Digital Image Processing Using MATLAB

SP_Tutorial2

- TA

崔子昂 cuiza@shanghaitech.edu.cn
李超凡 lichf1@shanghaitech.edu.cn

Outline

- **Color image processing**
  - -Digital image and color image
  - -Color balance
  - -HSI color space
  - -Demosaicing
- **Spatial filtering**
  - -Spatial noise
  - -Linear and nonlinear spatial filter
- **Intensity transformation**
  - -Histogram
  - -Log transformation and Gamma transformation
  - -Histogram processing

Digital image and color image

- What is Digital Image?
- Digital image compose of a finite number of elements – Pixel.
- A visual representation in form of a function ***f(x, y)***,

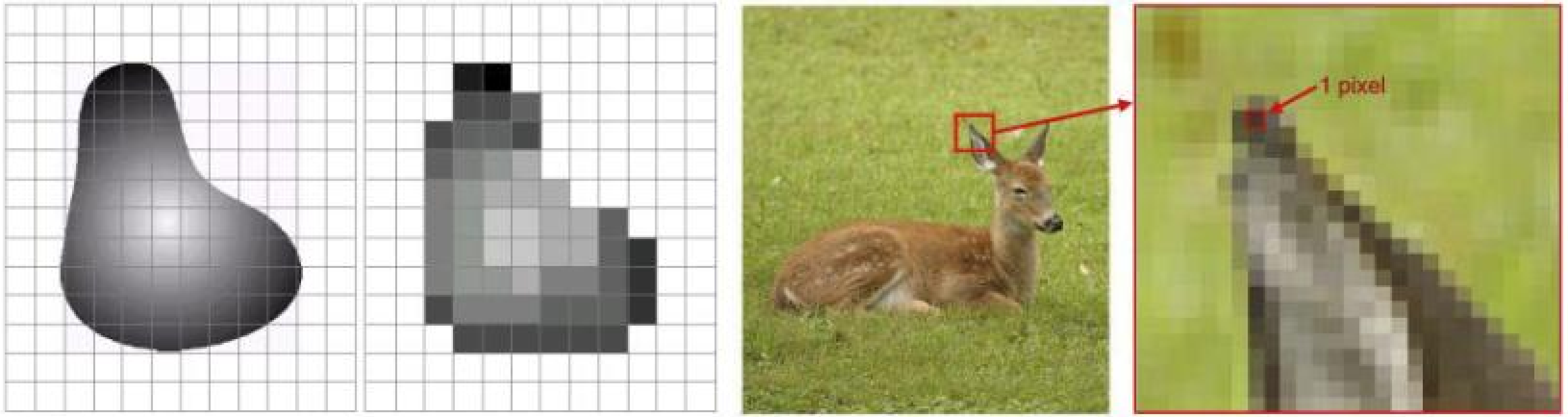
  -***f*** is related to the intensity or brightness at point

  -***(x, y)*** are spatial coordinates

  -***x***, ***y***, and the amplitude of ***f*** are finite and discrete quantities

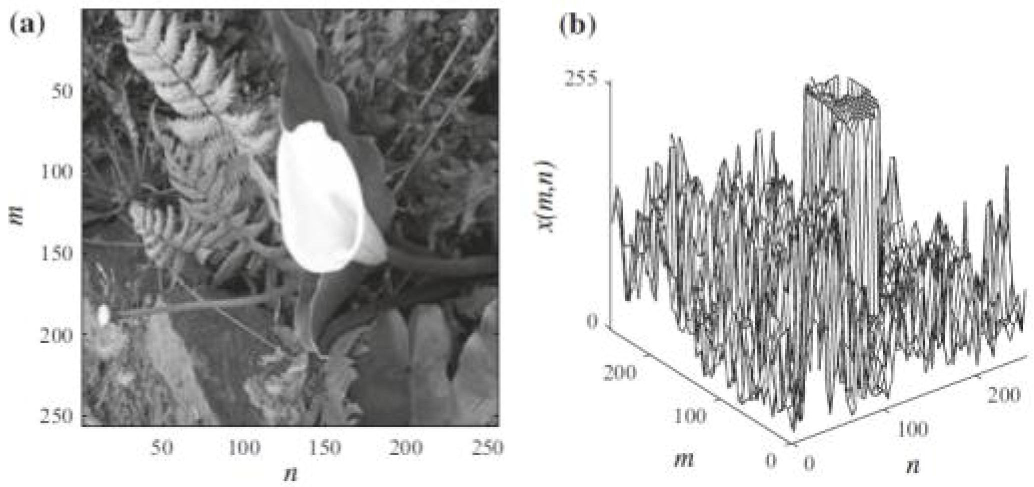
Digital image and color image

- What is Digital Image?



Digital image and color image

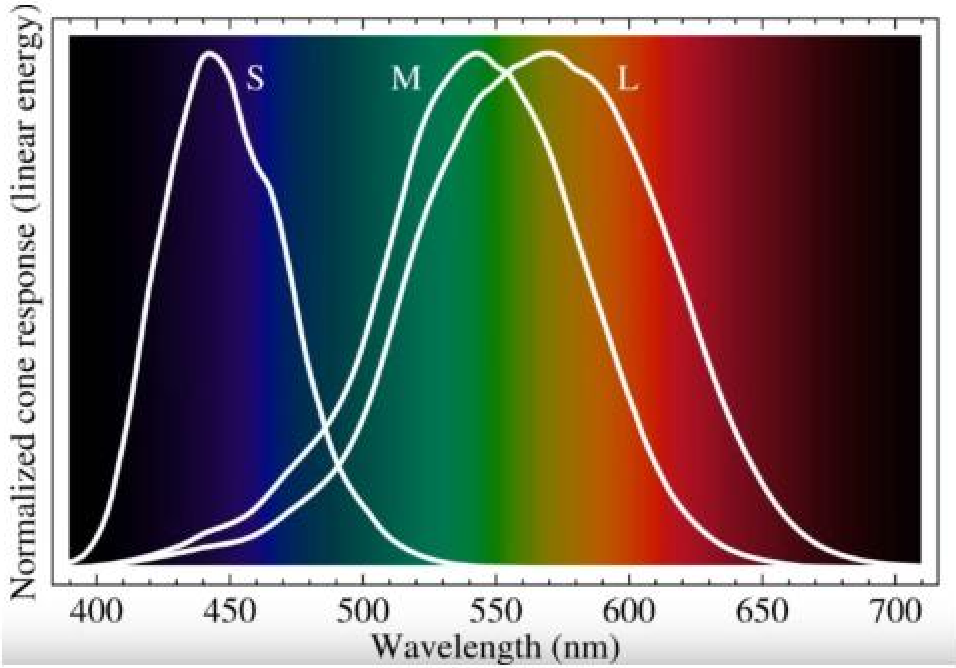
- What is Digital Image?



