数字图像处理

第二讲 空域处理（上）

王伟强

中国科学院大学计算机科学与技术学院

图像增强概述

- 增强的主要目的是对图像进行处理，使结果图像比原图像更适合于某一特定应用。 特定意味着技术是非常面向问题的。

- 图像增强方法可分为两大类:空域方法和频域方法。
  - 空域方法是基于图像中的像素的直接进行局部处理。
  - 频域技术是基于图像的傅里叶变换进行处理。

- 目前还没有图像增强的一般理论。
  - 当一幅图像增强效果通过人的视觉进行评估时，观察者是一个特定方法工作好坏的最终判断者，这使得比较不同方法的性能变得困难。
  - 若采用机器感知评判时，评价任务相对简单一些，如字符识别任务。

内容纲要

- 背景
- 灰度变换函数
- 图像直方图处理

背景

➢ 空间域处理的数学表示:

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

其中 $f(x,y)$: 输入图像

$g(x,y)$: 输出图像

T: 在$f(x,y)$上的操作，定义在(x,y)的某个邻域上

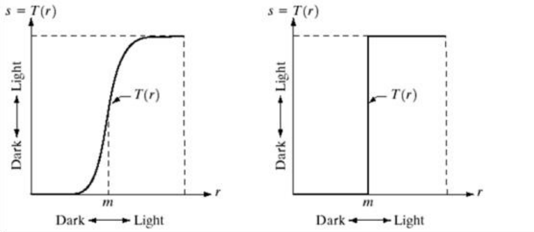
- T 可以在一组图像上进行操作，例如噪声消除
- 方形和矩形邻域是使用最多的，因为它们易实现，也可使用圆形区域。

➢ T最简单的情况是邻域大小为1×1。这时 g只依赖于 f在 (x,y)处的值，即

$$s = T(r),$$

我们称之为灰度变换或亮度变换，其中s与r分别为输出与输入的亮度值。

灰度变换


$s = T(r)$
Light
Dark
$T(r)$
r
m
Dark Light
$s = T(r)$
Light
Dark
$T(r)$
r
m
Dark Light

幂律变换

- $s = cr^{\gamma}$ 其中 c, γ 是正常数

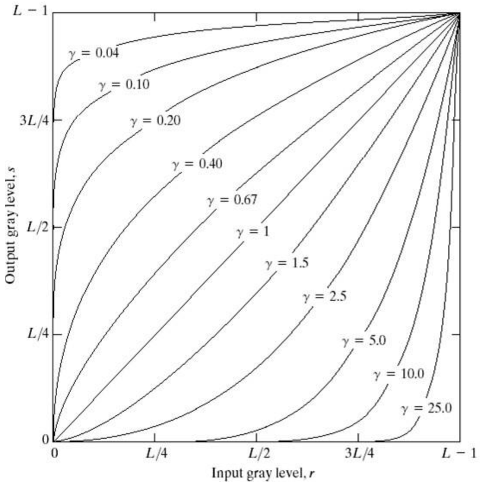
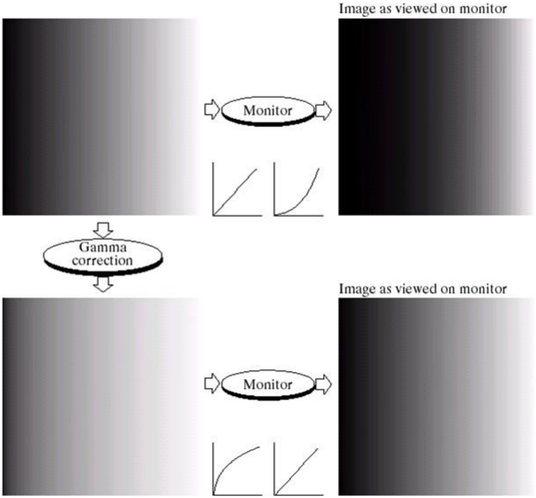


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

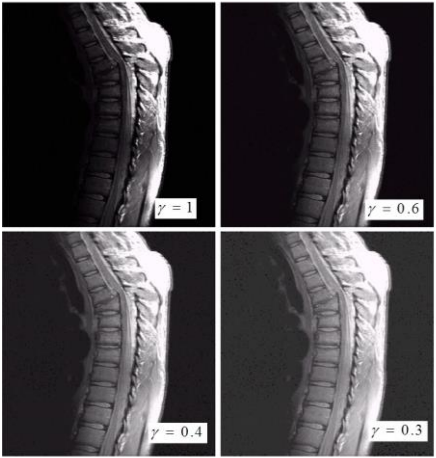
幂律变换：应用例子1 伽马校正



a b
c d

FIGURE 3.7
(a) Linear-wedge gray-scale image.
(b) Response of monitor to linear wedge.
(c) Gamma-corrected wedge.
(d) Output of monitor.

