

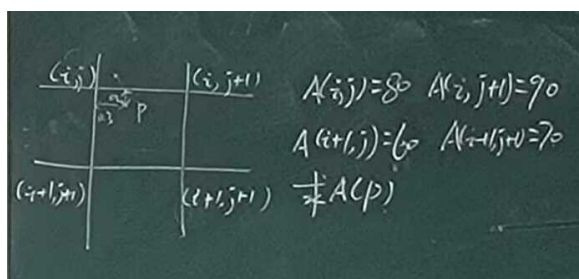
# 光电图像处理考试重点

宇宙安全声明：以下内容均为本人与诸多有志之士一同整理，不保证正确率，如有错误自行负责。

## 计算题

计算题基本是原题，公公又式式，重点

### 插值运算(双线性插值)



(已有两位豪杰指出板书的题目可能有问题)

已知图像中

$$A(i, j) = 80 \quad A(i, j+1) = 90$$

$$A(i+1, j) = 60 \quad A(i+1, j+1) = 70$$

求  $A(P)$  其中  $\alpha = 0.3$   $\beta = 0.2$

$\alpha$  和  $\beta$  分别表示什么意思?

$\alpha = 0.3$  表示所求点横向位置上距离左边的距离是整个宽度的 30%

$\beta = 0.2$  表示所求点在纵向位置上距离上边的距离是整个高度的 20%

解答:

横向插值:

$$A_1 = (1 - \alpha) \cdot A(i, j) + \alpha \cdot A(i, j+1)$$

$$= 0.7 \cdot 80 + 0.3 \cdot 90 = 83$$

$$A_2 = (1 - \alpha) \cdot A(i+1, j) + \alpha \cdot A(i+1, j+1)$$

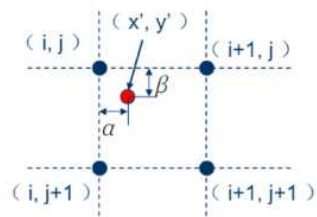
$$= 0.7 \cdot 60 + 0.3 \cdot 70 = 63$$

纵向插值 (其实就是双线性插值的最后结果)

$$A(P) = (1 - \beta)A_1 + \beta A_2 = 79$$

## ➤ 双线性插值

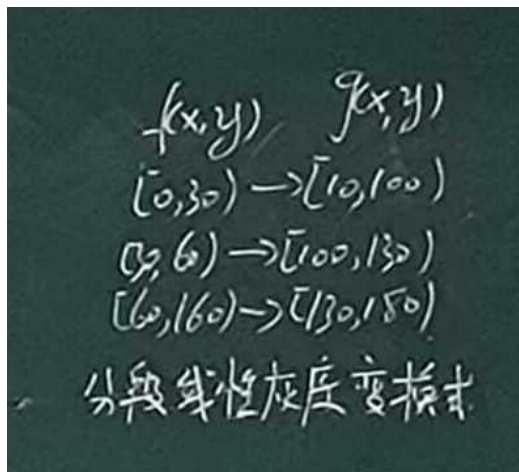
### —— Bilinear Interpolation



$$g(x', y') = (1-a)(1-\beta) \cdot g(i, j) \\ + (a)(1-\beta) \cdot g(i+1, j) \\ + (1-a)(\beta) \cdot g(i, j+1) \\ + a\beta \cdot g(i+1, j+1)$$

