

# 第一章：绪论 + 图像数字化

---

授课老师：李厚强 ([lihq@ustc.edu.cn](mailto:lihq@ustc.edu.cn))

周文罡 ([zhwg@ustc.edu.cn](mailto:zhwg@ustc.edu.cn))

李 礼 ([lil1@ustc.edu.cn](mailto:lil1@ustc.edu.cn))

胡 洋 ([eeeyhu@ustc.edu.cn](mailto:eeeyhu@ustc.edu.cn))



# 课程信息

---

## □ 上课时间：

- 周四/周五 6~9节（14:00~17:30）

## □ 课程成绩

- 期末考试（闭卷）占60%-70%
- 平时作业（共2-3次）占20%-30%
- 课堂测试占10%

## □ 参考教材

- 《Computer Vision: Algorithms and Applications》第二版

## □ 课程主页：<https://ustc-dia.github.io>

- 课件下载
- 发布作业、考试等通知



# 绪论

---

- 图像基础知识
- 图像分析概论
- 主要内容和安排

# 图像基础知识

## 图像

用各种**观测系统**以不同形式和手段观测**客观世界**而获得的，可以直接或**间接作用于人/机器**并进而产生视知觉的实体，包括但不限于照片、绘图、动画、文档等。简单来说，图像就是所有具有视觉效果的画面

