计算机视觉
Computer Vision

-- Statistics

钟 凡

zhongfan@sdu. edu. cn

描述一个像素

- 空间坐标 (x, y)
- 颜色值（RGB, YUV, …）

- (x, y, R, G, B)

多个像素?

- ……

多个像素

- 空间坐标 (x, y)
  - 图像的结构;
- 颜色值（RGB, YUV, …）
  - 像素颜色的分布（统计特征）；

直方图

Histogram

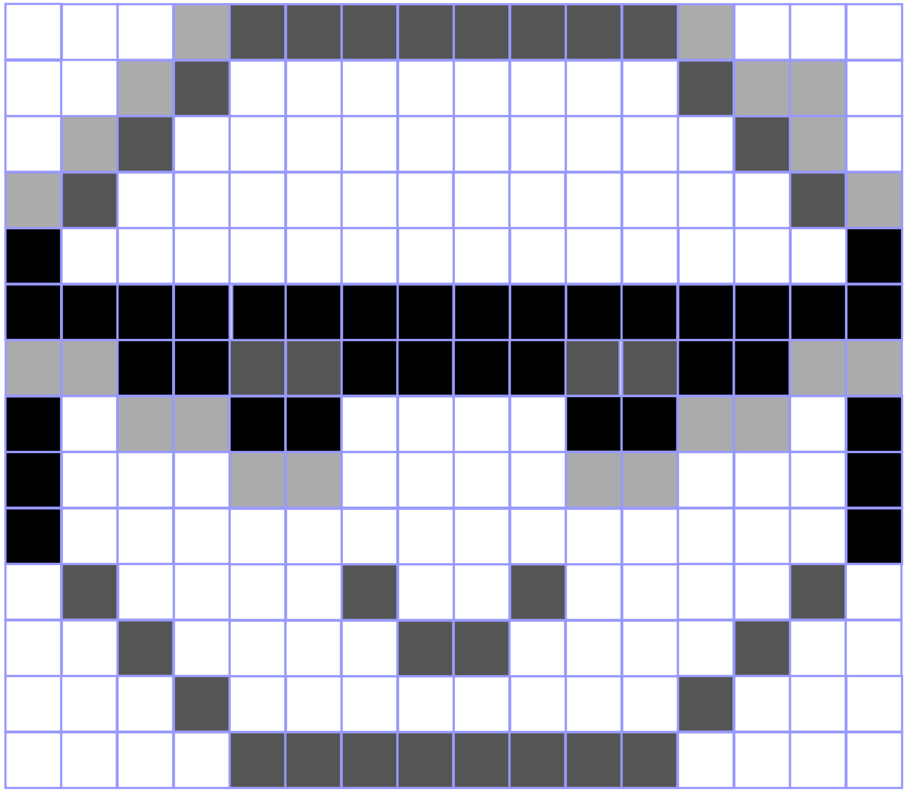
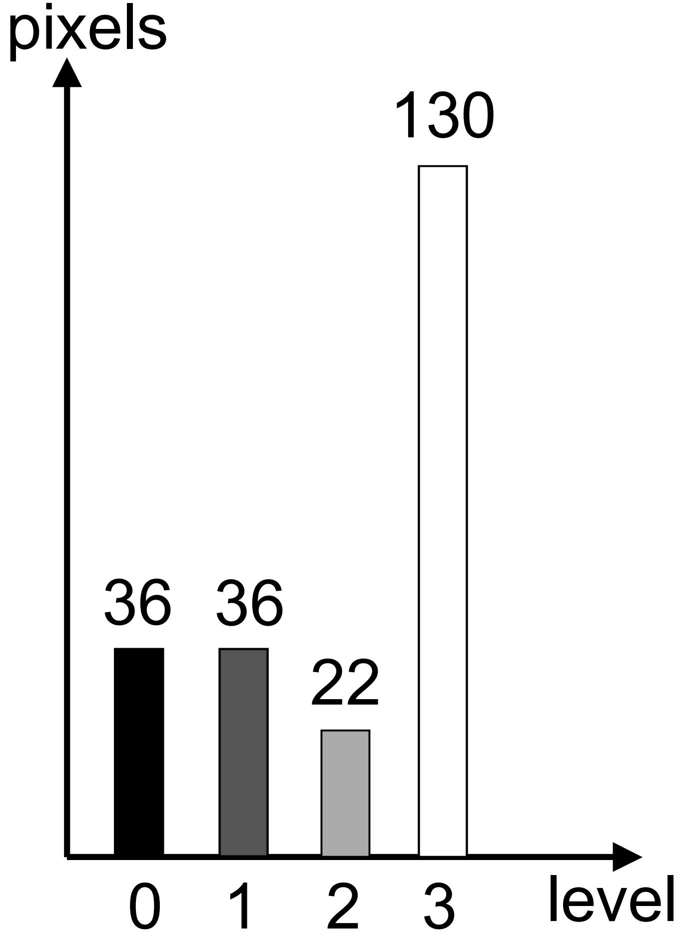


Image 16x14 = 224 pixels



Histogram Processing

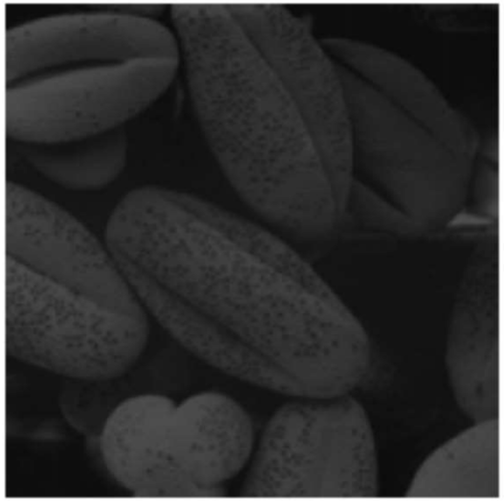
- The histogram of a digital image with gray levels from 0 to L-1 is a discrete function $h(r_k) = n_k$, where:
  - r_k is the kth gray level
  - n_k is the number of pixels in the image with that gray level
  - n is the total number of pixels in the image
  - $k = 0, 1, 2, \ldots, L-1$
- Normalized histogram: $p(r_k) = n_k / n$
  - sum of all components = 1
- The shape of the histogram of an image does provide useful information about the possibility for contrast enhancement

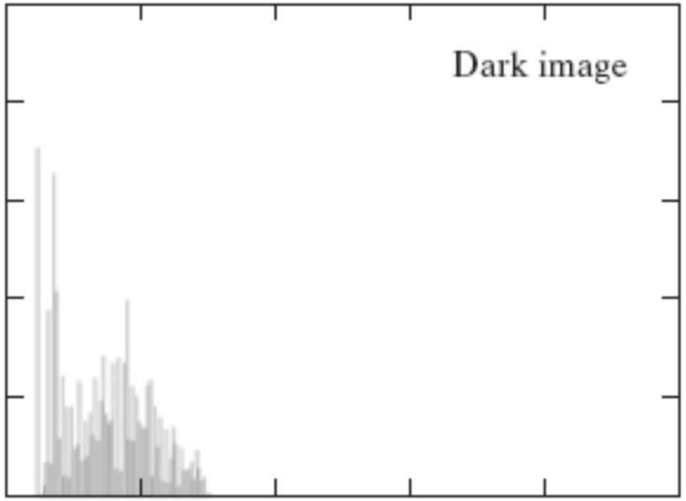
Calculate Histogram

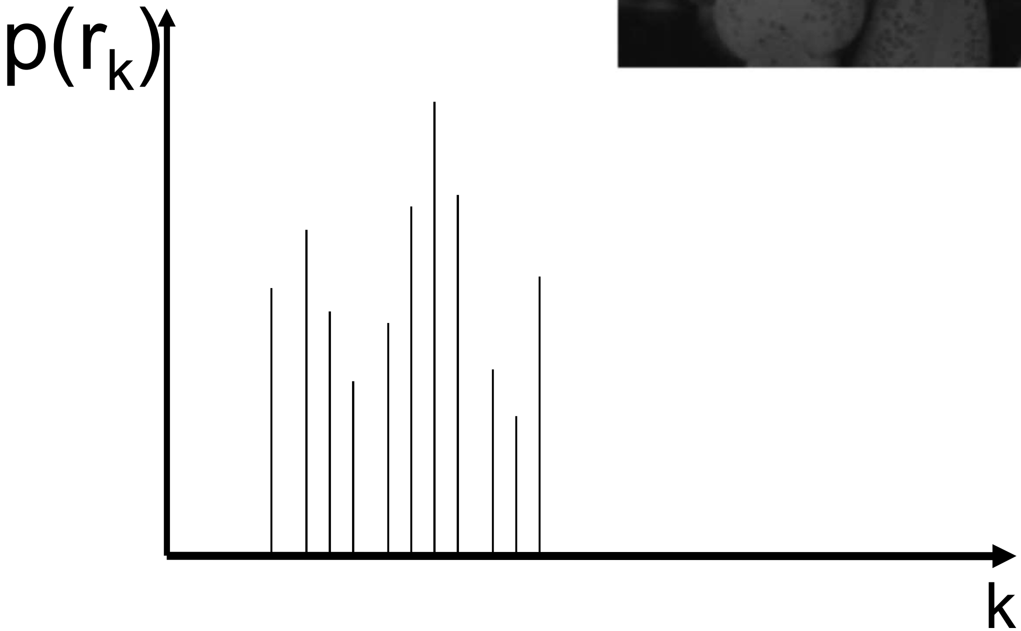
```
void calc_hist(uchar *data, int width, int height, int step, int H[256] )
{
    memset(H, 0, sizeof(H[0])*256);

    uchar *row=data;
    for(int yi=0; yi<height; ++yi, row+=step)
    {
       for(int xi=0; xi<width; ++xi)
        {
            H[ row[xi] ]++;
        }
    }
}
```

