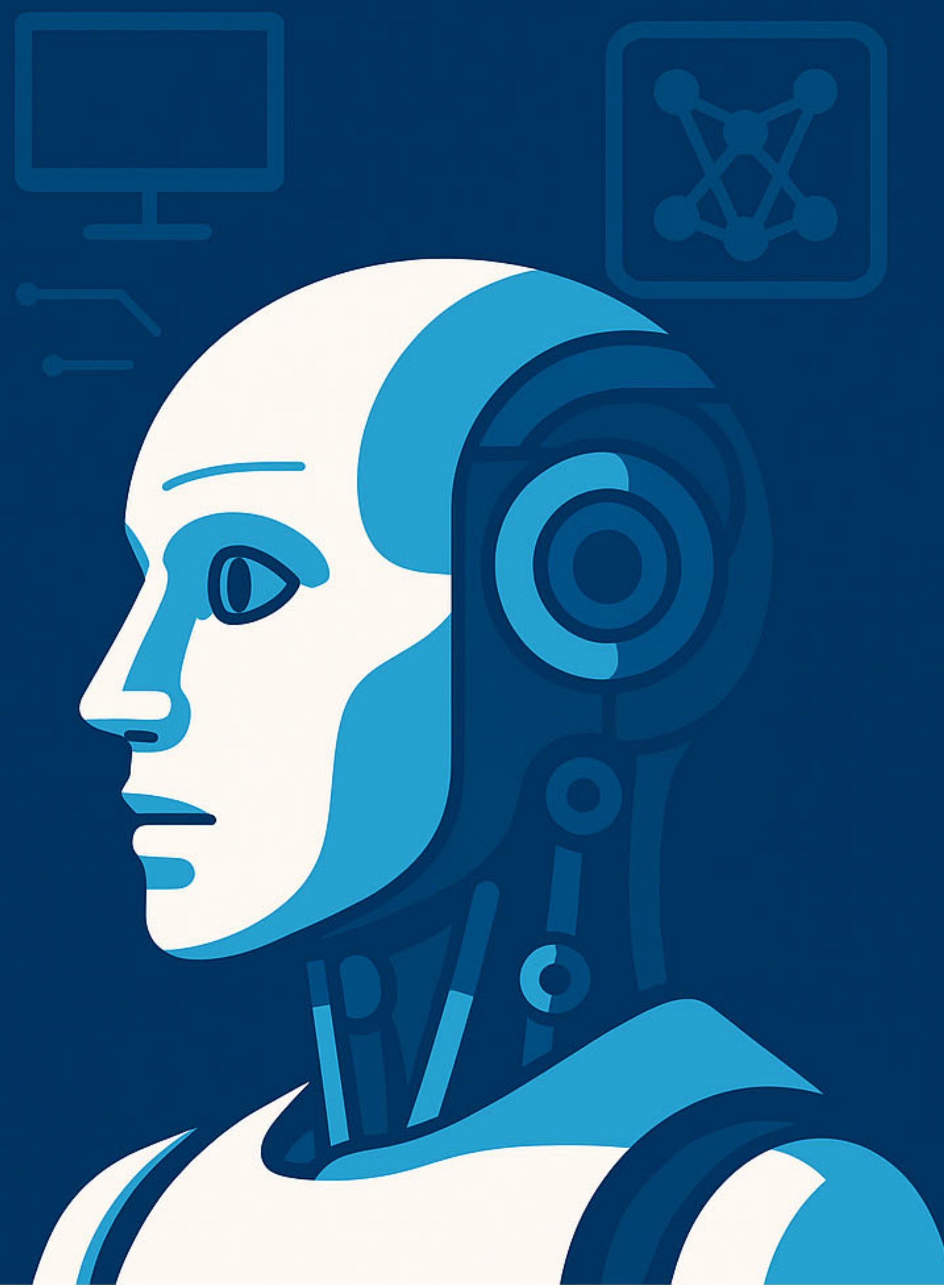
ME5411

ROBOT VISION AND AI



Dr. NG Hsiao Piau

Ng_h_p@nus.edu.sg

Lecture 3 (Part1)

Image Preprocessing

Topics

- Definitions
- Gray Level Transformation
- Histogram Equalization
- Geometric Transformation
- Noise and Filtering

Key takeaways

- Gray level transforms: negative, thresholding, contrast stretch
- Histogram equalization: enhances contrast
- Geometric transforms: pixel mapping + interpolation
- Noise types: Gaussian, impulse, quantization
- Filtering: smoothing and edge detection

1. Definitions

- Image pre-processing refers to operations with images at the lowest level of abstraction—both input and output are intensity images.

- The aim of pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing.

- Hence, image pre-processing can be classified into image enhancement and image restoration.

- After image enhancement, the result is more suitable than the original image for a specific application.
  - Problem oriented
- Two categories of approaches
  - Spatial Domain Methods
    - Spatial domain refers to the image plane itself
    - Direct manipulation of pixels in the image
    - Get to see the results after processing
    - Easy to implement but can be computational intensive
  - Frequency Domain Methods
    - Processing techniques are based on modifying the Fourier Transform of an image
    - Effective, quite imaginative as you do not see the results immediately after processing
- Image restoration takes a corrupt/noisy image and estimates the clean, original image.

Spatial Domain Methods

- Spatial domain methods operate directly on the pixels composing an image.
- Image processing function:

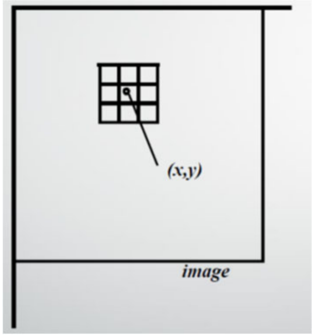
$$\boxed{g(x,y) = T[f(x,y)]}$$

f(*x*, *y*): input image

g(*x*, *y*): processed or output image

T: transfer function - an operator on *f*(*x*, *y*) defined over neighborhood of (*x*, *y*)

T can also operate on a set of input images (e.g. pixel-by-pixel averaging for noise removal)



Neighborhood about (*x*, *y*) is usually a square or rectangular area centered at (*x*, *y*). The centre of this sub-image moves from pixel to pixel during the operation. The operator is applied at each location to produce *g*(*x*, *y*).

Point Processing

- Point processing is a type of image enhancement, also known as pixel brightness transformation.
- Neighborhood is 1x1 == the size of pixel neighbor is 0. Point processing operates only one pixel.
- Transformation is defined pixel by pixel.
- The transformation *T* has been referred to as gray-scale transformation or gray-level mapping function.

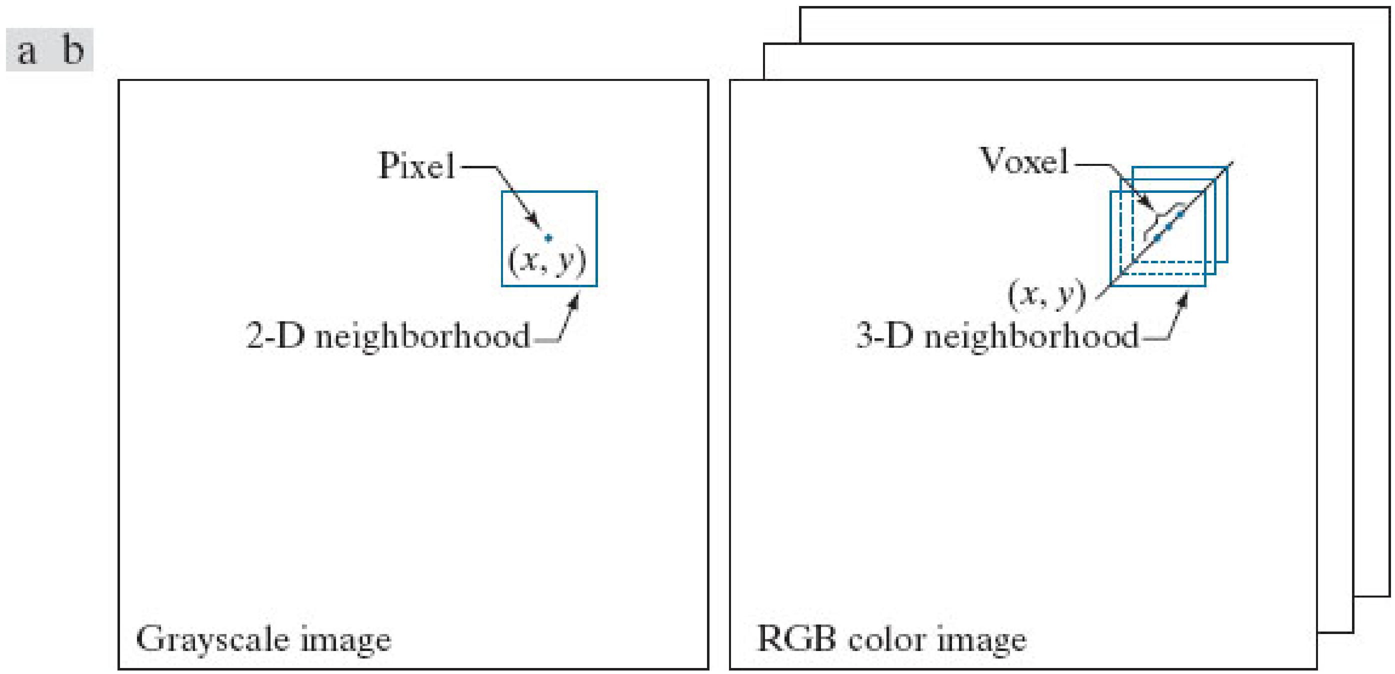
$$s = T(r)$$

r and *s* are the gray levels of *f*(x,y) and *g*(x,y) at (x,y) respectively.

- It changes the brightness of a pixel with no consideration of the pixel position.
- The mapping function can be specified in different ways, such as piecewise linear function, or based on the histogram of the input image.

Point Processing

- Color: p(*i*,*j*) = [*R*(*i*,*j*) *G*(*i*,*j*) *B*(*i*,*j*)]
  - RGB/8: *R*(i,j) = 0..255, *G*(i,j) = 0..255, *B*(i,j) = 0..255.
  - 8 bit of data per color channel.



Spatial neighborhoods for grayscale and RGB color images. Observe in (b) that a single pair of spatial coordinates, (x,y), addresses the same spatial location in all three images.

Source: Digital Image Processing By Gonzalez and Woods, Pearson.

