

图象工程（上）

图 象 处 理

（第3版）

章毓晋

清华大学电子工程系 100084 北京



第1单元 图象增强

- 第2章 空域增强：点操作
- 第3章 空域增强：模板操作
- 第4章 频域图象增强

图象增强技术是最基本和最常用的一大类图象处理技术，也常用于其他图象技术应用的预处理阶段。图象增强的目的是通过对图象的特定加工，以将被处理的图象转化为对具体应用来说视觉质量和效果更“好”或更“有用”的图象。



第2章 空域增强：点操作

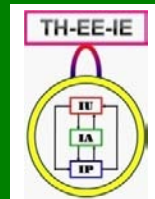
{每次运算基于单个像素}

2.1 图象坐标变换

2.2 图象间运算

2.3 图象灰度映射

2.4 直方图变换



2.1 图象坐标变换

{改变 $f(x, y)$ 中的 (x, y) }

2.1.1 基本坐标变换

2.1.2 坐标变换扩展

2.1.1 基本坐标变换

坐标 (x, y) , 如用齐次坐标, 则记为 $(x, y, 1)$

$$\mathbf{v}' = \mathbf{A}\mathbf{v}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\mathbf{v}' = [x' \quad y' \quad 1]^T$$

$$\mathbf{v} = [x \quad y \quad 1]^T$$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

2.1.1 基本坐标变换

平移变换

$$\mathbf{v}' = T\mathbf{v}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\mathbf{v}' = [x' \quad y' \quad 1]^T$$

$$\mathbf{v} = [x \quad y \quad 1]^T$$

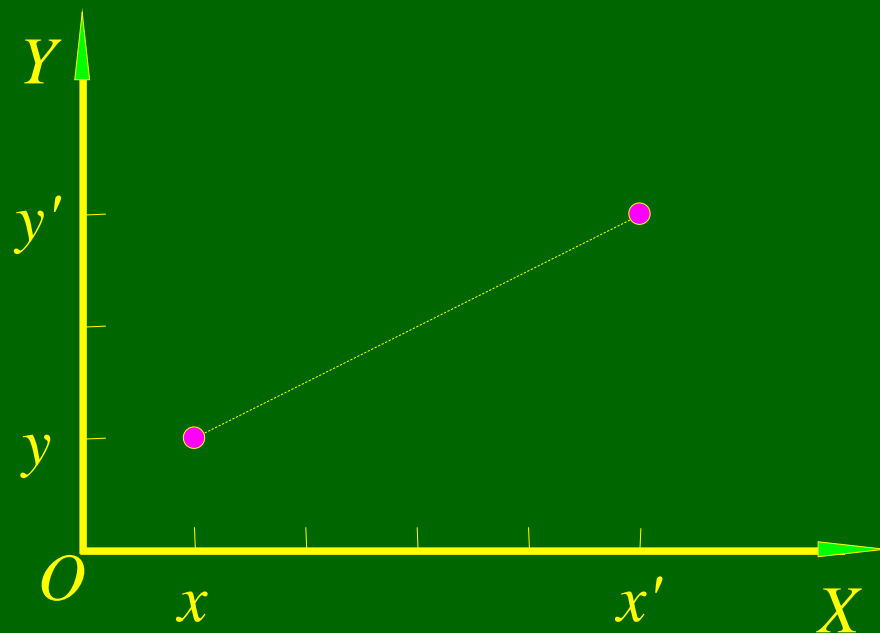
$$T = \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix}$$

2.1.1 基本坐标变换

平移变换

$$T = \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 4 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 3 \\ 1 \end{bmatrix}$$



2.1.1 基本坐标变换

放缩变换

$$\mathbf{v}' = S\mathbf{v}$$

$$\mathbf{v}' = [x' \quad y' \quad 1]^T$$

$$\mathbf{v} = [x \quad y \quad 1]^T$$

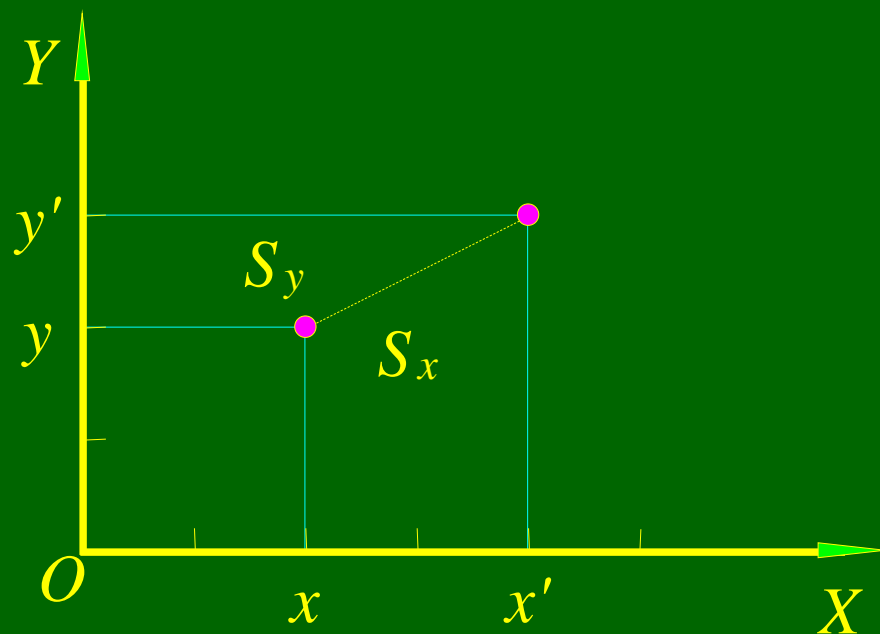
$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.1.1 基本坐标变换