(a) A 256X256 image with 256 gray levels; (b) its amplitude profile

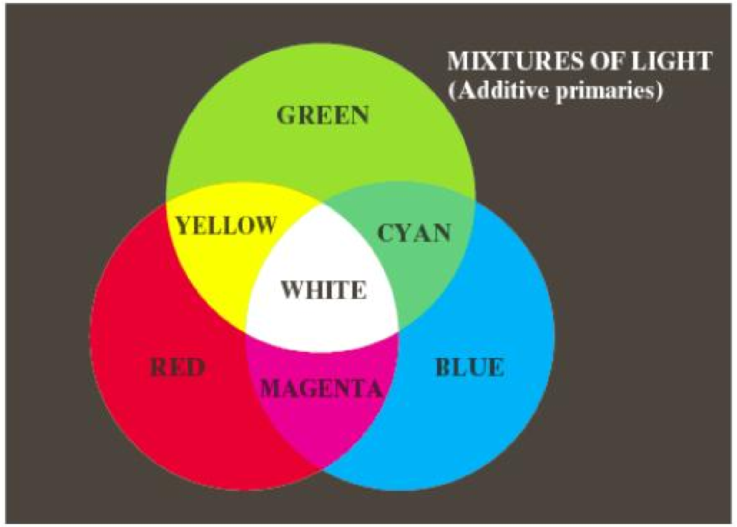
Digital image and color image

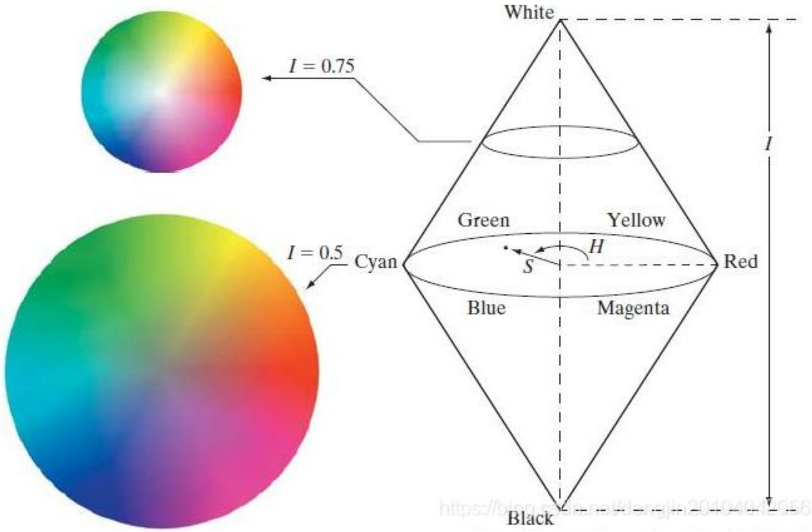
- Color image
- Human visual system color space – the LMS color space
- 3 types of cones sensitive to red, green and blue respectively



Digital image and color image

- Color image





$$\cancel{f(x, y)} \longrightarrow f(x, y, c)$$

Outline

- **Color image processing**

  -Digital image and color image

  -Color balance

  -HSI color space

  -Demosaicing

- **Spatial filtering**

  -Spatial noise

  -Linear and nonlinear spatial filter

- **Intensity transformation**

  -Histogram

  -Log transformation and Gamma transformation

  -Histogram processing

Color balance

- White balance

$$I(x,y) = 0.299 f_R(x,y) + 0.587 f_G(x,y) + 0.114 f_B(x,y)$$

$$k_R = \frac{\bar{I}}{\bar{f}_R} \quad k_G = \frac{\bar{I}}{\bar{f}_G} \quad k_B = \frac{\bar{I}}{\bar{f}_B}$$

$$\begin{bmatrix} g_R(x,y) \\ g_G(x,y) \\ g_B(x,y) \end{bmatrix} = \begin{bmatrix} k_R & & \\ & k_G & \\ & & k_B \end{bmatrix} \begin{bmatrix} f_R(x,y) \\ f_G(x,y) \\ f_B(x,y) \end{bmatrix}$$

- Calculate $I(x,y)$;
- Find means of I, f_R, f_G & f_B;
- Calculate coefficient k_R, k_G & k_B;
- $g(x,y) = k * f(x,y)$

Outline

- **Color image processing**

  -Digital image and color image

  -Color balance

  -HSI color space

  -Demosaicing

- **Spatial filtering**

  -Spatial noise

  -Linear and nonlinear spatial filter

- **Intensity transformation**

  -Histogram

  -Log transformation and Gamma transformation

  -Histogram processing

HSI color space

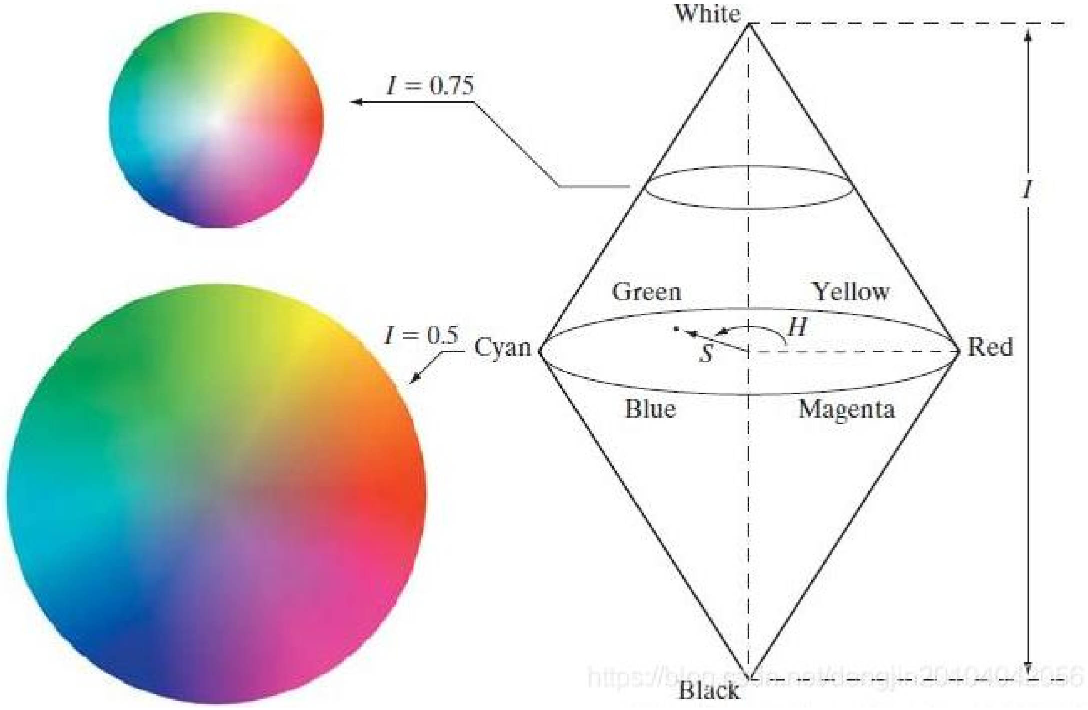
- RGB color space to HSI color space

$$I = \frac{1}{3}(R+G+B)$$

$$S = 1 - \frac{3min(R,G,B)}{R+G+B}$$

$$H = \begin{cases} \theta . G \geq B \\ 2\pi - \theta , G < B \end{cases}$$

$$\theta = cos^{-1}\left[\frac{\frac{1}{2}[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right]$$