幂律变换：应用例子2 对比度增强



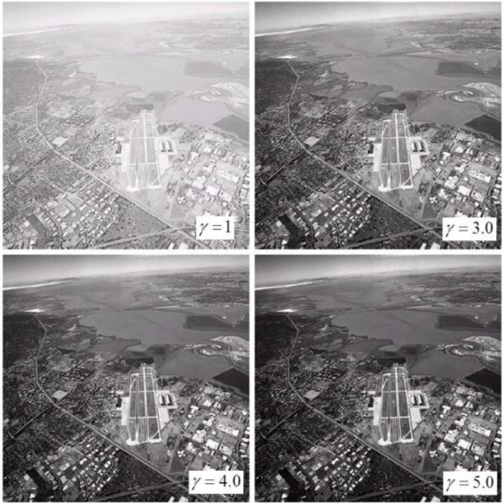
a b
c d

FIGURE 3.8 (a) Magnetic resonance (MR) image of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

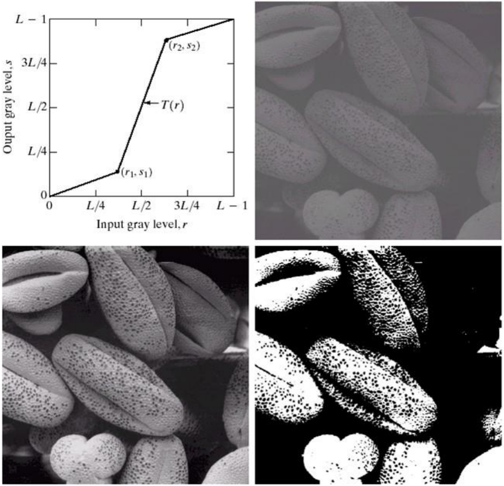
幂律变换：应用例子3 对比度增强

a b
c d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0$, and 5.0, respectively. (Original image for this example courtesy of NASA.)



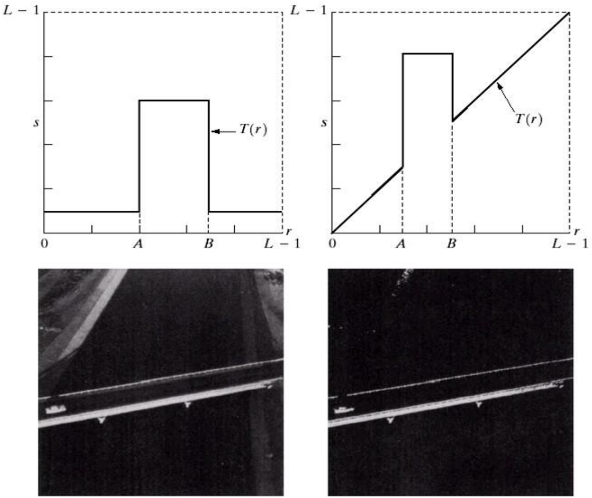
幂律变换：应用例子4 对比度增强



a b
c d

FIGURE 3.10 Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (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.)

幂律变换：应用例子5 灰度级分层



a b
c d

FIGURE 3.11 (a) This transformation highlights range [A, B] of gray levels and reduces all others to a constant level. (b) This transformation highlights range [A, B] but preserves all other levels. (c) An image. (d) Result of using the transformation in (a).

幂律变换：应用例子6 比特位面图像（1）

- 比特位面图像
  - 它可以突出特定位对图像总外观的贡献。
  - 图像中的每个像素由8位二进制数表示。
  - 图像由8个1位平面组成，最小有效位平面为0位平面，最大有效位平面为7位平面。

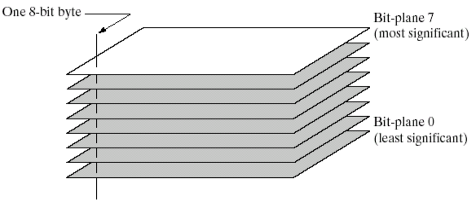


FIGURE 3.12 Bit-plane representation of an 8-bit image.