放缩变换

$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 1 \end{bmatrix}$$



2.1.1 基本坐标变换

放缩变换

$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



2.1.1 基本坐标变换

旋转变换

$$\mathbf{v}' = \mathbf{R}_\gamma \mathbf{v}$$

$$\mathbf{v}' = [x' \quad y' \quad 1]^T$$

$$\mathbf{v} = [x \quad y \quad 1]^T$$

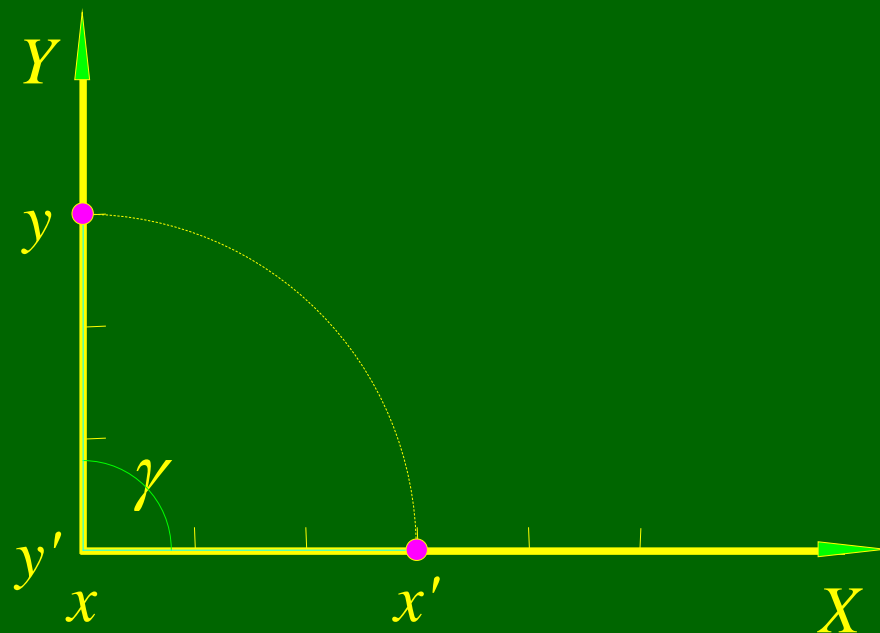
$$\mathbf{R}_\gamma = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.1.1 基本坐标变换

旋转变换

$$R_{\gamma} = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}$$



2.1.1 基本坐标变换

旋转变换

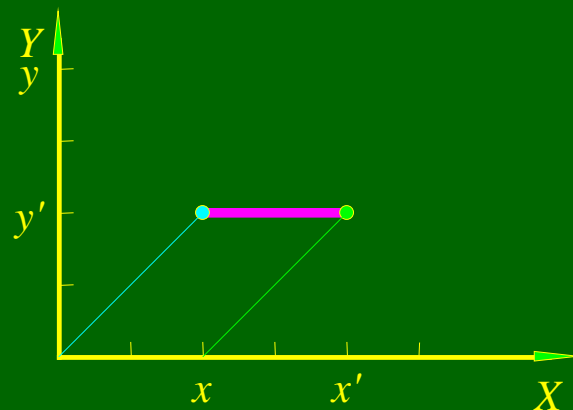
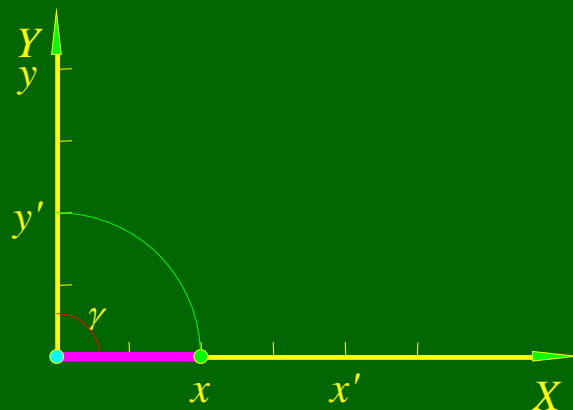
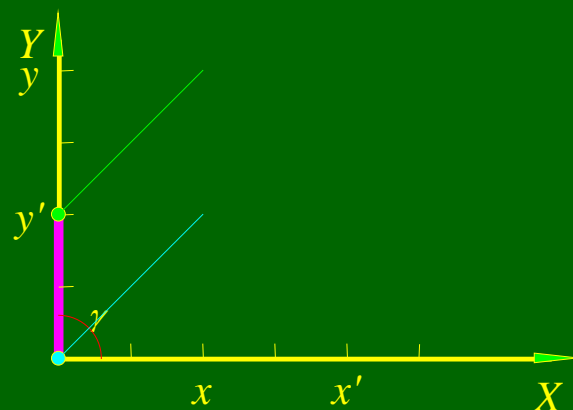
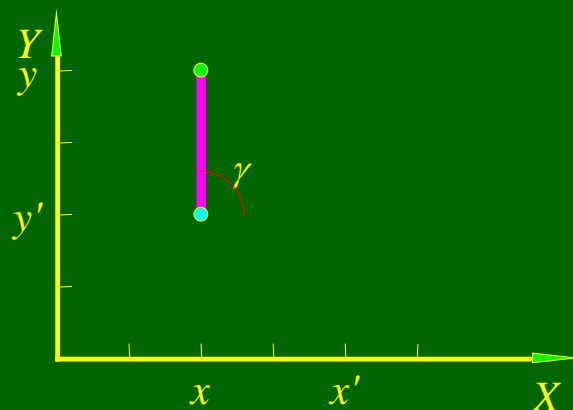
$$\gamma = -45^\circ \quad \mathbf{R} = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 & 0 \\ \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

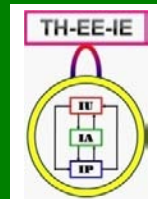


2.1.1 基本坐标变换

旋转轴不在原点

绕非原点
旋转
=
平移至原点
+
绕原点旋转
+
平移回去





2.1.2 坐标变换扩展

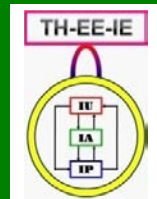
变换级连

对一个坐标为 \mathbf{v} 的点的平移、放缩、绕Z轴旋转变换（级联起来）可表示为：

$$\mathbf{v}' = \mathbf{R}_\gamma [\mathbf{S}(\mathbf{T}\mathbf{v})] = \mathbf{A}\mathbf{v}$$

等价于用单个变换矩阵 \mathbf{A} 对点 \mathbf{v} 进行变换

这些矩阵的运算次序一般不可互换



2.1.2 坐标变换扩展

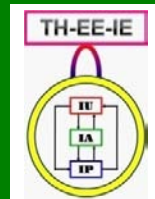
变换级连

级连示例

$$T = \begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{bmatrix} \quad S = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad v = [1 \quad 2 \quad 1]^T$$

$$v'_{ST} = [9 \quad 1 \quad 21]^T \quad v'_{TS} = [5 \quad 8 \quad 1]^T$$

先平移后放缩 \neq 先放缩后平移



2.1.2 坐标变换扩展

反变换

逆矩阵

$$T^{-1} = \begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

•对比
(2.1.4)

$$S^{-1} = \begin{bmatrix} 1/S_x & 0 & 0 \\ 0 & 1/S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