HSI color space

- HSI color space to RGB color space

$$1^\circ if\ H \in [0^\circ, 120^\circ):$$
$$H = H$$
$$\begin{cases} B = I(1-S) \\ R = I[1 + \frac{S\cos H}{\cos(60^\circ - H)}] \\ G = 3I - (R+B) \end{cases}$$

$$2^\circ if\ H \in [120^\circ, 240^\circ):$$
$$H = H - 120^\circ$$
$$\begin{cases} R = I(1-S) \\ G = I[1 + \frac{S\cos H}{\cos(60^\circ - H)}] \\ B = 3I - (R+G) \end{cases}$$

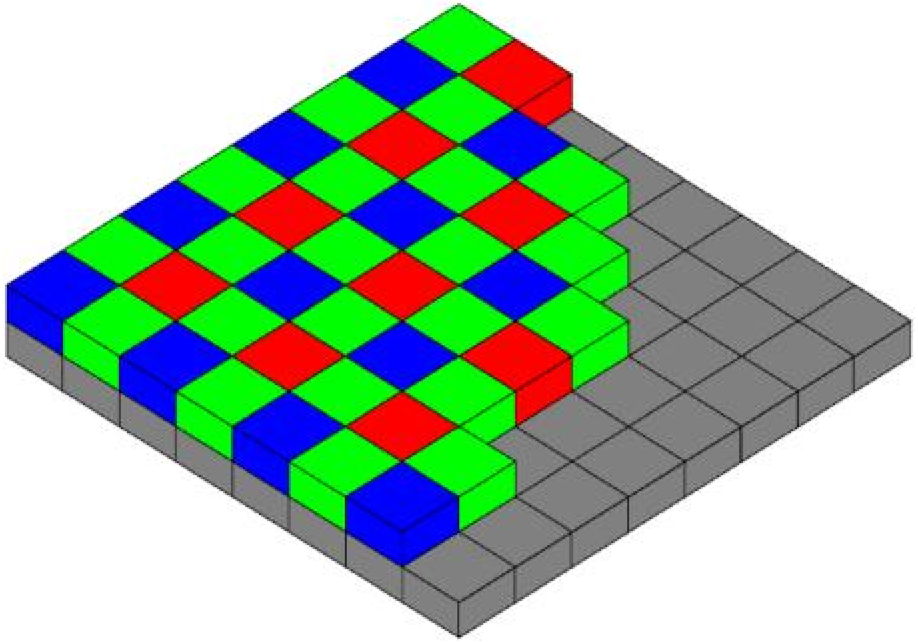
$$3^\circ if\ H \in [240^\circ, 360^\circ):$$
$$H = H - 240^\circ$$
$$\begin{cases} G = I(1-S) \\ B = I[1 + \frac{S\cos H}{\cos(60^\circ - H)}] \\ R = 3I - (G+B) \end{cases}$$

Outline

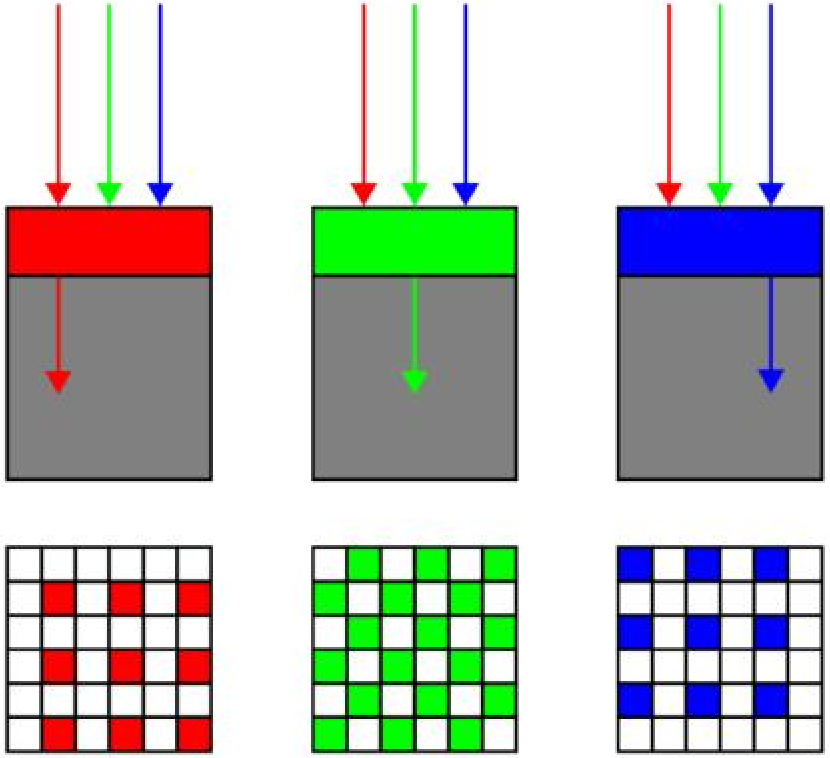
- **Color image processing**
  - -Digital image and color image
  - -Color balance
  - -HSI color space
  - -Demosaicing
- **Spatial filtering**
  - -Spatial noise
  - -Linear and nonlinear spatial filter
- **Intensity transformation**
  - -Histogram
  - -Log transformation and Gamma transformation
  - -Histogram processing

Demosaicing

- Bayer filter

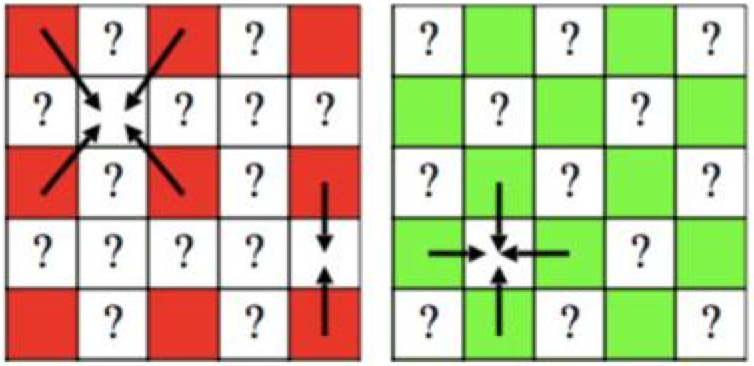


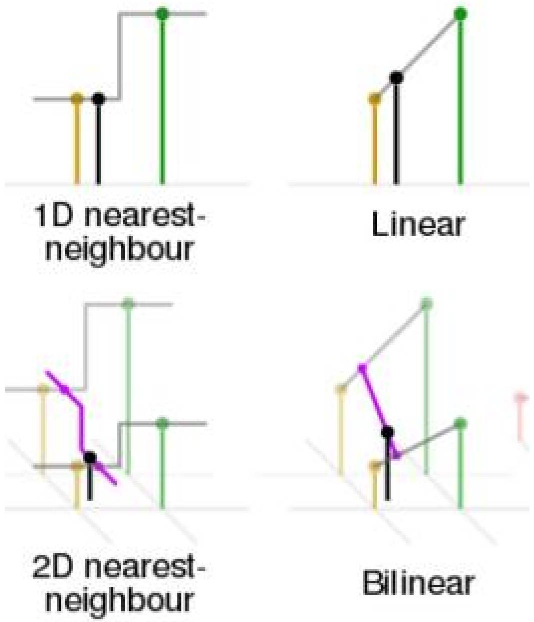
$$f(x, y) \rightarrow f(x, y, c)$$



Demosaicing

- Color image reconstruction





- Nearest-neighbour interpolation (最近相邻插值)
- Bilinear interpolation (双线性插值)

Demosaicing

- Some tips and functions may used in your homework:
- img_R = img(:, :, 1);
- RGB_img = cat(3, img_R, img_G, img_B);
- uint8 & double:

  -uint8 range from 0 to 255;

  -double range from 0 to 1;

  -If you use double(img), the intensity range of the image will not change. It can only change data type.