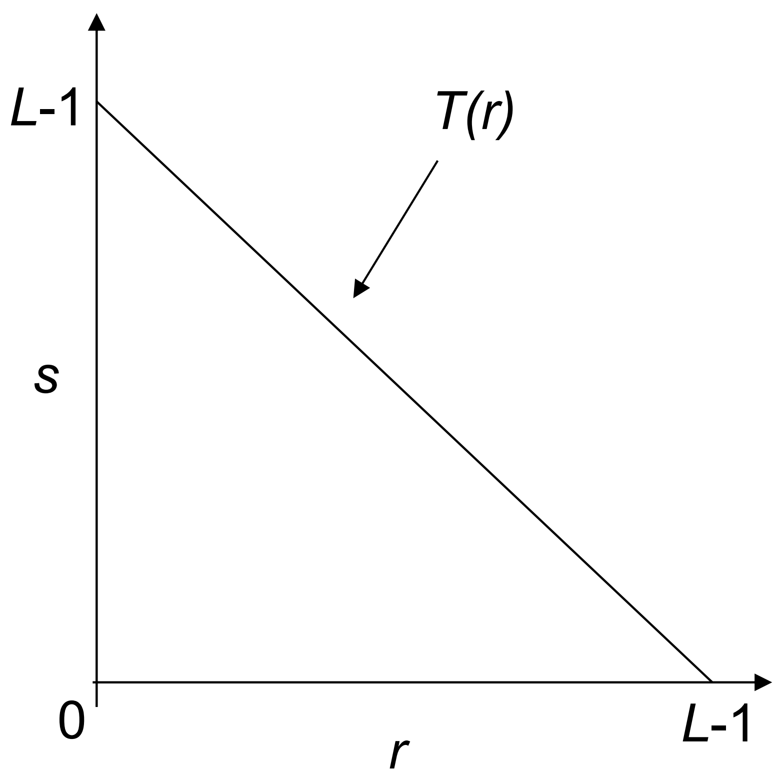
2. Gray-Scale Transformation

- The most common gray scale transformation are
  - Negative transformation
  - Brightness thresholding that results in a black and white image
  - Piecewise linear function that enhances the image contrast between two specific brightness values.

Negative Transformation

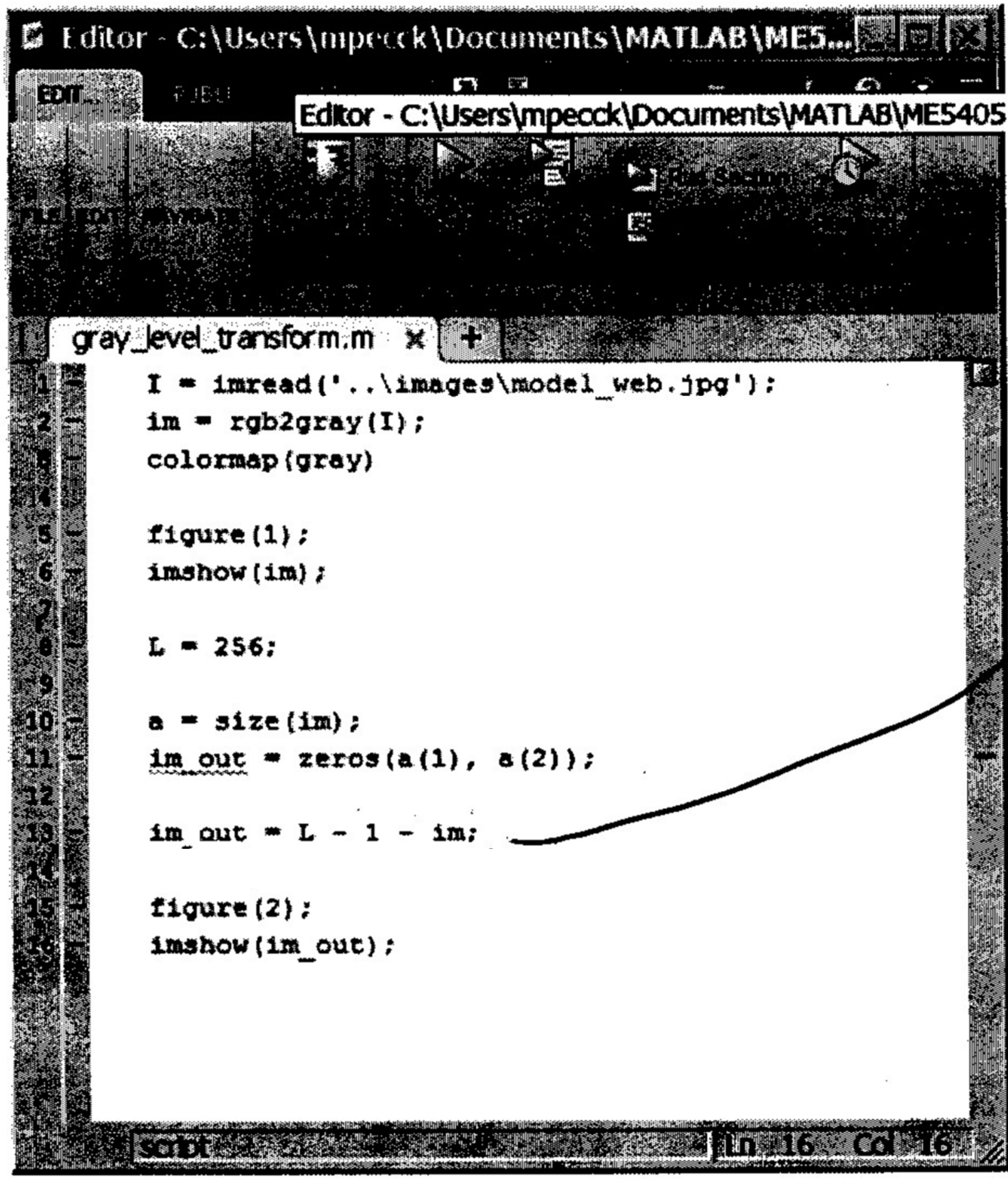
- Also known as negative image.
  It inverts the “color” of an image.

$$\boxed{s = T(r) = L - 1 - r}$$



p(i,j)

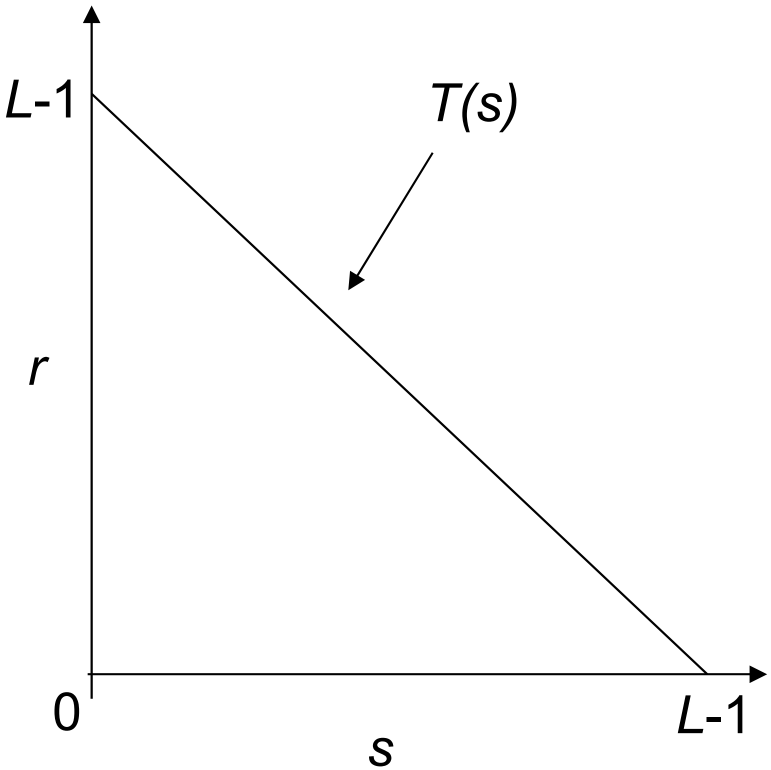
T(p(i,j))




```
for i = 1:a(1)
    for j = 1:a(2)
        im_out(i,j) = L-1-im(i,j)
    end
end
```

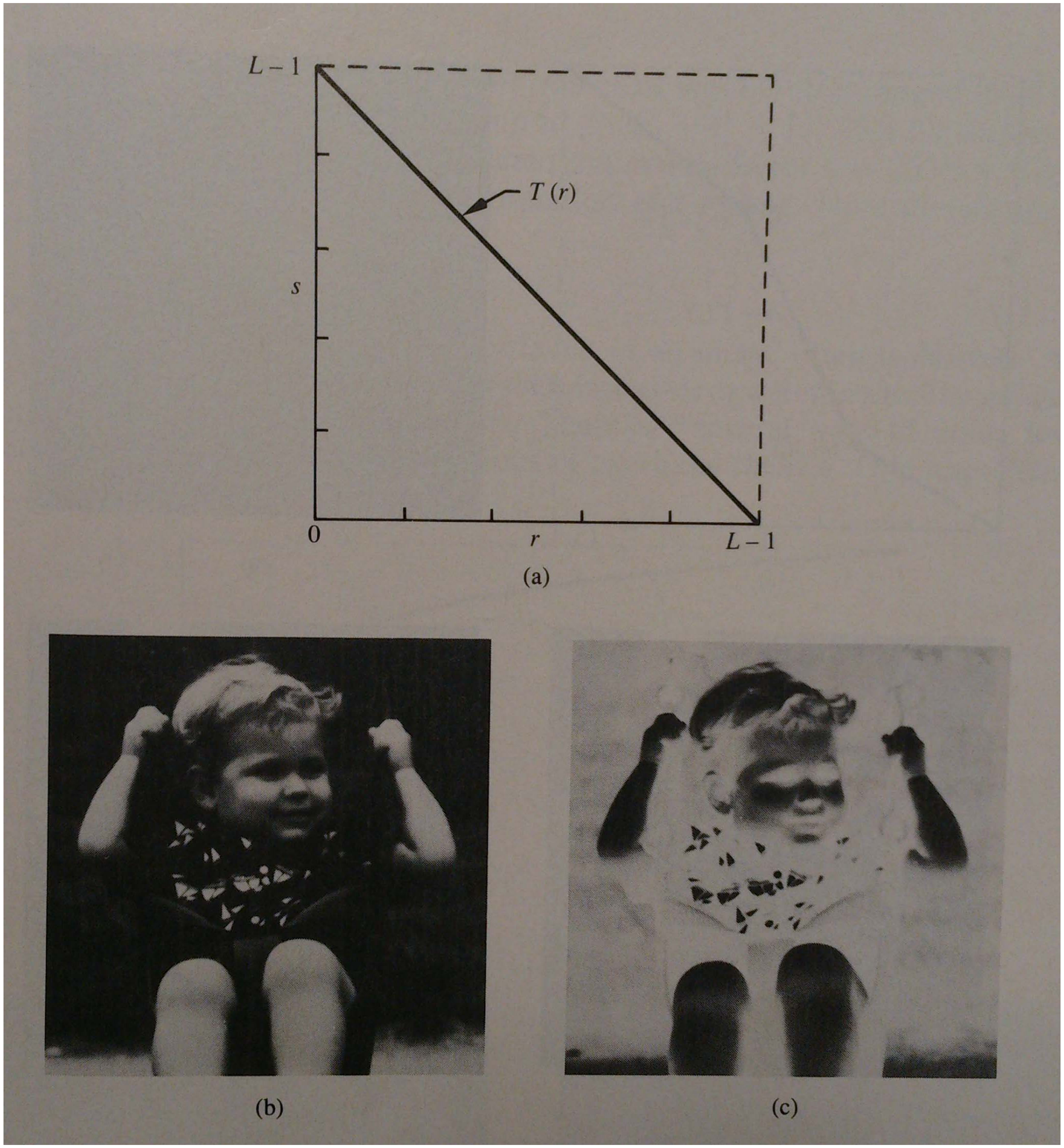
Negative Transformation

- Can we recover the original image by inverting the gray scale of a negative image?