**Deepseek:** 通过绘画、摄影、扫描或数字技术等方式，在二维平面上创建的视觉表示或视觉印象。它本质上是将现实世界或想象中的场景、物体、人物或抽象概念，转化为视觉形式的结果

## 图像和信息

图像是一种承载信息的媒介或媒体，人类从外界（客观世界）获得的信息约有80%来自视觉系统

## 图像表示

2-D数组或矩阵 :  $f(x, y)$

$x$  、  $y$ : 2-D空间 $XY$ 中坐标点的位置

$f$ : 代表图像在 $(x, y)$ 的性质  $F$  的数值

图像矩阵 $F$ 的性质 : 可对应不同物理量, 灰度图像里用灰度表示

## 图像单元

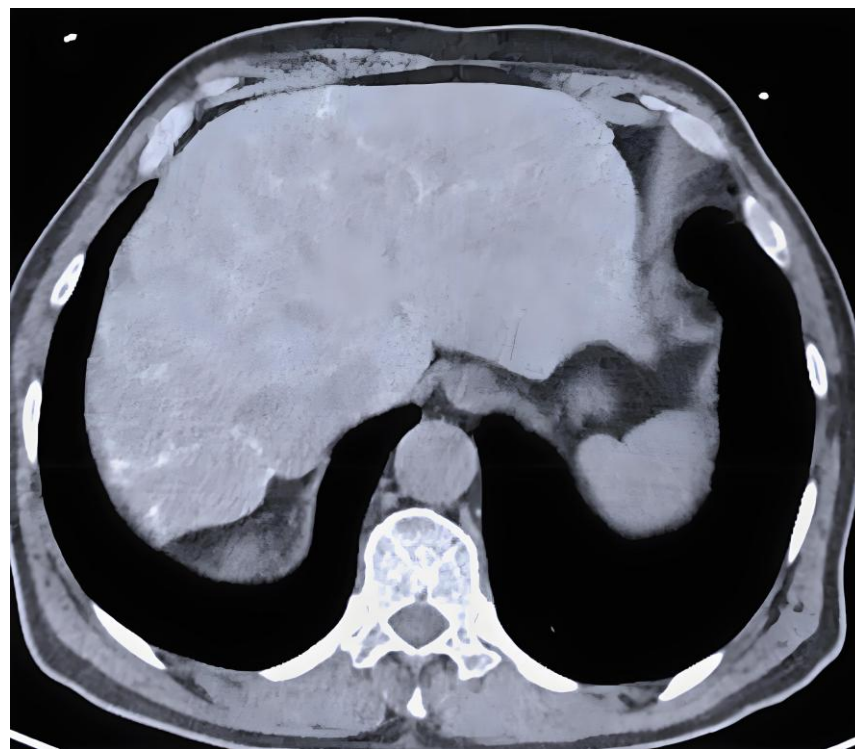
一幅图像是许多图像单元的集合体

2-D图像： 像素（picture element）  
英文里常用pixel表示

3-D图像： 体素（volume element）  
英文里常用voxel表示

$$f(x, y) \rightarrow f(x, y, z), f(x, y, t)$$

# 图像基础知识





# 绪论

---

- 图像基础知识
- 图像分析概论
  - 图像分析定义和研究内容
  - 图像分析系统
- 主要内容和安排

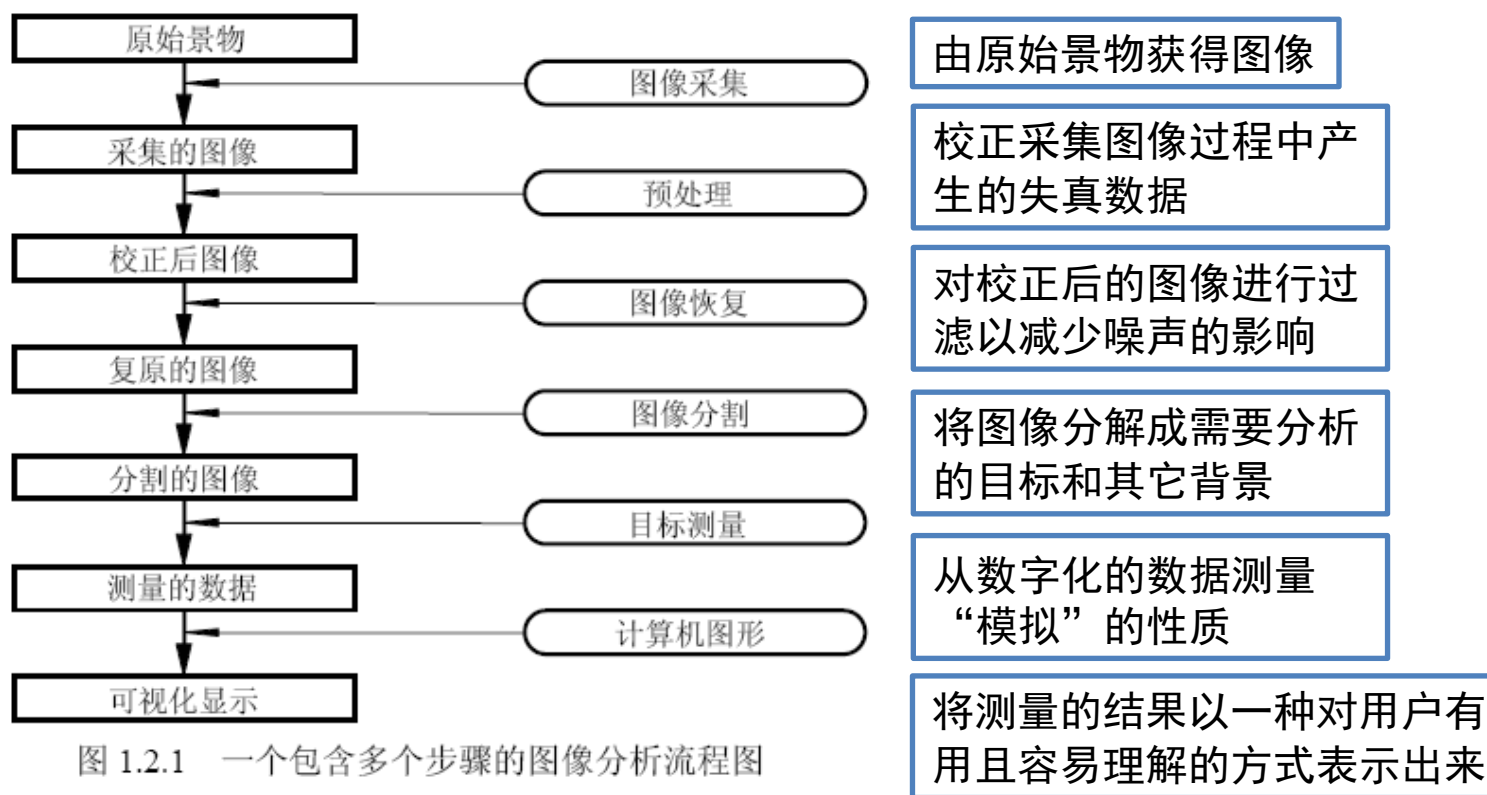


# 图像分析定义和研究内容

## □ 图像分析的定义

- 图像分析的中心问题是：将具有若干兆字节的灰度图像或彩色图像简化成只有若干个有意义和有用的数字[Russ 2006]
- 从图像出发，对其中感兴趣目标进行**检测**、**提取**、**表达**、**描述**和**测量**，从而获取客观信息，输出数据结果的过程和技术
- Deepseek：从图像中提取有意义的信息和数据的过程。它超越了单纯“看”图像，而是通过一系列方法和技术来量化和理解图像中的内容

## 图像分析应用流程示例



## ☐ Segment Anything

<https://segment-anything.com/demo#>

# 绪论

---

## □ 图像和图像工程

## □ 图像分析概论

### ■ 图像分析定义和研究内容

### ■ 图像分析系统

## □ 主要内容和安排



# 主要内容和安排

- 绪论、图像数字化, 图像变换, 形态学 (2次)
- 图像预处理, 图像增强, 图像恢复 (1次)
- 深度学习(CNN, RNN, Transformer):图像增强和图像恢复 (1次)
- 图像表达: 边缘检测、局部特征、全局特征、图像预训练 (2次)
- 图像分割:经典+深度学习 (1次)
- 图像识别:经典+深度学习(FasterRCNN, YoLO, DETR)(2次)
- 图像生成:概率图模型+深度学习生成 (1次)
- 运动分析:光流方程等+基于深度学习的运动分析 (1次)
- 目标跟踪:经典+深度学习 (1次)
- 3D图像分析:SLAM, Structure from motion, depth estimation, 3D reconstruction, point cloud analysis (2次)
- Image-based rendering: Nerf, Gaussiansplatting (1次)