•对比
(2.1.5)

$$R_{\gamma}^{-1} = \begin{bmatrix} \cos(-\gamma) & \sin(-\gamma) & 0 \\ -\sin(-\gamma) & \cos(-\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

•对比
(2.1.6)



2.1.2 坐标变换扩展

其他3-点映射变换

拉伸变换

$$\mathbf{L} = \begin{bmatrix} L & 0 & 0 \\ 0 & 1/L & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

剪切变换

$$\mathbf{J}_h = \begin{bmatrix} 1 & J_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{J}_v = \begin{bmatrix} 1 & 0 & 0 \\ J_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.1.2 坐标变换扩展

基本坐标变换

平移变换 放缩变换 旋转变换
拉伸变换 剪切变换

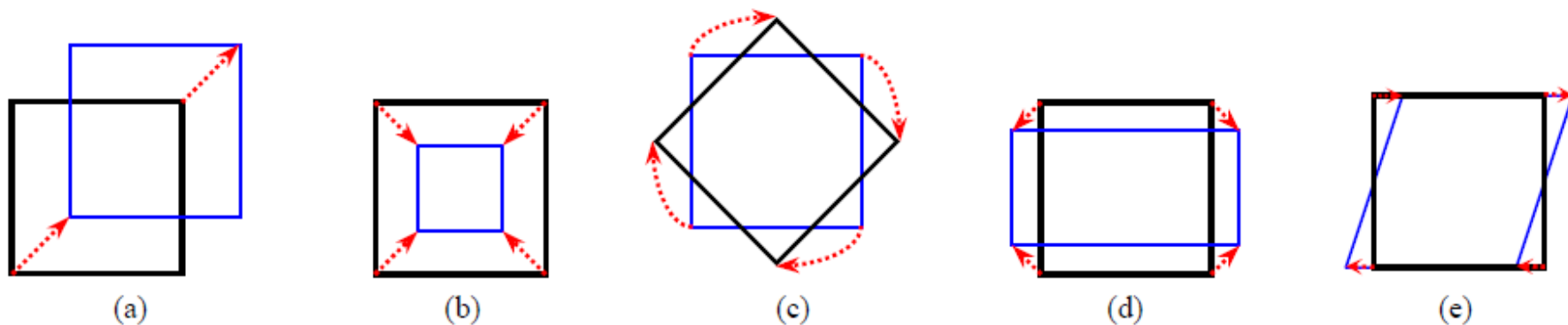


图 2.1.2 五种典型的坐标变换示意

2.1.2 坐标变换扩展

旋转变换的分解

$$\mathbf{R} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -\tan \theta & 1/\cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} x'' &= x' \\ y'' &= y' / \cos \theta - x' \tan \theta \end{aligned}$$

$$\begin{aligned} x' &= x \cos \theta + y \sin \theta \\ y' &= y \end{aligned}$$

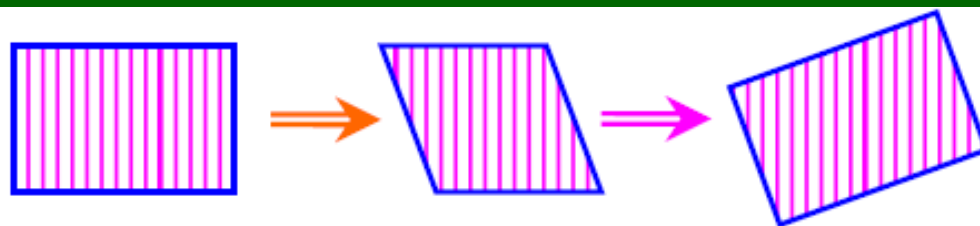
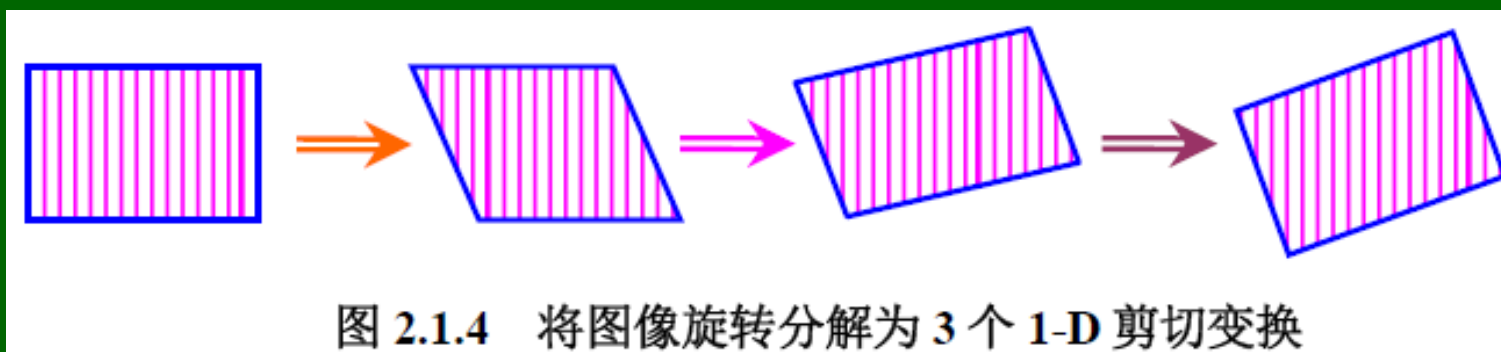


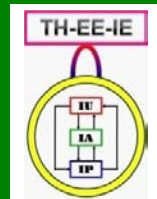
图 2.1.3 通过结合剪切以及水平和垂直方向的放缩来实现快速旋转

2.1.2 坐标变换扩展

旋转变换的分解

$$R = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & \tan(\theta/2) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ -\sin \theta & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & \tan(\theta/2) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



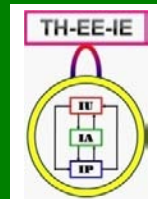


2.2 图象间运算

{将图象作为运算单元}

2.2.1 算术和逻辑运算

2.2.2 图象间算术运算的应用

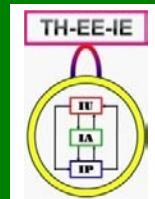


2.2.1 算术和逻辑运算

1. 算术运算

- (1) 加法: 记为 $p + q$
- (2) 减法: 记为 $p - q$
- (3) 乘法: 记为 $p * q$ (也可写为 pq 或 $p \times q$)
- (4) 除法: 记为 $p \div q$