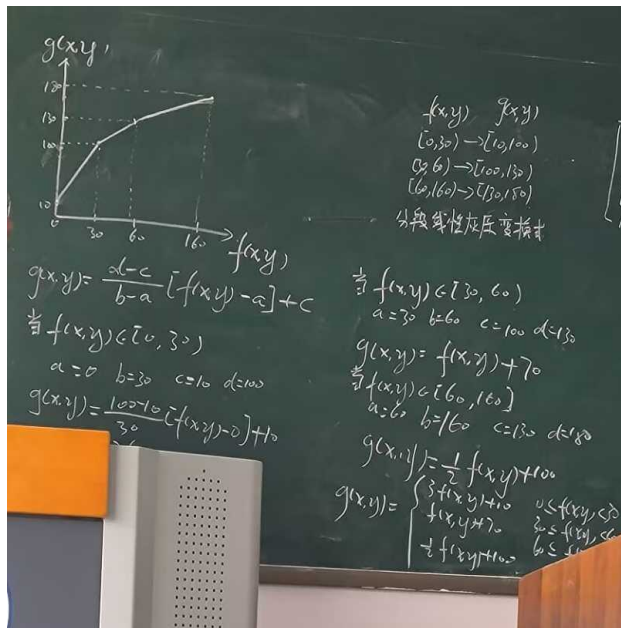
$$x' = i + a, \quad y' = j + \beta$$

## 分段线性灰度

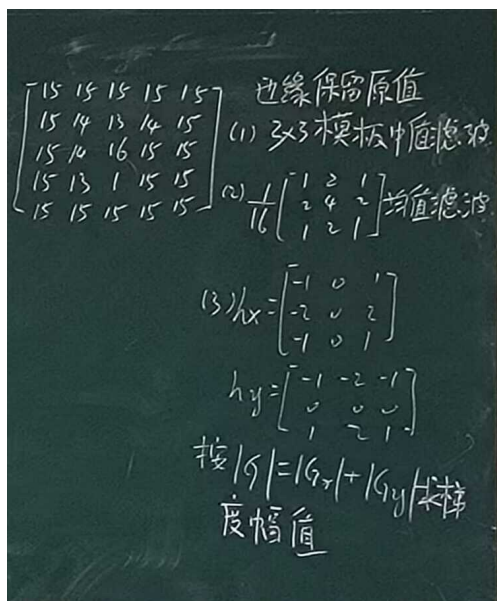


$$g(x, y) = \frac{d-c}{b-a} [f(x, y) - a] + c$$

即分段求解一元一次方程的表达式



## 模板运算(卷积)+梯度幅度



(1) 中值滤波

(中值滤波是一种非线性滤波，属于模板排序)

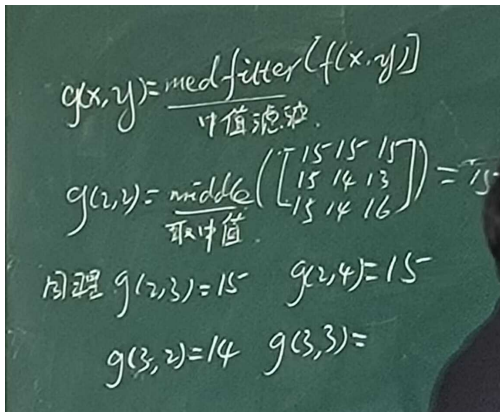
## 中值滤波——算例

例如：取 $3 \times 3$ 方形窗口

212	200	198		212	200	198
206	<u>202</u>	201	→	206	<u>205</u>	201
208	<u>205</u>	207		208	205	207

从小到大排列，取中间值

198 200 201 202 205 206 207 208 212



(2) 均值滤波

(均值滤波是一种线性滤波)

$$g(x, y) = \frac{\sum_{(s,t) \in N(x,y)} w(s,t) f(s,t)}{\sum_{(s,t) \in N(x,y)} w(s,t)}$$

(3) 梯度幅值

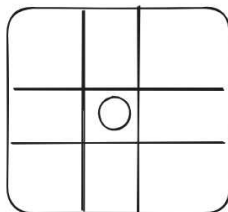
平方根开方法 (定义):

梯度的幅度用  $G[f(x,y)]$  表示:

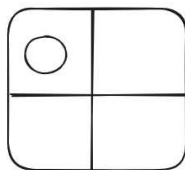
$$G[f(x,y)] = \left[ \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right]^{1/2}$$

绝对值相加法 (近似): (考试时使用这种方法计算)

类似Sobel算子的3x3  
卷积计算结果是放在中心  
如果题目中没有要求保留边缘  
就必须在原图像外一圈补0



类似Roberts算子的2x2卷积计算结果是  
放在左上角，如果题目没有要求保留边缘  
就必须在原图像的某两侧补0



使用算子（给定 Sobel 算子或 Roberts 算子等）卷积

（即  $h(x)$  和  $h(y)$ ，题目会给出）

分别得到每个像素点的  $|G(x)|$  和  $|G(y)|$ ，相加得到每个像素点的  $|G|$ ，即代表每个像素点的梯度幅度。

注意：在使用 Sobel 算子计算图像的梯度时，对于图像边缘的处理方式并不是固定不变的，而是取决于具体的实现和设置。常见的边缘处理方法包括补 0（zero-padding）或者保留原始值【本题题干要求】（即不进行填充，只对可以完全覆盖卷积核的区域进行计算）。这意味着在某些实现中，边缘可能会被补 0 以保持输出图像尺寸与输入相同；而在其他情况下，可能只会计算那些能够应用完整卷积核的位置的梯度【符合本题题干要求】，这会导致输出图像尺寸小于输入图像尺寸

\_\_Qwen3.0

## 直方图均衡化与规定化

### 题目 1 均衡化

例：设图像有  $64 \times 64 = 4096$  个像素，有 8 个灰度级，灰度分布如表所示。进行直方图均衡化。

$r_k$	$n_k$	$p(r_k)$
$r_0=0$	790	0.19
$r_1=1/7$	1023	0.25
$r_2=2/7$	850	0.21
$r_3=3/7$	656	0.16
$r_4=4/7$	329	0.08
$r_5=5/7$	245	0.06
$r_6=6/7$	122	0.03
$r_7=1$	81	0.02