## □ 数字图像基础

- 成像过程
- 数字化
- 数字图像基础

## □ 数字图像基础

### ■ 成像过程

✓ 视觉过程

✓ 成像变换

✓ 视觉系统

### ■ 数字化

### ■ 数字图像基础

# 视觉过程

## □ 光学过程

- 视网膜成像为倒影
- 晶状体可以改变形状，调节聚焦焦距（范围：14~17mm），远离变扁，靠近加厚
- 中央凹：视网膜中视觉（辨色力、分辨力）最敏锐的区域

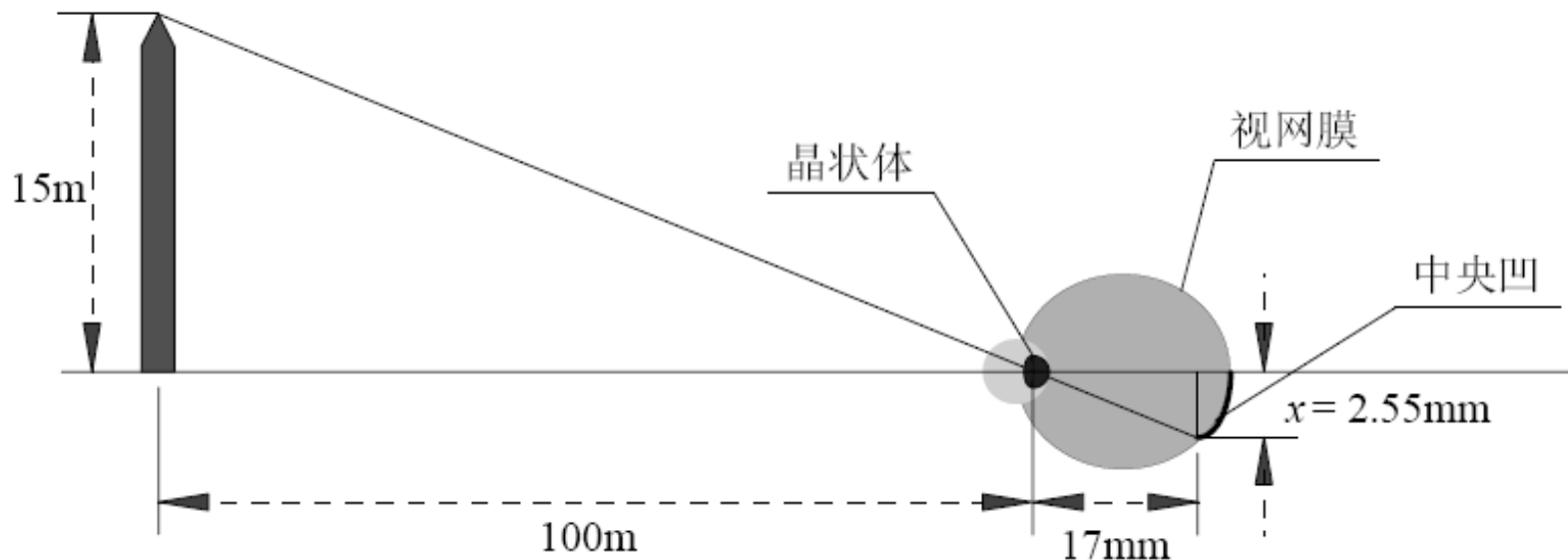


图 2.1.1 人眼水平横截面示意图

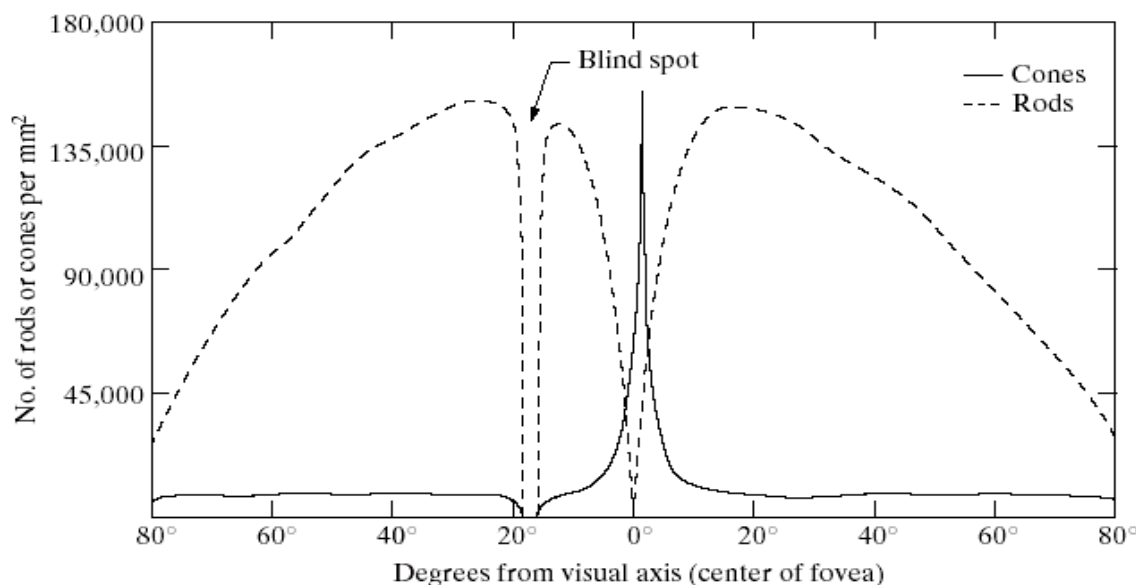
$$15 / 100 = 2.55 / 17$$



## □ 化学过程

### ■ 视网膜上布满两类光感受器：锥细胞和柱细胞

- ✓ 锥细胞(cone): 数量少 (600~700万), 对颜色高度敏感
- ✓ 锥细胞视觉: 明视觉或亮光视觉(photopic vision)
- ✓ 柱细胞(rod): 数量多 (7500~15000万), 分辨率比较低不感受颜色, 对低照度较敏感
- ✓ 柱细胞视觉: 暗视觉或微光视觉 (scotopic vision)