对整幅图象的算术和逻辑运算是逐象素进行的, 即在两幅图象的对应(位置)象素间进行



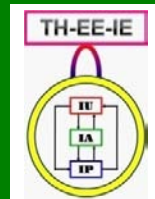
2.2.1 算术和逻辑运算

2. 逻辑运算

- (1) 补 (COMPLEMENT) : 记为 $\text{NOT } q$
- (2) 与 (AND) : 记为 $p \text{ AND } q$
- (3) 或 (OR) : 记为 $p \text{ OR } q$
- (4) 异或 (XOR) : 记为 $p \text{ XOR } q$

{例2.2.1, 图2.2.1}

- 组合基本逻辑运算
{例2.2.2, 图2.2.2}



2.2.2 图象间算术运算的应用

1. 图象间加法的应用

{ 去除采集噪声 }

模型

$$g(x, y) = f(x, y) + e(x, y)$$

运算

$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

均值

$$E\{\bar{g}(x, y)\} = f(x, y)$$

均方差

$$\sigma_{\bar{g}(x, y)} = \sqrt{1/M} \times \sigma_{e(x, y)}$$

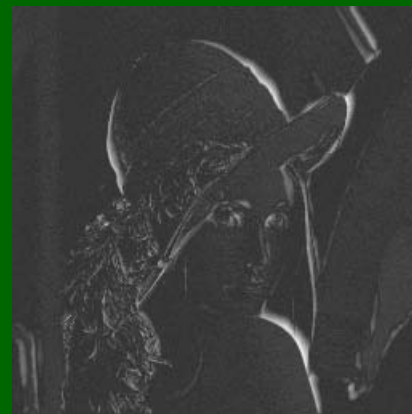
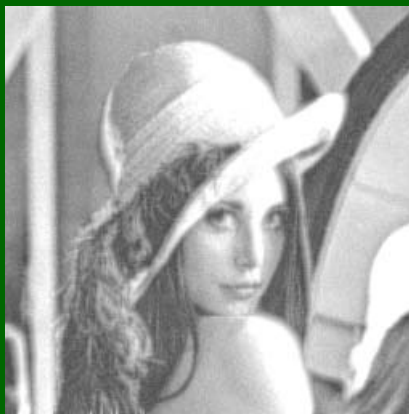
{ 图2.2.3 }