Negative Transformation

- Example – invert colors of a color image



Obtaining the negative of image:

(a) gray-level transformation function, *r* and *s* denote the input and output gray levels respectively;

(b) an image; and (c) negative of the image.

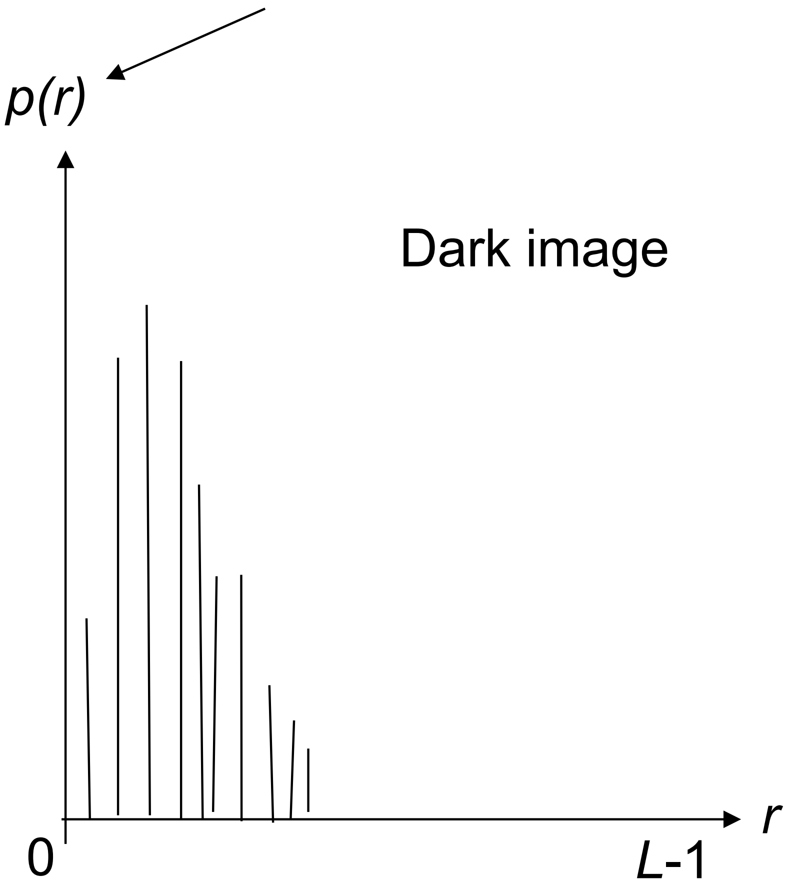
Histogram

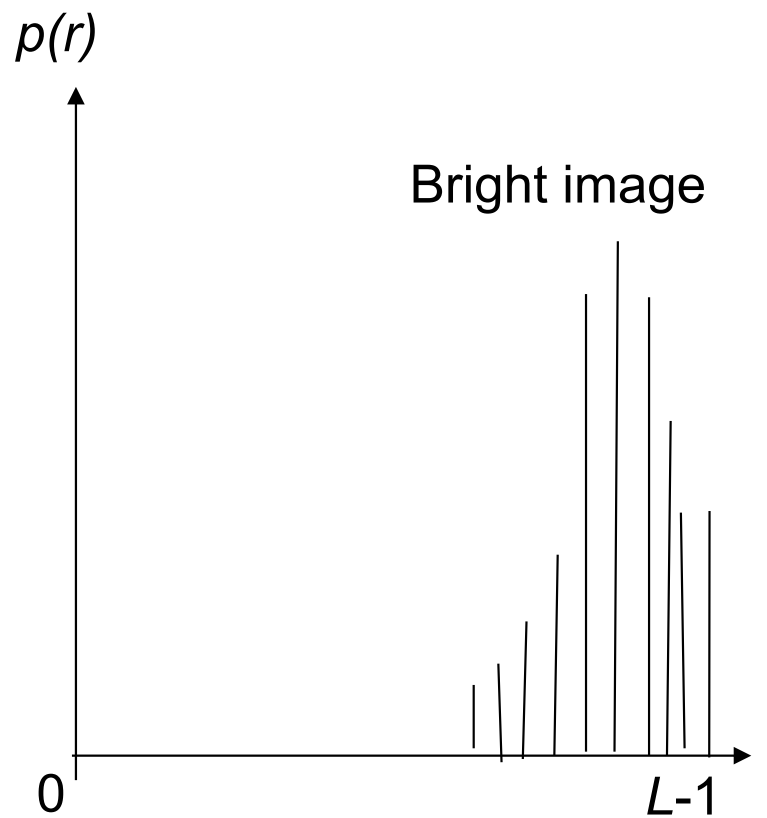
- The histogram of an image shows the distribution of the pixel values in the image over the intensity (or dynamic) range.
- For a 8-bit gray scale image, the pixel values typically from 0 to 255.
- The *i*th item of the histogram is $p(i) = n_i / N$, $i = 0..255$.
- It represents the probability of the a randomly chosen pixel has the gray level *i*, where n_i is the number of pixels of gray level *i*, and *N* is the total number of pixels in the image. (probability density function).

Histogram: Dark and Bright Images

- Brightness refers to the overall lightness or darkness of an image.

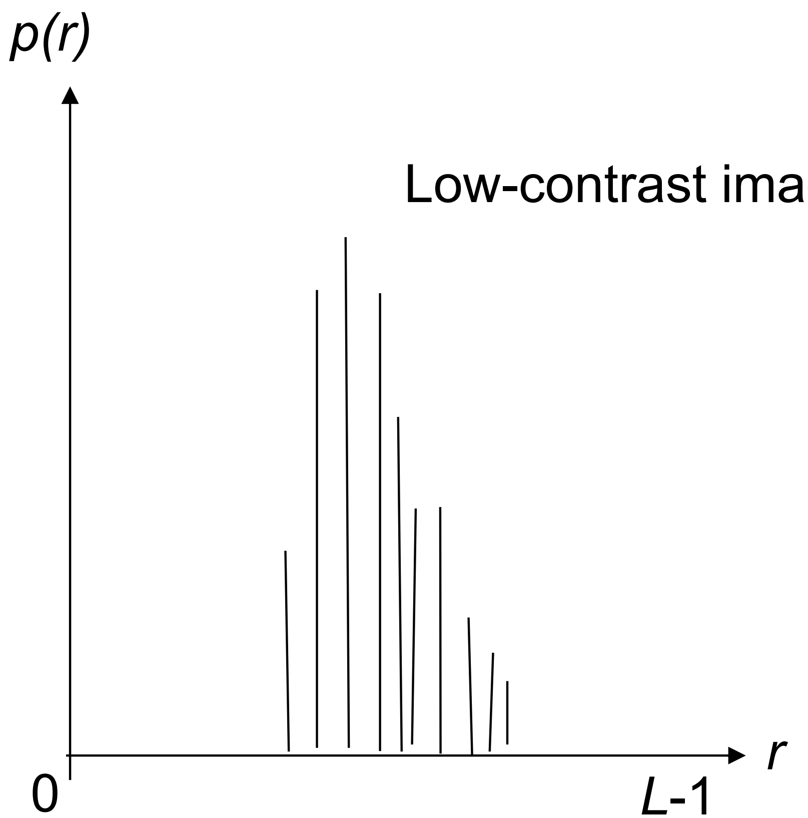
Probability density function (pdf)
= Number of pixels with intensity *r* / Total number of pixels

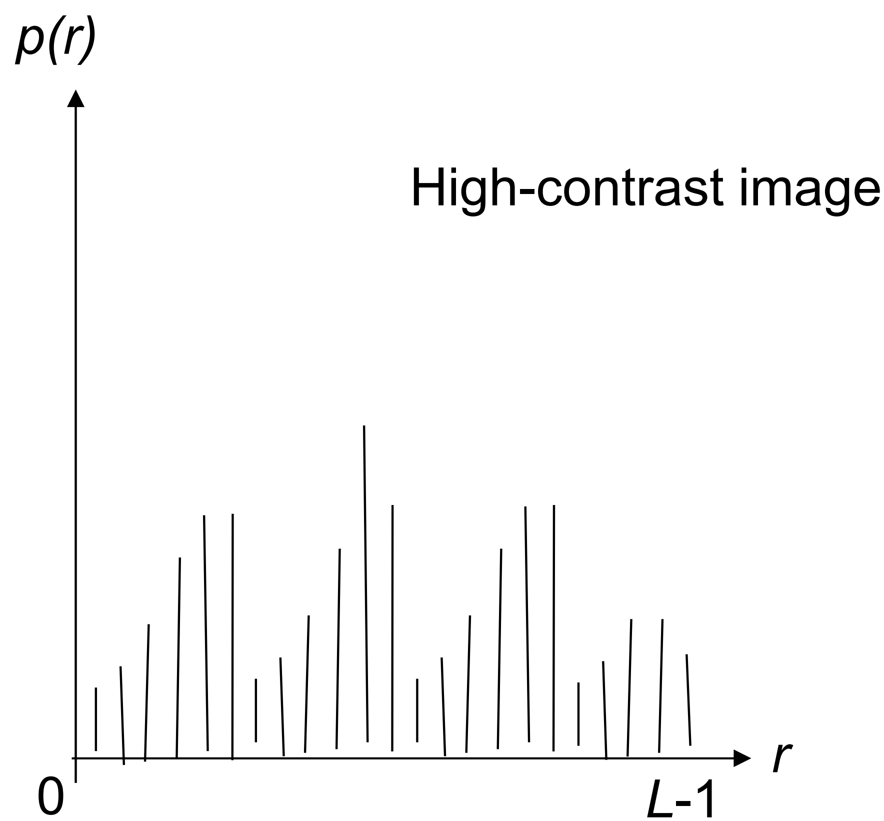




Histogram: Low-contrast and High-contrast Images

- When there are no sharp differences between black and white in an image, the image lacks contrast or does not have sufficient contrast.