幂律变换：应用例子6 比特位面图像（2）

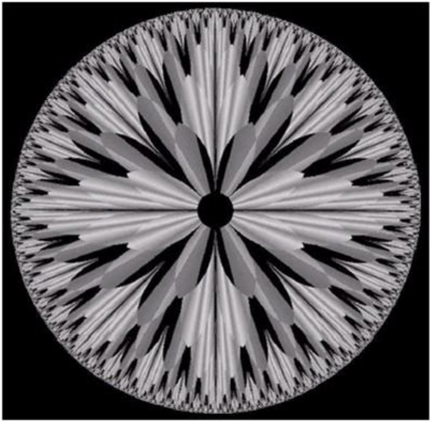


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

幂律变换：应用例子6 比特位面图像（3）

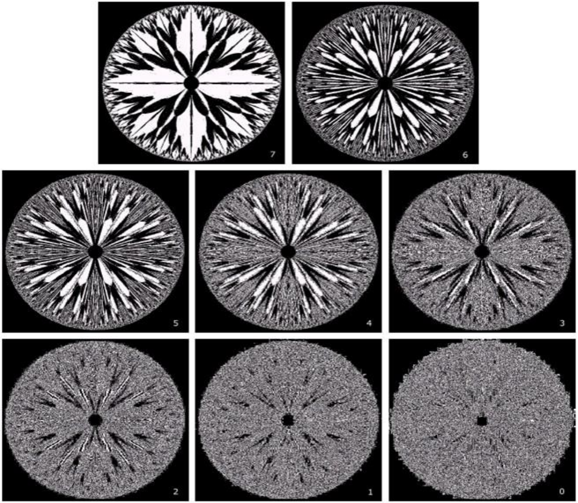
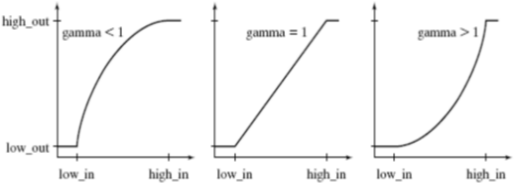


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

图像处理工具箱(IPT)中的函数imadjust

- g = imadjust(f, [low_in, high_in], [low_out, high_out], gamma)
  - ✓这个函数将图像f中的强度值映射到g中的新强度值.
  - ✓使low_in和high_in之间的值映射到 low out和high out之间的值。
  - ✓比low_in还低的和比high_in还高的将会被裁减;就是说比low_in还低的值对应到low_out，比high_in还高的值映射到high_out.

IPT中的函数 imadjust



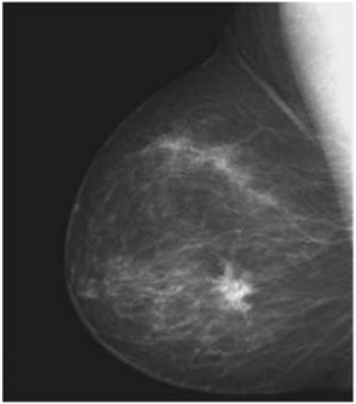
a b c

FIGURE 3.2 The various mappings available in function `imadjust`.

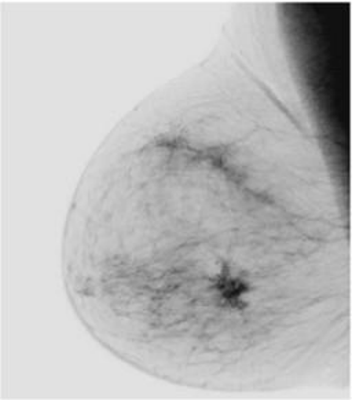
- ✓ 参数gamma指定了将f中强度值映射成g中强度值的映射曲线的形状
- ✓ 如果gamma小于1，该映射倾向于产生更高的亮度输出值，如左侧图所示。
- ✓ 如果gamma大于1，该映射倾向于产生更低的的亮度输出值，如右侧侧图所示。
- ✓ 如果函数参数中省略了它，默认为gamma=1的线性映射

应用举例

```
I=imread('Fig0303(a)(breast).tif');
imshow(I)
```

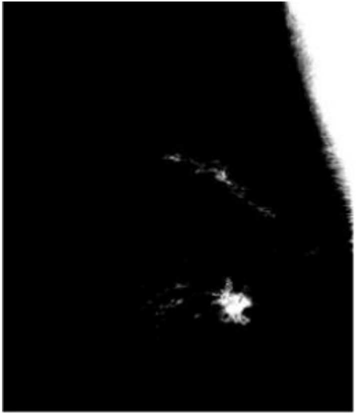


```
G=imadjust(I,[0 1],[1 0]);
imshow(G)
```

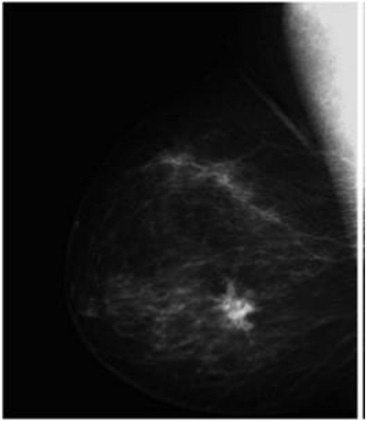


应用举例

G=imadjust(I,[0.5 0.75],[0 1]);
imshow(G)