2.2.2 图象间算术运算的应用

2. 图象间减法的应用

两图相减运算可把差异凸显出来

$$g(x, y) = f(x, y) - h(x, y)$$



{例2.2.4}

2.2.2 图象间算术运算的应用

3. 图象间乘法和除法的应用

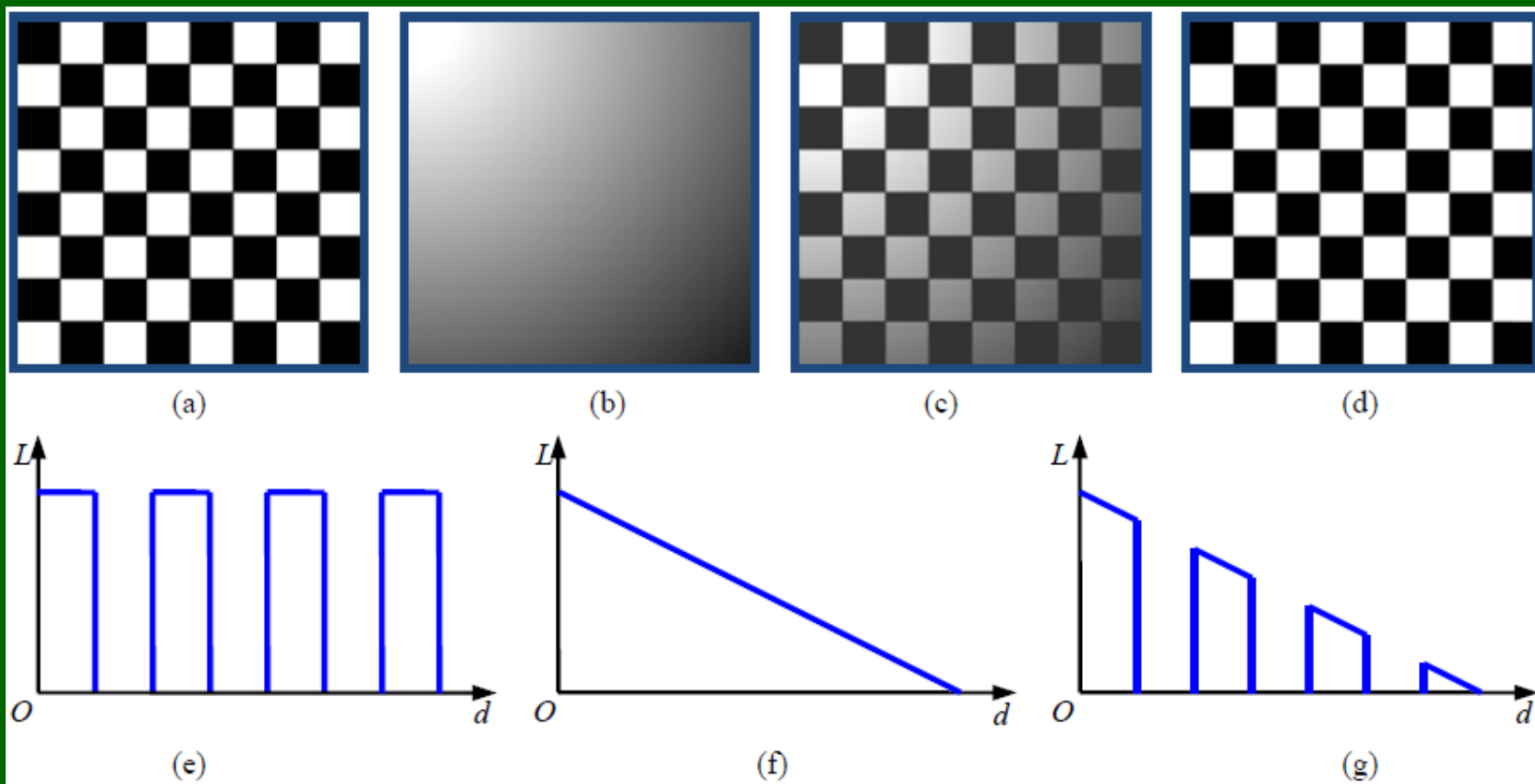


图 2.2.5 利用图像除法来校正照明的非均匀性



2.3 图象灰度映射

{将 $f(x, y)$ 中的每个象素灰度按映射 E_H 操作，
直接变换以得到 $g(x, y)$ }

2.3.1 灰度映射原理

2.3.2 典型灰度映射

2.3.1 灰度映射原理

灰度映射函数

{图2.3.2}

$$t = E_H(s)$$

增强灰度值

原始灰度值

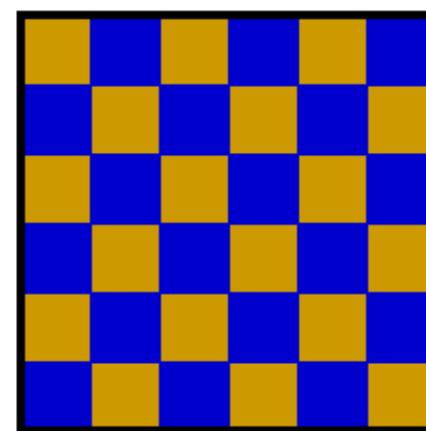
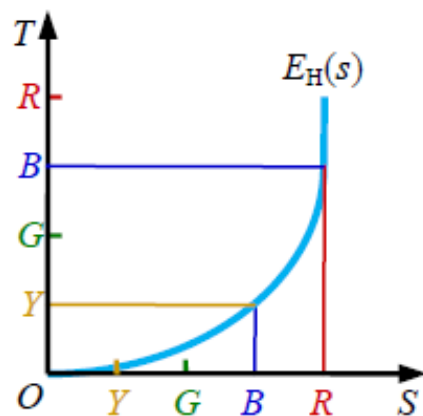
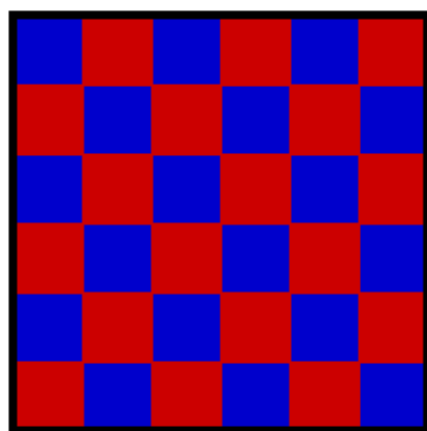


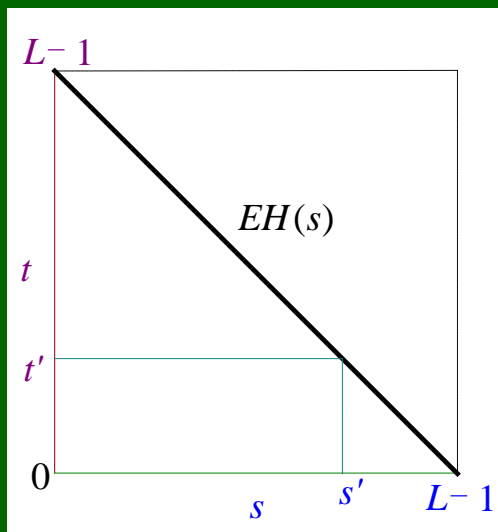
图 2.3.1 图像灰度映射原理

2.3.2 典型灰度映射

1、图象求反

(照片和底片)

$$t = L - 1 - s$$

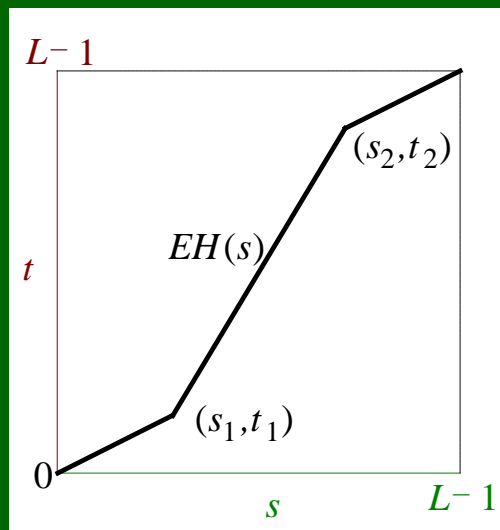


2.3.2 典型灰度映射

2、增强对比度

各段斜率不同

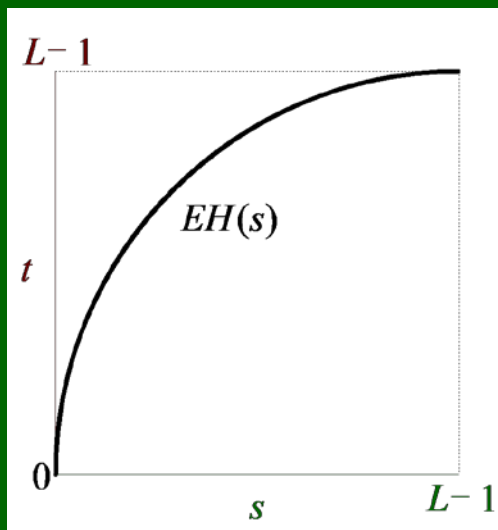
$$t = \begin{cases} as & 0 < s < s_1 \\ bs + d & s_1 < s < s_2 \\ cs + e & s_2 < s < L-1 \end{cases}$$



2.3.2 典型灰度映射

3、动态范围压缩

$$t = \sqrt{(L-1)^2 - (s - L + 1)^2}$$



2.3.2 典型灰度映射

4、伽马校正

$$t = Cs^\gamma$$

(指数变换)