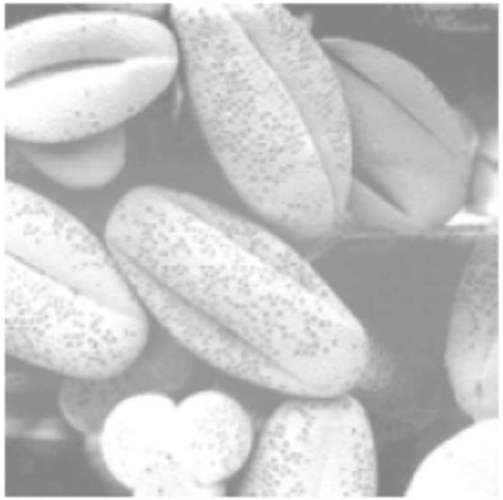
Dark Image

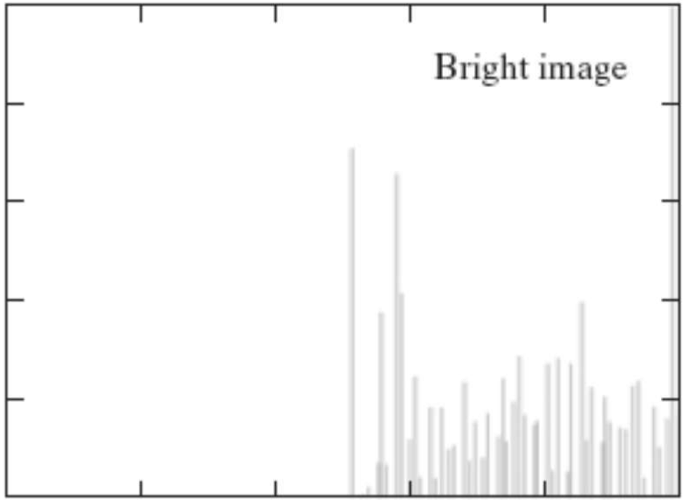


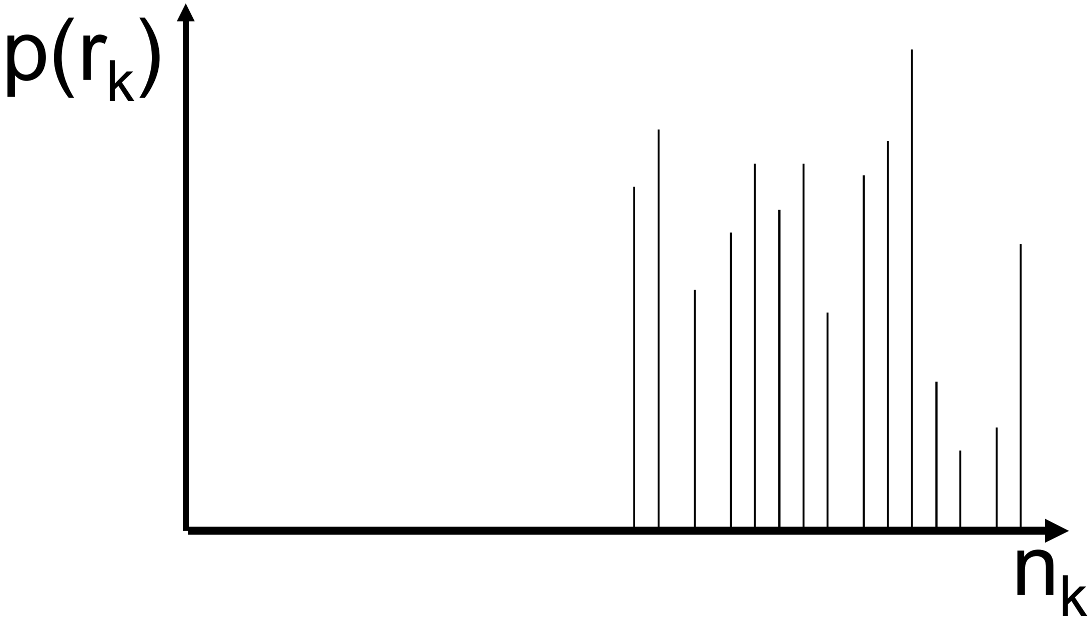




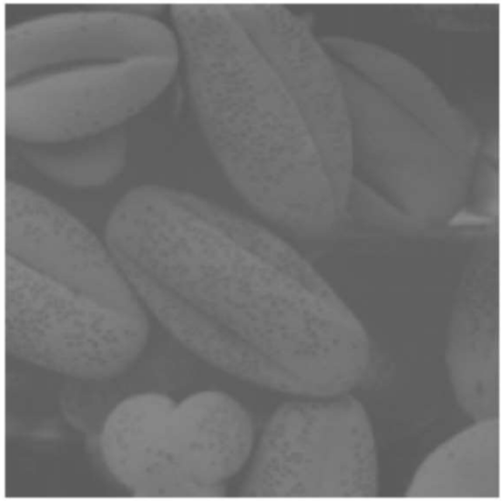
Bright Image

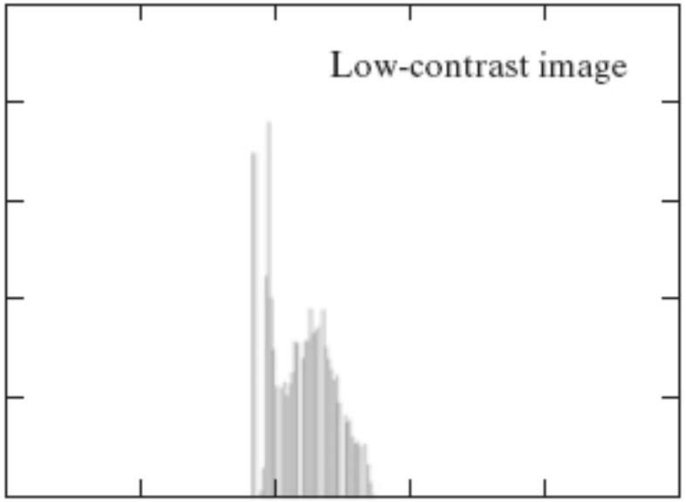


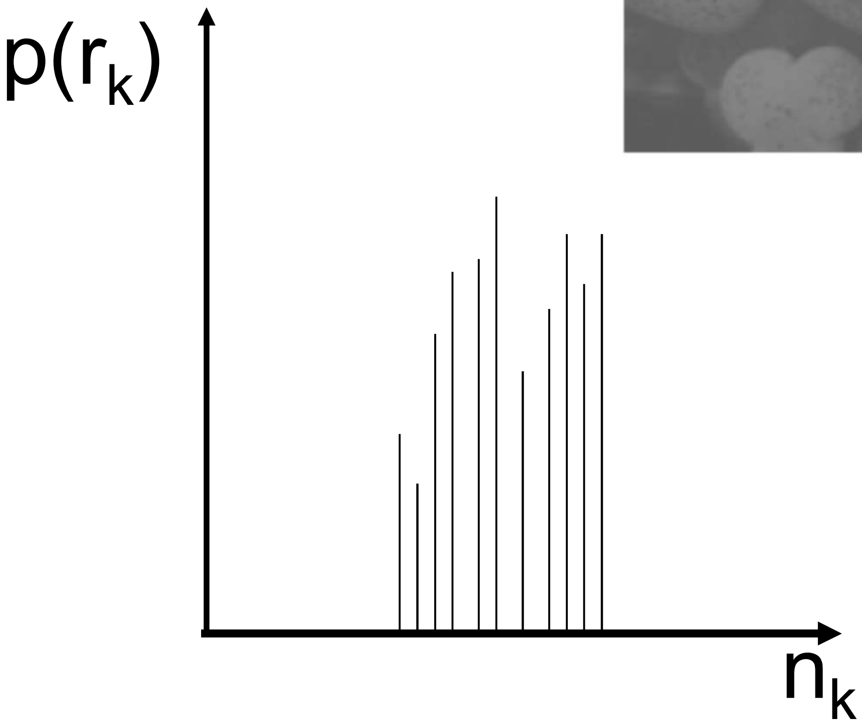




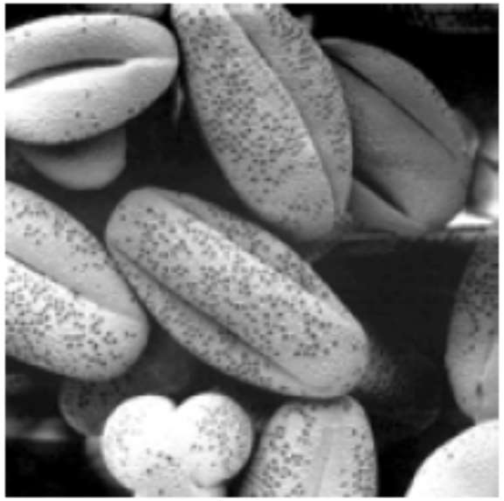
Low-Contrast Image

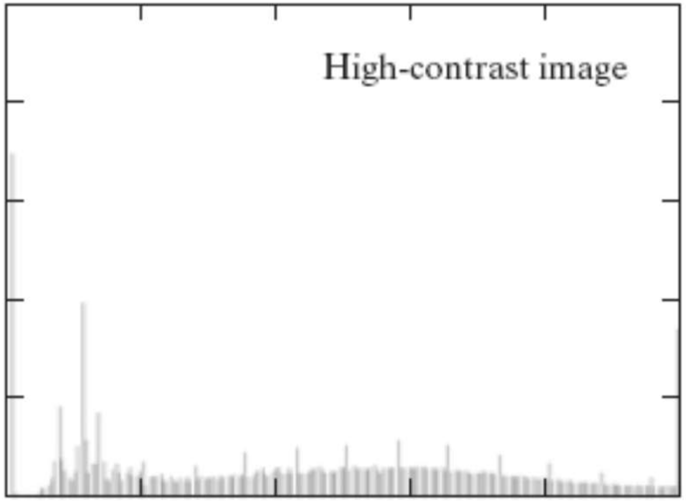


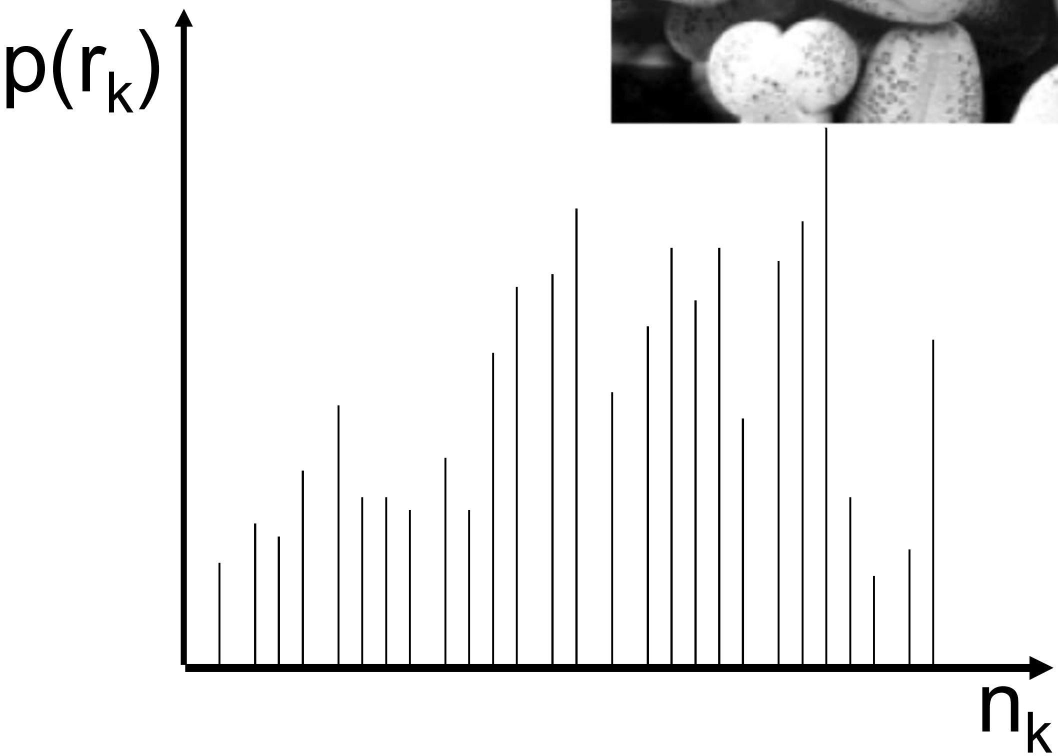




High-Contrast Image



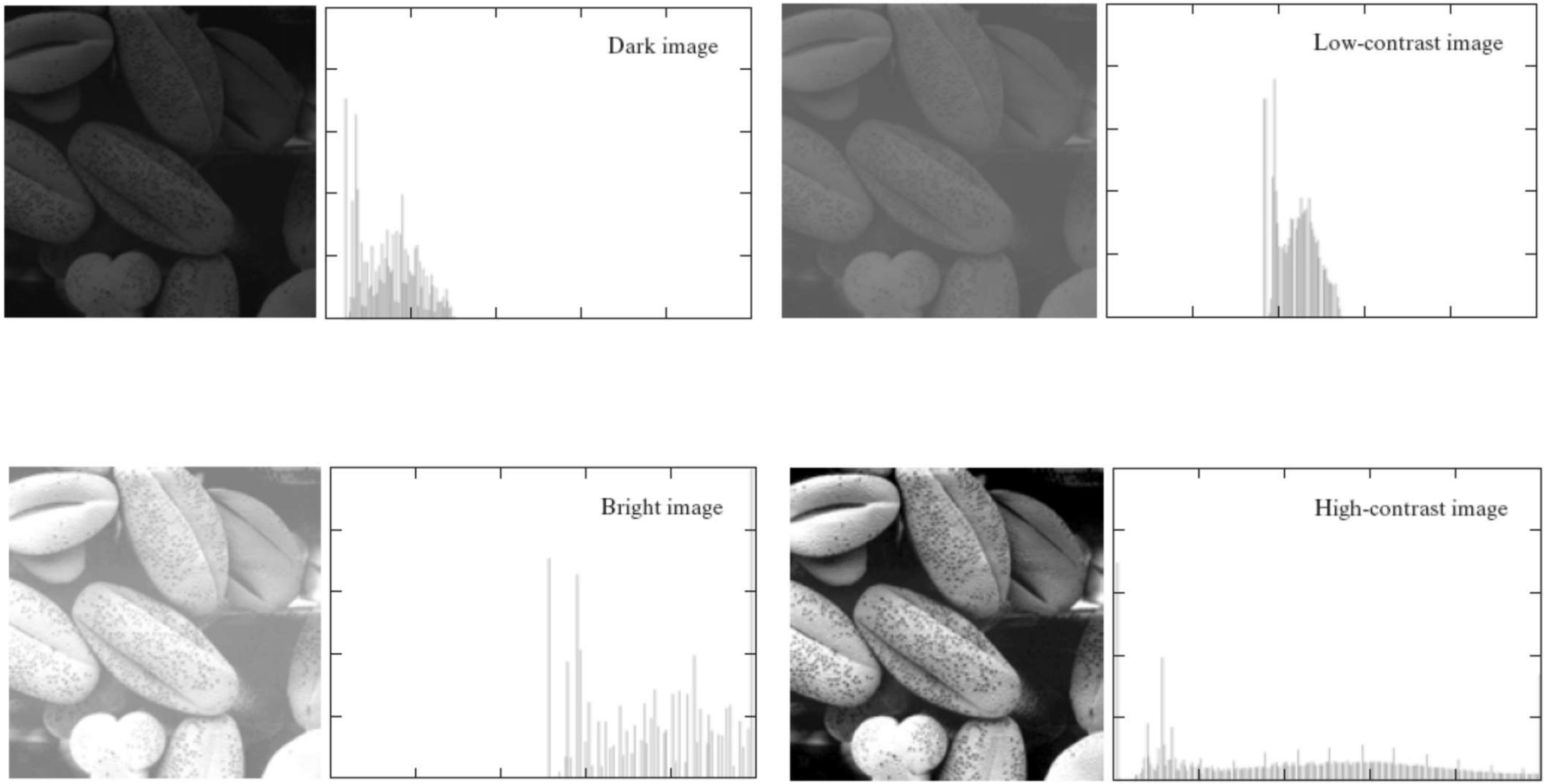




Histogram Processing

- Histogram Equalization
- Histogram Matching
- Local Enhancement
- Enhancement Using Histogram Statistics

Which looks better?

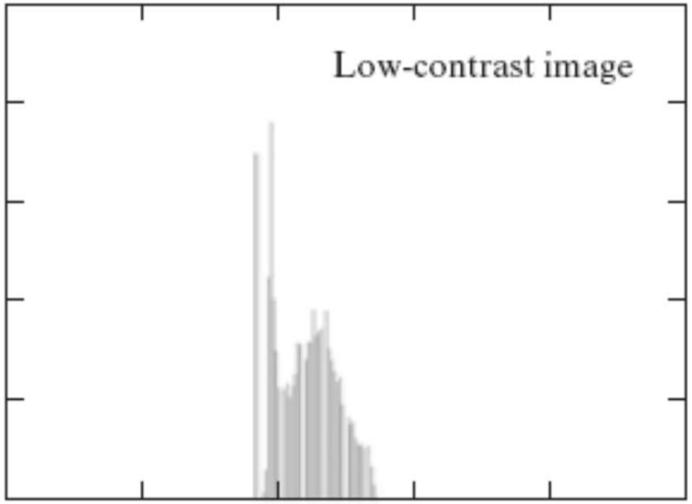


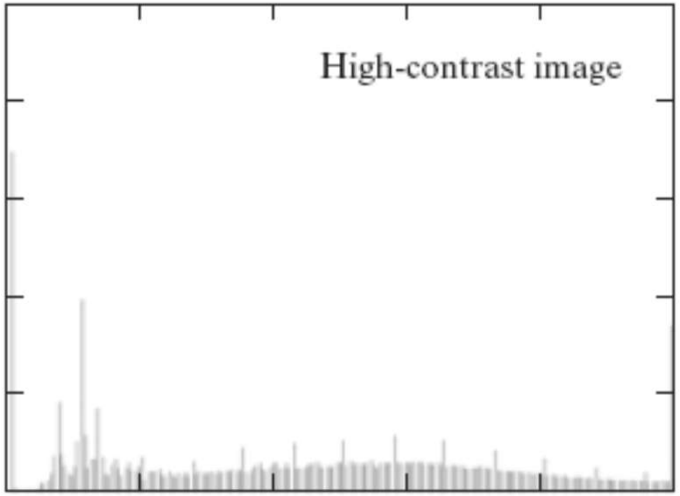
Which looks better?

- The gray levels of pixels :
  - Occupy the entire range of possible gray levels
  - Distributed uniformly

Histogram Equalization

- **For the above purpose…**





How to achieve?