  -You should use im2double(img) to change the intensity range to [0,1]

Outline

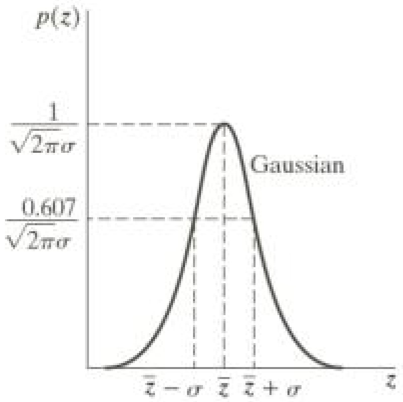
- **Color image processing**
  - -Digital image and color image
  - -Color balance
  - -HSI color space
  - -Demosaicing
- **Spatial filtering**
  - -Spatial noise
  - -Linear and nonlinear spatial filter
- **Intensity transformation**
  - -Histogram
  - -Log transformation and Gamma transformation
  - -Histogram processing

Spatial noise

- Gaussian Noise

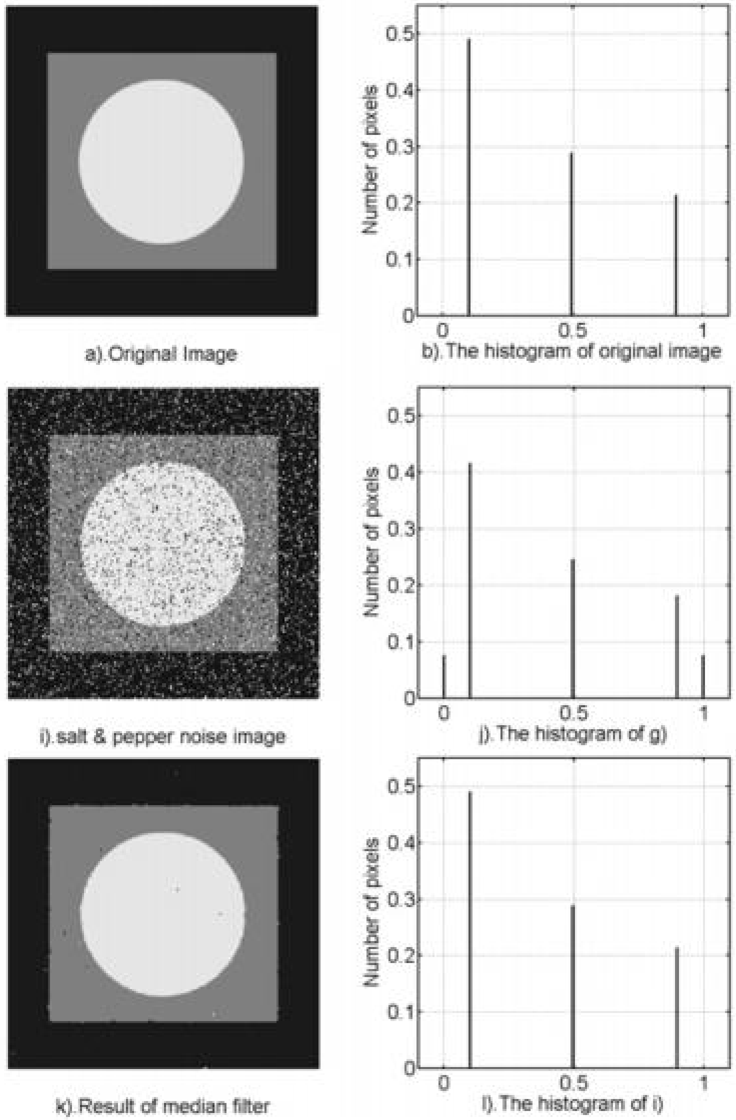
$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$mean = \mu\, ; variance = \sigma^2$$



- Salt-and-pepper Noise(椒盐噪声)

  Pulse Noise(冲激噪声)



Outline

- **Color image processing**
  - -Digital image and color image
  - -Color balance
  - -HSI color space
  - -Demosaicing
- **Spatial filtering**
  - -Spatial noise
  - -Linear and nonlinear spatial filter
- **Intensity transformation**
  - -Histogram
  - -Log transformation and Gamma transformation
  - -Histogram processing

Spatial filter

- A spatial filter is directly applied on the image.
- A spatial filter is also called spatial masks(掩模), kernels(核), templates(模板) or windows(窗口).
- A spatial filter consists of

a) neighborhood

b) a predefined operation

- A spatial filter can be linear and nonlinear

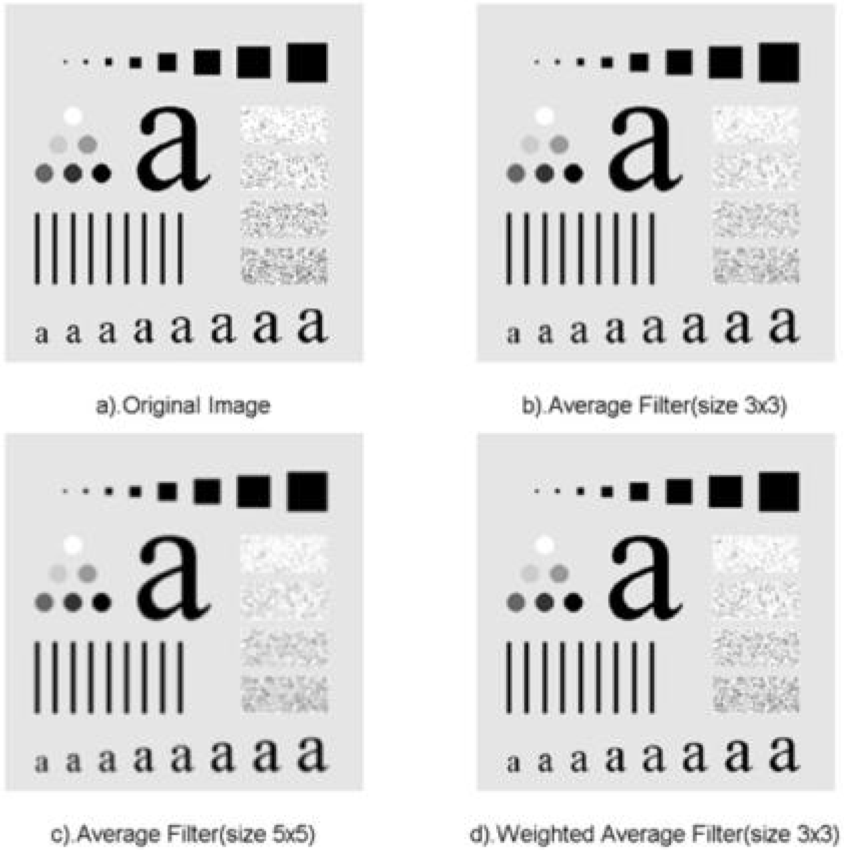
Spatial filter

- Linear Spatial Filter: Average Filter

$$(1/9)*\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} \qquad (1/16)*\begin{bmatrix}1&2&1\\2&4&2\\1&2&1\end{bmatrix}$$