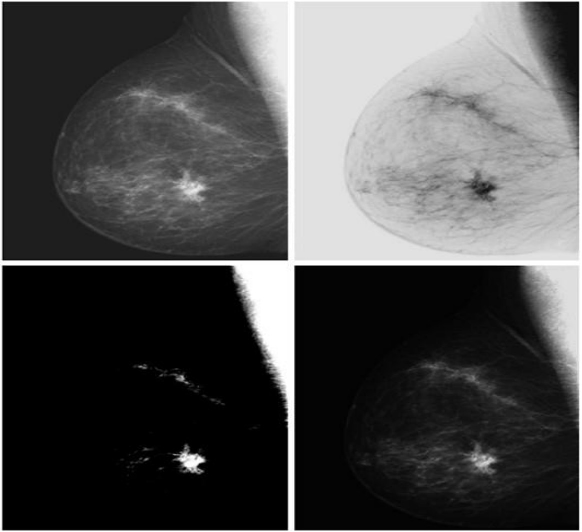
G=imadjust(I,[],[],2);
imshow(G)



Imadjust函数实验结果图

a b
c d

FIGURE 3.3 (a) Original digital mammogram. (b) Negative image. (c) Result of expanding the intensity range [0.5, 0.75]. (d) Result of enhancing the image with gamma = 2. (Original image courtesy of G. E. Medical Systems.)



对数变换（1）

- 对数变换和对比拉伸变换是动态范围处理的基本工具。
- 对数变换是用这个数学表达式实现的：

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

  - 其中c是常数.
  - 该变换的形状类似于gamma曲线。
  - 请注意，伽马曲线的形状是**可变的**，而对数函数的形状是**固定的**。
  - 另外，**为什么要加1呢？**

对数变换（2）

- 对数变换的主要用途之一是压缩动态范围

  - 例如，傅里叶谱(第4章)的值在$[0, 10^6]$或更高的范围内并不少见。当显示在一个线性缩放到8位的显示器上时，高值占据了显示器的主导地位，从而丢失了低强度值的视觉细节频谱。通过计算log，动态范围例如10^6减少到大约14，这更易于显示管理。

  - 执行对数转换时，通常希望将压缩后的值恢复到显示的全部范围。对于8位，在MATLAB中最简单的方法是使用这个语句

```
gs= im2uint8(mat2gray(g));
```

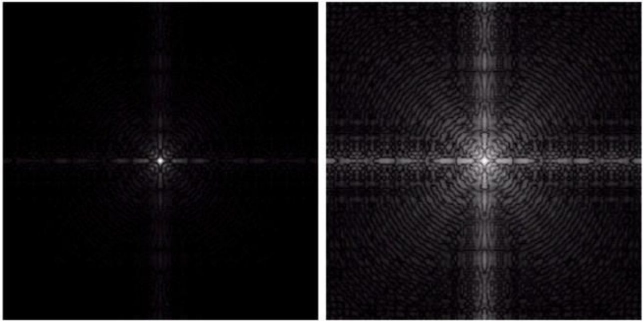
使用**mat2gray**将值映射到[0,1]范围，而**im2uint8**将值映射到[0,255]范围。

对数变换（3）

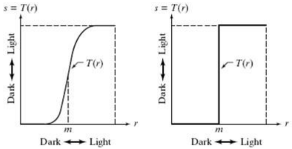
```
>> g = im2uint8(mat2gray(log(1 + double(f))));
>> imshow(g)
```

a b

FIGURE 3.5 (a) Fourier spectrum. (b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.



对比拉伸变换



a b
FIGURE 3.4 (a) Contrast-stretching transformation. (b) Thresholding transformation.

- 一个对比拉伸变换函数可以定义为

$$s = T(r) = \frac{1}{1 + (m/r)^E}$$

- 它将小于m的输入值压缩到输出图像中较窄的暗区范围内；
- 类似地，它将m以上的值压缩成输出中的窄带亮区；
- 其中r为输入图像的强度，s为输出图像中对应的强度值，E控制函数的斜率，m为参数。
- 在MATLAB中，可以用下面这个表达式处理整幅图像

g= 1./(1 + (m./(double(f) + eps)).^E)