- **Design a transformation function so that the result image occupy the entire range of gray level.**

HE Transformation Function

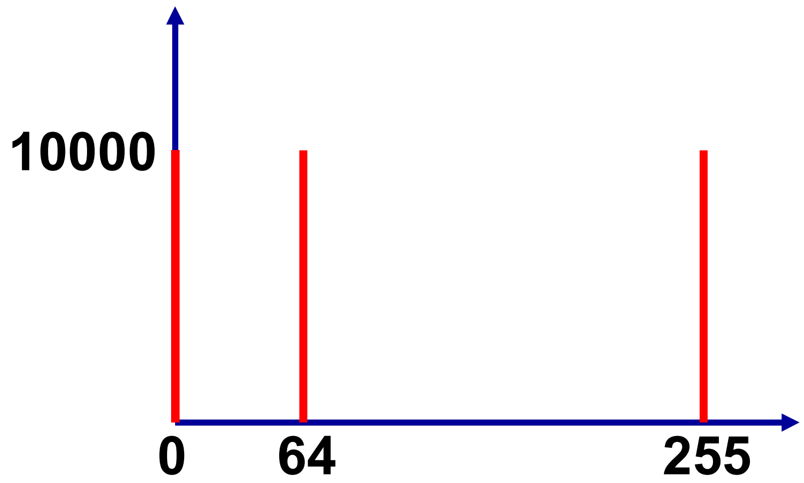
- **Normalization r → [0, 1]**

- **The gray-level transformation function used in histogram equalization:**

$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} = \sum_{j=0}^{k} p_r(r_j)$$

Example

- An image (300X100)
  - Dark



Example

$T(0) = 10000/30000 * 255 = 85$

$T(1) = T(0) + 0/30000 = 85$

$T(2) = T(1) + 0/30000 = 85$

…

$T(63) = T(62) + 0/30000 = 85$

$T(64) = (10000/30000 + 10000/30000)*255 = 170$

$T(65) = T(64) + 0/30000 = 170$

…

$T(253) = T(252) + 0/30000 = 170$

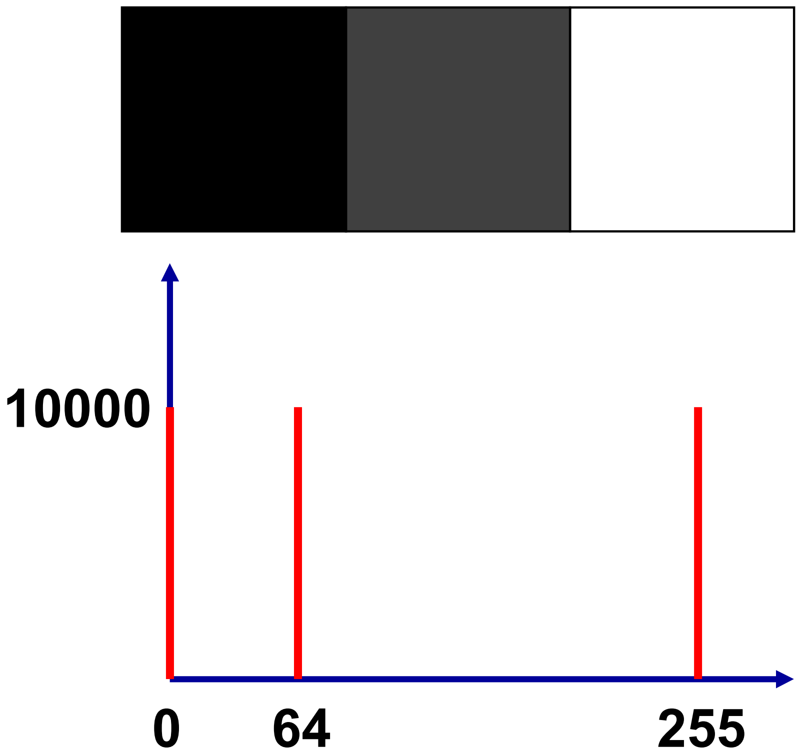
$T(254) = T(253) + 0/30000 = 170$

$T(255) = (10000/30000 + 10000/30000 + 10000/30000)*255 = 255$

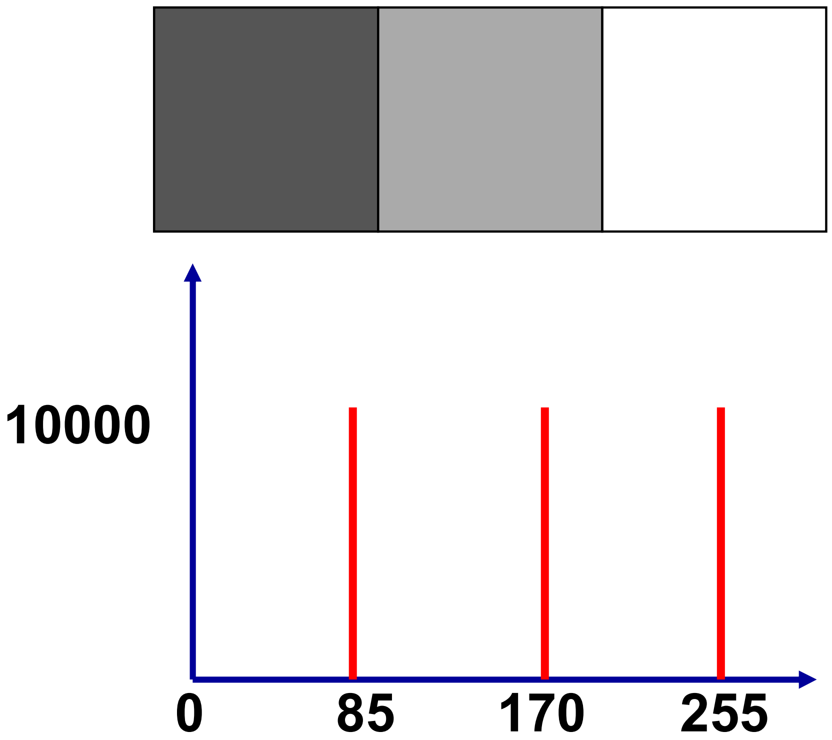
Example

- T(0)   = 85   …   T(63)   = 85
- T(64) = 170   …   T(254) = 170
- T(255) = 255

Original image



Enhanced image



Uniform?