解答：

8个灰级，总计 $64*64=4096$ 点

$D_A$	$n_k$	$P_r(r_k)$	$S$	取成整数倍	均衡后直方图
0	790	0.19	0.19	1/7(0.14)	0.19
1/7	1023	0.25	0.44	3/7(0.428)	0.25
2/7	850	0.21	0.65	5/7(0.714)	0.21
3/7	656	0.16	0.81	6/7(0.857)	0.16+0.08=0.24
4/7	329	0.08	0.89	6/7(0.857)	
5/7	245	0.06	0.95	7/7(1.00)	0.06+0.03+0.02=0.11
6/7	122	0.03	0.98	7/7(1.00)	
7/7	81	0.02	1.00	7/7(1.00)	

取整方法 1:  $int((L-1)g+0.5)$  (其中 L 为灰度级，本题为 8)

取整方法 2: 四舍五入  $((L-1) \cdot g)$  (个人不推荐用这种方法)

## 题目 2 规定化

例：设图象有 $64*64=4096$ 个像素，有8个灰度级，灰度分布如表所示。进行直方图规定化。

$r_k$	$n_k$	$p(r_k)$	$P(z)$	$s_k$
$r_0=0$	790	0.19	0	$s_0$
$r_1=1/7$	1023	0.25	0	$s_1$
$r_2=2/7$	850	0.21	0	$s_2$
$r_3=3/7$	656	0.16	0.15	$s_3$
$r_4=4/7$	329	0.08	0.20	$s_4$
$r_5=5/7$	245	0.06	0.30	$s_5$
$r_6=6/7$	122	0.03	0.20	$s_6$
$r_7=1$	81	0.02	0.15	$s_7$

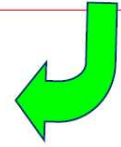
解答：





### 4.3.3 直方图规定化

#### 5 确定变换关系



$r_k$	$n_k$	$p(r_k)$	$V_1$ 计算	$V_2$ 计算	$P(z)$	$s_k$
$r_0=0$	790	0.19	0.19	0	0	$s_0$
$r_1=1/7$	1023	0.25	0.44	0	0	$s_1$
$r_2=2/7$	850	0.21	0.65	0	0	$s_2$
$r_3=3/7$	656	0.16	0.81	0.15	0.15	$s_3$
$r_4=4/7$	329	0.08	0.89	0.35	0.20	$s_4$
$r_5=5/7$	245	0.06	0.95	0.65	0.30	$s_5$
$r_6=6/7$	122	0.03	0.98	0.85	0.20	$s_6$
$r_7=1$	81	0.02	1.00	1.00	0.15	$s_7$



### 4.3.3 直方图规定化

序号	运算	步骤和结果							
1	原始图像灰度级	0	1	2	3	4	5	6	7
2	各灰度级像素	790	1023	850	656	329	245	122	81
3	原始直方图 $P(r)$	0.19	0.25	0.21	0.16	0.08	0.06	0.03	0.02
4	原始累积直方图 $V_1$	0.19	0.44	0.65	0.81	0.89	0.95	0.98	1.00
5	规定直方图 $P(z)$	0	0	0	0.15	0.20	0.30	0.20	0.15
6	规定累积直方图 $V_2$	0	0	0	0.15	0.35	0.65	0.85	1.00
7	映射 $ V_2 - V_1 $ 最小	3	4	5	6	6	7	7	7
8	确定映射关系	0->3	1->4	2->5	3,4->6		5,6,7->7		
9	变换后直方图	0	0	0	0.19	0.25	0.21	0.24	0.11

请注意，第7步中，是以 $V_1$ 作为锚点，依次与各个 $V_2$ 相比较。

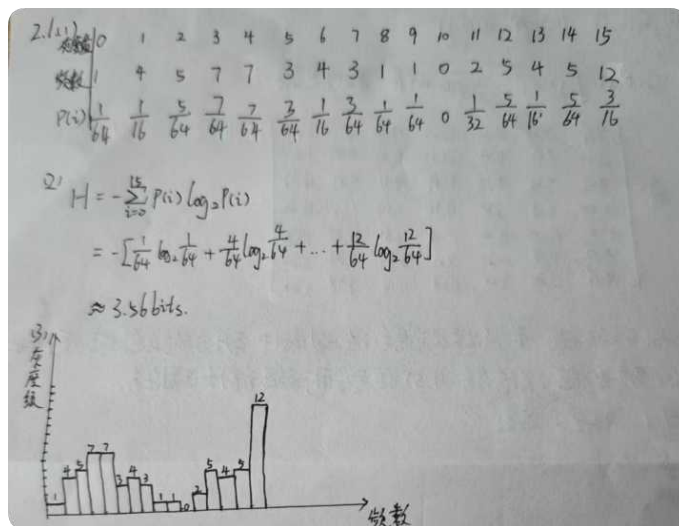
## 频数+一阶差分+熵的计算+直方图+梯度

2.1 给出一幅4bit 的图像A（见下图）。

$$A = \begin{bmatrix} 11 & 14 & 4 & 15 & 1 & 15 & 15 & 15 \\ 12 & 12 & 9 & 14 & 4 & 15 & 12 & 13 \\ 0 & 2 & 6 & 15 & 3 & 7 & 1 & 13 \\ 4 & 2 & 8 & 2 & 5 & 3 & 3 & 15 \\ 5 & 3 & 3 & 15 & 4 & 2 & 6 & 1 \\ 11 & 15 & 4 & 14 & 7 & 15 & 13 & 15 \\ 12 & 13 & 6 & 15 & 3 & 14 & 12 & 14 \\ 4 & 2 & 6 & 4 & 1 & 7 & 3 & 5 \end{bmatrix}$$