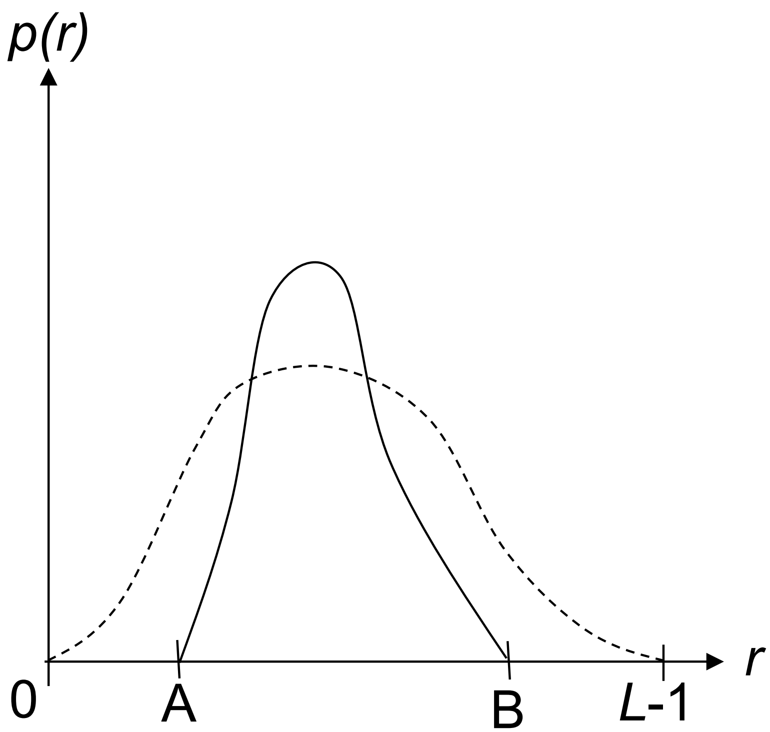
Histogram Processing

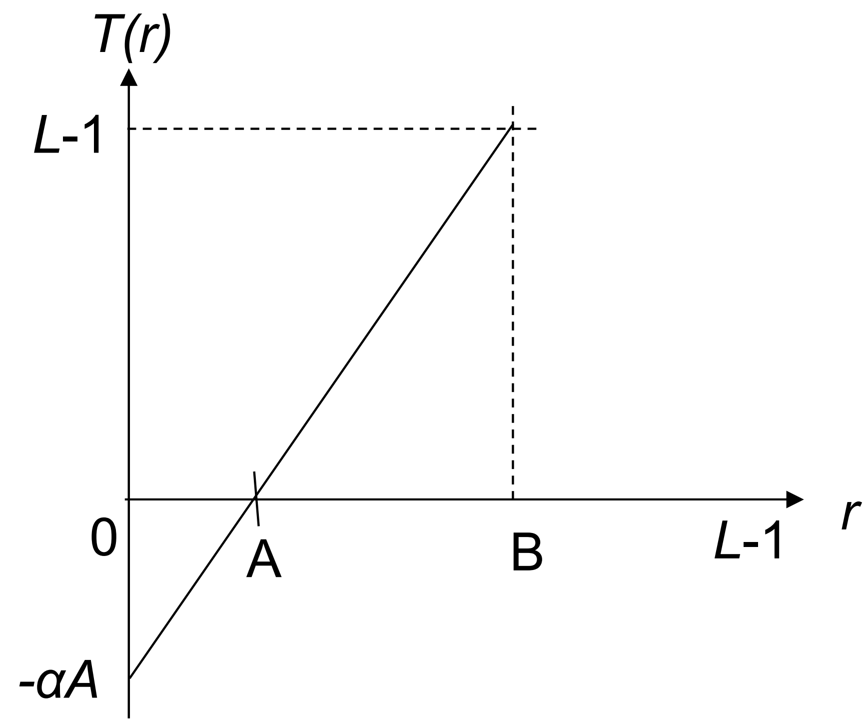
- Applying a linear function to transform the histogram.

$$f(r) = \alpha r + \beta$$

$$0 \equiv f(\mathrm{A}) = \alpha\mathrm{A} + \beta \Rightarrow \beta = -\alpha\mathrm{A}$$

$$L - 1 \equiv f(\mathrm{B}) = \alpha\mathrm{B} + \beta \Rightarrow \alpha = \frac{L\text{-}1}{\mathrm{B} - \mathrm{A}}$$

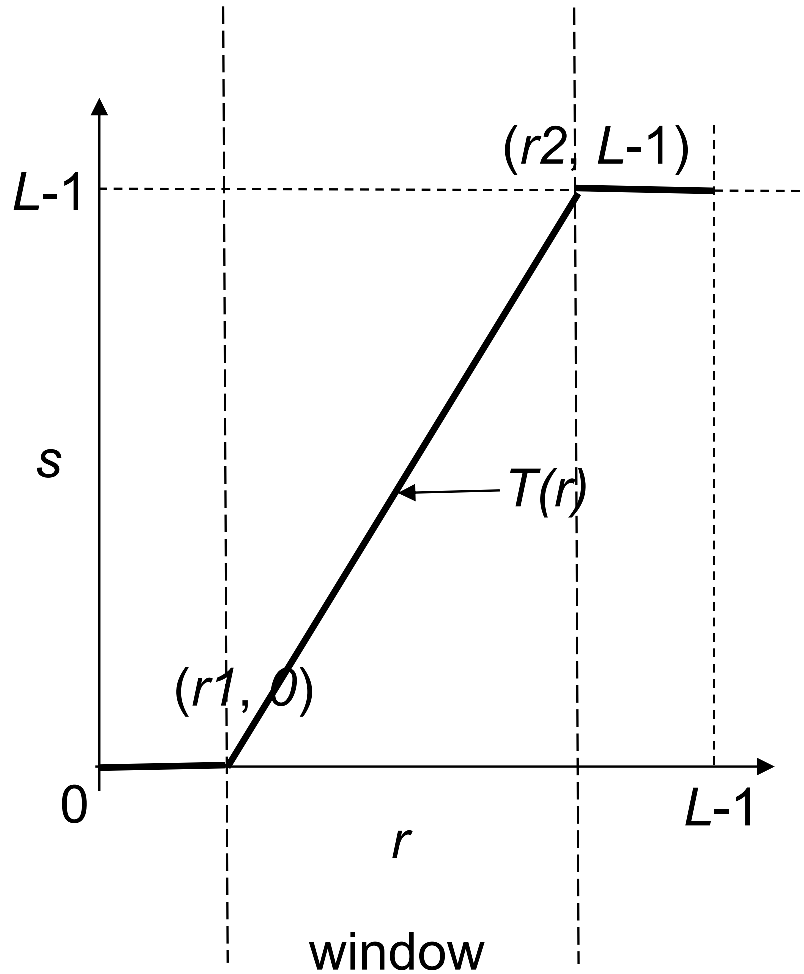


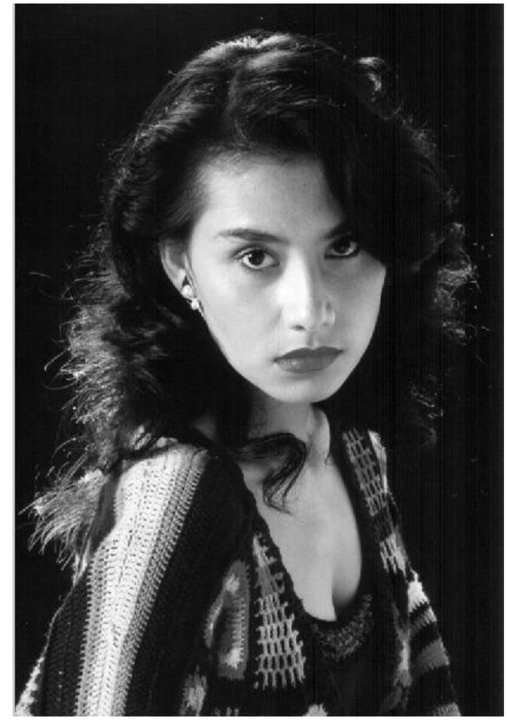


Contrast Stretching

- Contrast stretching is used to change the contrast or brightness of an image.
- In contrast stretching,
  - pixel values below a specified value are considered as black (or, have a value 0),
  - pixel values above another specified value are considered as white, and
  - pixel values in between these two values are considered as shades of gray.
- This is a linear mapping of a subset of pixel values to the entire range of grays from black to white.
- This will produce an image of higher contrast, but some details are lost.

