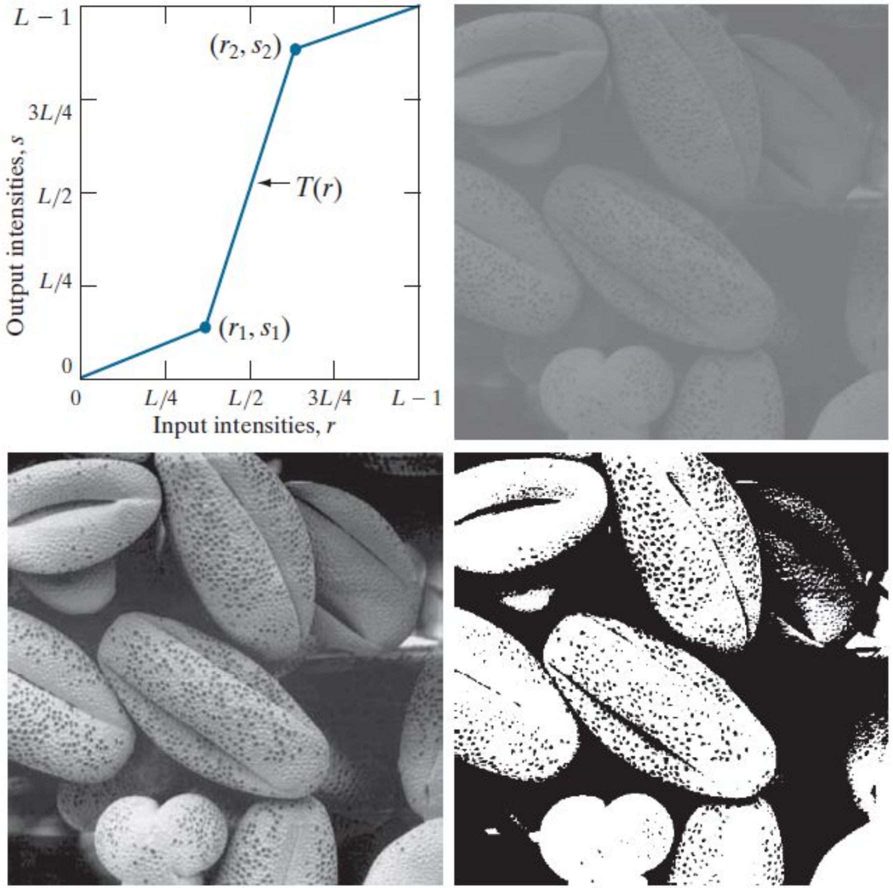
If the ROI/Image has more darker values, use gamma<1

Piecewise Linear Transformation

We can create complex functions, but it requires considerable user input

- Contrast Stretching
- Intensity-level Slicing
- Bit-plane Slicing

Contrast Stretching



a b
c d

Contrast stretching. (a) Piecewise linear transformation function. (b) A low-contrast electron microscope image of pollen, magnified 700 times. (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.)

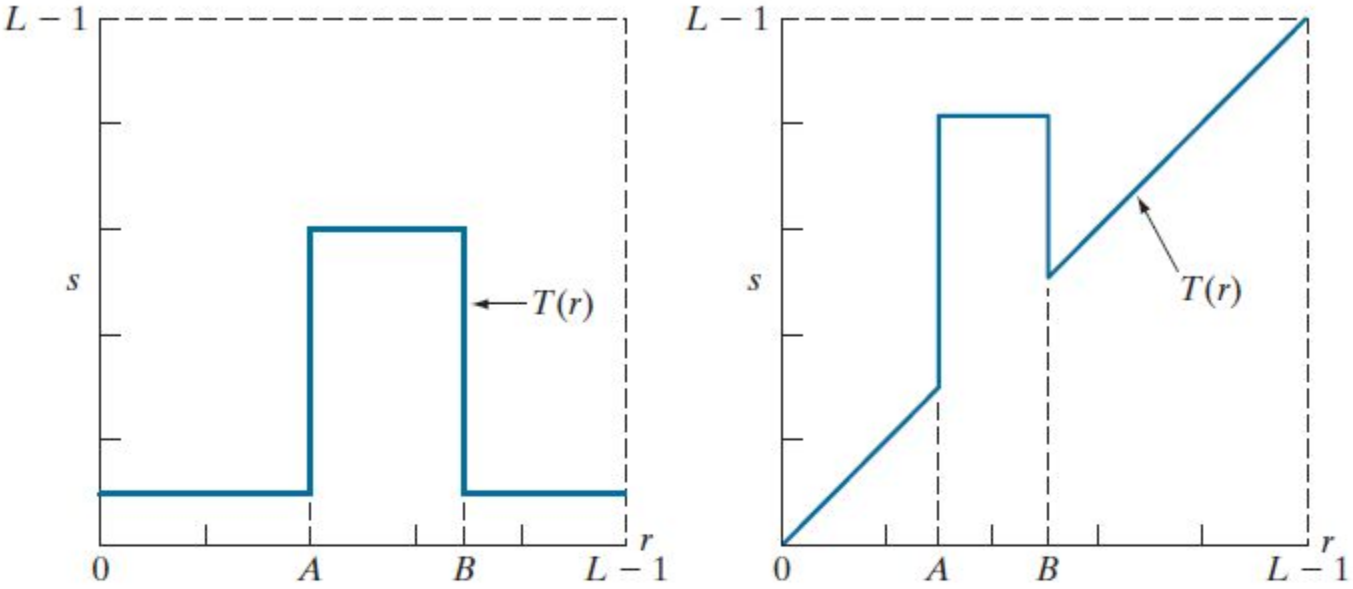
$$(r_1, s_1) = (r_{min}, 0) \text{ and } (r_2, s_2) = (r_{max}, L-1)$$