- Nonlinear Spatial Filter: Median Filter

Order-statistic Filter(统计排序滤波器)



a).Original Image

b).Average Filter(size 3x3)

c).Average Filter(size 5x5)

d).Weighted Average Filter(size 3x3)

$$g(x,y) = median\{m \times n\ pixel\ neighbouring\ around\ I(x,y)\}$$

Spatial filter

- Try these built-in functions:
- imnoise(input_image, type, para);
- fspecial(type, para);
- imfilter(input_image, spatial_mask, …);
- medfilt2(input_image, [m n]);
- Other functions...

Outline

- **Color image processing**

  -Digital image and color image

  -Color balance

  -HSI color space

  -Demosaicing
- **Spatial filtering**

  -Spatial noise

  -Linear and nonlinear spatial filter
- **Intensity transformation**

  -Histogram

  -Log transformation and Gamma transformation

  -Histogram processing

Histogram

$$h(r_k) = n_k$$

Where r_k : the kth intensity value in the level range of [0, L-1]

n_k : the number of pixels in the image with intensity r_k

Normalized Histogram（归一化直方图）

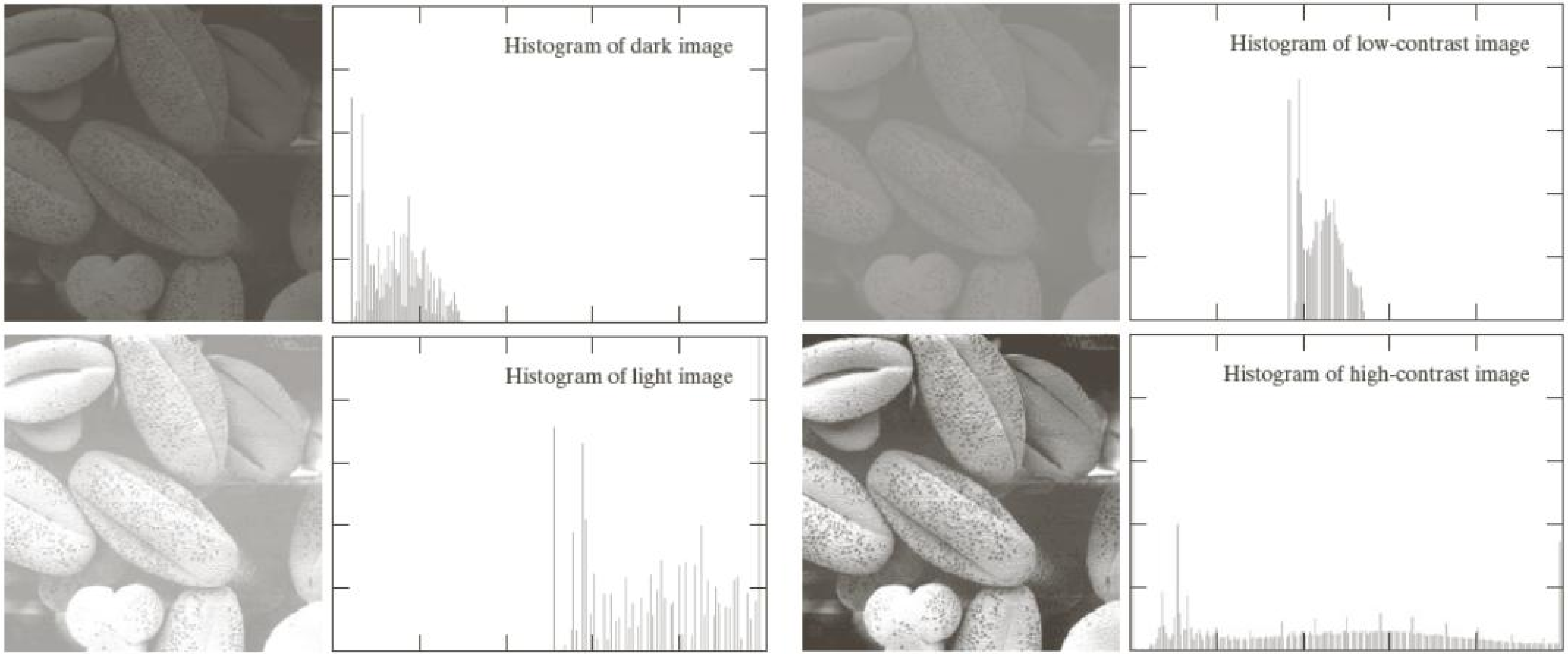
$$p(r_k) = \frac{n_k}{MN}$$

Where $p(r_k)$: the probability of occurrence of intensity r_k in an image

M, N: the row and column dimensions of the image

Histogram

- Histogram examples



Histogram

Histogram properties:

- Describe the number or probability of intensity, no location(spatial) information;
- Can be same as other images;
- $\sum_0^{L-1} n_k = M \cdot N \;\; or \;\; \sum_0^1 p(r_k) = 1$
- If region $C = A \cup B$,A and B are disjoint, $H_c = H_A + H_B$