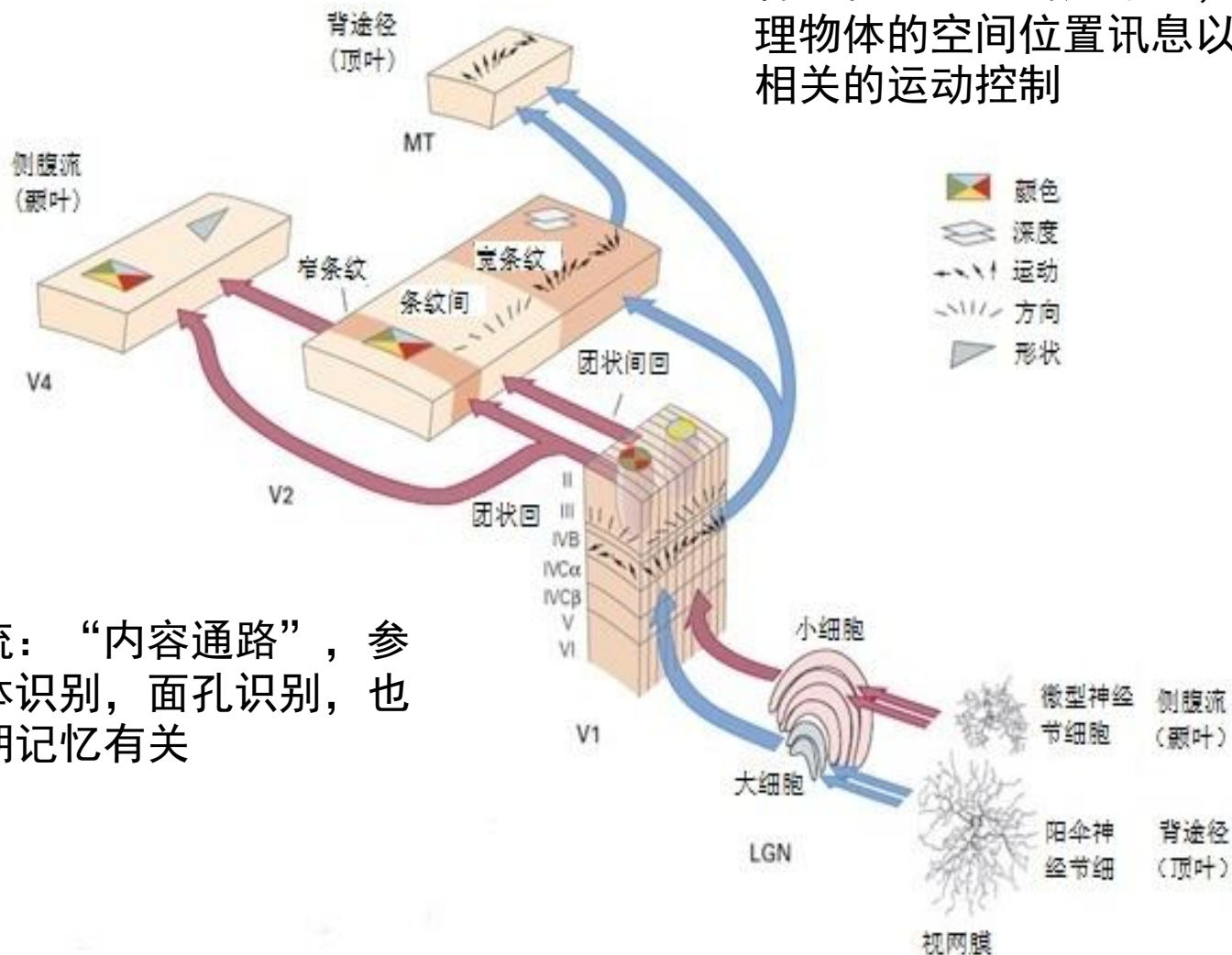
# 视觉过程

## □ 神经处理过程

- 每个视网膜接收单元都与一个神经元细胞借助**突触**(synapse)相连
- 每个神经元细胞借助其它的**突触**与其它细胞连接，从而构成光神经(optical nerve)网络
- 光神经进一步与大脑中的侧区域(side region of the brain)连接，并到达大脑中的纹状皮层(striate cortex, V1)和纹外皮层(extrastriate cortex, V2, V3, V4, V5)
- 对光刺激产生的响应经过一系列处理最终形成关于场景的表像，从而将对光的感受转化为对景物的知觉