- E控制函数的斜率？探索：图像是对称的？E的取值可以是任意的吗？

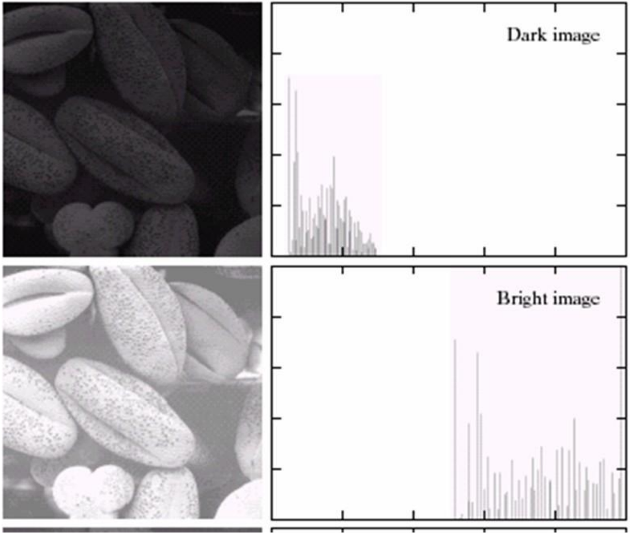
直方图处理

- 亮度变换通常是基于从图像亮度直方图中提取的信息进行的，直方图在很多图像处理方面（图像增强、压缩、分割和描述）都起到基础的作用。
- 一幅亮度取值在[0,G]范围的数字图像，将它量化为L个亮度级别的直方图可以定义为一个离散函数

$$h(r_k) = n_k$$

其中r_k为区间[0,G]中的第k个强度级别类，n_k为图像中强度级别为r_k的像素数目。uint8类图像的G值为255,uint16类图像的G值为65535,double类图像的G值为1.0。

生成绘制图像直方图



图像的直方图

- 一幅图像的直方图$h(r_k)$可以表示为一个向量，将其所有分量除以该图像的像素总数，可得到对应的归一化直方图，即

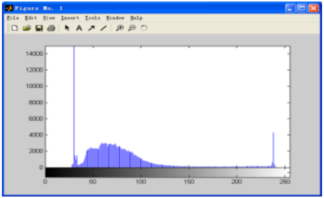
$$p(r_k)=\frac{h(r_k)}{N}=\frac{n_k}{N},N=\sum_{k=1}^{L}n_k,k=1,2,...,L$$

从基本概率论知识可知，$p(r_k)$可看作为在一幅图像中强度级r_k发生概率的估计值。

Matlab中图像直方图函数

- h=imhist(f,b)
  - 其中f为输入图像，h为输入图像的直方图，b为生成直方图所用量化级别的数目。
  - 如参数中不包含b，则默认使用b = 256。
  - 量化级别是对亮度范围的一种子划分。
- 例子

```
f=imread('Fig3 8 a.tif');
imhist(f)
```



- 计算归一化直方图:

```
p=imhist(f,b)/numel(f)
```

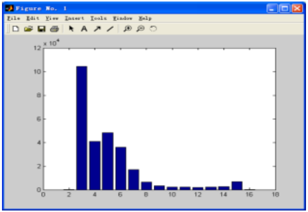
用bar绘制图像直方图

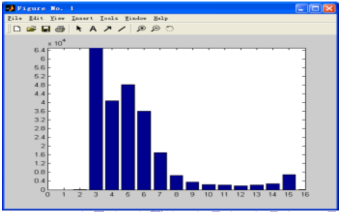
➢ 我们通常用柱状图绘制直方图，可使用matlab函数bar(horz, v, width)实现。

- 其中v是一个行向量，包含要绘制的点的纵坐标。
- horz是一个维数与v相同的向量，它包含水平尺度的增量。如省略了horz，则水平轴的单位是从0到长度v的长度。
- 宽度是介于0和1之间的一个数。

```
s=imread('Fig0303(a)(breast).tif');
h1=imhist(s,16);
horz=1:16;
bar(horz,h1,0.8)
```

```
axis([0 16 0 65000]);
set(gca,'xtick',0:1:16);
set(gca,'ytick',0:4000:65000);
```





用stem绘制图像直方图

- 函数stem类似于bar。语法是

  stem(horz, v, 'color_linestyle_marker', 'fill')

  - 其中v，horz与bar中的参数描述相同。
  - Color_linestyle_marker是下表中值的三元组。

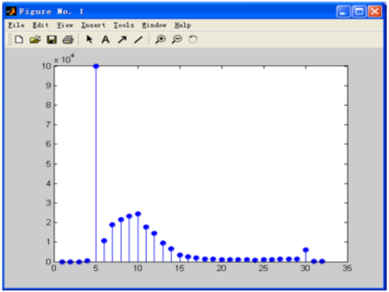
Symbol	Color	Symbol	Line Style	Symbol	Marker
k	Black	-	Solid	+	Plus sign
w	White	--	Dashed	o	Circle
r	Red	:	Dotted	*	Asterisk
g	Green	-.	Dash-dot	.	Point
b	Blue	none	No line	x	Cross
c	Cyan			s	Square
y	Yellow			d	Diamond
m	Magenta			none	No marker

TABLE 3.1 Attributes for functions stem and plot. The none attribute is applicable only to function plot, and must be specified individually. See the syntax for function plot below.