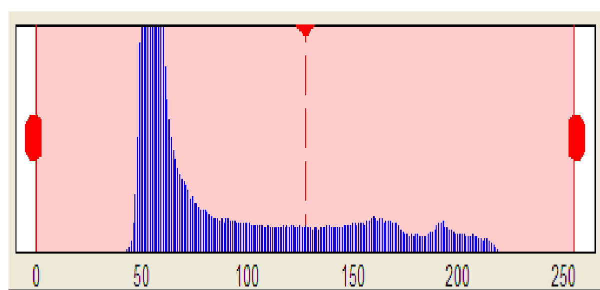
Contrast Stretching

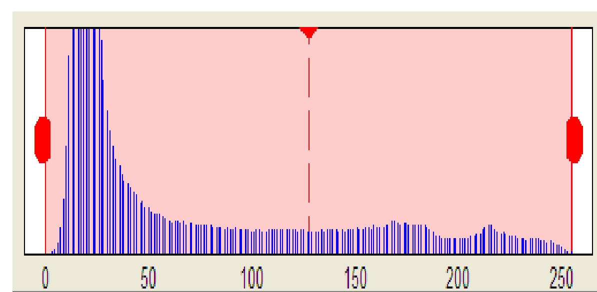




Original gray scale image

Image after contrast stretching





Contrast Stretching

- Using Matlab

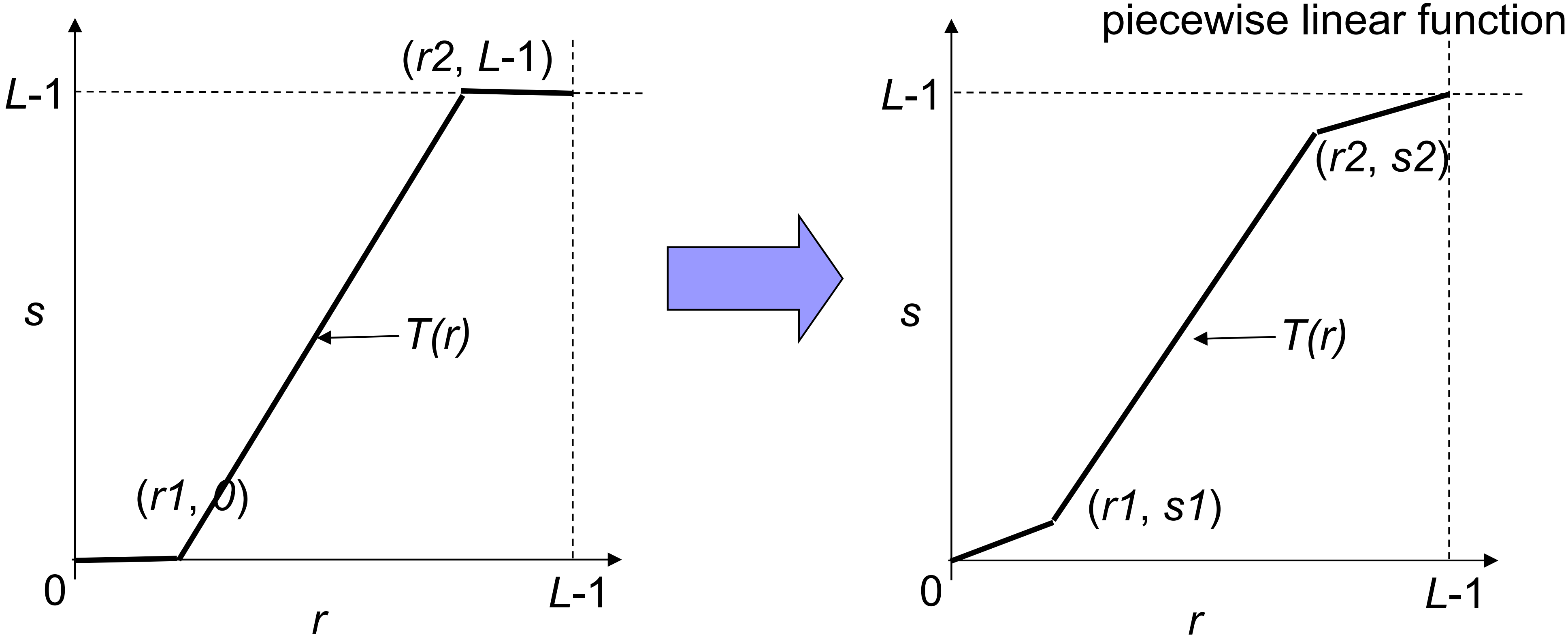
```
% read the input image from disk
model = imread('model_web.jpg');

% convert the input color image to 256 gray level image
modelGray = rgb2gray(model);
colormap(gray(256));

% using image tool to adjust image contrast
imtool(modelGray);
```

Contrast Stretching

- To reduce the loss in details of image.

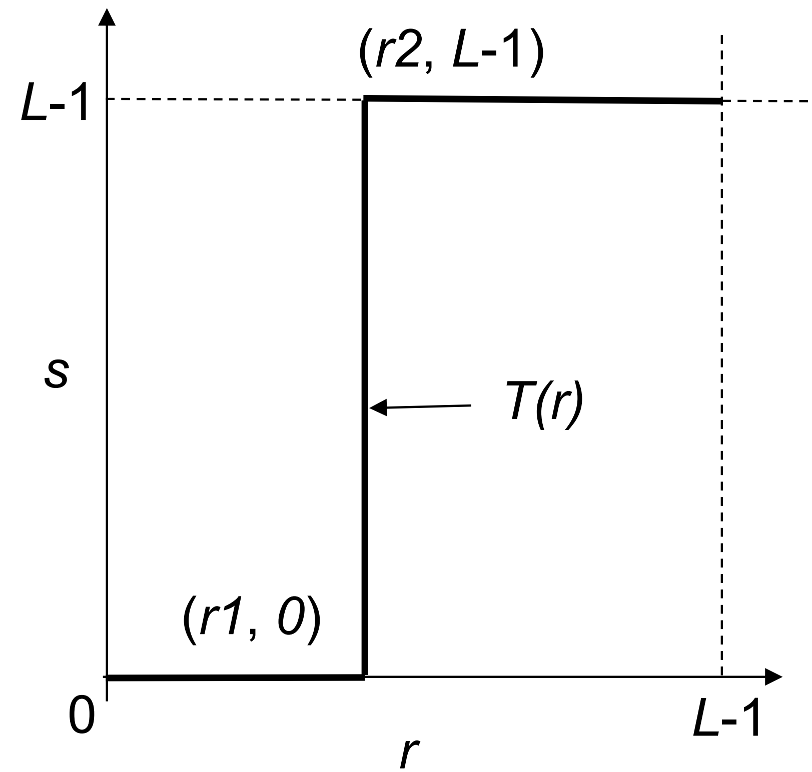


Piecewise Linear Function

- A piecewise linear function is a function composed of straight line segments, defined by different linear expressions over specific intervals of its domain.
- It is a generalization of typical contrast stretching.

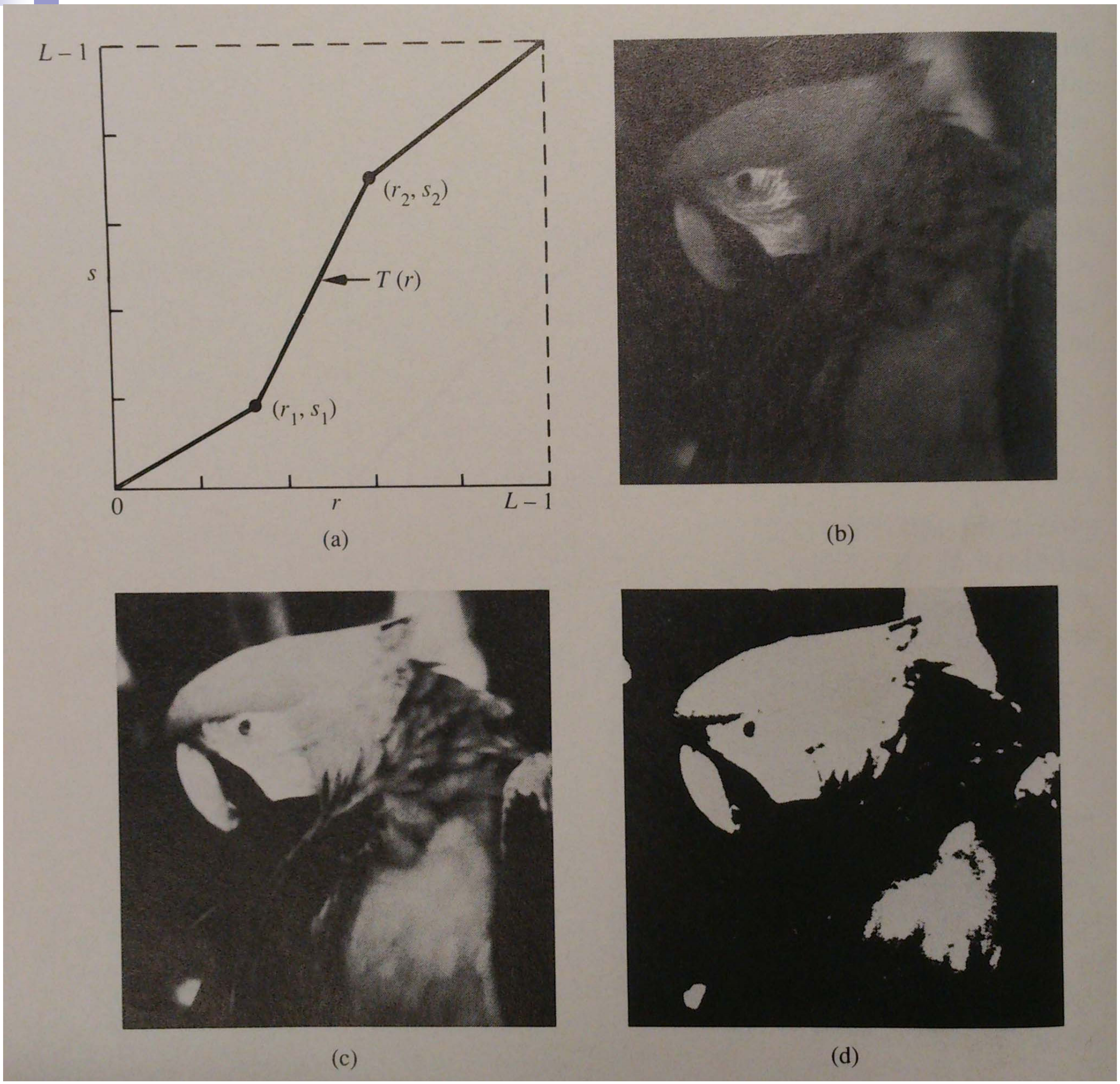
Brightness Thresholding

- It is a special case of piecewise linear function, and a simple way for image segmentation.
- A thresholding function will map all pixel values below a specified threshold to zero and all above to 255 for a 8-bit gray scale image.
- Brightness thresholding will result in a black-and-white image.