- …

Transformation Functions

- **Consider continuous functions and transformations of the form**
  - $s = T(r) \quad r \in [0, 1]$
- **Subjected to that**
  - $T(r)$: **single-valued, monotonically increasing for** $r \in [0, 1]$
  - $T(r) \in [0, 1]$ **for** $r \in [0, 1]$
  - **Inverse transformation**
    - $r = T^{-1}(s),\ s \in [0, 1]$
- **Probability density function of** s

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

Transformation Functions

- **Probability density function of s**

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

- **Transformation function**

$$\boxed{s = T(r) = \int_0^r p_r(w)dw}$$

  - **Which will satisfy that**

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = \frac{d}{dr}\left[\int_0^r p_r(w)dw\right] = p_r(r)$$

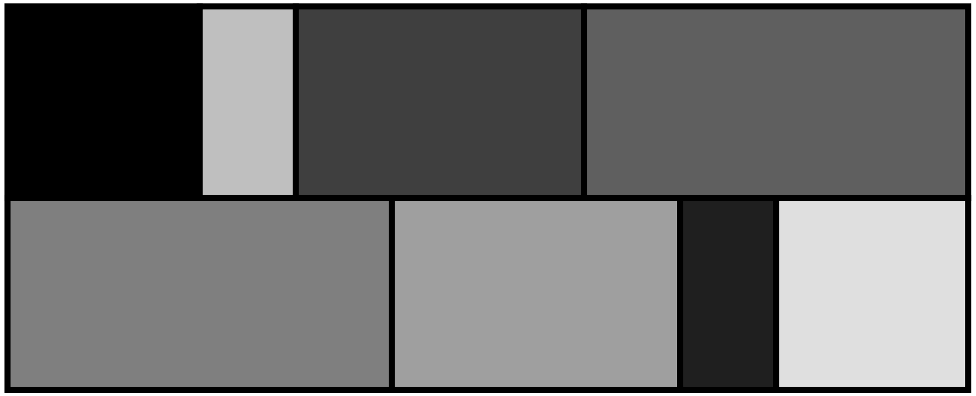
  - **Then s has**

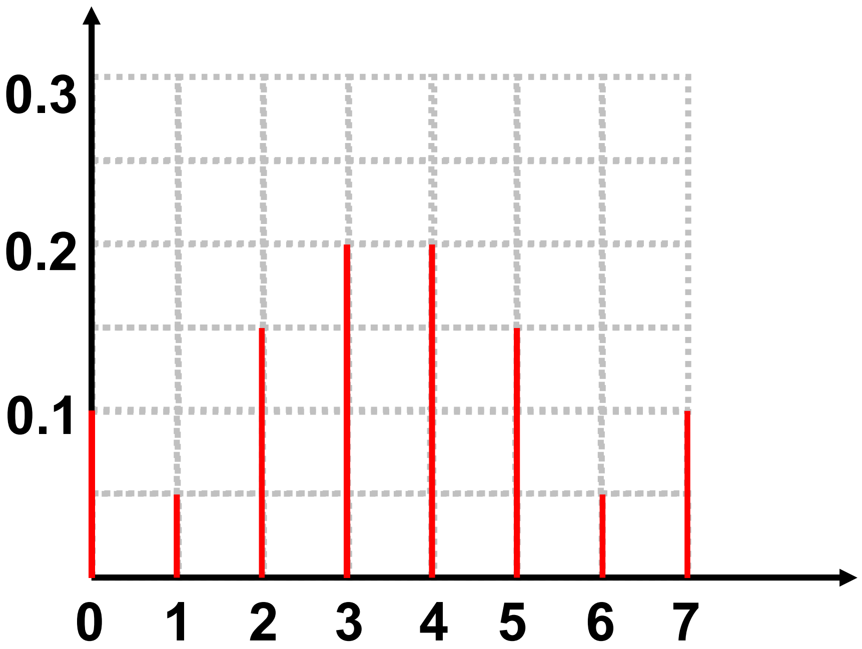
$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = p_r(r)\frac{1}{p_r(r)} = 1$$

The case of discrete

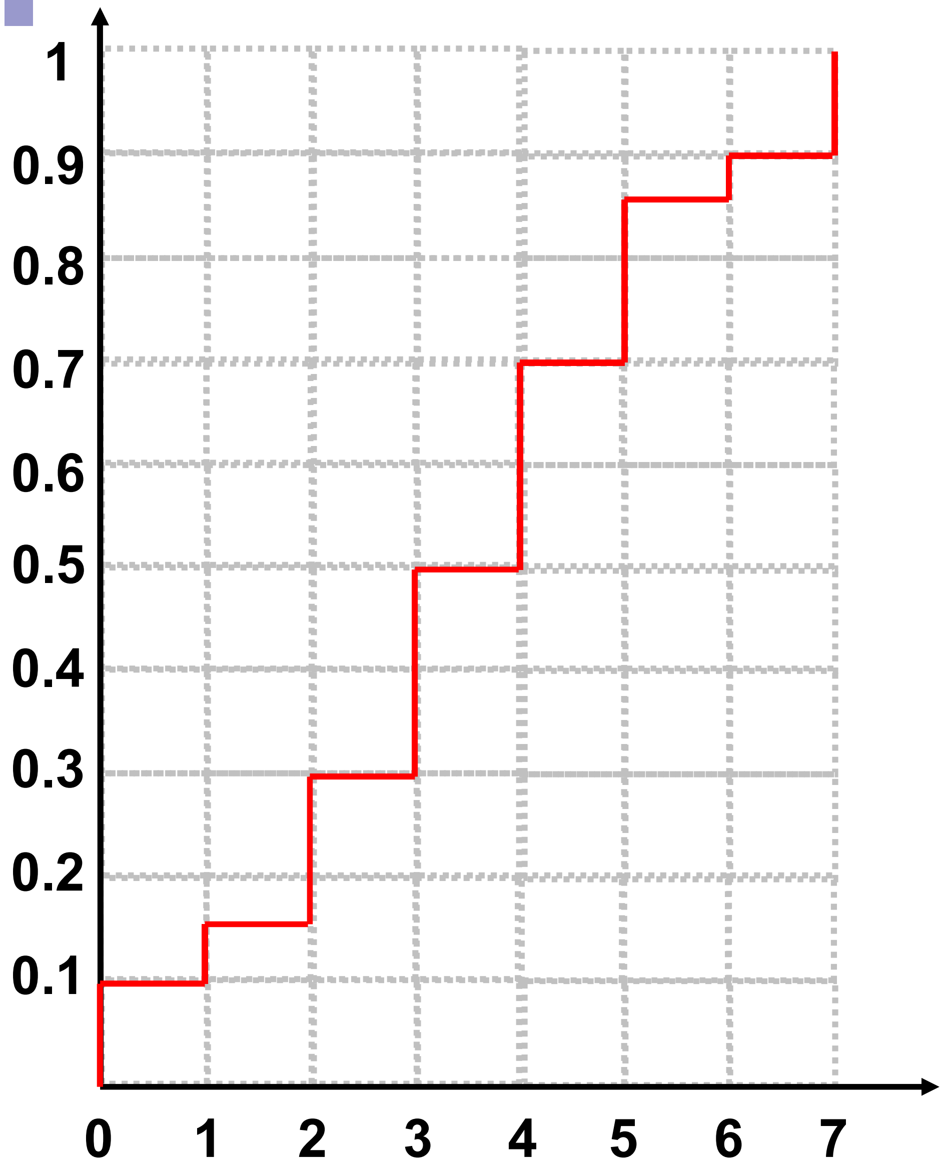
- Still uniform?

Example





Histogram of original image



Cumulate histogram

$T(0) = 0.1 \times 7 = 0.7 \rightarrow 1$

$T(1) = 0.15 \times 7 = 1.05 \rightarrow 1$

$T(2) = 0.3 \times 7 = 2.1 \rightarrow 2$

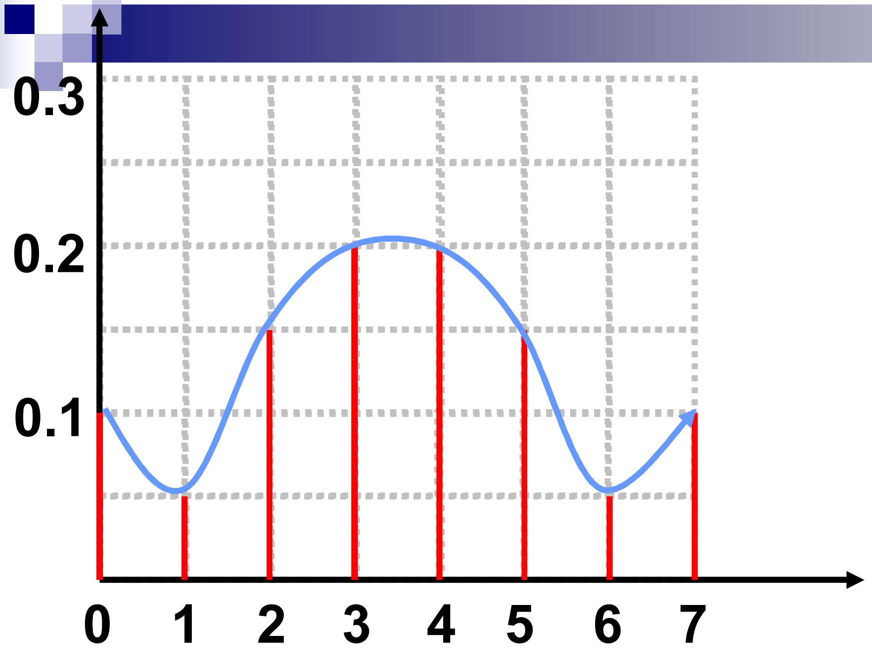
$T(3) = 0.5 \times 7 = 3.5 \rightarrow 4$

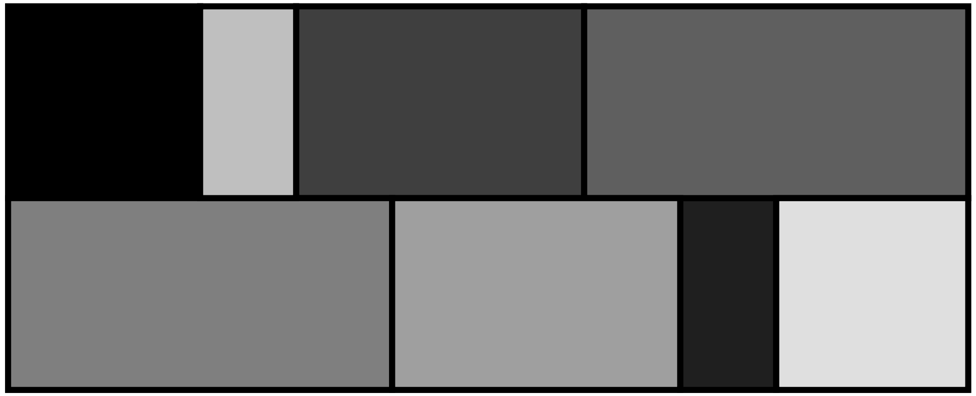
$T(4) = 0.7 \times 7 = 4.9 \rightarrow 5$

$T(5) = 0.85 \times 7 = 5.95 \rightarrow 6$

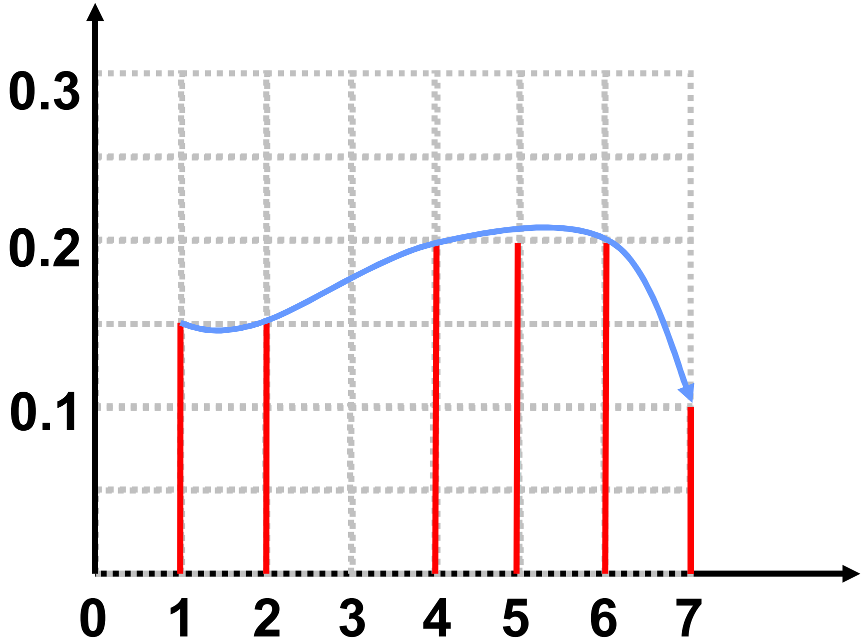
$T(6) = 0.9 \times 7 = 6.3 \rightarrow 6$

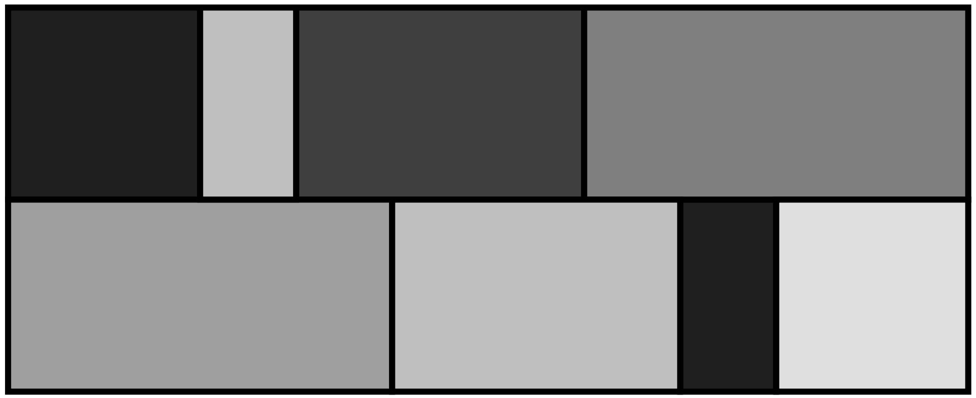
$T(7) = 1 \times 7 = 7 \rightarrow 7$





T(0) → 1, T(1) → 1,
T(2) → 2, T(3) → 4,
T(4) → 5, T(5) → 6,
T(6) → 6, T(7) → 7.