使用stem绘制直方图例子

- 例如
  stem(v, ’rs’)
  - 生成一个线段图，其中线和标记是红色的，线是虚线，标记是正方形。
  - 如果指定 ‘fill’ ，且使用的标记是圆形、正方形或菱形，则标记用指定的颜色填充。
  - 默认的颜色为黑色，默认的线型为实线，默认的标记为圆形。

```
s=imread(’Fig0303(a)(breast).tif’);
hi=imhist(s,32);
stem(hi,’b-o’,’fill’);
```



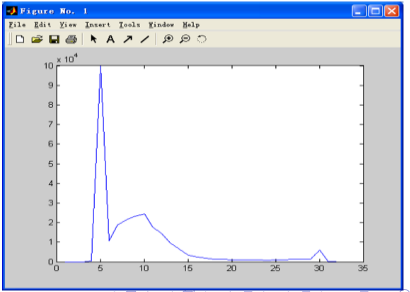
用plot绘制图像直方图

➢ 函数plot是用直线连接一组点。语法是

plot(horz, v, 'color_ linestyle_ marker')

- 其中参数如前面stem函数的定义相同。.
- 函数plot是常用来绘制函数图像，而且可以将多组函数绘制在一个坐标系下。

```
s=imread('Fig0303(a)(breast).tif');
hi=imhist(s,32);
plot(hi);
```



直方图均衡（1）

➤ 直方图均衡:

- 目的：改善图像的对比度。
- 结果：均衡后的的图像具有几乎均匀的像素值分布。
- 优点：无参数的算法

➤ 寻找变换函数:

- 假设r被归一化为区间[0,1]，$r=0$表示黑色，$r=1$表示白色

$$s=T(r)\,,\ 0\leqslant r\leqslant 1$$

- 转换函数T要求满足下列条件:

① $T(r)$在区间 [0,1]上是单值的且单调递增的

② $0\leqslant T(r)\leqslant 1$，对于$0\leqslant r\leqslant 1$

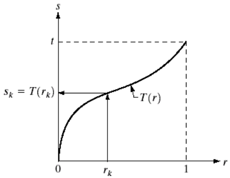


FIGURE 3.16 A gray-level transformation function that is both single valued and monotonically increasing.

直方图均衡（2）

- 直方图均衡化是基于一个随机变量的概率密度函数的变换。
- 令$p_s(s)$和$p_r(r)$分别为随机变量s和r的概率密度函数。
- 如果 $p_r(r)$和$T(r)$是已知，则可得到变换后的变量s的概率密度函数$p_s(s)$ 通过下面的计算

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

- 我们定义一个转换函数

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

其中w是积分的虚变量，方程右边可看作随机变量w的**累积分布函数**。

直方图均衡（3）

➢ 针对给定的变换函数$T(r)$，我们有

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

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

$$p_s(s) = p_r(r)|\frac{dr}{ds}| = p_r(r)|\frac{1}{p_r(r)}| = 1, 0 \le s \le 1$$

- 可知在给定的映射变换下，输出随机变量s服从均匀概率分布。
- $T(r)$虽然依赖于$p_r(r)$，但是得到的$p_s(s)$总是均匀的.

直方图均衡（4）

➢ 对于离散的情况， 图像中亮度级 r_k 出现的概率为

$$p_r(r_k) = \frac{n_k}{n} \quad , k = 0, 1, 2, \ldots, L - 1$$

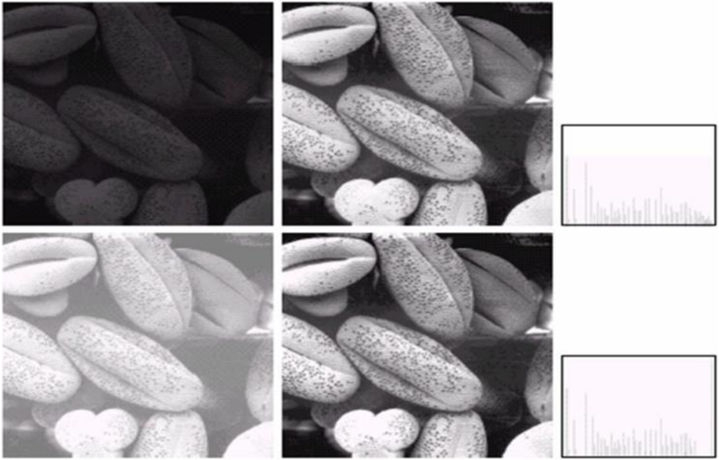
其中n_k为输入图像中r_k级的像素数目， n为从图像中总的像素数目。

➢ 此时，变换函数为

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}, k = 0, 1, 2, \ldots, L - 1$$