变换的结果是输入中较窄的低灰度范围被映射到输出中较宽的灰度范围，而同时输入中较宽的高灰度范围被映射到输出中较窄的灰度范围

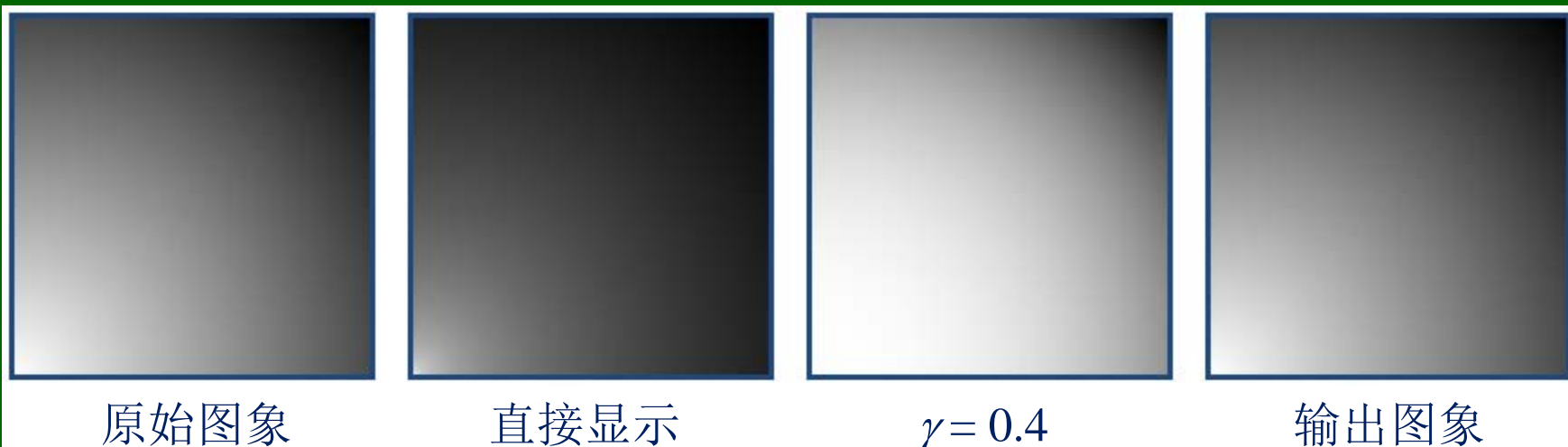


图 2.3.5 伽玛校正示例



2.4 直方图变换

{ 以概率论为基础的，通过改变图象的直方图来改变图象中像素的灰度，以达到图象增强的目标。也常称直方图修正 }

2.4.1 直方图均衡化

2.4.2 直方图规定化

2.4.1 直方图均衡化

灰度统计直方图

1-D的离散函数 (以灰度值为自变量)

提供了图象像素的灰度值分布情况

计算:

设置一个
有 L 个元素的数
组, 对原图的灰
度值进行统计

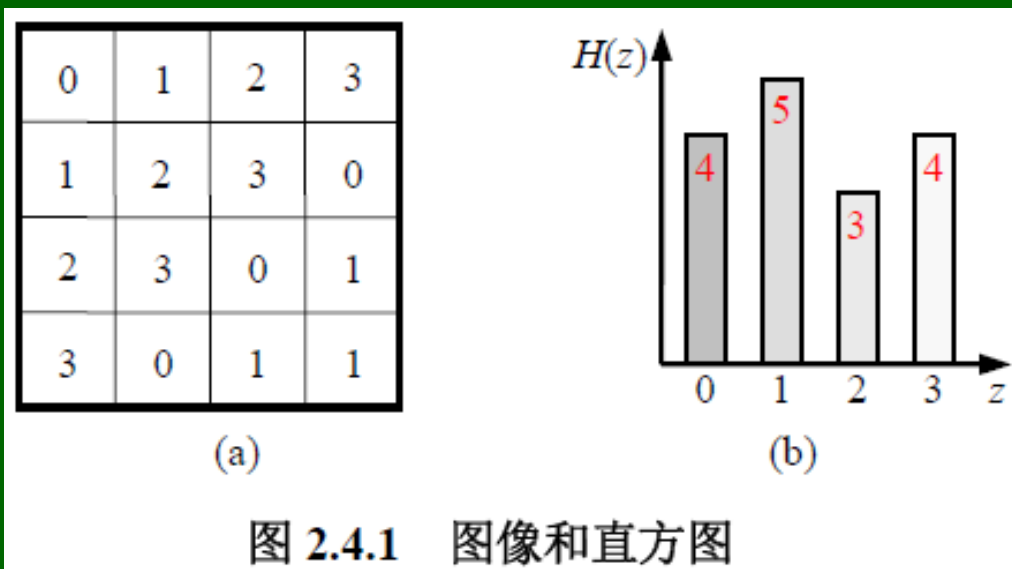
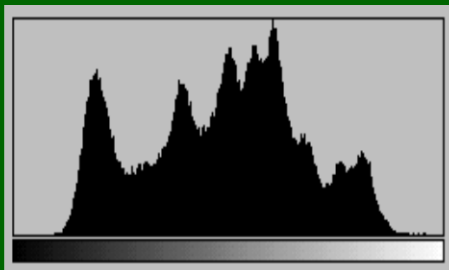


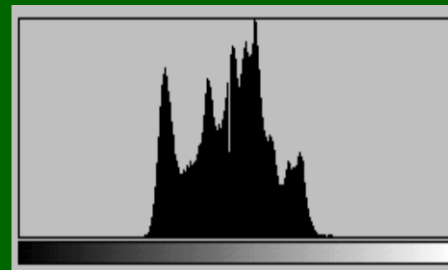
图 2.4.1 图像和直方图

2.4.1 直方图均衡化

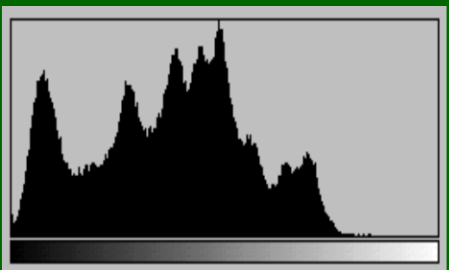
直方图与图象的视觉效果密切相关



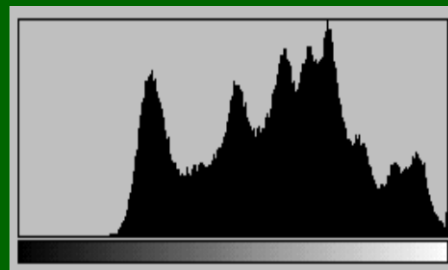
动态范围正常



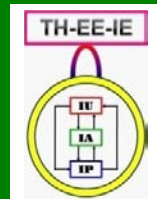
动态范围偏小



整体偏暗



整体偏亮



2.4.1 直方图均衡化

直方图均衡化原理

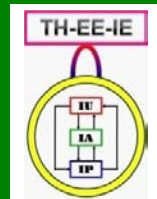
借助直方图变换实现（归一的）灰度映射
均衡化（线性化）基本思想

变换原始图象的直方图为均匀分布

==> 大动态范围

使像素灰度值的动态范围最大

==> 增强图象整体对比度（反差）



2.4.1 直方图均衡化

归一化直方图

$$p_s(s_k) = n_k / n \quad \begin{matrix} 0 \leq s_k \leq L-1 \\ k = 0, 1, \dots, L-1 \end{matrix}$$