$$(r_1, s_1) = (m, 0) \text{ and } (r_2, s_2) = (m, L-1)$$

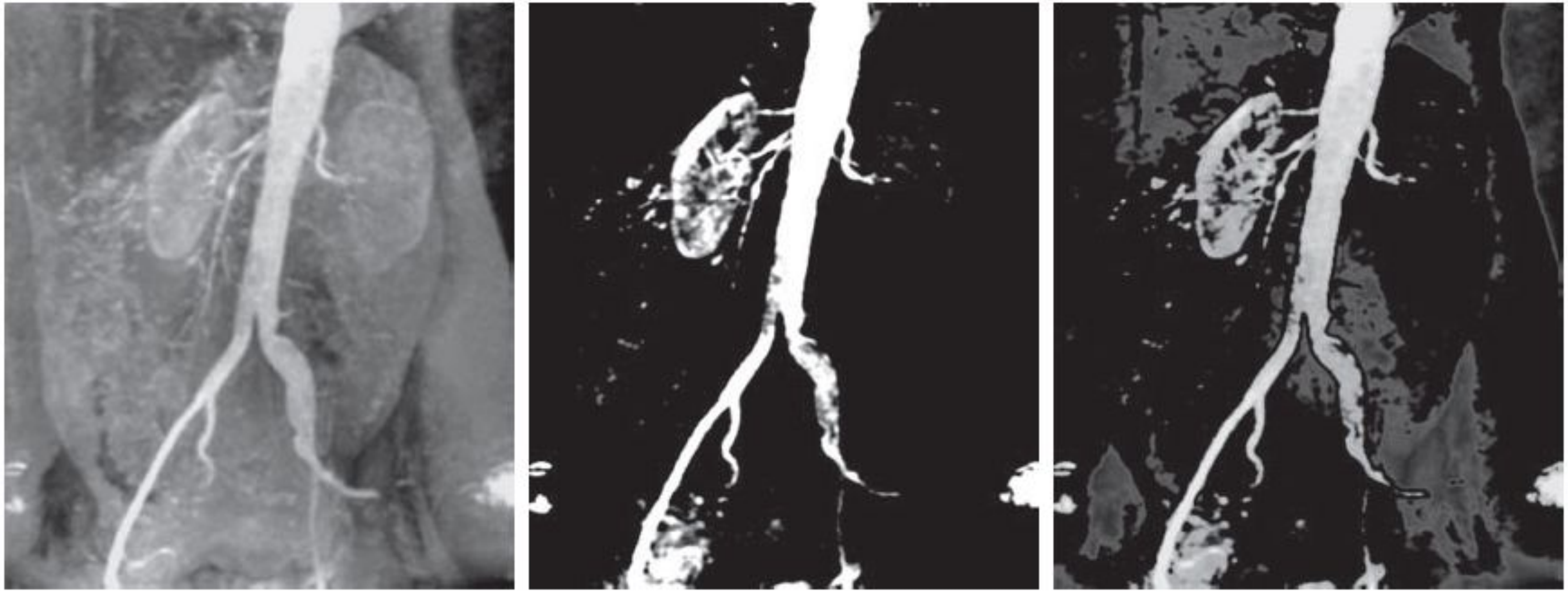
m=mean pixel value

Intensity Level Slicing: Highlighting or Suppressing certain intensity levels

a b



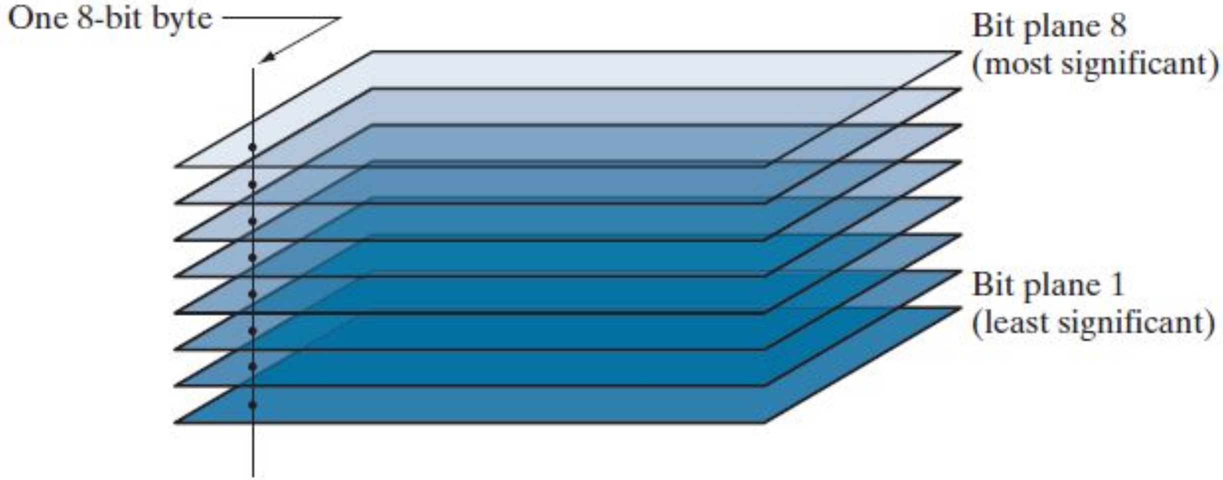
(a) This transformation function highlights range $[A, B]$ and reduces all other intensities to a lower level. (b) This function highlights range $[A, B]$ and leaves other intensities unchanged.