# 视觉过程：脑视觉信息处理通路

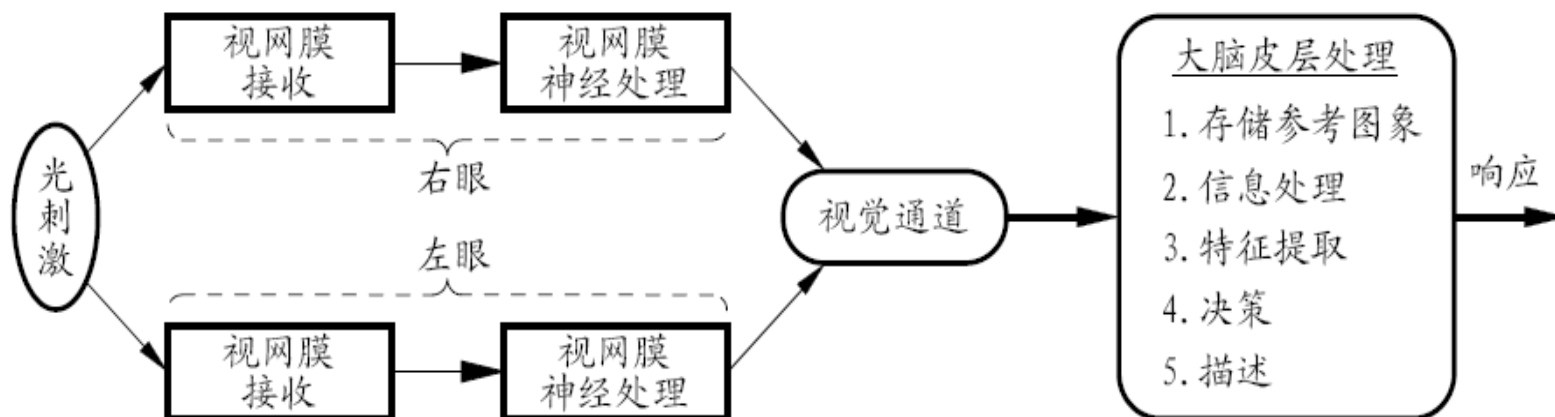
背途径：“空间通路”，处理物体的空间位置讯息以及相关的运动控制



腹侧流：“内容通路”，参与物体识别，面孔识别，也与长期记忆有关

# 视觉过程

## □ 整体视觉过程

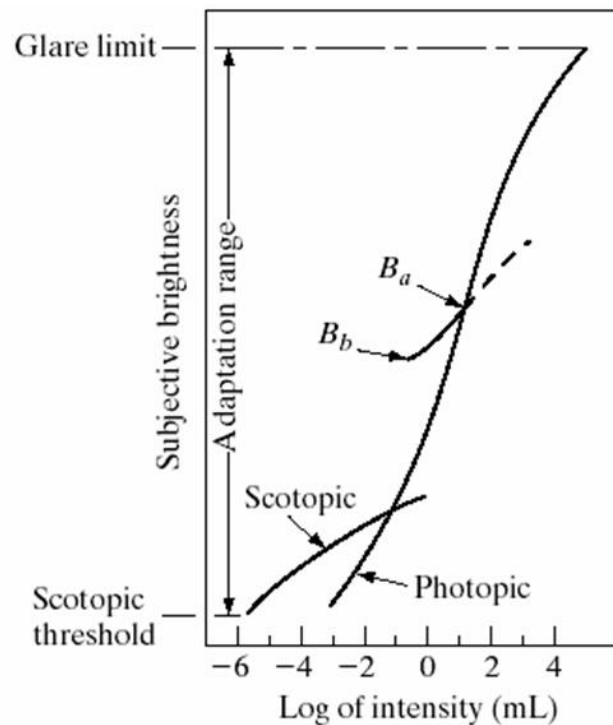
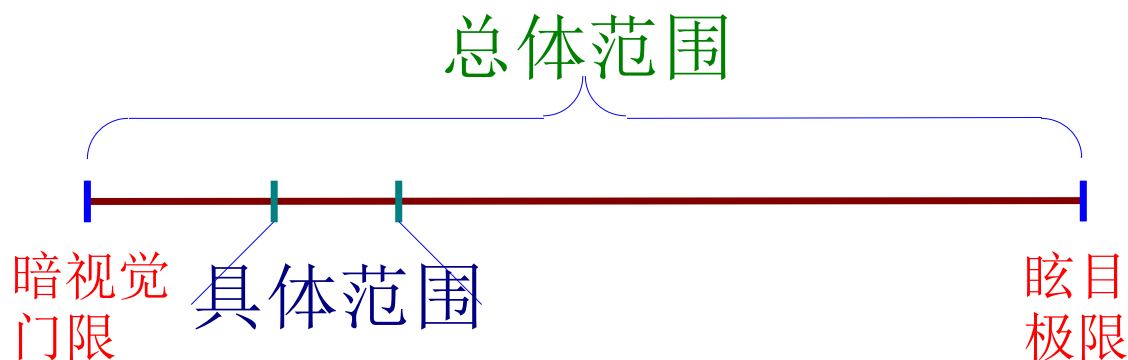


视觉 = “视” + “觉”