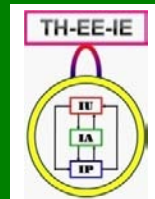
增强函数

(1) $E_H(s)$: 单值单增函数, $0 \leq s \leq L-1$

各灰度级在变换后仍保持排列次序

(2) $0 \leq E_H(s) \leq L-1$

变换前后灰度值动态范围一致



2.4.1 直方图均衡化

- 满足条件的增强函数——累积直方图
1-D的离散函数
提供了图象像素灰度值的累积分布情况

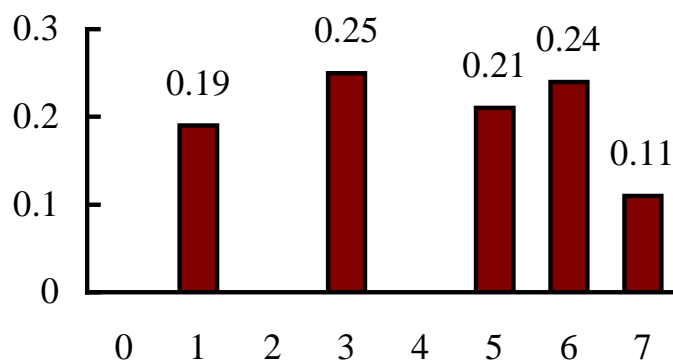
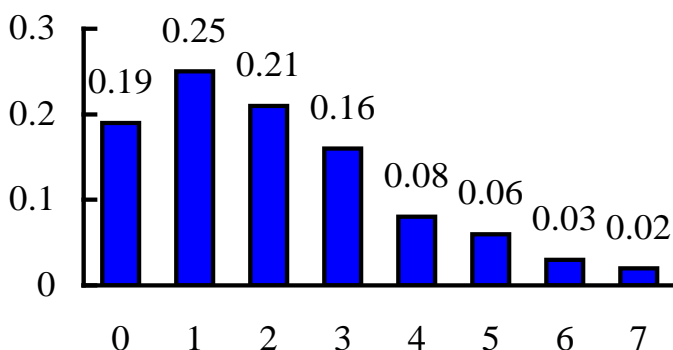
$$t_k = E_H(s_k) = \sum_{i=0}^k \frac{n_i}{N} = \sum_{i=0}^k p_s(s_i)$$

- (1) t_k 是 k 的单值单增函数
- (2) 灰度取值范围一致, $0 \leq t_k \leq 1$
- (3) 将 s 的分布转换为 t 的均匀分布 {表2.4.1}

2.4.1 直方图均衡化

直方图均衡化示例

为什么不平?



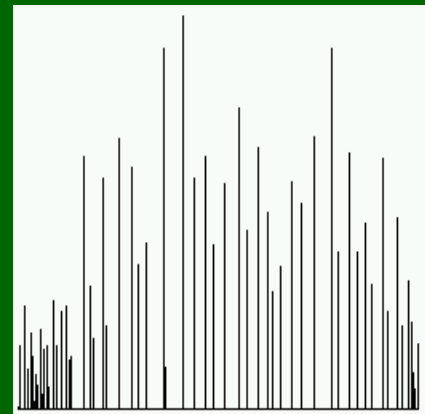
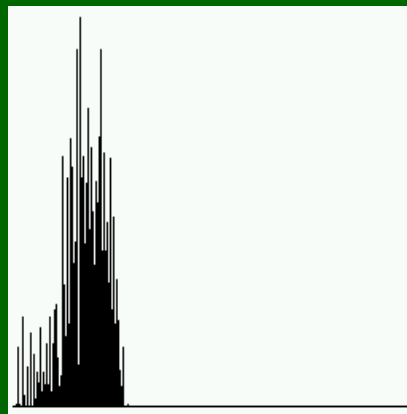
2.4.1 直方图均衡化

直方图均衡化的效果

图象对比



直方图对比



- (1) 能将同一灰度的像素区分开来吗?
- (2) 为什么直方图各条 (bin) 间有空隙?



2.4.2 直方图规定化

借助直方图变换实现规定/特定的灰度映射

(1) 对原始直方图进行灰度均衡化

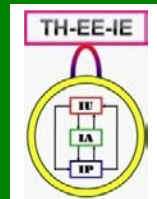
$$t_k = EH_s(s_i) = \sum_{i=0}^k p_s(s_i)$$

(2) 规定所需要的直方图，计算能使规定直方图均衡化的变换

$$v_l = EH_u(u_j) = \sum_{j=0}^l p_u(u_j)$$

(3) 将原始直方图对应映射到规定直方图

三个步骤



2.4.2 直方图规定化

两种映射/对应规则

(1) 单映射规则

$$\left| \sum_{i=0}^k p_s(s_i) - \sum_{j=0}^l p_u(u_j) \right| \quad \begin{array}{l} k = 0, 1, \dots, M-1 \\ l = 0, 1, \dots, N-1 \end{array}$$

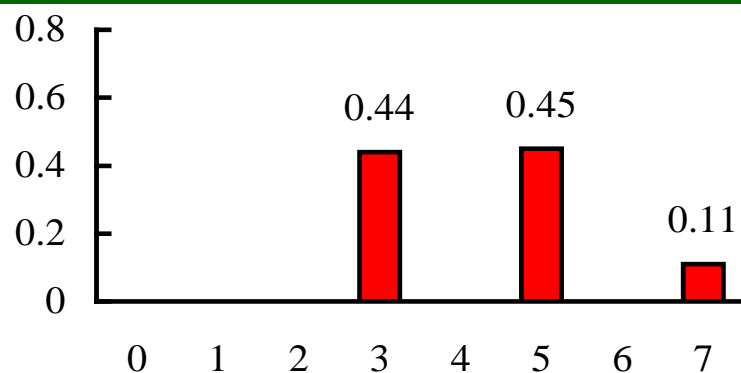
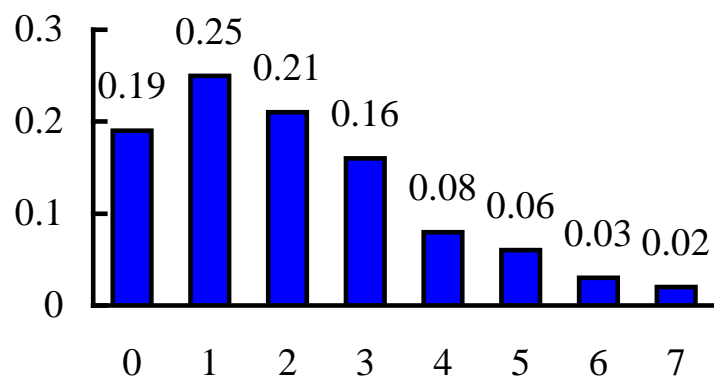
(2) 组映射规则 ($I(l)$: 整数函数)

$$\left| \sum_{i=0}^{I(l)} p_s(s_i) - \sum_{j=0}^l p_u(u_j) \right| \quad l = 0, 1, \dots, N-1$$