$r_1 = r_2$

$$s = T(r) = \begin{cases} 0 & r < r_1 \\ L-1 & r \geq r_2 \end{cases} \text{ or } s = T(r) = \begin{cases} 0 & r \leq r_1 \\ L-1 & r > r_2 \end{cases}$$



Contrast stretching:

(a) gray-level transformation function;

(b) a low-contrast image;

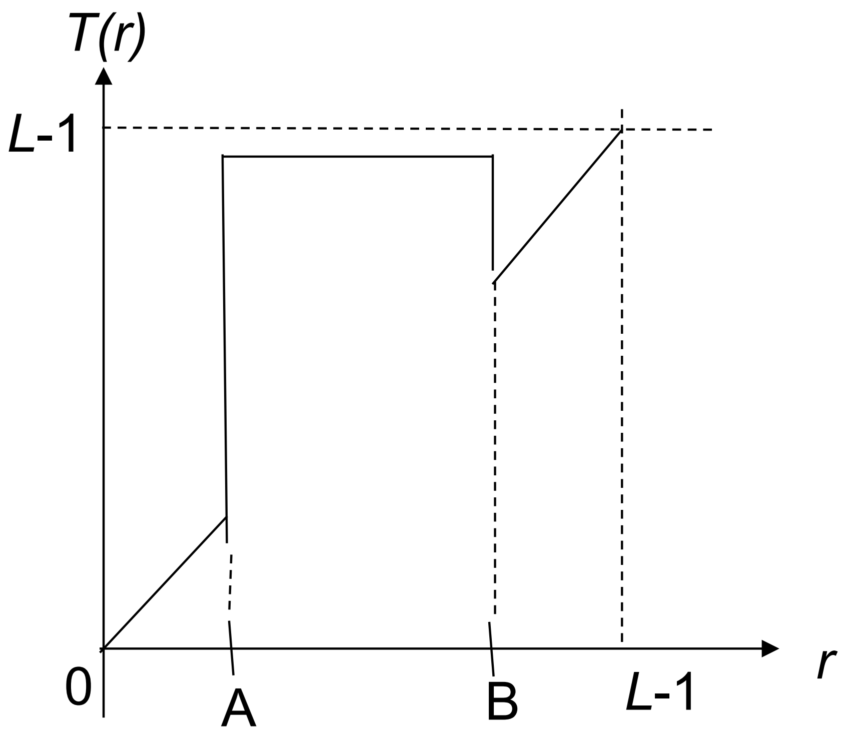
(c) result of contrast stretching;

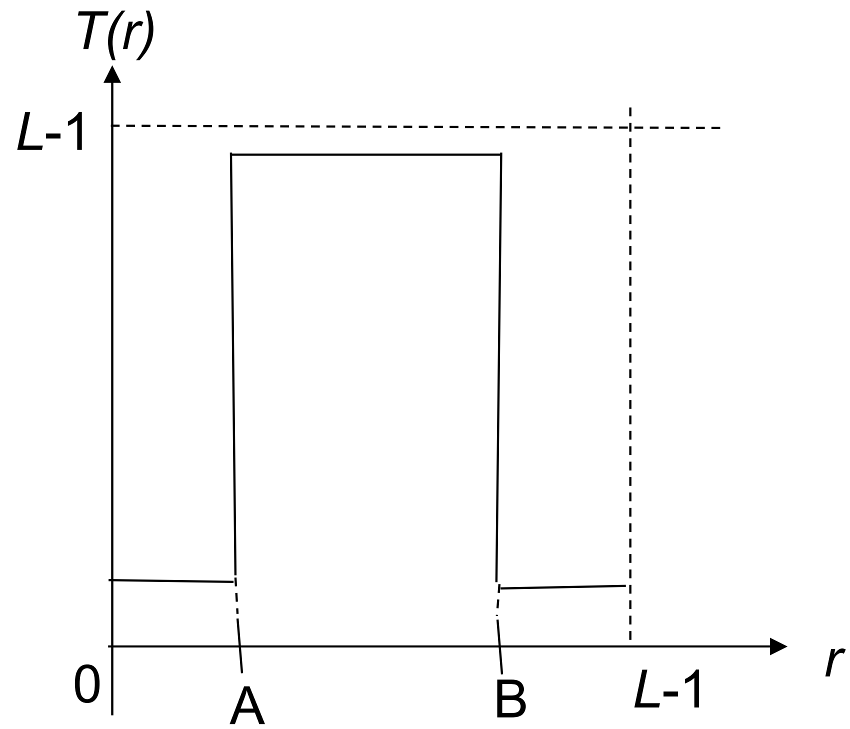
(d) result of thresholding.

The threshold is set at $r = L/2$ with output set at L-1 (white) for any gray level in the input image of $L/2$ or higher and at 0 (black) for all other values.

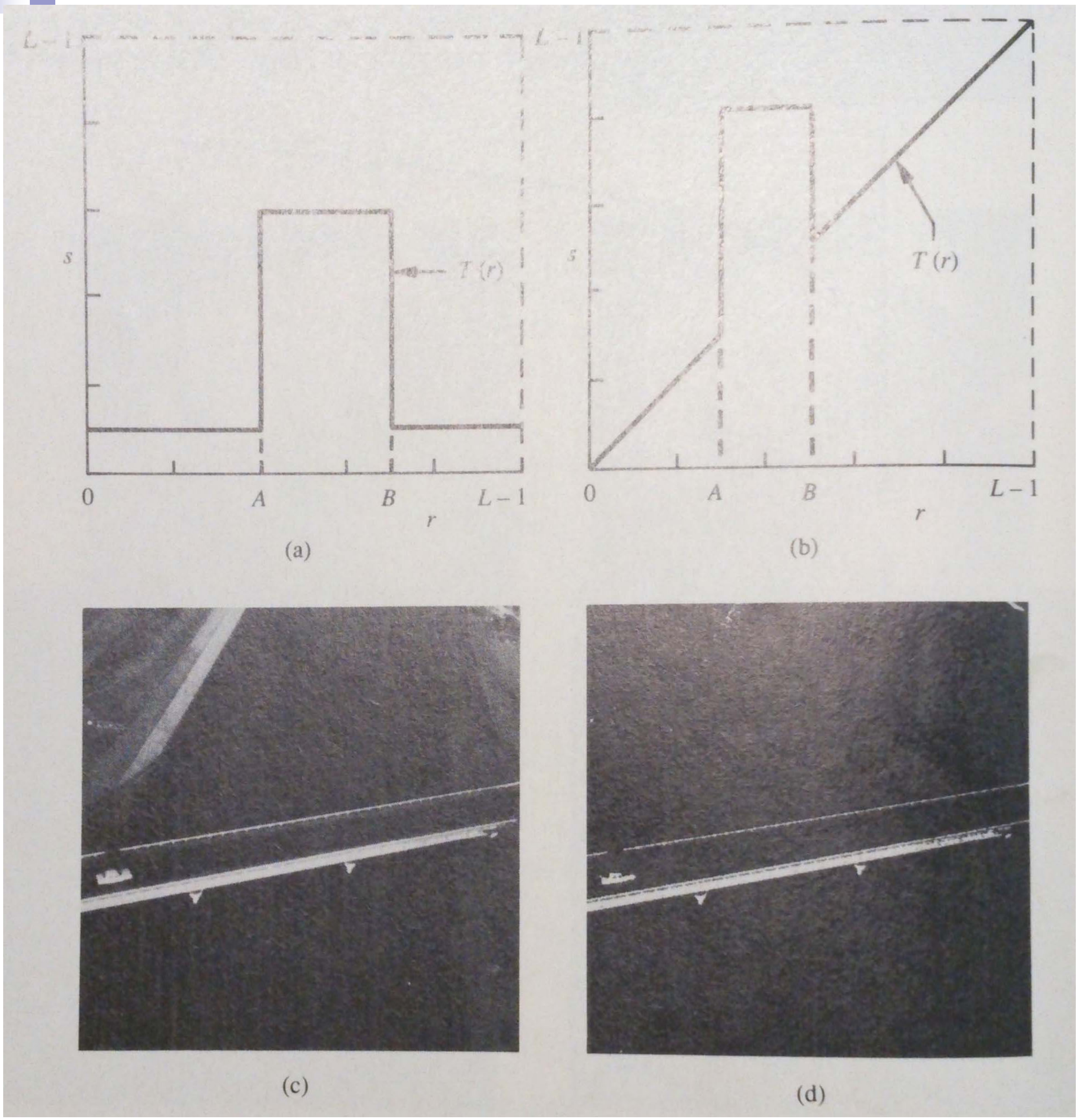
Gray Level Slicing

- To highlight the gray scales between [A,B].

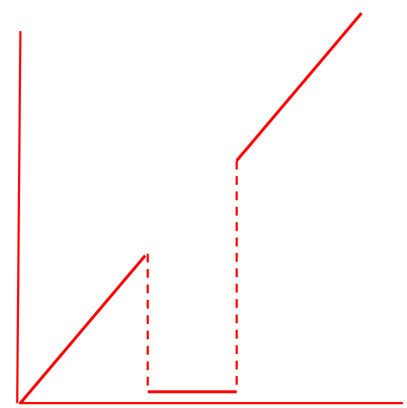




Note that gray scales outside [A,B] are diminished



Gray level slicing (or intensity-level slicing):
(a) transformation function that highlights a range [A,B] of intensities while diminishing all others to a constant, low level; (b) transformation function that highlights a range [A,B] of intensities but preserves all others; (c) an image; (d) result of using the transformation below with the intensities between A and B <u>darkened</u>.