Why not uniform?

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = \frac{d}{dr}\left[\int_0^r p_r(w)dw\right] = p_r(r)$$

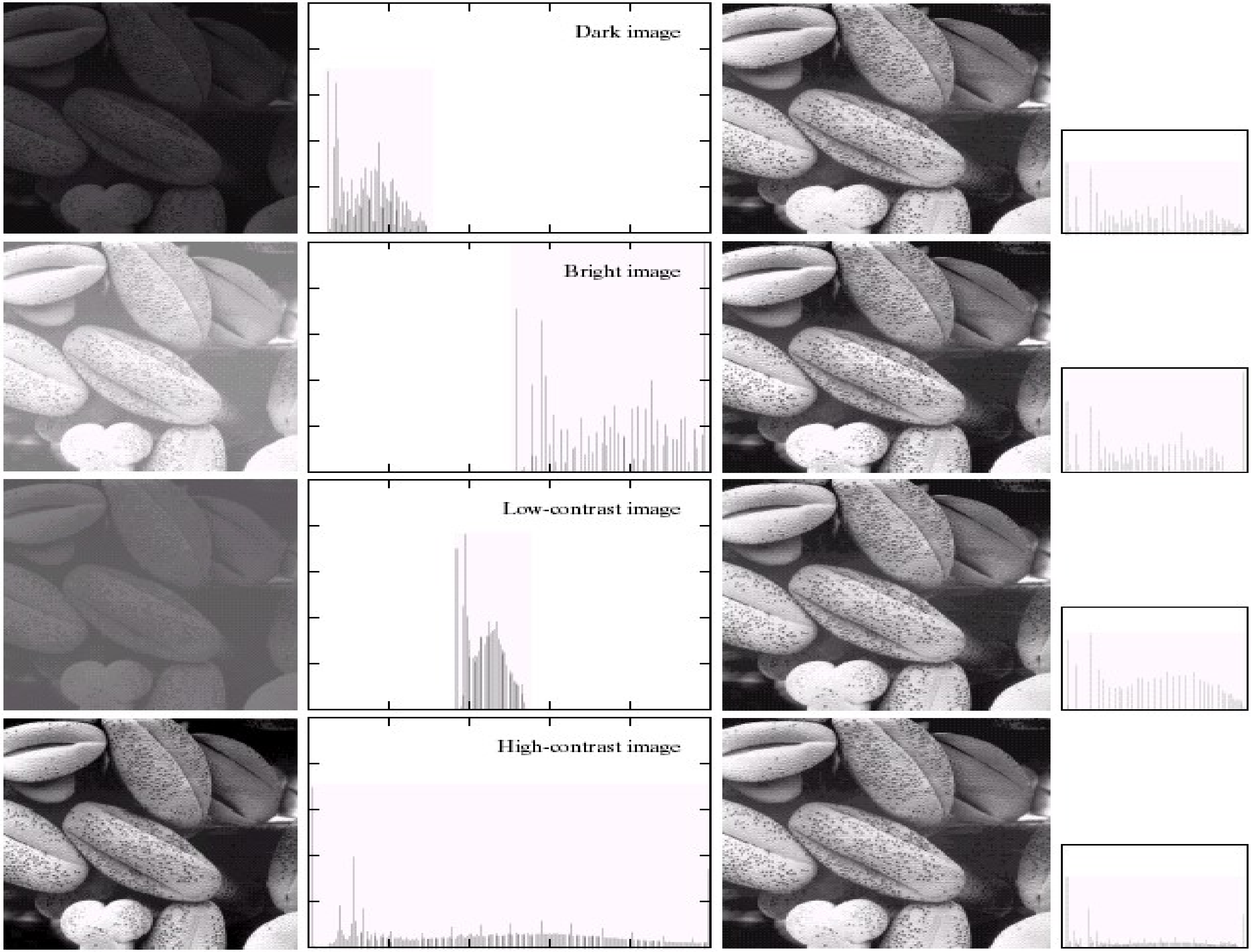
$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = p_r(r)\frac{1}{p_r(r)} = 1$$

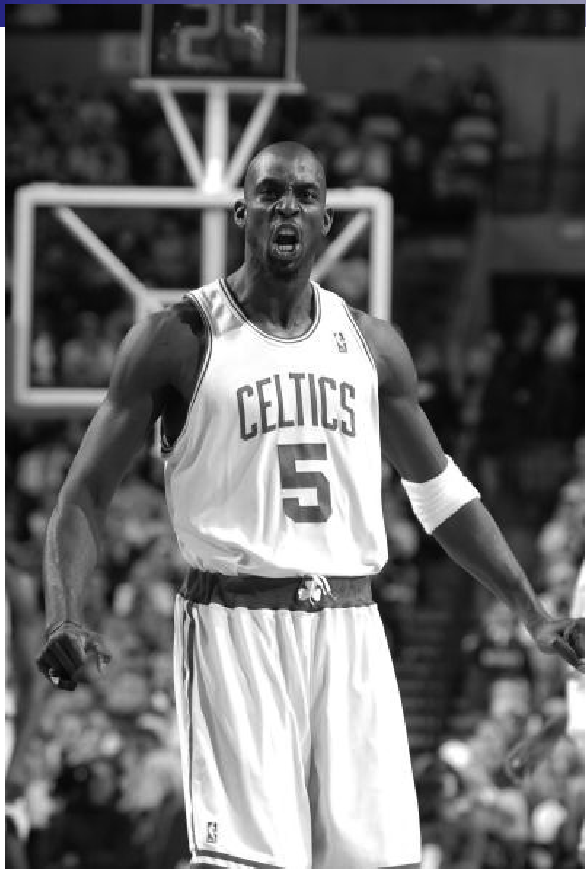
Histogram Equalization

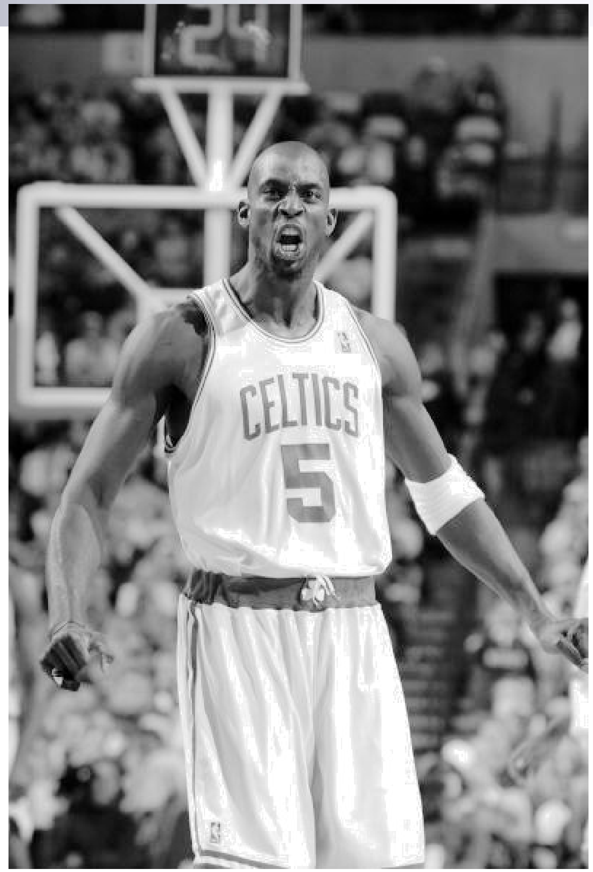
- **The transformation :**

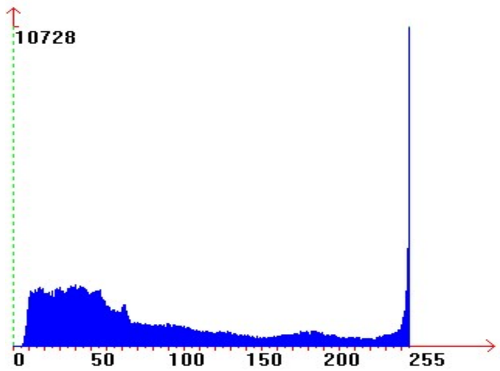
$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} = \sum_{j=0}^{k} p_r(r_j)$$

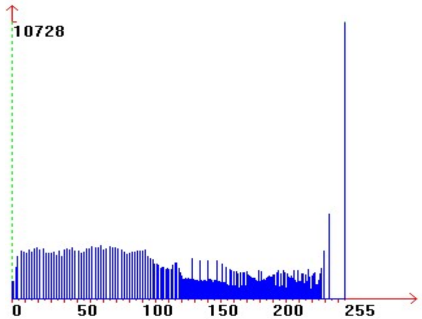
  - **This will not produce a uniform histogram, but will tend to spread the histogram of the input image**
  - **Advantages:**
    - **Gray-level values cover entire scale (contrast enhancement)**
    - **Fully automatic**