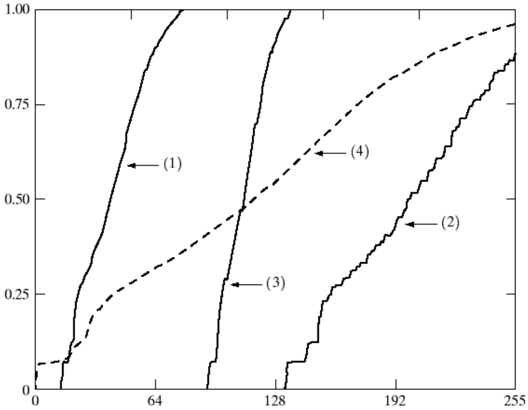
利用该映射，可将将输入图像中r_k级的每个像素映射到s_k级对应的像素，从而得到均衡后输出图像。

直方图均衡（5）



直方图均衡（6）

FIGURE 3.18 Transformation functions (1) through (4) were obtained from the histograms of the images in Fig.3.17(a), using Eq. (3.3-8).



直方图均衡（7）

- 直方图均衡化是通过函数histeq在工具箱中实现的，函数的语法如下

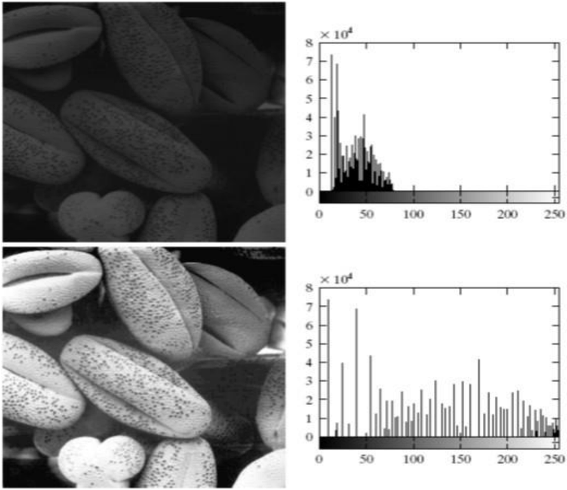
$$g = \text{histeq}(f, nlev)$$

  - 其中 f 为输入图像， g 为均衡后的输出图像，nlev为输出图像指定的亮度级别数。
  - 如果nlev等于L(输入图像中亮度级总数)，那么histeq直接实现转换函数。如果nlev小于L，那么histeq将尝试分布这些级别使它们近似于一个平坦的直方图。

Matlab中的直方图均衡演示代码

```
>> f = imread('Fig0308(a)(pollen).tif');
>> imshow(f);
>> figure,imhist(f);
>> ylim('auto');
>> g = histeq(f,256);
>> figure, imshow(g);
>> figure, imhist(g);
>> ylim('auto');
```

Matlab中的直方图均衡的效果

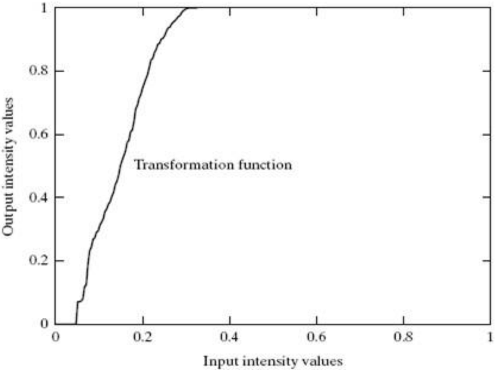


a b
c d

FIGURE 3.8 Illustration of histogram equalization. (a) Input image, and (b) its histogram. (c) Histogram-equalized image, and (d) its histogram. The improvement between (a) and (c) is quite visible. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra.)

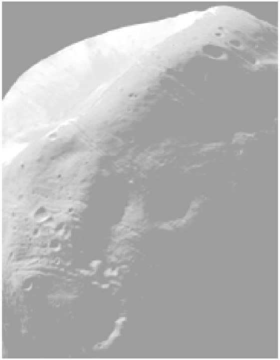
直方图均衡中的采用的映射函数曲线

FIGURE 3.9 Transformation function used to map the intensity values from the input image in Fig. 3.8(a) to the values of the output image in Fig. 3.8(c).