- (1) 逐个统计各灰度级出现的频数；
- (2) 计算该图像的熵；
- (3) 画出A 的直方图。

解答：



计算熵的公式：

$$H(A) = -\sum_{i=1}^M P_i \log_2 P_i$$

哦

解答：（x 方向不看第一列，从最后一排往上减，y 方向不看第一行，从最右边往左边减）



是的，不单独考差分，考的是卷积运算

注意：这种计算方法**不会单独考**，会以上述【模板运算（卷积）+梯度幅度】第三问的形式出现，也就是说你会做那道题就行了。



-2	5	-1	3	0	-3	-2
-10	-3	1	-1	-8	-11	0
0	2	-13	2	-4	2	2
1	-5	13	-1	-1	3	-14
12	1	-1	3	13	7	14
-2	2	1	-4	-1	-1	-1
-11	0	-11	-2	-7	-9	-9

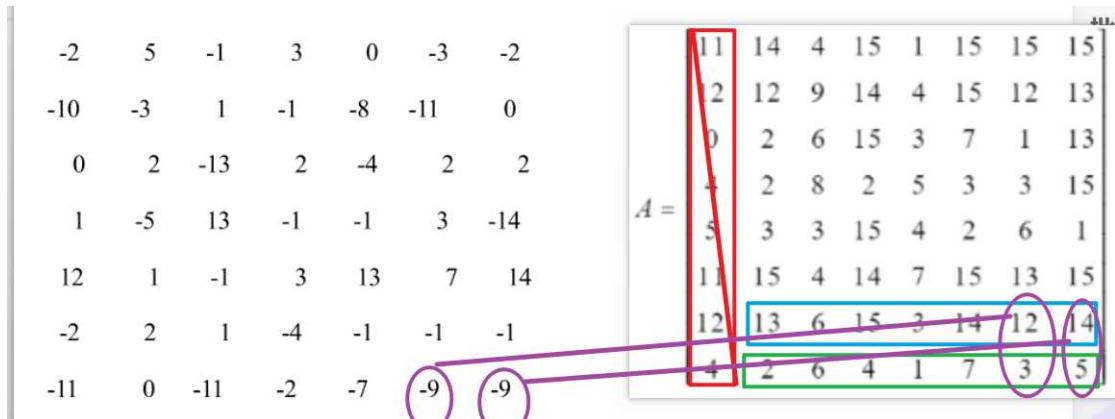
沿 y 方向一阶差分 `direc_y =`

0	-3	5	-10	11	-3	1
2	4	9	-12	4	-6	12
-2	6	-6	3	-2	0	12
-2	0	12	-11	-2	4	-5
4	-11	10	-7	8	-2	2
1	-7	9	-12	11	-2	2
-2	4	-2	-3	6	-4	2

两个方向梯度绝对值之和求幅值 `amp1 =`

2	8	6	13	11	6	3
12	7	10	13	12	17	12
2	8	19	5	6	2	14
3	5	25	12	3	7	19
16	12	11	10	21	9	16
3	9	10	16	12	3	3

直白解释：



## 霍夫曼编码（哈夫曼编码）

09:01 15.0 KB/S 4G 5G 52

### 8.2 霍夫曼编码

霍夫曼 (Huffman) 编码

33/60

通过减少编码冗余来达到数据压缩的目的。

#### 基本原理

将在图像中出现次数多的像素值给一个短的编码，将出现次数少的像素值给一个长的编码。

#### 举例说明：

aaaa bbb cc d eeeee fffffff (共 $22 \times 8 = 176$  bit)

4 3 2 1 5 7

f=0 e=10 a=110 b=1111 c=11100 d=11101

110, 110, 110, 110, 1111, 1111, 1111, 11100, 11100, 11101,

10, 10, 10, 10, 10, 0, 0, 0, 0, 0, 0, 0

(共  $7 \times 5 + 2 \times 4 \times 3 + 3 \times 4 + 2 \times 5 + 1 \times 5 = 56$  bit)

- 基本思想是统计各个符号出现的概率  $P_k$ 。
- 建立一个概率统计表。

将最经常出现(概率大的)的符号用最短的编码；最少出现的符号用最长的编码。

#### 步骤：

- $P_k$ 由大到小排列
- 最小两个概率相加，形成一个新的概率集合，再按1)重排，直至只有两个概率
- 分配码字

输入 输入概率 第一步

S <sub>1</sub>	0.4	0.4
S <sub>2</sub>	0.3	0.3
S <sub>3</sub>	0.1	0.1
S <sub>4</sub>	0.1	0.1
S <sub>5</sub>	0.06	0.1
S <sub>6</sub>	0.04	