Outline

- **Color image processing**

  -Digital image and color image

  -Color balance

  -HSI color space

  -Demosaicing

- **Spatial filtering**

  -Spatial noise

  -Linear and nonlinear spatial filter

- **Intensity transformation**

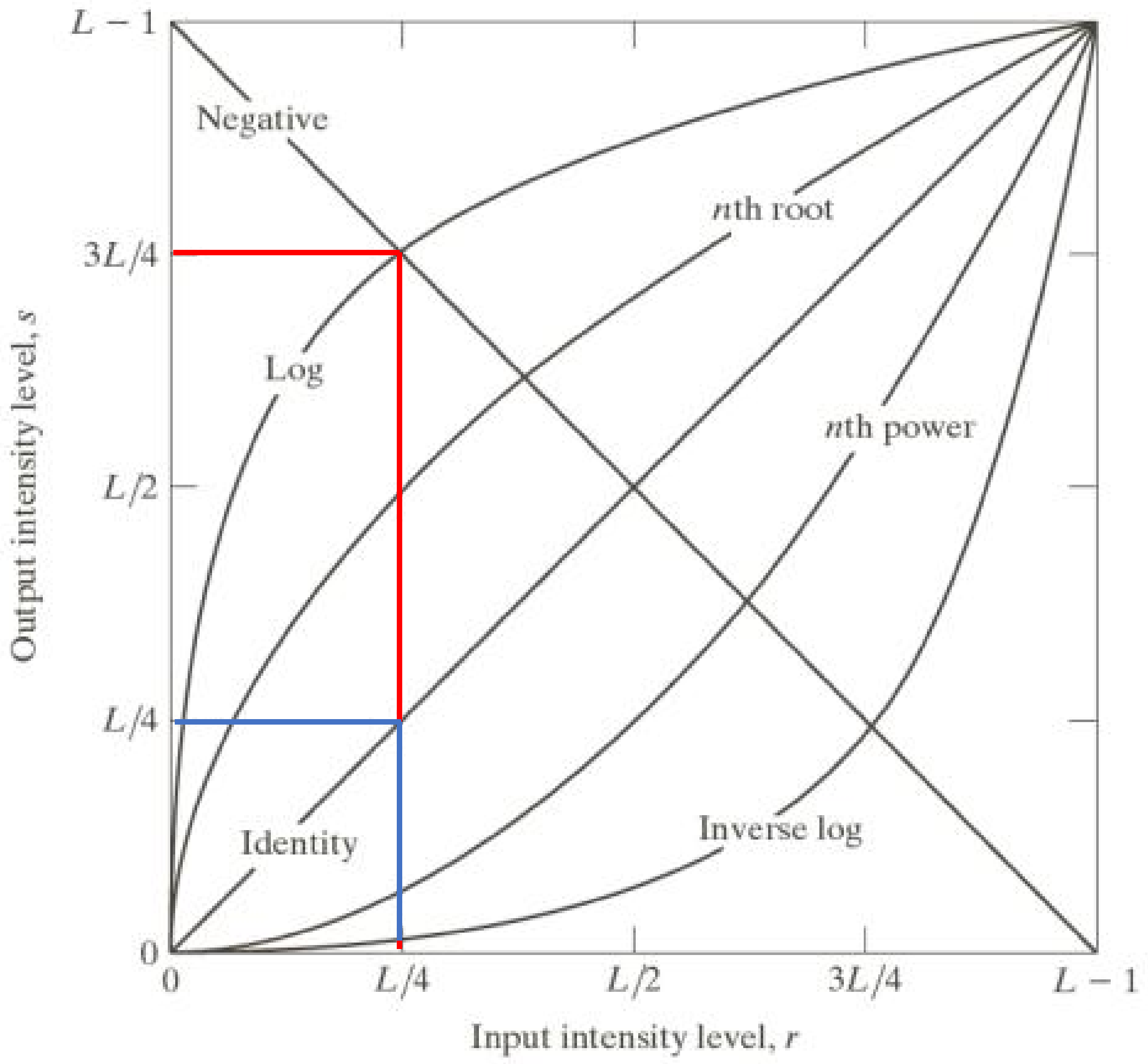
  -Histogram

  -Log transformation and Gamma transformation

  -Histogram processing

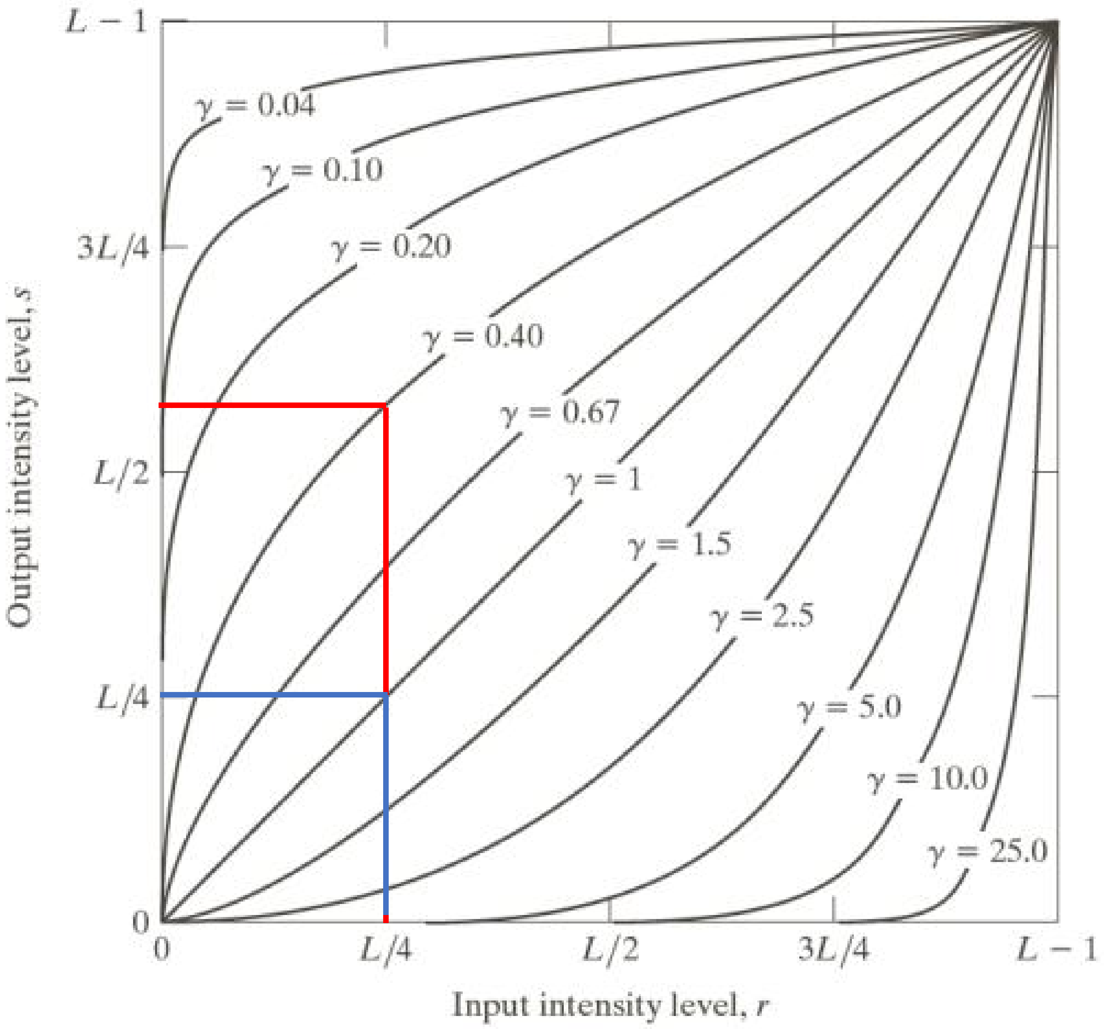
Log transformation

- $s = c\log(1 + r)$
- $s, r \in [0, L - 1]$



Gamma transformation

- $s = c \cdot r^{\gamma}$
- $s, r \in [0, L-1]$



Outline

- **Color image processing**

  -Digital image and color image

  -Color balance

  -HSI color space

  -Demosaicing

- **Spatial filtering**

  -Spatial noise

  -Linear and nonlinear spatial filter