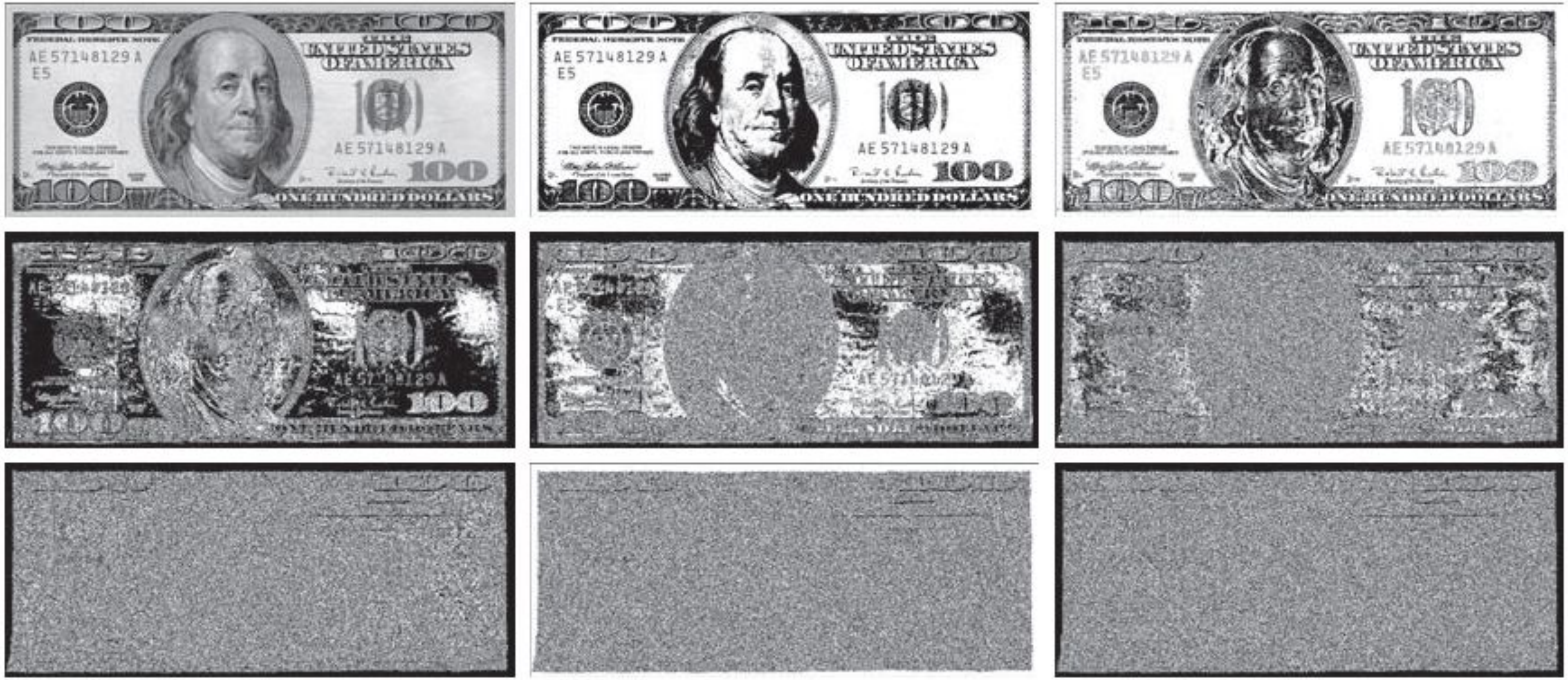


a b c

(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 range set near black, so that the 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.)

Bit-planes of an 8-bit image.





a	b	c
d	e	f
g	h	i

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

Reconstructed Image: 128*B8+64*B7+32*B6+16*B5+8*B4+4*B3+2*B2+1*B1

If all 8 planes are available

a b c

Image reconstructed from bit planes: (a) 8 and 7; (b) 8, 7, and 6; (c) 8, 7, 6, and 5.

Bit-plane Slicing

Depending upon different bit planes available, different reconstructed images can be created. If a particular bit plane isn’t available, the corresponding term gets removed.