输入 输入概率 第一步 第二步

S <sub>1</sub>	0.4	0.4	0.4
S <sub>2</sub>	0.3	0.3	0.3
S <sub>3</sub>	0.1	0.1	0.2
S <sub>4</sub>	0.1	0.1	0.1
S <sub>5</sub>	0.06	0.1	
S <sub>6</sub>	0.04		



输入 概率 第1步 第2步 第3步

S <sub>1</sub>	0.4	0.4	0.4	0.4
S <sub>2</sub>	0.3	0.3	0.3	0.3
S <sub>3</sub>	0.1	0.1	0.2	0.3
S <sub>4</sub>	0.1	0.1	0.1	
S <sub>5</sub>	0.06	0.1		
S <sub>6</sub>	0.04			



输入 概率 第1步 第2步 第3步 第4步

S <sub>1</sub>	0.4	0.4	0.4	0.4	0.6
S <sub>2</sub>	0.3	0.3	0.3	0.3	0.4
S <sub>3</sub>	0.1	0.1	0.2	0.3	
S <sub>4</sub>	0.1	0.1	0.1		
S <sub>5</sub>	0.06	0.1			
S <sub>6</sub>	0.04				



输入 概率 第1步 第2步 第3步 第4步

S <sub>1</sub>	0.4	0.4	0.4	0.4	0.6	0
S <sub>2</sub>	0.3	0.3	0.3	0.3	0.4	1
S <sub>3</sub>	0.1	0.1	0.2	0.3	1	
S <sub>4</sub>	0.1	0.1	0.1			
S <sub>5</sub>	0.06	0.1				
S <sub>6</sub>	0.04					

平均码制长度+编码效率+压缩比

题目 1 平均码制长度+编码效率+熵的计算

(2) 平均码字长度:

$\beta_i$  是第*i*种字符的长度(二进制代码的位数),  $P_i$  是第*i*种字符出现的概率, 则平均码长为:

$$L=\sum_{i=1}^M \beta_i P_i$$

短语	符号	频数	概率(频率)	代码长度	编码
Because	B	1	0.03	5	01001
I'm	I	6	0.17	3	001
Bad	D	15	0.43	1	1
Come on	C	2	0.06	4	0101
It	I	1	0.03	5	01000
Really	R	6	0.17	3	000
You know	Y	4	0.11	3	011

编码效率:  $\eta = \frac{H}{L} \times 100\%$  ( $H$ 为熵,  $L$ 为平均码长)

$$\text{压缩比 } r = \frac{n}{L} = \frac{\Sigma(\text{频数})\log_2(\text{代码数量})}{\Sigma(\text{频率} \cdot \text{代码长度})}$$

压缩率计算勘误:

$\log 0.11) = 2.29$   
 $35 \times \log_2 7 = 98.3$ , 编码后: 81位,  
 $\eta = 2.29 / 2.32 \times 100\% = 98.7\%$   
 $98.3/81 \approx 1.213$ 。



编码后的那个数据大小是每个二进制数算1bit, 那个because 是5bit  $\times$  1 (频数), 然后每个加起来就出来了



前面的那个编码大小是总频数  $\times$  log以2为底的符号数量



09:14

他那个压缩率应该是算的不对, 应该是  $81/98.3 \times 100\%$



压缩比 ( $>1$ ) 和压缩率 ( $<1$ ) 是互为倒数的关系。

解答:

平均码长:

$$L=0.03 \times 5 + 0.17 \times 3 + 0.43 \times 1 + 0.06 \times 4 + 0.03 \times 5 \\ + 0.17 \times 3 + 0.11 \times 3 \\ = 2.32$$

熵:

$$H = -(0.03 \times \log_2 0.03 + 0.17 \times \log_2 0.17 + 0.43 \times \log_2 0.43 + \\ 0.06 \times \log_2 0.06 + 0.03 \times \log_2 0.03 + 0.17 \times \log_2 0.17 + \\ 0.11 \times \log_2 0.11) = 2.29$$

编码前:  $35 \times \log_2 7 = 98.3$ , 编码后: 81位,

编码效率:  $\eta = 2.29 / 2.32 \times 100\% = 98.7\%$

压缩率:  $98.3/81 \approx 1.213$ 。

(注意上面压缩率应该是压缩比, 写错了)

## 简答题 / 分析题

简答题和分析题都是不确定, 只能靠广撒网有印象了。

### 人眼结构相当于相机什么结构, 起到什么作用?

膜晶状体——相机镜头, 透镜, 改变焦距

瞳孔——相机光圈, 进光量

### 人眼有哪两种细胞, 有什么作用?

- a. **锥状细胞**: 在强光条件下发挥作用, 能分辨颜色和细节, 具有高分辨率, 是明视觉的基础
- b. **杆状细胞**: 对弱光敏感, 不能分辨颜色, 主要负责暗视觉和周边视觉。