{表2.4.2}

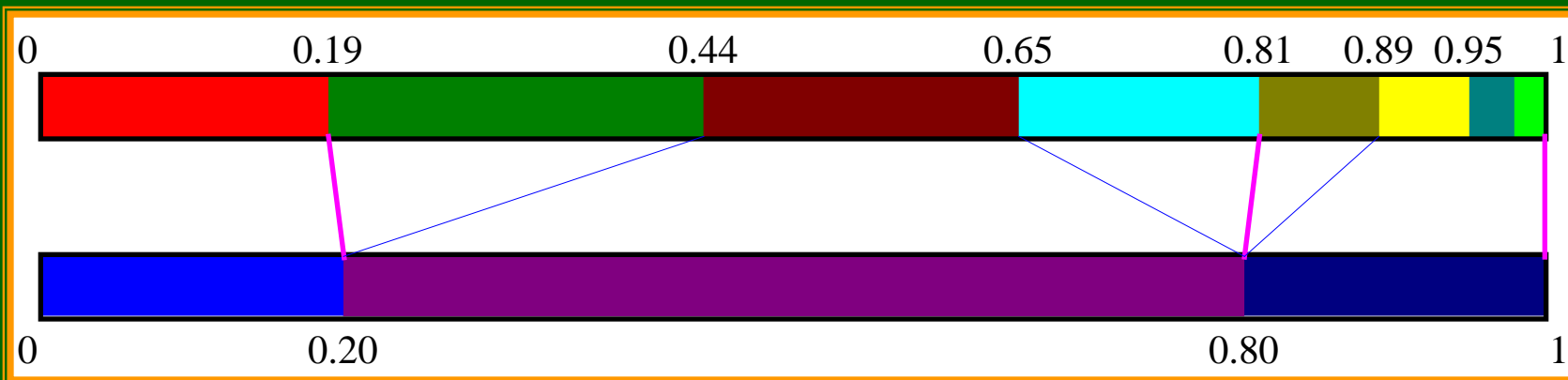
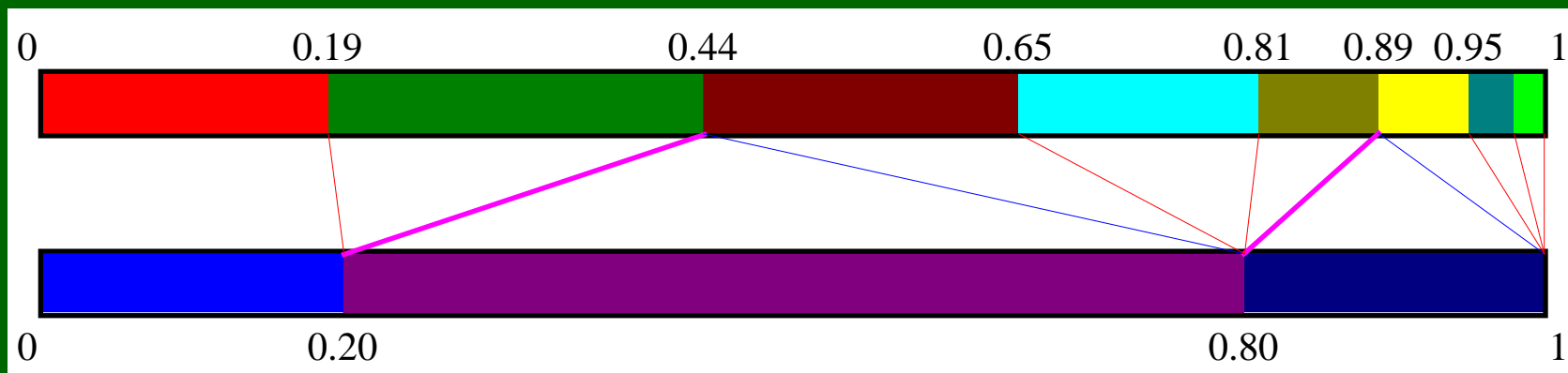
2.4.2 直方图规定化

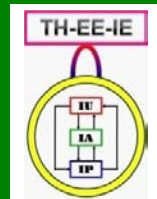
直方图规定化示例



2.4.2 直方图规定化

单映射规则 vs. 组映射规则





2.4.2 直方图规定化

映射误差

对应映射间数值的差值（取绝对值）的和

单映射规则：最大误差 $p_u(u_j)/2$

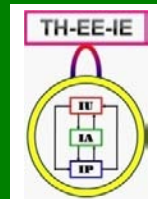
组映射规则：最大误差 $p_s(s_i)/2$

$$\because N \leq M, \therefore p_s(s_i)/2 \leq p_u(u_j)/2$$

单映射规则：有偏的映射规则

组映射规则：统计无偏的映射规则

{P.49: 映射误差及期望值}



2.4.2 直方图规定化

直方图规定化 vs. 直方图均衡化

- 直方图均衡化：
 - 自动增强
 - 效果不易控制
 - 总得到全图增强的结果
- 直方图规定化：
 - 有选择地增强
 - 须给定需要的直方图
 - 可得到特定增强的结果



联系信息



- ☞ 通信地址：北京清华大学电子工程系
- ☞ 邮政编码：100084
- ☞ 办公地址：清华大学，罗姆楼，6层305室
- ☞ 办公电话：(010) 62798540
- ☞ 传真号码：(010) 62770317
- ☞ 电子邮件：zhang-yj@tsinghua.edu.cn
- ☞ 个人主页：oa.ee.tsinghua.edu.cn/~zhangyujin/