- **Intensity transformation**

  -Histogram

  -Log transformation and Gamma transformation

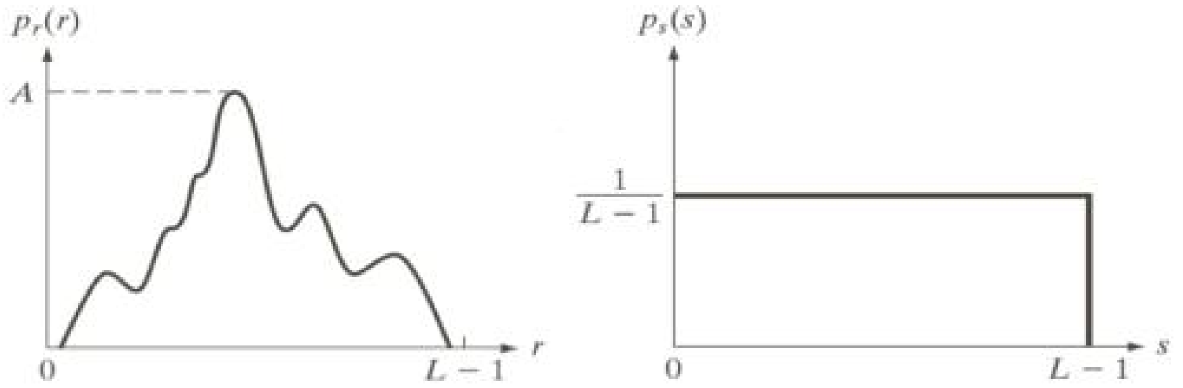
  -Histogram processing

Histogram equalization

- Uniform Probability density function : $p_s(s) = \frac{1}{L-1}$
- The probability density function (PDF) of s is

$$p_s(s) = p_r(r) \cdot \frac{dr}{ds} \Rightarrow p_r(r) \cdot \frac{dr}{ds} = \frac{1}{L-1} \Rightarrow (L-1)p_r(r) \cdot dr = ds$$

- Transformation function : $s = T(r) = (L-1)\int_0^r p_r(w)dw$

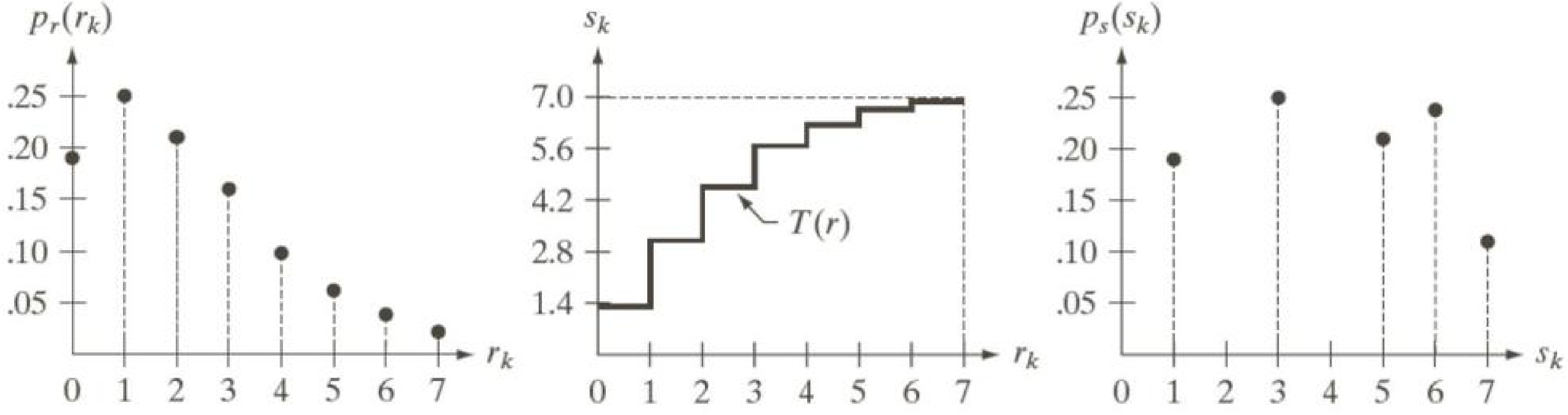


Histogram equalization

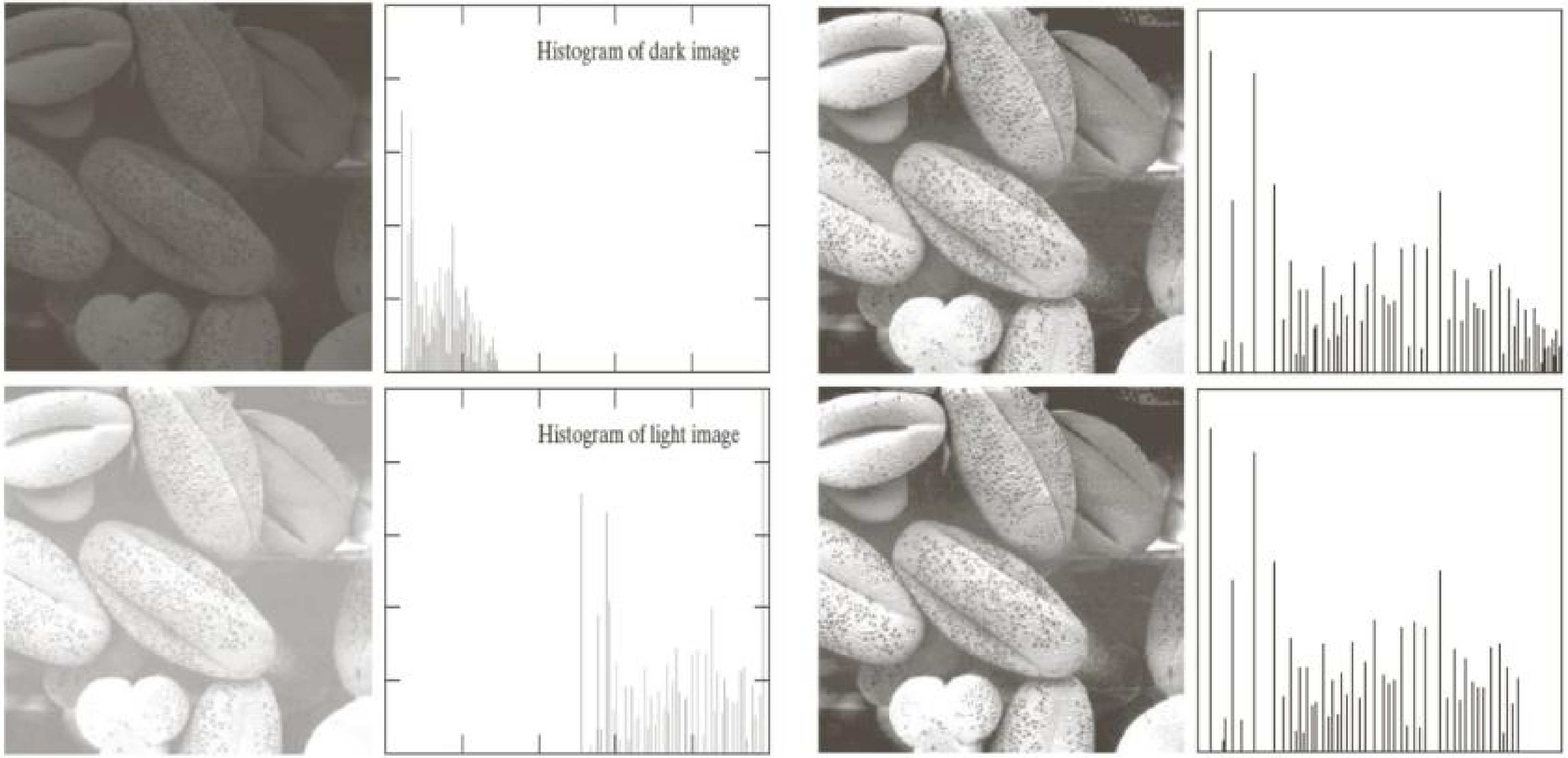
$$s = T(r) = (L-1)\int_0^r p_r(w)dw = (L-1)\sum_{j=0}^{k} p_r(r_j) = (L-1)\sum_{j=0}^{k}\frac{n_j}{MN} = \frac{L-1}{MN}\sum_{j=0}^{k} n_j \; \mathrm{k}$$
$$= 1, 2, \ldots, \mathrm{L}-1$$