### 人眼成像过程



### 视觉特性

视觉的空间特性

灰度分辨特性

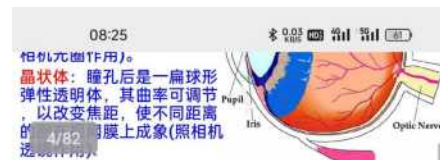


视觉的时间特性

马赫带效应

对数特性

侧抑制效应



### 2.1.1 人眼的构造与机理

视细胞：视网膜上集中了大量视细胞，分为两类：

锥状细胞：明视细胞，在强光下检测亮度和颜色；

杆(柱)状细胞：暗视细胞，在弱光下检测亮度，无色彩感觉。

其中，每个锥状视细胞连接着一个视神经末梢，故分辨率高，分辨细节、颜色；多个杆状视细胞连接着一个视神经末梢，故分辨率低，仅分辨图的轮廓。

人眼成像过程：



### 2.1.2 人眼的视觉特性与错觉

视觉的空间特性

人眼的空间分辨能力为1'(1/60度)；

灰度分辨特性

灰度分辨能力大约64级。

视觉的时间特性

活动图像的帧频至少是15帧/s的时候，人眼才有图像连贯的感觉。

### 2.1.2 人眼的视觉特性与错觉

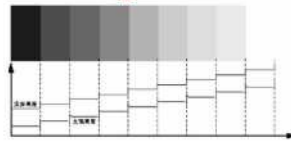
马赫带 (Mach Band) 效应



有图像连贯的感觉。

### 2.1.2 人眼的视觉特性与错觉

马赫带 (Mach Band) 效应



马赫带 (Mach band) 马赫发现的一种明度对比现象。它是一种主观的边缘对比效应。当观察两块亮度不同的区域时，边界处亮度对比加强，使轮廓表现得特别明显。

### 2.1.2 人眼的视觉特性与错觉

对数特性

主观亮度S与实际亮度B之间的关系  $S = K \ln B + k_0$

人眼亮度感觉之应用：若一幅原图像经过处理，恢复后得到重现图像，重现图像的亮度不必等于原图像的亮度，只要保证二者的对比度相同，就能给人以真实的视觉。

对一个色觉正常的人来说，颜色刺激面积非常小时就不能识别颜色了——第三色盲。

### 2.1.2 人眼的视觉特性与错觉

侧抑制效应

某个视觉信号并不是单纯由一个视细胞感光产生，而是由许多空间相邻的视细胞的信号加权求和产生。

视觉系统所感觉到的物体的形状并不是简单的投影到视网膜上的形状。对形状的感觉受到物体自身形状及周围背景的影响，也有心理因素的作用。

## 色度学三个基本特征

色度学是一门以光学、视觉生理、视觉心理等学科为基础的综合性学科，是一门研究彩色计量的科学，其任务是研究人眼彩色视觉的定性和定量规律及应用

色彩：彩色和非彩色(灰色)

色彩三个特征：色调、饱和度和亮度

色调是颜色的基本相貌，取决于物体发出的或反色的光线的主导波长

饱和度指彩色的深浅或纯洁程度，饱和度越高，纯度越高

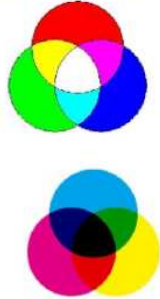
亮度表示物体颜色深浅明暗的特征量，光反射比越高，亮度越高。

## 物理配色，两种配色法，对应应用场合

### 物理配色机理

· 加色法：红、绿、蓝 (RGB)  
红色+绿色=黄色， 红色+蓝色 = 品红， 绿色+蓝色 = 青色  
红色+绿色+蓝色 = 白色  
通用配色公式： $C=rR+gG+bB$   
电视、监视器等设备利用加混色原理

2. 减色法：青、品、黄、黑 (CMYK)  
黄色=白色-蓝色， 青色=白色-红色， 品红=白色-绿色  
黑色=白色-蓝色-红色-绿色  
印刷、染织行业利用减混色原理



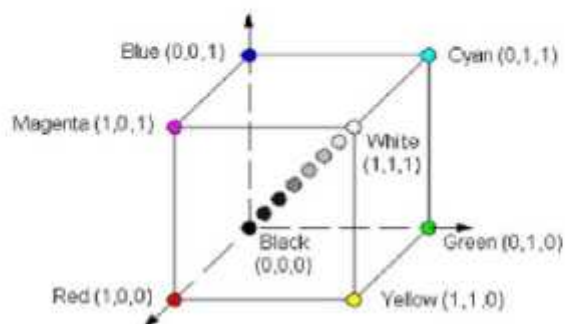
## 颜色模型

- RGB
- HSI



## (1) RGB颜色模型

RGB颜色模型就是三维直角坐标颜色系统的一个单位正方体，原点为黑色，距离原点最远的顶点(1,1,1)对应的颜色为白色。两点之间的连线即正方体的的对角线称为灰色线。