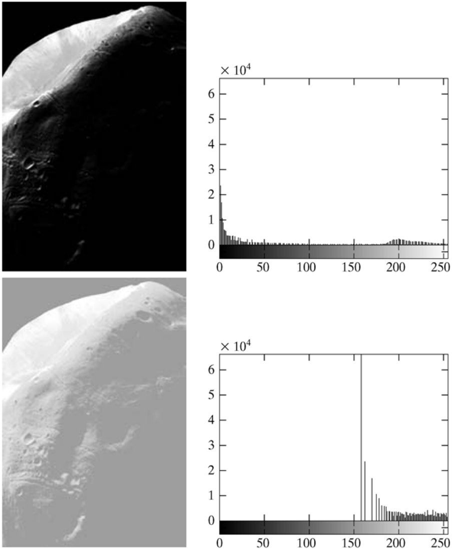
直方图均衡的一个失败例子

```
f=imread('Fig0310(a)(Moon Phobos).tif');
f1=histeq(f,256);
imshow(f1);
```



- 结果表明，直方图均衡化在这种情况下并没有产生特别好的效果。
- 通过研究均衡后图像的直方图可以看出原因。

直方图均衡还不够完美！



a b
c d

FIGURE 3.10 (a) Image of the Mars moon Phobos. (b) Histogram. (c) Histogram-equalized image. (d) Histogram of (c). (Original image courtesy of NASA).

直方图匹配（1）

- 直方图匹配类似于直方图均衡化，只不过我们不希望输出图像具有扁平的直方图，而是希望输出图像的直方图具有指定的形状。
- 考虑一下标准化到区间[0, 1]的连续亮度级，令 r 和 z 分别表示输入和输出图像的亮度级。输入图像的亮度分布服从概率密度函数 $p_r(r)$ ，输入图像的亮度分布服从指定的概率密度函数 $p_z(z)$ 。

直方图匹配（2）

- 在前面的讨论中，我们知道通过如下变换:

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

可以得到理想的均衡化直方图 $p_s(s)$。

- 假设现在我们定义了一个变量z具有如下属性

$$H(z) = \int_0^z p_z(w)dw = s$$

- 由上面两个等式，我们可以推出

$$z = H^{-1}(s) = H^{-1}(T(r))$$

- 我们可以从输入图像找到 $T(r)$ (通过前面讨论的直方图均衡化计算), 所以只要我们能找到H^{-1}, 就可以利用红色标记的等式计算出变换后的亮度值z，它具有我们指定的的概率密度函数 $p_z(z)$。

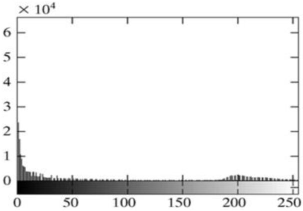
直方图匹配（3）

- 工具箱使用histeq中的以下语法实现直方图匹配:

$$g = histeq(f, hspec)$$

  - 其中f为输入图像，hspec为指定直方图(指定值的行向量)，g为输出图像，其直方图近似于指定直方图hspec。
  - 这个向量应该包含与等距容器相对应的整数计数。histeq的一个特性是，当长度(hspec)远小于f中强度级别的数量时，g的直方图通常更符合hspec。

利用直方图匹配解决前面的图像（1）



- 本例中直方图均衡化的问题主要由于原始图像中像素的高度集中在0附近，因此一种合理的方法是修改该图像的直方图，使其不具有此属性。
- 解决这种情况的一种可能方法就是使用直方图匹配。我们希望输出图像保持原始图像的直方图的一般形状，同时将源图像在灰度级低端的密集情况进行改善，使其变得分布更广泛、分散。
- 我们注意到原始图像的直方图基本上是双峰的，一个大的模式在原点，另一个小的模式在灰度的高端。例如，可以使用多模态高斯函数对目标直方图进行建模。

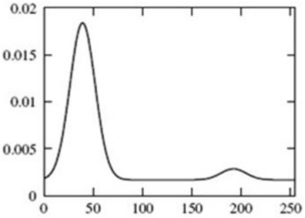
利用直方图匹配解决前面的图像（2）

- 我们用下面的函数twomodegauss计算一个归一化到单位面积的双峰高斯函数，用它可以作为一个指定的直方图。

$$p(x) = k + \frac{A_1}{\sqrt{2\pi}\sigma_1}\exp(-\frac{(x-m_1)^2}{2\sigma_1^2}) + \frac{A_2}{\sqrt{2\pi}\sigma_2}\exp(-\frac{(x-m_2)^2}{2\sigma_2^2})$$

- 下面的交互函数接受键盘输入，并绘制得到的高斯函数。有关函数输入和str2num的说明，请参阅2.10.5节。注意如何设置图的限制。

函数 manualhist



利用直方图匹配解决前面的图像（3）

- 用可用下面的matlab语句生成具有指定直方图的图像

```
>> g = histeq(f, p);
```

- matlab实验代码：

```
>> f=imread('Fig0310(a)(Moon Phobos).tif');
>> g=histeq(f,manualhist);
输入m1, sig1, m2, sig2, A1, A2, k或x退出;
>> imshow(g);
```

利用直方图匹配解决前面的图像（4）

- 实验的结果

a b
c

FIGURE 3.11
(a) Specified histogram.
(b) Result of enhancement by histogram matching.
(c) Histogram of (b).