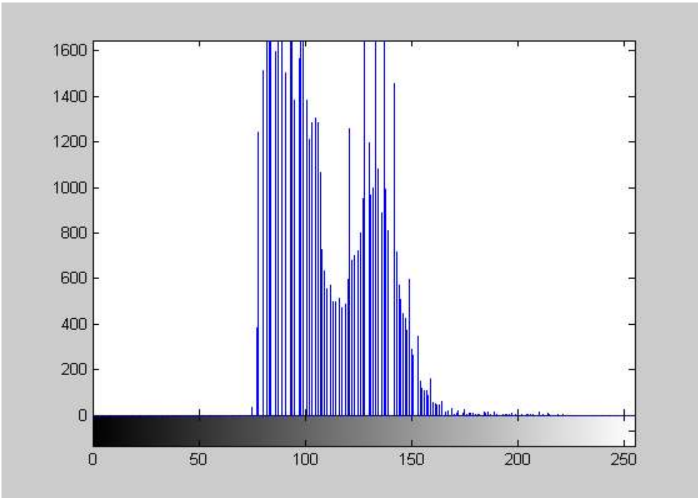

Dark image
Bright image
Low-contrast image
High-contrast image

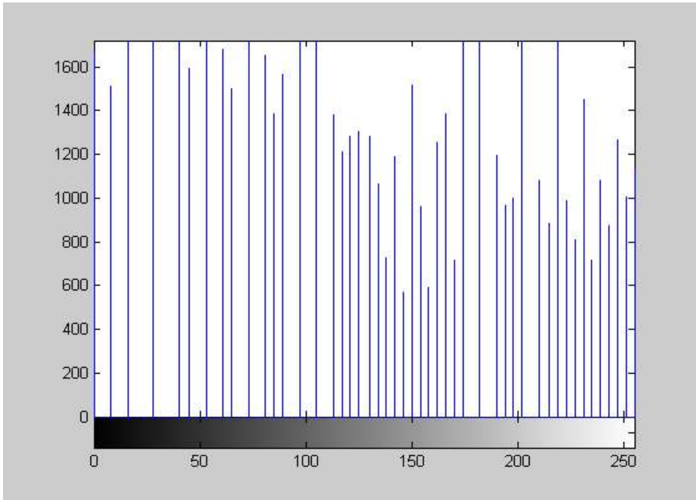

24
CELTICS
5

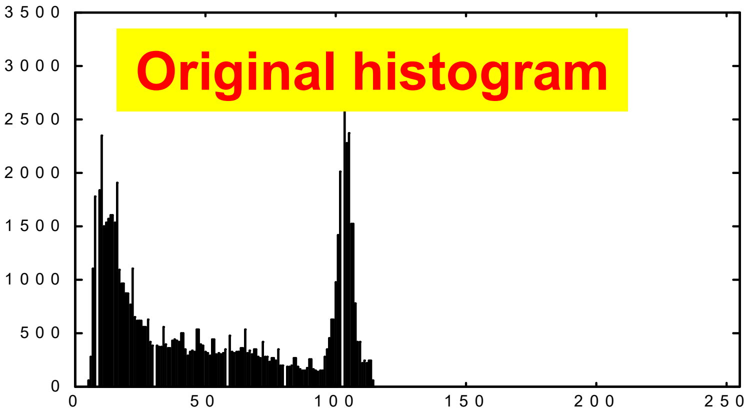

24
CELTICS
5

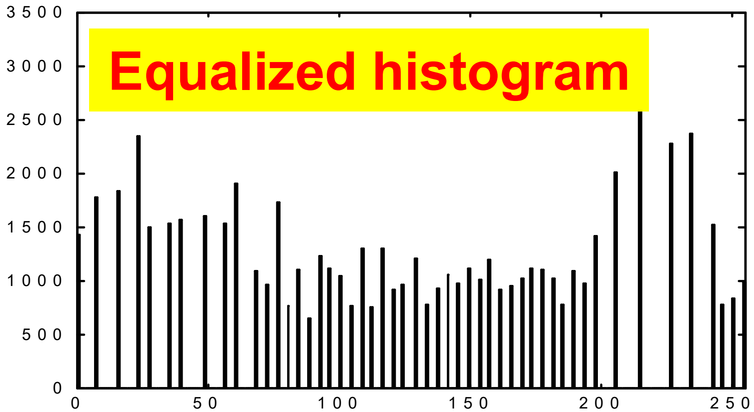

10728
0
50
100
150
200
255


10728
0
50
100
150
200
255


1600
1400
1200
1000
800
600
400
200
0
0
50
100
150
200
250


1600
1400
1200
1000
800
600
400
200
0
0
50
100
150
200
250


Original histogram
3500
3000
2500
2000
1500
1000
500
0
0
50
100
150
200
250


Equalized histogram
3500
3000
2500
2000
1500
1000
500
0
0
50
100
150
200
250

Histogram Processing

- Histogram Equalization
- **Histogram Matching**
- Local Enhancement
- Enhancement Using Histogram Statistics

Histogram Matching

Original

Equalized

Matched

Histogram Matching

- Special request for histogram of enhanced image
- → Histogram matching (or histogram specification)
  - Histogram equalization does not allow interactive image enhancement and generates only one result: an approximation to a uniform histogram.
  - Sometimes though, we need to be able to specify particular histogram shapes capable of highlighting certain gray-level ranges.

Procedure for Histogram Matching

- **Equalize the levels of the original image using:**

$$s = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n}$$

- **Specify the desired density function and obtain the transformation function G(z):**

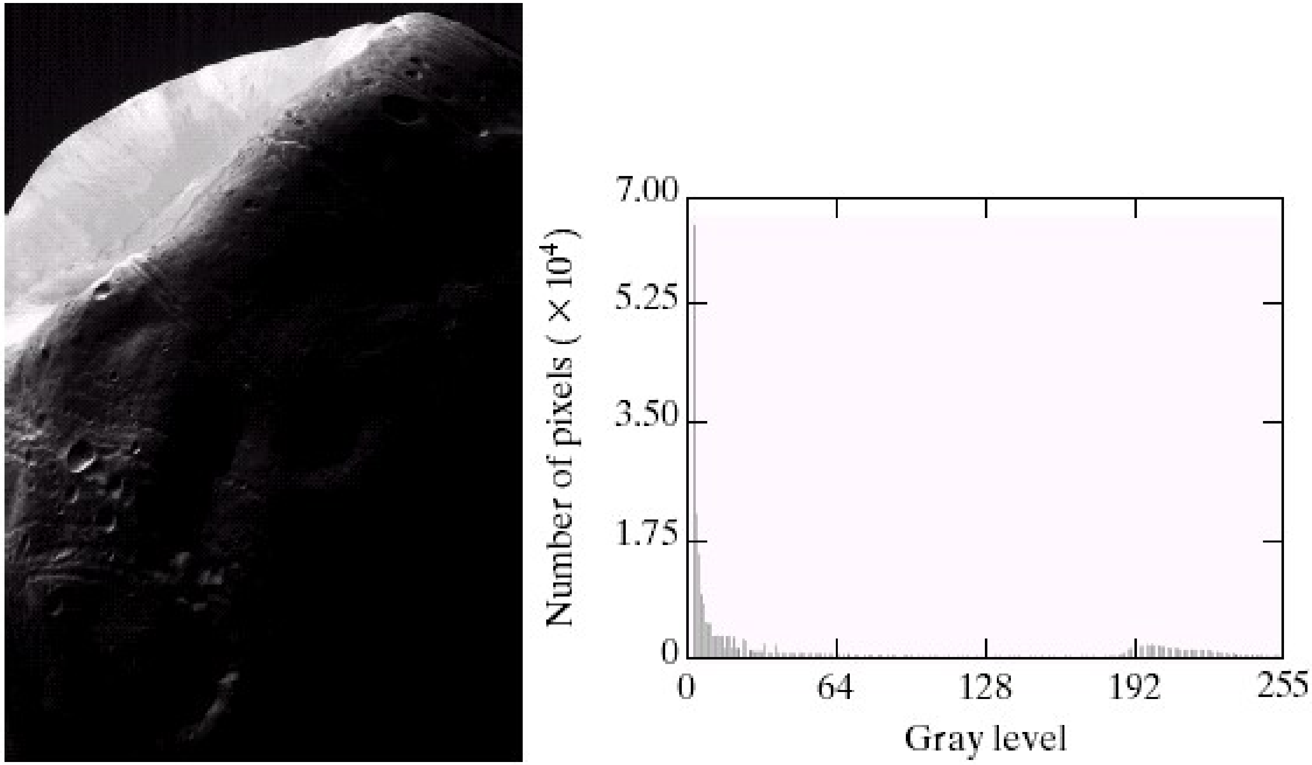
$$v = G(z) = \sum_{0}^{z} p_z(w) \approx \sum_{i=0}^{z} \frac{n_i}{n}$$

- **Apply the inverse transformation function $z = G^{-1}(s)$ to the levels obtained in step 1.**
- **The new, processed version of the original image consists of gray levels characterized by the specified density $p_z(z)$. In essence:**

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

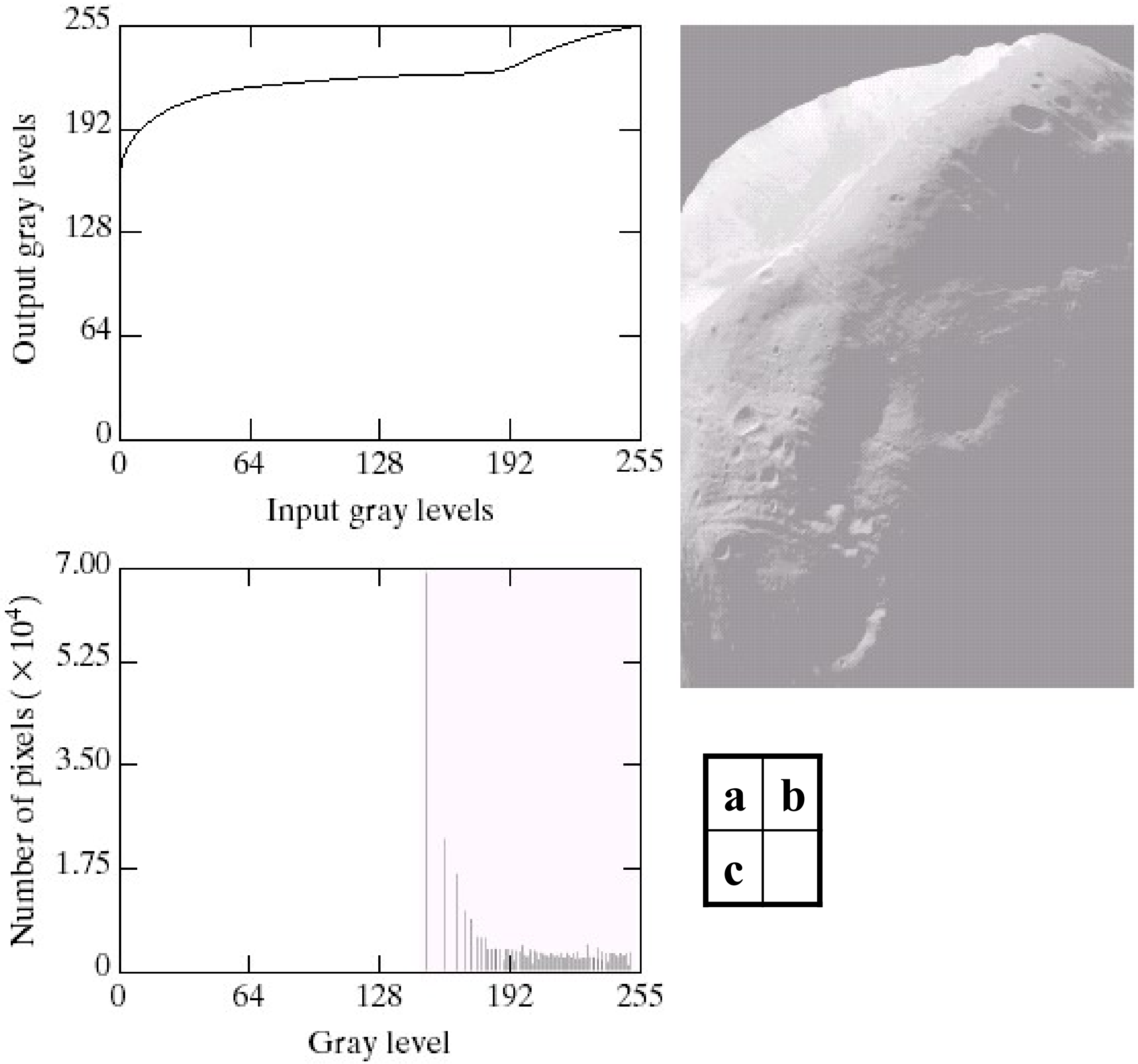
Histogram Matching

- The principal difficulty in applying the histogram specification method to image enhancement lies in being able to construct a meaningful histogram. So:
  - Either a particular probability density function (such as a Gaussian density) is specified and then a histogram is formed by digitizing the given function,
  - Or a histogram shape is specified on a graphic device and then is fed into the processor executing the histogram specification algorithm.