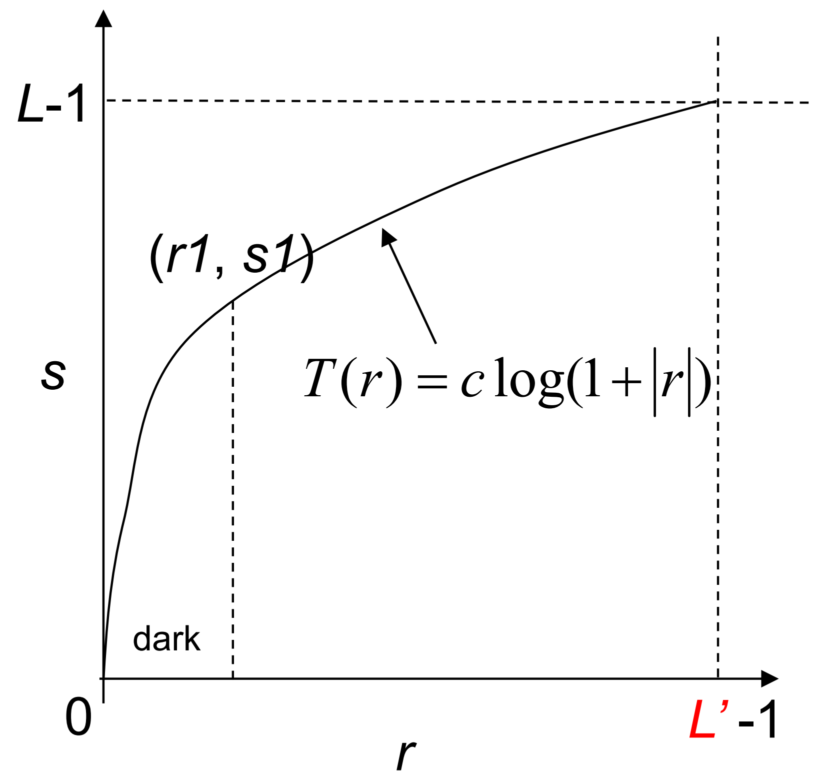


Dynamic Range Compression

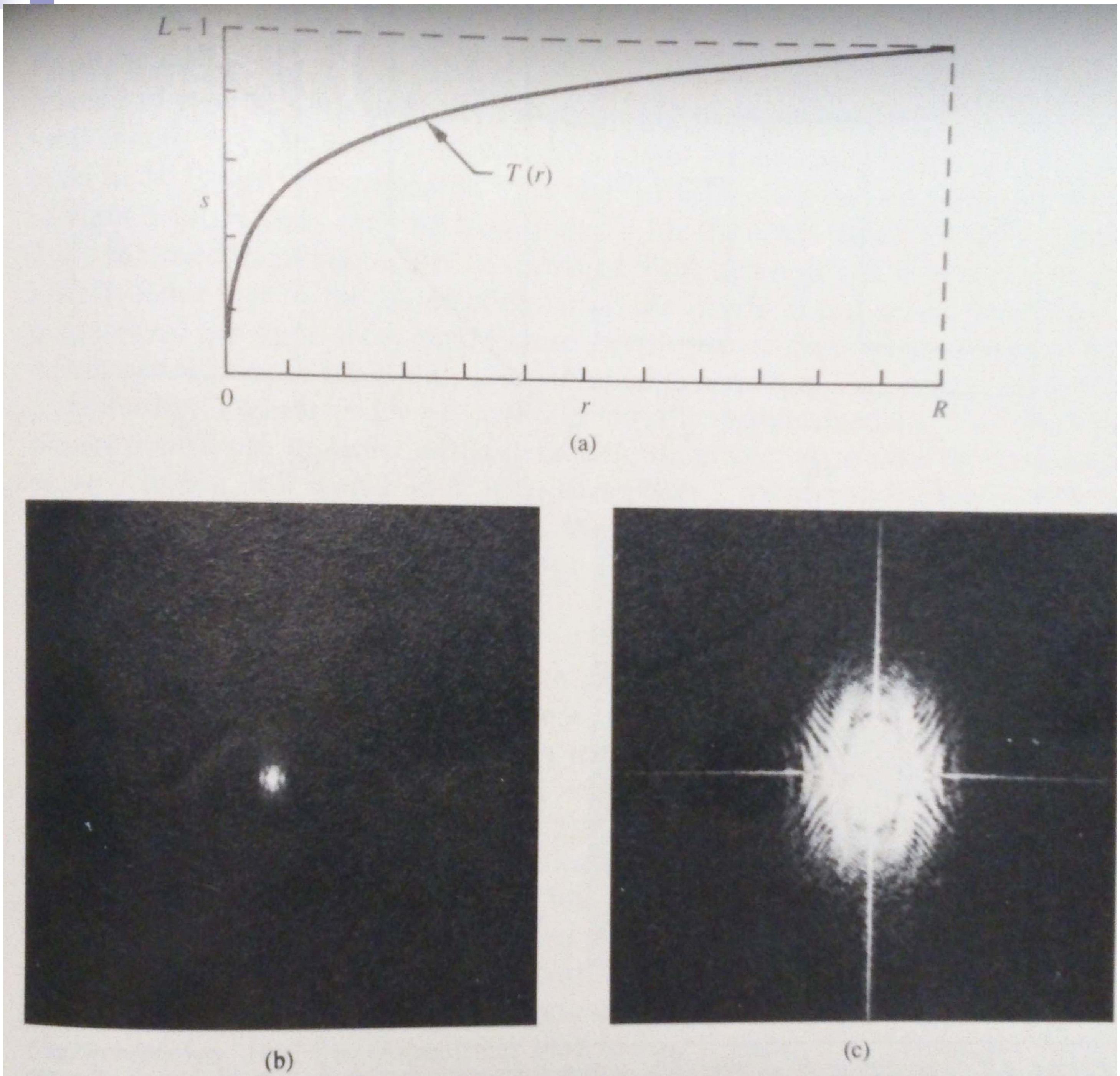
- Application of nonlinear functions
  - Compression of dynamic range
    - The dynamic range of a processed image may exceed the capability of display device – only the brightness parts of image are visible.
    - *c*: scaling constant; logarithm function performs desired compression.

Dynamic range is the range of the different between the lightest light and darkest dark of an image.

Nonlinear function



If log(1 +|r|) yields 0 to 6.4, setting c = 255/6.4 will scale the range to 0 to 255 gray levels.



Compression of dynamic range:

(a) logarithm gray-level transformation function;

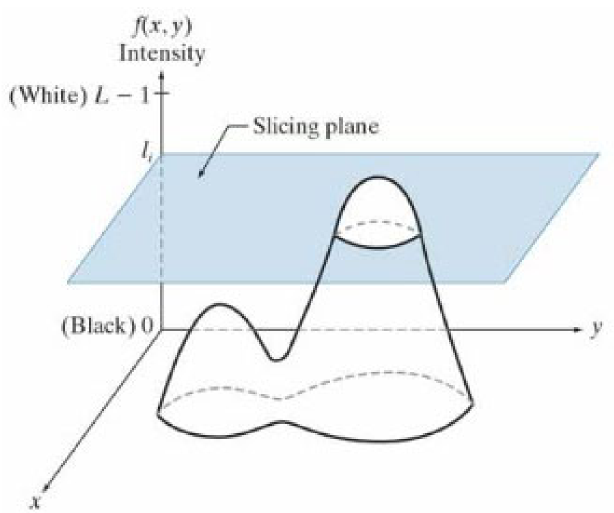
(b) image with large dynamic range (pixel values ranging from 0 to 2.5×10^6;

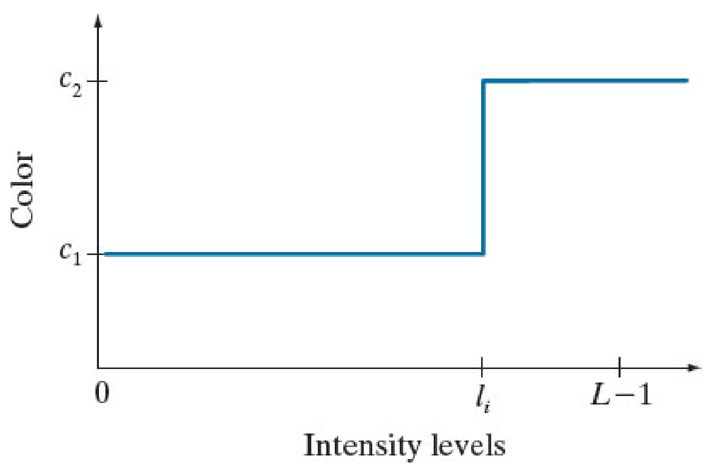
(c) result after transformation.

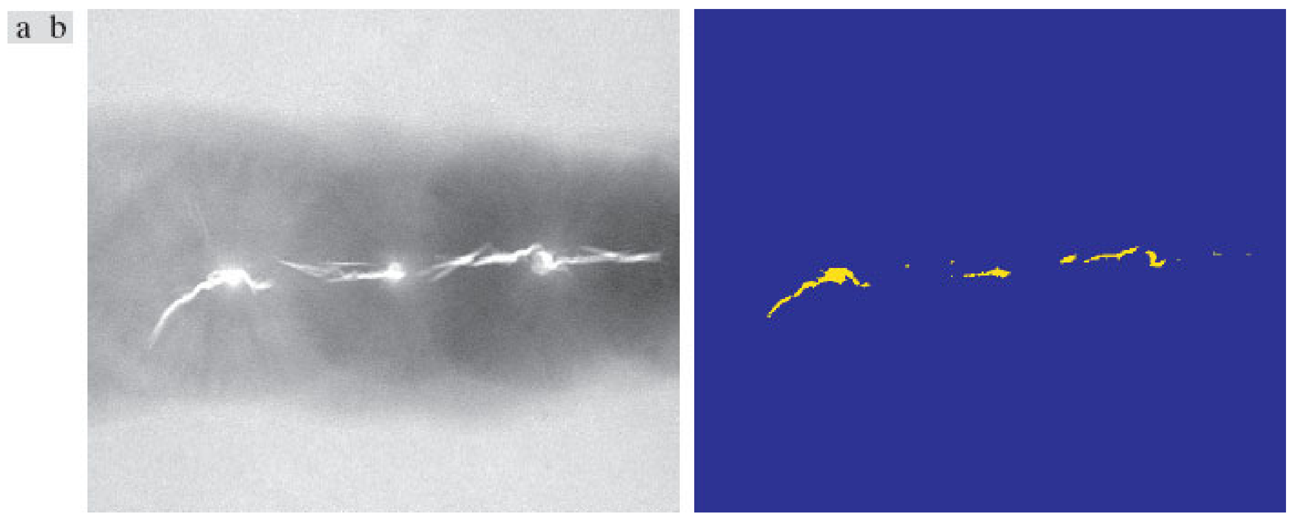
HDR –
High-Dynamic-Range photography

Pseudocolor Image Processing – intensity (or density) slicing

Pseudo color Image Processing is a digital image processing technique where colors are assigned to grayscale images (intensity images) to enhance their visualization and interpretation.







(a) X-ray image of a weld. (b) Result of color coding. (Original image courtesy of X-T E K Systems, Ltd.)

Source: Digital Image Processing By Gonzalez and Woods, Pearson.

3. Histogram Equalization

- Histogram equalization is a widely used image technique in image enhancement for increasing contrast.
- It is a method of contrast adjustment using the image’s histogram.
- It involves finding a gray level transformation function that creates an output image with a uniform or nearly uniform histogram.
- The transformation replaces each intensity in the input image by a new one.
- Transformation matrix is applied to the whole image at once.

Histogram Equalization

- The aim is to create an image with equally distributed brightness levels over the whole brightness scale
- An ideal equalized image has an equal number of pixels at all brightness levels, resulting in a straight horizontal line on the histogram graph.

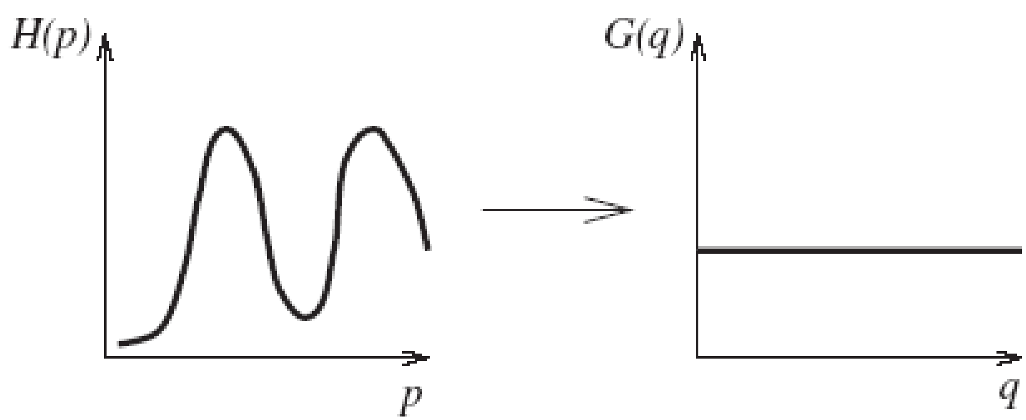


Figure 5.2: Histogram equalization.