## 2.2.2 颜色模型

### (2) HSI颜色模型

HSI模型反映了人的视觉系统感知颜色的方式。其中**H(Hue)**表示色调，与混合光谱中主要光波长相联系；**S(Saturation)**表示饱和度，与一定色调的纯度有关；**I(Intensity)**表示明度，与物体的反射率成正比。它比RGB颜色模型更符合人的视觉特性



## 彩色图像，假彩色，真彩色和伪彩色，根本区别

### 伪彩色图像：

是指经过伪彩色处理而形成的彩色图像。其像素值是所谓的索引值，是按照灰度值进行彩色指定的结果，其色彩并不一定忠实于外界景物的真实色彩。

### 假彩色图像：

将一种彩色转换为另一种彩色，遥感多波段图像合成的彩色图像。

### 真彩色图像：

是忠实于外界景色的色彩的图像，其像素一般是颜色的真实值。

## 传感器 CCD，CMOS 的区别与工作流程

CMOS 每个像素包含了独立放大器与 A/D 转换电路(PPT 原话)

CCD 的像素结构通过将电荷依次传递到串联寄存器中进行统一处理。

工作流程：

相同部分：

- 光电转换：光电二极管受光产生电荷。
- 电荷存储：像素内将电荷转为电压。
- 信号读出：行列选址读取电压信号。

不同部分：

CCD：电荷通过一系列的 MOS 电容实现像素间的连续转移。

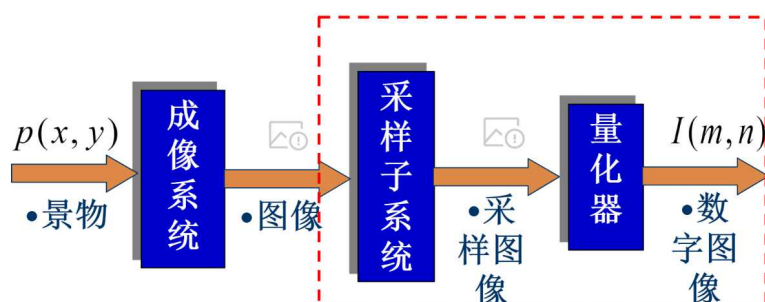
特点：灵敏度高，噪音小，信噪比大。但是生产工艺复杂、成本高、功耗高。(PPT 原话)

CMOS：电荷通过独立金属导线传递至各个像素独立的放大器和模数转换器。

特点： 功耗低（不到 CCD 的 1/3）、成本低。但是噪音比较大、灵敏度较低、对光源要求高。(PPT 原话)



## 图像数字化的过程



### 图像数字化流程

图像数字化即模拟图像离散化过程，包括采样和量化两个步骤。

## 位图格式

宋涛貌似提了一嘴位图格式，了解一下即可。



debbie. bmp

BMP是一种与设备无关的位图格式，一般采用非压缩模式。

256 × 256, 65KB



Water. tiff

TIFF是最复杂的一种位图文件格式，应用于对图像质量要求较高的图像的存储与转换。

640 × 480, 905KB

## 代数运算和逻辑运算区别

代数运算：

加运算：图像叠加，二次曝光，图像衔接。（PPT 原话）

减运算：同一场景的两幅图像变化检测，运动检测，去除一幅图像中不需要的加性图案，如缓慢变化的背景阴影，周期性噪声（PPT 原话）

乘运算：掩膜作用，用于过滤部分图像，生成光照特定条件下的图像（PPT 原话）

除运算：校正照明的非均匀性。（PPT 原话）

逻辑运算：

求反：获得一个阴图像（PPT 原话）

异或：获得相交子图像（PPT 原话）

或：合并子图像（PPT 原话）

与：求两个子图像的相交子图像（PPT 原话）

## 旋转变换

宋涛说：几何变换中旋转变换知道怎么计算。

然而计算题已经说了上述那些，此题作为计算题概率不大。

**旋转变换：绕(x0,y0)点旋转α度**

设： $a(x,y)-x_0 = (x-x_0) * \cos(\alpha) - (y-y_0) * \sin(\alpha);$

$b(x,y)-y_0 = (x-x_0) * \sin(\alpha) + (y-y_0) * \cos(\alpha);$

**用齐次矩阵表示：**

$$\begin{bmatrix} a(x,y) \\ b(x,y) \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha & x_0 \\ \sin \alpha & \cos \alpha & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x-x_0 \\ y-y_0 \\ 1 \end{bmatrix}$$

## 空间域灰度变换

M 哥记录宋涛：非线性变换的两个特点（两个非线性变换的特点）。



Presented with xmind AI

直方图均衡化，减少灰度层次。

我记录的 ST 说的：平滑和锐化根据例子来。



M 哥记录的 ST 说的：5 章考计算题(上文已经提到)

## 平滑

去除图像中一些不重要的细节减小噪声。

线性平滑滤波：模板卷积

非线性平滑滤波：模板排序

$$\text{例: } f_1(x, y) = \begin{bmatrix} 100 & 100 & 100 \\ 100 & 200 & 100 \\ 100 & 100 & 100 \end{bmatrix} \quad f_2(x, y) = \begin{bmatrix} 100 & 100 & 100 \\ 100 & 140 & 100 \\ 100 & 100 & 100 \end{bmatrix}$$