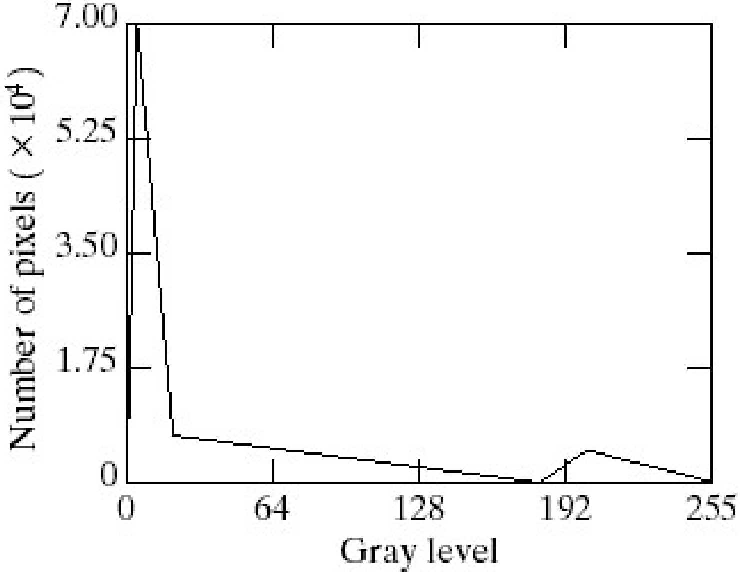


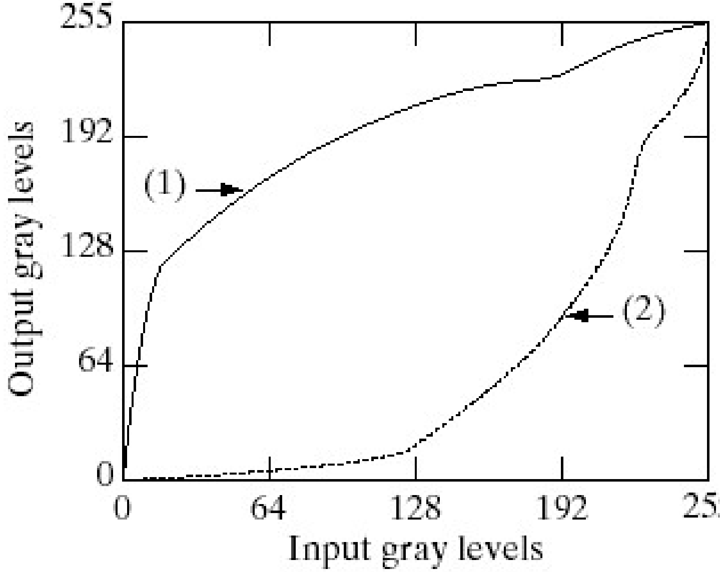
a | b

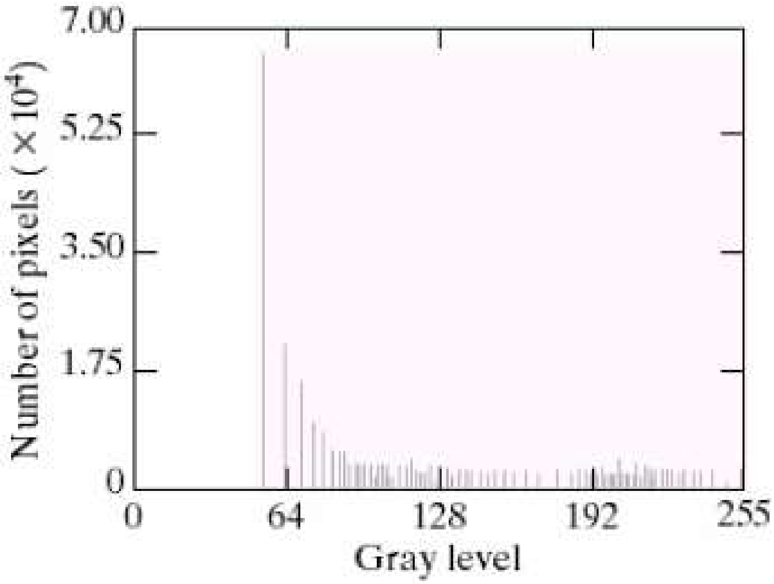
(a) Image of the Mars moon Photos taken by NASA’s Mars Global Surveyor. (b) Histogram. (Original image courtesy of NASA.)



(a) Transformation function for histogram equalization. (b) Histogram equalized image (note the was held out appearance). (c) Histogram of (b).







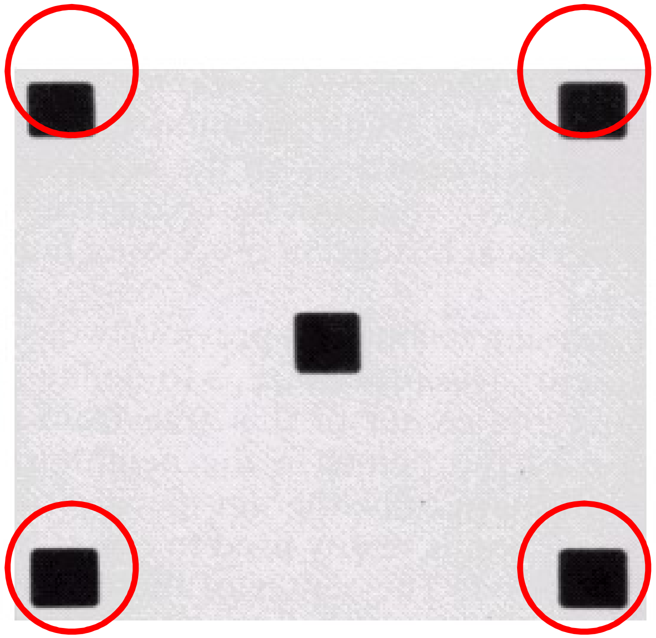
(a) Specified histogram. (b) Curve (1) is from Eq. (3.3-14), using the histogram in (a); curve (2) was obtained using the iterative procedure in Eq. (3.3-17). (c) Enhanced image using mappings from curve (2). (d) Histogram of (c).

a	b
c	d

Histogram Processing

- Histogram Equalization
- Histogram Matching
- **Local Enhancement**
- Enhancement Using Histogram Statistics

Local Enhancement



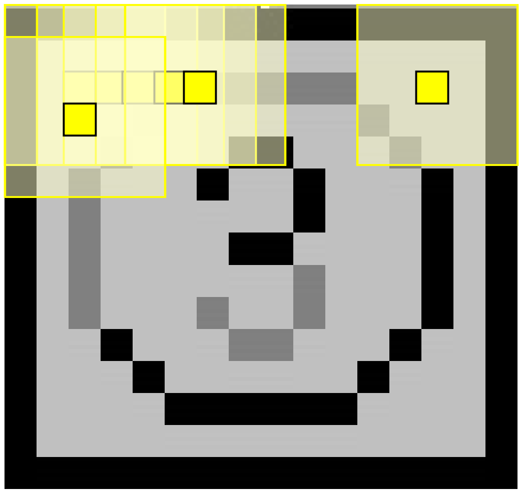
Special request for local image…

Local Enhancement

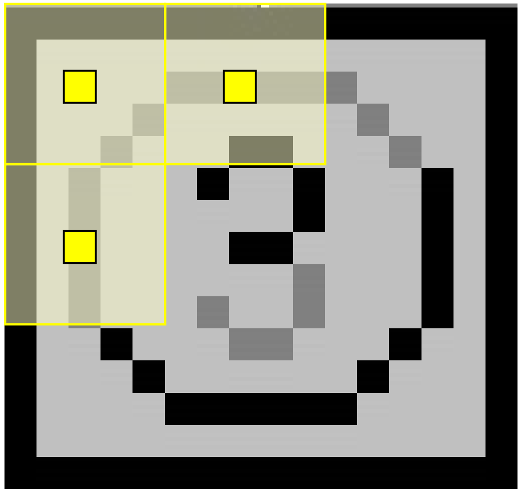
- Previous methods were global.
  - When it is necessary to enhance details over smaller areas (local)?
- To devise transformation functions based on the gray-level distribution in the neighborhood of every pixel

Local Enhancement

Pixel-to-pixel translation



Non-overlapping region



Reducing Computational Cost?

- **Additive construction of local histograms…**

Beyond Enhancement

- Histogram as image/region descriptor
  - Image retrieval （图像检索）
  - Object tracking （目标跟踪）
- Histogram as probability distribution function (pdf)
  - Segmentation/matting （分割/抠图）

Histogram of Color Images

- **By concatenation or using 3D table**

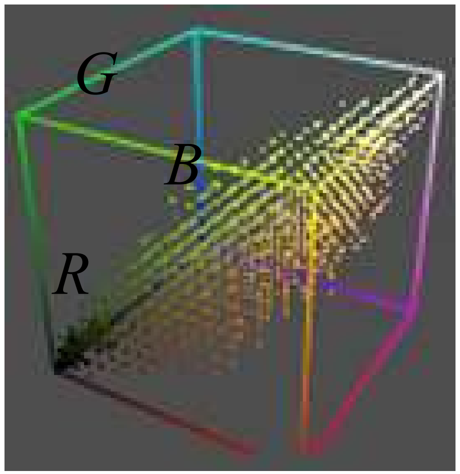
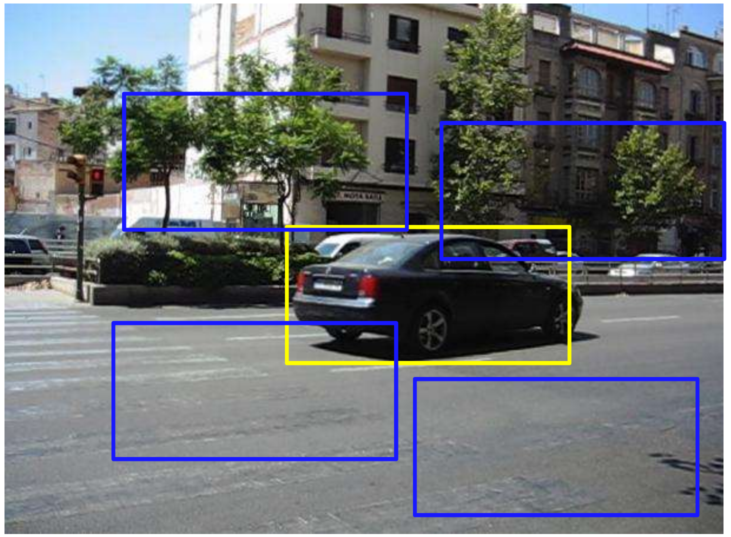


Image Retrieval by Histogram

- Using the histogram of each image as index

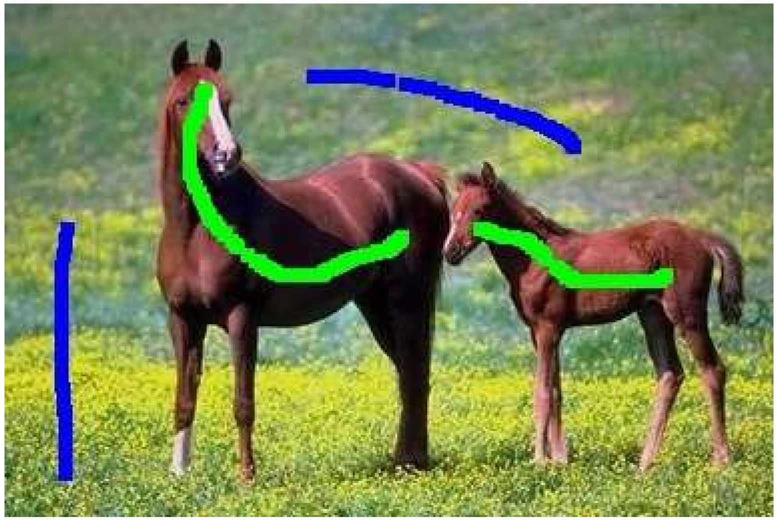
Object Tracking

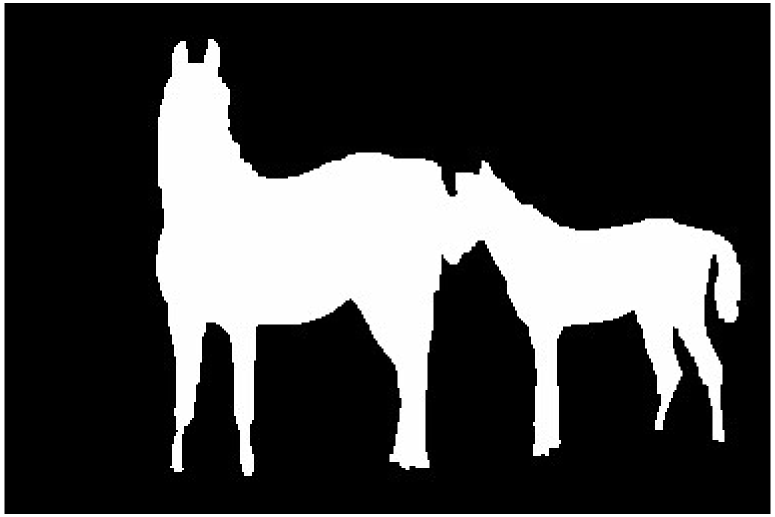
- Search a region whose histogram best matched with the previous frame…



Interactive Image Segmentation

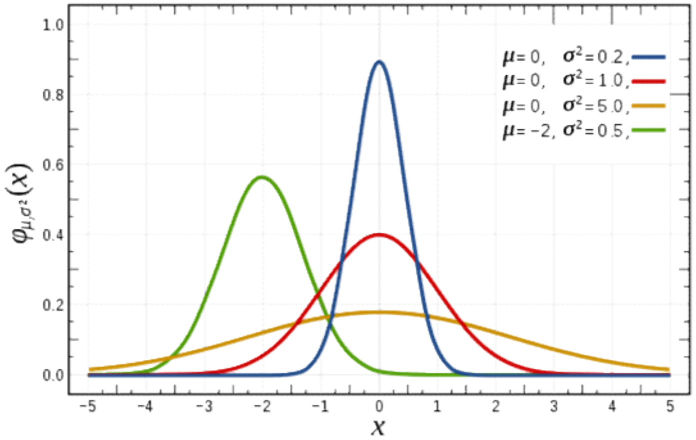
- **Histogram to model color distribution of foreground and background**