# 视觉系统对光的感知特点

## □ 人眼适应的亮度范围

- 总体范围大：从暗视觉门限到眩目极限之间的范围在 $10^{10}$ 量级
- 具体范围小：一般范围在 $10^2$ 量级



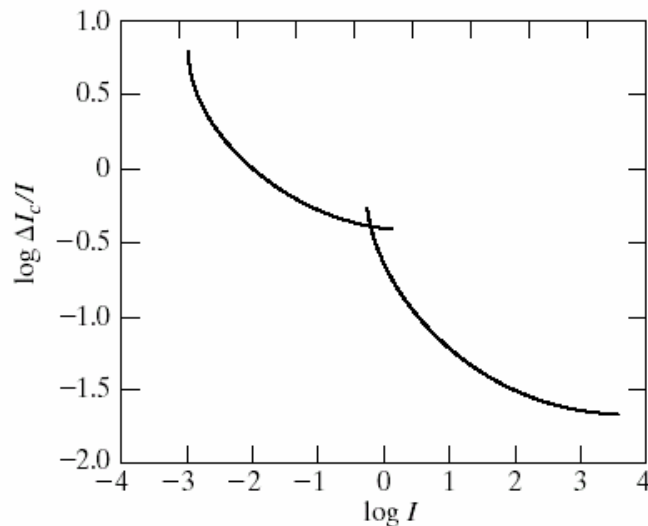
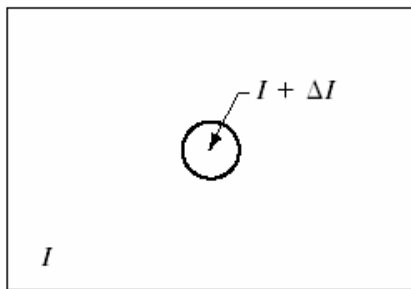
# 视觉系统对光的感知特点

## □ 亮度变化的感知

- 人类视觉系统对**亮度变化**的感知比对亮度本身要敏感
- 人类视觉系统对光强度的响应不是线性的，而是**对数**形式的

## □ 亮度适应和分辨

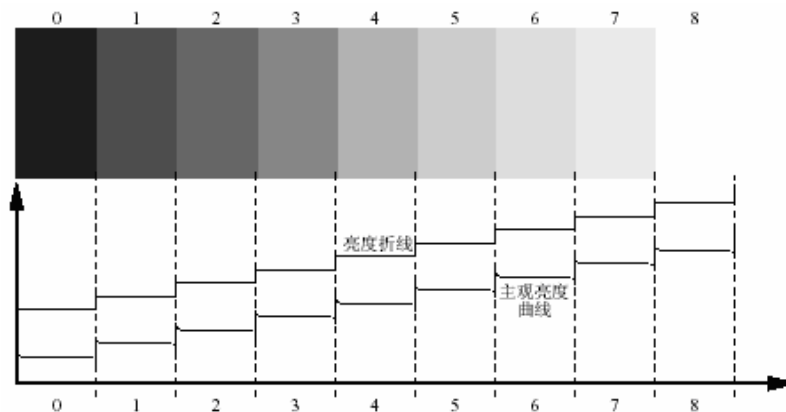
- 低照度级别，亮度分辨率较差；高照度级别，亮度分辨率较高
- 韦伯率



# 视觉系统对光的感知特点

## □ 亮度错觉

- 马赫带：在亮度变化部分附近的亮区和暗区分别感知到一条更暗和更亮的条带



- 同时对比度

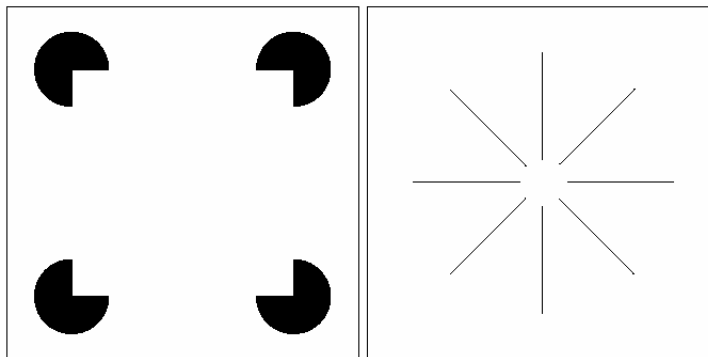


# 视觉系统对光的感知特点

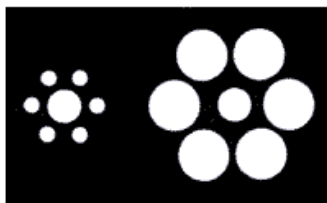
## □ 视觉错觉

- 由视觉系统特性特性、心理、知识等方面有关，形成机制尚未完全了解

填充

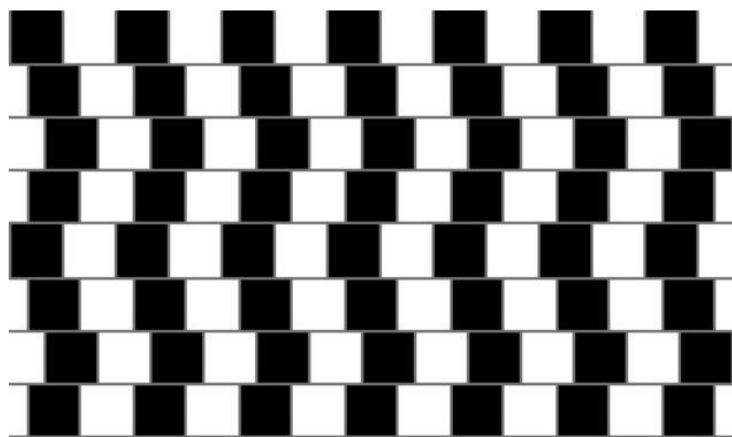
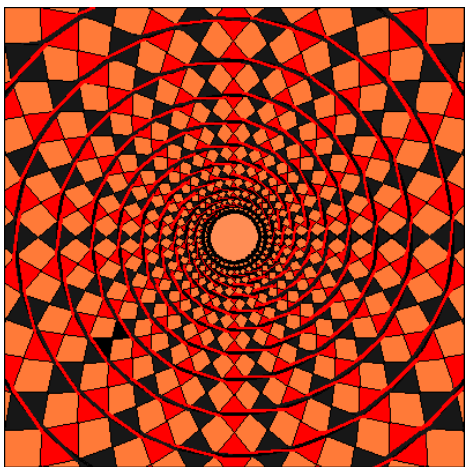