取阈值  $T=40$ ，分别对  $f_1$  和  $f_2$  进行滤波

$$g_1(x, y) = \begin{bmatrix} 100 & 100 & 100 \\ 100 & 111 & 100 \\ 100 & 100 & 100 \end{bmatrix} \quad g_2(x, y) = \begin{bmatrix} 100 & 100 & 100 \\ 100 & 140 & 100 \\ 100 & 100 & 100 \end{bmatrix}$$

(本人认为计算题不可能再考此题)

## 锐化

原理：采用差分或梯度计算来检测边缘，在原图上叠加该检测效果有勾边效果。(PPT原话)

## 图像分割

本人和 M 哥都记录：

阈值分割法，哪种图像更适合，哪种不适合，要说出区别



### 7.1.1 阈值分割的原理

设输入图像为  $f(x, y)$ ，输出图像为  $f'(x, y)$ ，则阈值分割过程如下：首先确定一个阈值  $T$ ，对于图像中的每个像素，若其灰度值大于  $T$ ，则将其置为目标点(值为1)，否则置为背景点(值为0)，或者相反，从而将图像分为目标区域与背景区域。用公式可表示为

$$f'(x, y) = \begin{cases} 1 & , \quad f(x, y) \geq T \\ 0 & , \quad f(x, y) < T \end{cases}$$

$$\text{或 } f'(x, y) = \begin{cases} 1 & , \quad f(x, y) \leq T \\ 0 & , \quad f(x, y) > T \end{cases}$$

全局阈值：当图像目标与背景之间具有高对比度时，利用全局阈值可以成功地分割图像。目标与背景之间具有鲜明的对比，直方图表现出双峰性质（PPT 原话）

确定全局阈值：极小点阈值法、迭代阈值法、最优阈值法、OTSU 阈值法、最大熵法、 $p$  参数法等（PPT 原话）

动态阈值(自适应阈值)：由于光照不均匀等因素的影响，图像背景的灰度值并不恒定，目标与背景的对比度在图像中也会有变化，此时应采用动态阈值。

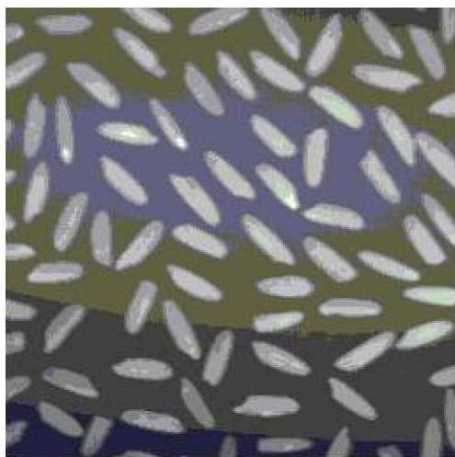
在实际应用中，也可以将目标像素置为255，背景像素置为0，或者相反。当图像中含有多个目标且灰度差别较大时，可以设置多个阈值实现多阈值分割。多阈值分割可表示为

$$g(x,y)=\begin{cases} 0 & f(x,y)\leq T_1 \\ k & T_k < f(x,y)\leq T_{k+1} \\ 255 & f(x,y)>T_m \end{cases} \quad k=1,2,\dots,K-1$$

式中： $T_k$ 为一系列分割阈值； $k$ 为赋予每个目标区域的标号； $m$ 为分割后的目标区域数减1。

可能会这么考：（本题目纯属猜测）

下面两张图，分别适合选用哪种阈值进行图像分割，分析原因。



## 第八章（25/60）压缩编码系统评价



### 8.1.4 压缩编码系统评价

信息熵具有有个基本性质:

- (1) 非负性, 即  $H(A) \geq 0$
- (2) 当信源字符以等概率分布时, 其熵值最大, 即

$$p_1 = p_2 = \cdots = p_M = \frac{1}{M} \text{ 时, } H_{\max}(A) = \log_2^M$$



### 8.1.4 压缩编码系统评价

(2) 平均码字长度:

$\beta_i$  是第*i*种字符的长度(二进制代码的位数),  $P_i$  是第*i*种字符出现的概率, 则平均码长为:

$$L = \sum_{i=1}^M \beta_i P_i$$



#### 8.1.4 压缩编码系统评价

##### (3) 编码效率

$$\eta = \frac{H}{L} \times 100\%$$

##### (4) 压缩比

$$r = \frac{n}{L}$$

少年，当你看到这里的时候，《光电图像处理》不说考百分，至少能及格了。祝愿不在补考考场和重修教室里面看到你。

希望你在通过考试之余，能有意识地在生活的方方面面运用这些知识，当你看见别人在抖音短视频里面使用美颜特效，能顺口说出滤波的时候，也许这门课的意义就达到了。

Now, open your new eyes!