Histogram Equalization

- *H(p)* = input histogram, input gray-scale range = [p_0, p_k]
- To find a monotonic transformation *q=T(p)* such that *G(q)* is uniform over the output range of [q_0, q_k].
- The histograms can be treated as a discrete probability density function. (Equation 1)
- Suppose that the image has *N* rows and columns, the equalized histogram *G(q)* corresponds to the uniform probability density function *f*. (Equation 2)
- The equalized histogram can be obtained for the continuous probability density function (Equation 3).
- The desired pixel brightness transformation *T* is the cumulative histogram (Equation 4).
- The discrete approximation to the continuous *T(p)* is given in Equation 5.

$$\sum_{i=0}^{k} G(q_i) = \sum_{i=0}^{k} H(p_i) \qquad (1)$$

$$f = \frac{N^2}{(q_k - q_0)} \qquad (2)$$

$$N^2 \int_{q_0}^{q} \frac{1}{q_k - q_0} ds = \frac{N^2 (q - q_0)}{q_k - q_0} = \int_{p_0}^{p} H(s) ds \qquad (3)$$

$$q = T(p) = \frac{q_k - q_0}{N^2} \int_{p_0}^{p} H(s) ds + q_0 \qquad (4)$$

$$q = T(p) = \frac{q_k - q_0}{N^2} \sum_{i=p_0}^{p} H(i) + q_0 \qquad (5)$$

Algorithm 5.1: Histogram equalization

1. For an $N \times M$ image of G gray-levels (often 256), create an array H of length G initialized with 0 values.
2. Form the image histogram: Scan every pixel and increment the relevant member of H—if pixel p has intensity g_p, perform

$$H[g_p] = H[g_p] + 1 \, .$$

3. Form the cumulative image histogram H_c:

$$\begin{aligned} H_c[0] &= H[0] \, , \\ H_c[p] &= H_c[p-1] + H[p] \, , \quad p = 1, 2, \ldots, G-1 \, . \end{aligned}$$

4. Set

$$T[p] = \text{round}\left(\frac{G-1}{NM} H_c[p]\right) \, .$$

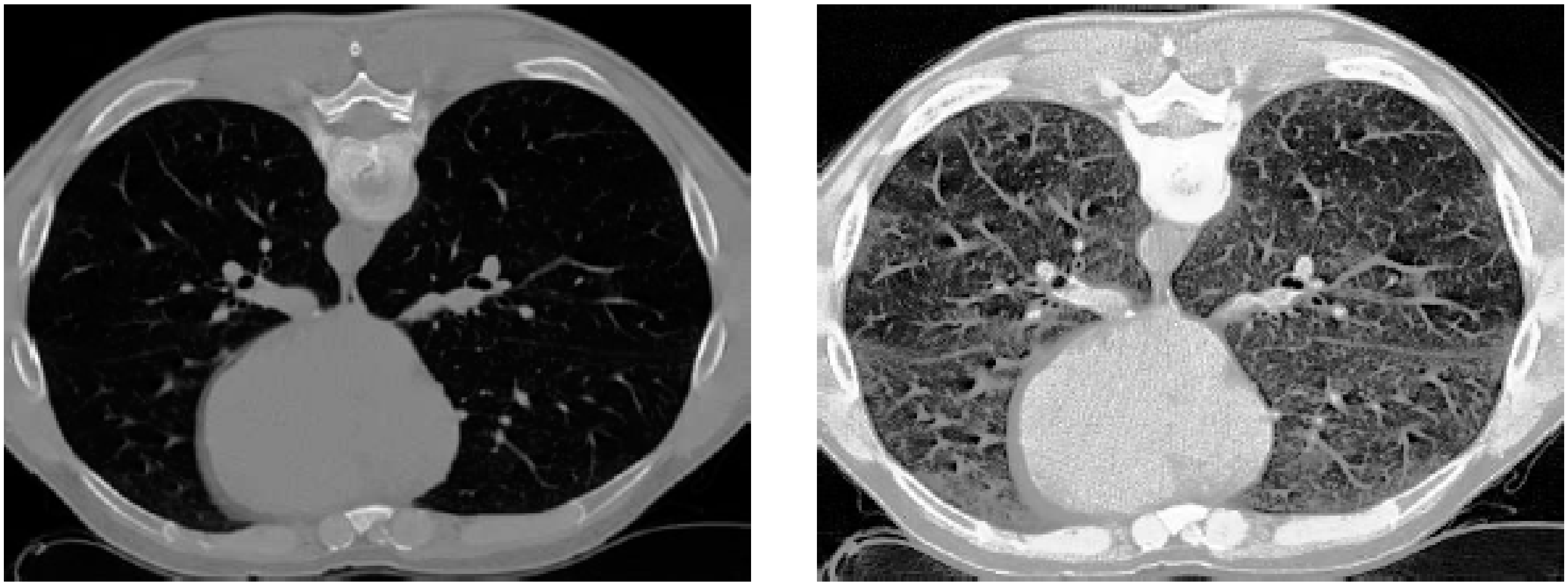
   (This step obviously lends itself to more efficient implementation by constructing a look-up table of the multiples of $(G-1)/NM$, and making comparisons with the values in H_c, which are monotonically increasing.)

5. Rescan the image and write an output image with gray-levels g_q, setting

$$g_q = T[g_p] \, .$$

Histogram Equalization

- Example



(a) (b)

Figure 5.3: Histogram equalization. (a) Original image. (b) Equalized image.

Histogram Equalization

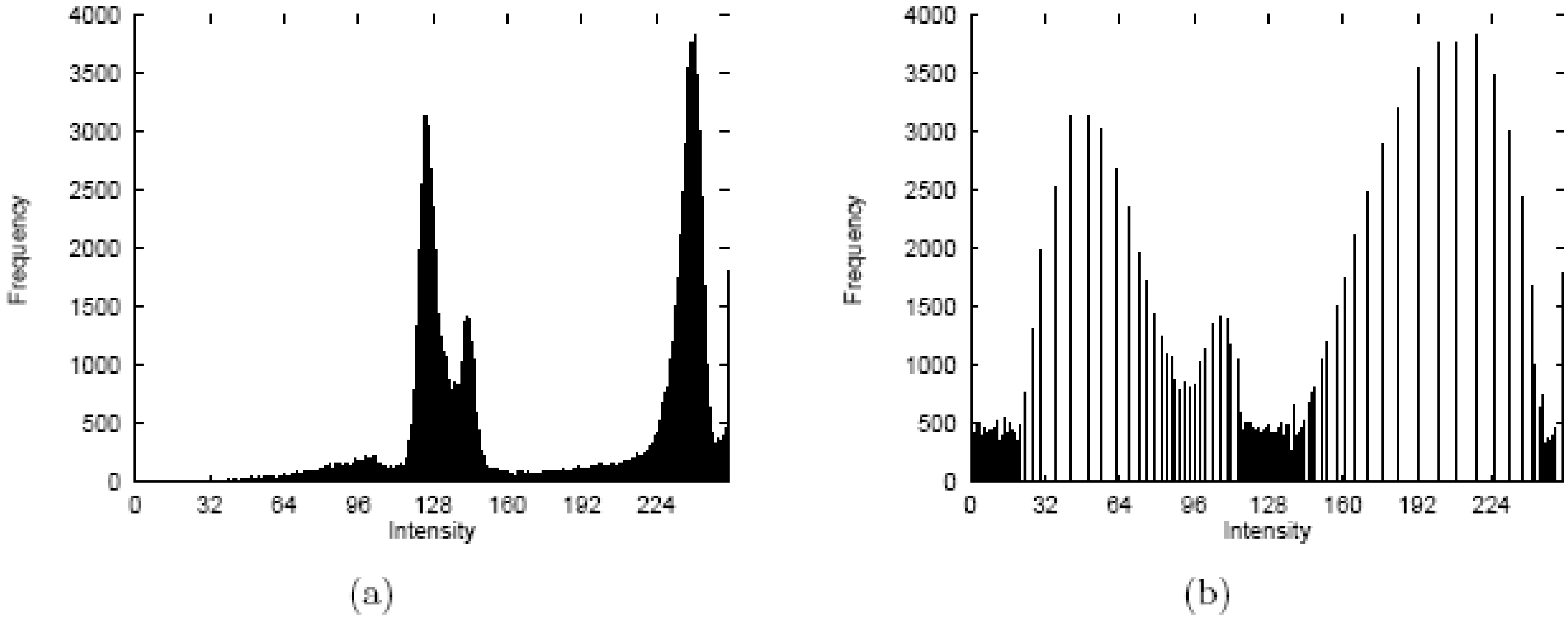


Figure 5.4: Histogram equalization: Original and equalized histograms corresponding to Figure 5.3a,b.

Matlab Implementation of Histogram Equalization

function [im_out, H, Hc, T] = hist_eq(im)

input:

im [m x n] input image

output:

im_out [m x n] equalized image

H [1x256] histogram of the input image

Hc [1x256] cumulative histogram of the input function

T [1x256] transformation function of the intensity

Matlab Implementation of Histogram Equalization

https://www.mathworks.com/help/images/histogram-equalization.html

Histogram Equalization

- Useful in images with both background and foreground are dark or bright.
  - It can reveal good detail in over or under-exposed photographs.
  - The method will provide good view of hard tissue (bone) in x-ray image.
- Disadvantage: global application leads to indiscriminate modification of image
  - The method may increase contrast of background noise and decrease usable signal.
  - It could not support the need to highligh certain gray level range.

Histogram Specification

- Histogram Specification is a generalized version of histogram equalization
- A “target” histogram that actually define the desired shape of the image histogram is specified.
- A nonlinear stretch operation is applied to force the image histogram to have that shape.

Summary

- Image most frequently represented as multidimensional array can be added, subtracted and made equal to another image.
- Logical operation can also be perform on image.
- Gray-scale transformations change brightness without regard to position in the image. The common gray-scale transformations are piecewise linear function, negative transformation and brightness thresholding.
- The goal of histogram equalization is to create an image with equally distributed brightness levels over the whole brightness scale.