几何特性





# 视觉系统对光的感知特点



Rubin Vase





# 成像变换

## □ 投影成像

- 将3-D客观场景投影到2-D图像平面

## □ 成像过程

- 两个坐标系统
  - ✓ 世界坐标系  $XYZ$
  - ✓ 图像平面  $xy$

从  $XYZ$  到  $xy$

# 成像变换

## □ 成像过程

### ■ 三维坐标系

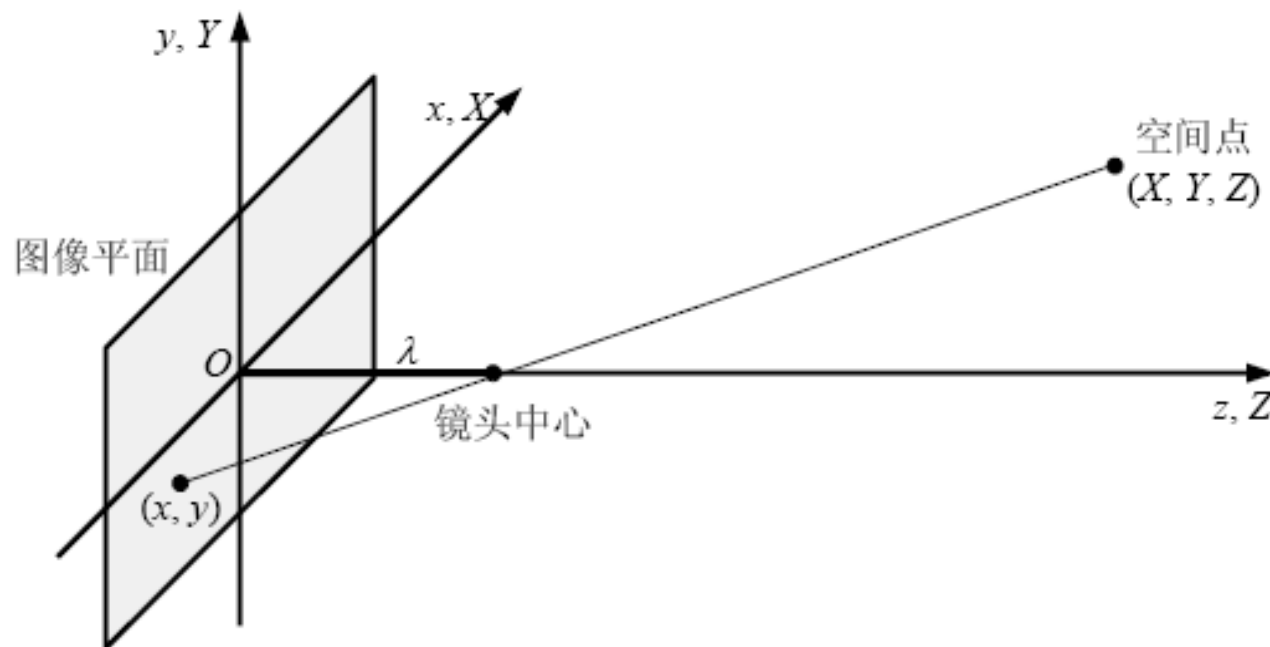


图 2.2.1 投影变换成像示意图

3-D点投影后的  
图像平面坐标

$$x = \frac{\lambda X}{\lambda - Z}$$

$$y = \frac{\lambda Y}{\lambda - Z}$$

非线性投影等式  
(分母含变量Z)

# 成像变换

## □ 齐次坐标

■ 可用来将前述非线性（分母中含变量 $Z$ ）等式表示成线性矩阵形式

✓ 笛卡尔坐标:  $\mathbf{w} = [X \quad Y \quad Z]^T$

✓ 齐次坐标:  $\mathbf{w}_h = [kX \quad kY \quad kZ \quad k]^T$  ( $k$  为任意非零常数)

齐次坐标  $\Rightarrow$  笛卡尔坐标: 用第4个坐标量去除前3个坐标量

- 可以表示两条平行线在无穷远处相交
- 可以使用矩阵乘法运算表示仿射变换

# 成像变换

## □ 齐次坐标透视变换

$$\mathbf{c}_h = \mathbf{P}\mathbf{w}_h = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/\lambda & 1 \end{bmatrix} \begin{bmatrix} kX \\ kY \\ kZ \\ k \end{bmatrix} = \begin{bmatrix} kX \\ kY \\ kZ \\ -kZ/\lambda + k \end{bmatrix}$$

齐次形式的  
摄像机坐标

用第4项分别去除前3个项

$$\mathbf{c} = \begin{bmatrix} x & y & z \end{bmatrix}^T = \begin{bmatrix} \frac{\lambda X}{\lambda - Z} & \frac{\lambda Y}{\lambda - Z} & \frac{\lambda Z}{\lambda - Z} \end{bmatrix}^T$$

# 成像变换

## □ 逆投影变换

- 根据2-D图像坐标来确定3-D客观景物的坐标

$$w_h = P^{-1} c_h \quad P^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/\lambda & 1 \end{bmatrix}$$