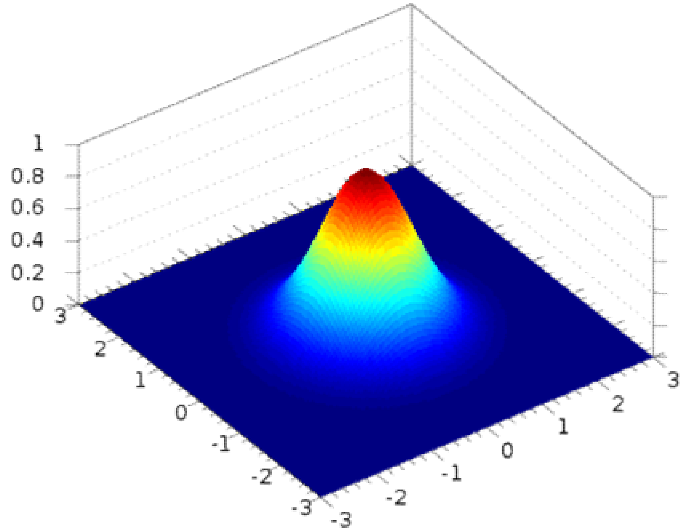




高斯与高斯混合模型

高斯模型





$$\frac{1}{\sigma_i\sqrt{2\pi}}\exp\left(-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right)$$

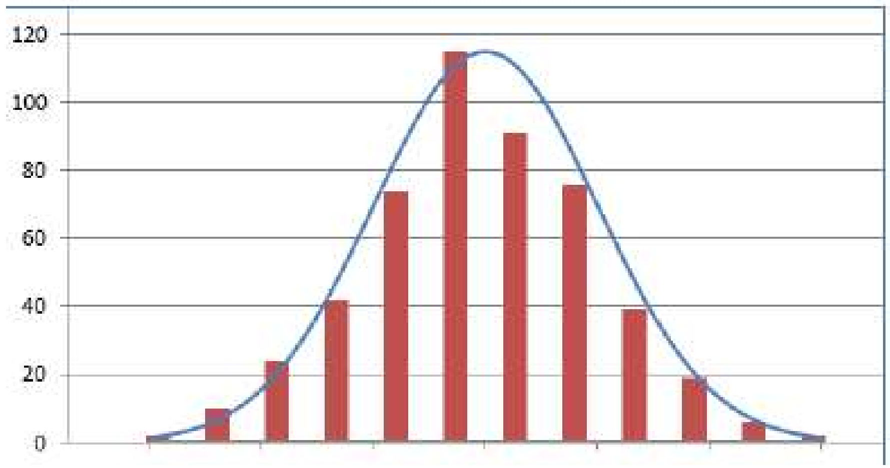
一维

$$\frac{1}{\sqrt{(2\pi)^K|\Sigma_i|}}\exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu}_i)^{\mathrm{T}}{\Sigma_i}^{-1}(\vec{x}-\vec{\mu}_i)\right)$$

多维

高斯与直方图

- 都可以用于像素亮度/颜色分布的描述



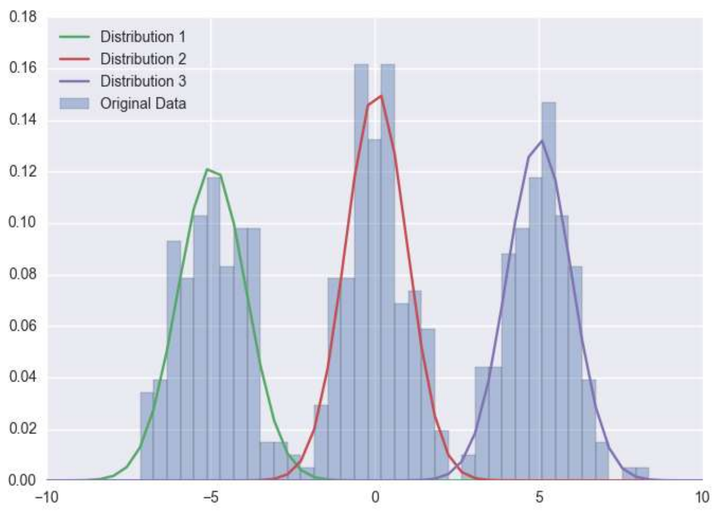
从直方图估计高斯模型？

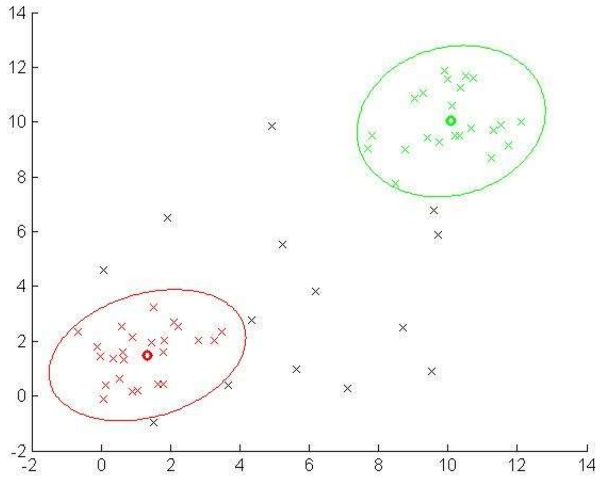
多维数据

拟合误差

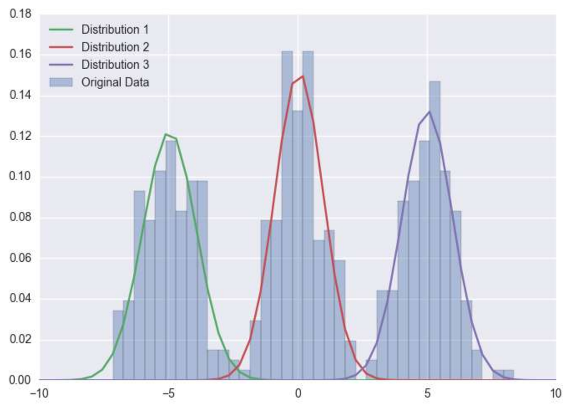
- 用高斯拟合直方图的误差取决于?

多峰分布


Distribution 1
Distribution 2
Distribution 3
Original Data
0.18
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−10
−5
0
5
10


14
12
10
8
6
4
2
0
-2
-2
0
2
4
6
8
10
12
14

高斯混合模型



- 用多个高斯拟合更复杂的数据分布

$$p(\vec{x}) = \sum_{i=1}^{K} \phi_i \mathcal{N}(\vec{x} \mid \vec{\mu}_i, \Sigma_i)$$

$$\mathcal{N}(\vec{x} \mid \vec{\mu}_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^K |\Sigma_i|}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu}_i)^{\mathrm{T}} {\Sigma_i}^{-1} (\vec{x} - \vec{\mu}_i)\right)$$

$$\sum_{i=1}^{K} \phi_i = 1$$

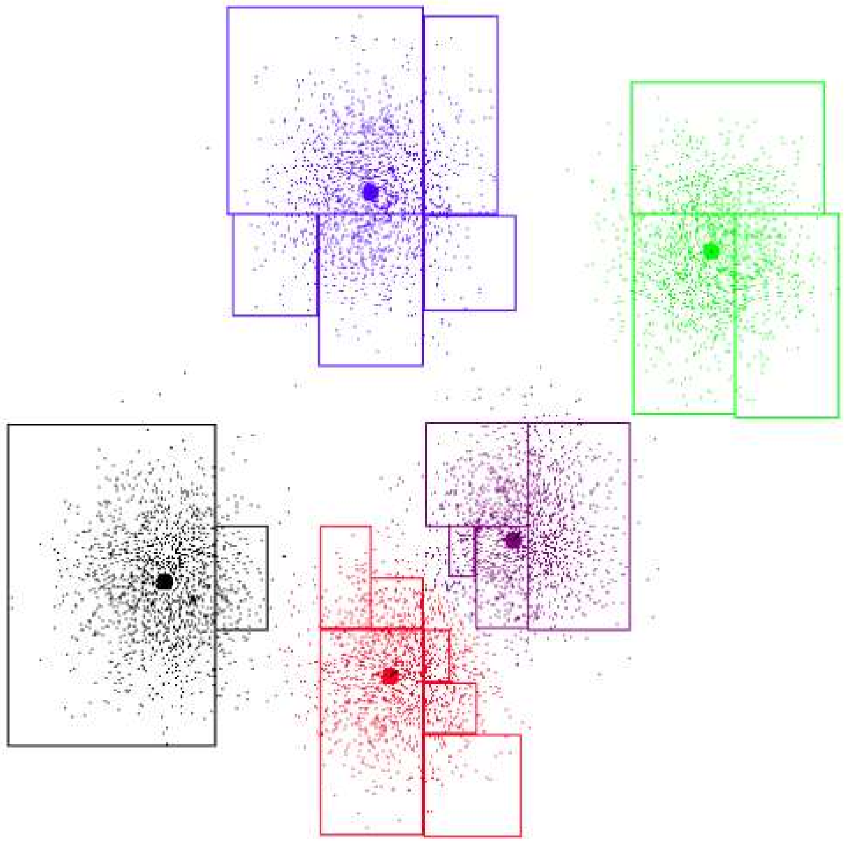
参数估计

- 已知K，如何计算K个高斯模型的参数，使得拟合误差最小？

soft k-means clustering

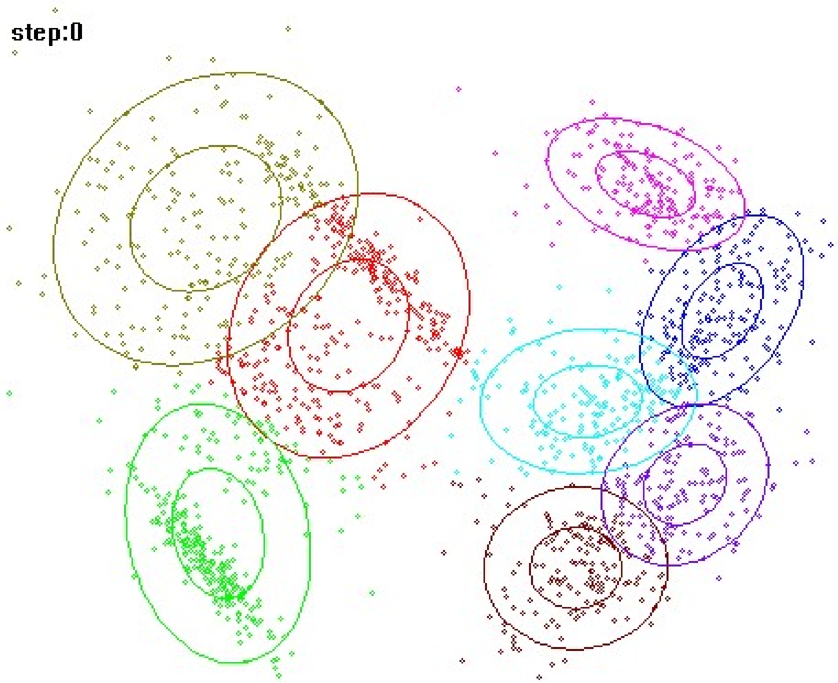
K-Means

1. 由用户指定类别个数k. (e.g. k=5)
2. 随机地选择k个数据点作为每一类中心（均值）的初始位置；
3. 将每个点分类到与其最近的中心所性的类；
4. 用每一类包含的数据的均值更新其中心
5. 转到第3步，迭代致中心不再变化



参数估计

- 假设第i个样本属于第j个高斯的概率（权重）为w_{ij}
- 迭代更新模型参数和w_{ij}，直至收敛



思考

- 直方图与高斯混合模型的优缺点？可以结合交互图像分割进行讨论。