r_k	n_k	$p_r(r_k)$	s_k		s_k	$p_s(s_k)$
0	790	0.19	1.33	1	0	0
1	1023	0.25	3.08	3	1	0.19
2	850	0.21	4.55	5	2	0
3	656	0.16	5.67	6	3	0.25
4	329	0.08	6.23	6	4	0
5	245	0.06	6.65	7	5	0.21
6	122	0.03	6.86	7	6	0.24
7	81	0.02	7.00	7	7	0.11

Histogram equalization



Histogram equalization



Histogram matching

- Generate a processed image with a specified histogram:

  For input: $s = T(r) = (L-1)\int_0^r p_r(w)dw$

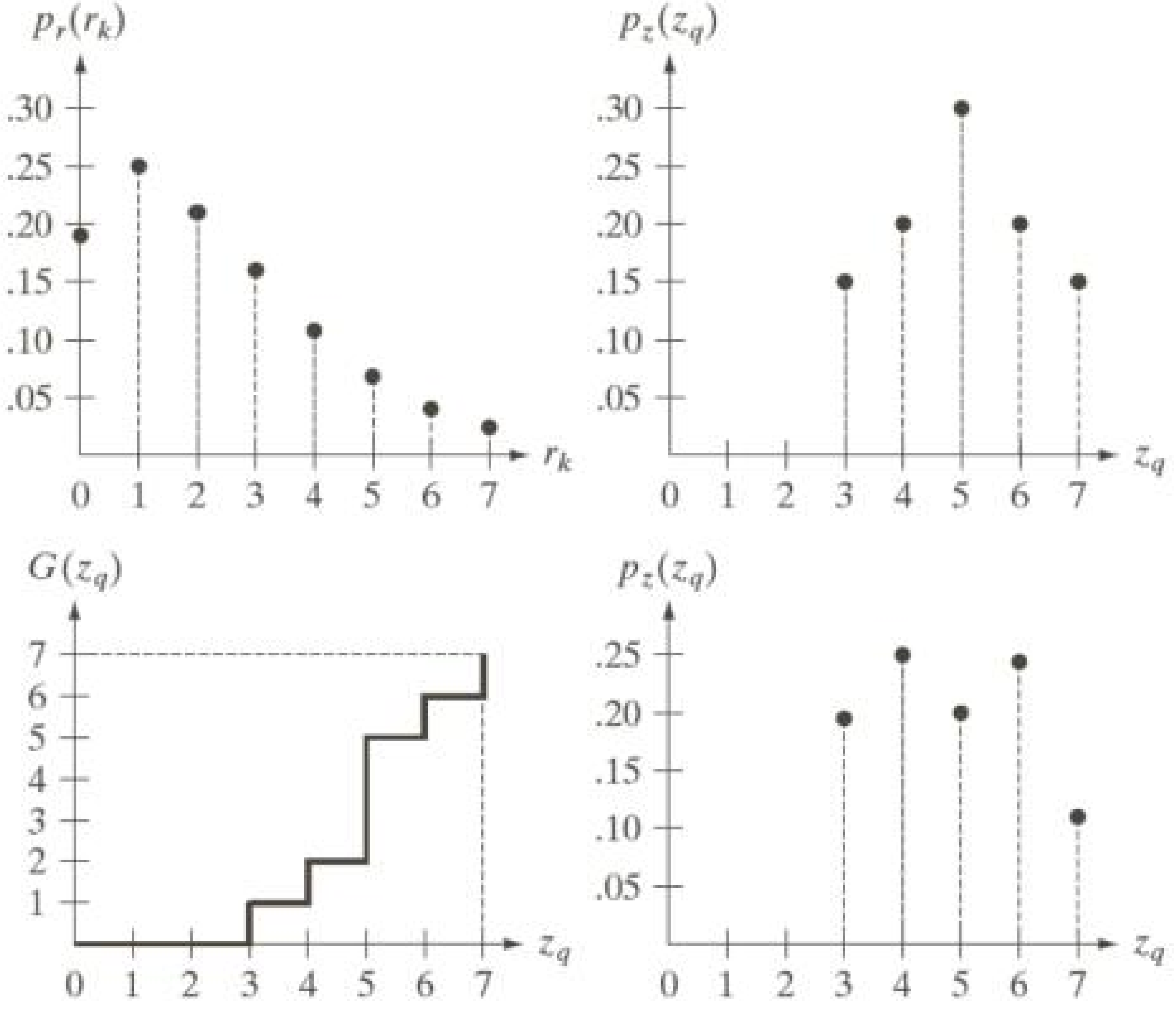
  For output: $G(z) = (L-1)\int_0^z p_z(t)dt$

  Therefore, $z = G^{-1}(s) = G^{-1}[T(r)]$

Histogram matching

r_k	$p(r_k)$	$s_k=T(r_k)$	z_q	$p(z_q)$	$s_k=G(z_q)$	$s_k \rightarrow z_k$	$r_k \rightarrow z_k$	z_k	$p(z_k)$
0	0. 19	1	0	0	0	0→0, 1, 2	0→3	0	0
1	0. 25	3	1	0	0	1→3	1→4	1	0
2	0. 21	5	2	0	0	2→4	2→5	2	0
3	0. 16	6	3	0. 15	1		3→6	3	0. 19
4	0. 08	6	4	0. 20	2		4→6	4	0. 25
5	0. 06	7	5	0. 30	5	5→5	5→7	5	0. 21
6	0. 03	7	6	0. 20	6	6→6	6→7	6	0. 24
7	0. 02	7	7	0. 15	7	7→7	7→7	7	0. 11

Histogram matching



Histogram matching