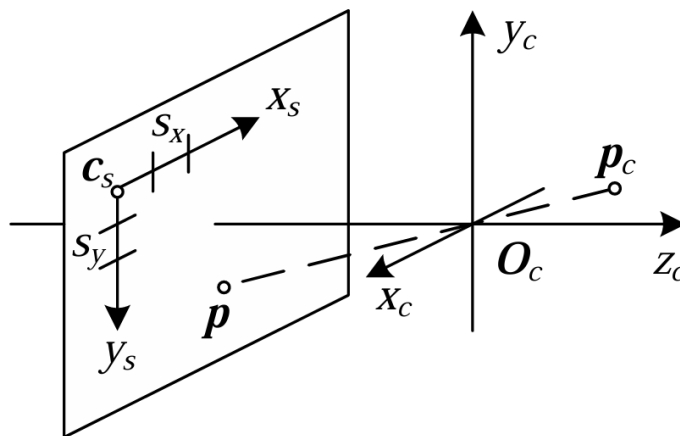
- 从3D客观场景映射到图像平面的投影变换是一个一到多的变换
- 每一个2D图像平面上的坐标点对应过空间点 (X, Y, Z) 的一条直线

# 成像变换

□ 进一步，如何将投影形成的坐标 $(x, y)$ 变成整数坐标 $(x_s, y_s)$ ?

$$\mathbf{p} = \begin{bmatrix} \mathbf{R}_s & \mathbf{c}_s \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \mathbf{M}_s \bar{\mathbf{x}}_s$$

相机内参矩阵  $\tilde{\mathbf{x}}_s = \alpha \mathbf{M}_s^{-1} \mathbf{p}_c = \mathbf{K} \mathbf{p}_c$

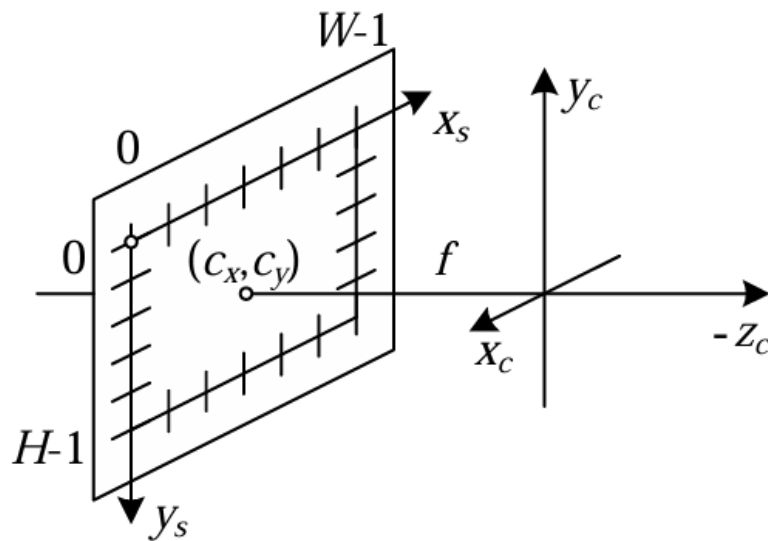


# 成像变换

## □ 相机内参矩阵

- 现代制造技术使得图像平面之间的轴间倾斜可以忽略不计

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$





# 成像变换

## □ 相机外参矩阵

- 相机在世界坐标系中的位置和方向
- 外参矩阵通常由一个3x3的旋转矩阵和一个3x1的平移向量组成，形式为

$$[R \mid T] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}$$

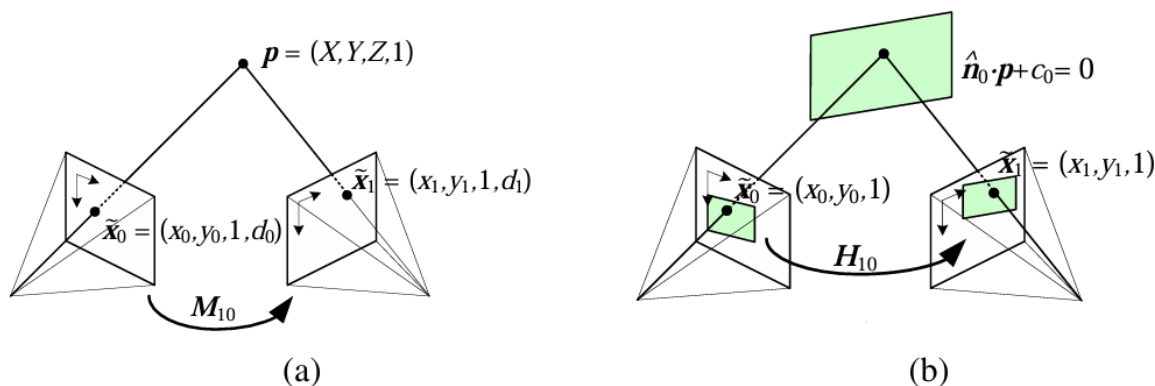
# 成像变换

## □ 相机与相机之间的变换

$$\tilde{\mathbf{x}}_0 \sim \tilde{\mathbf{K}}_0 \mathbf{E}_0 \mathbf{p} = \tilde{\mathbf{P}}_0 \mathbf{p}.$$

$$\mathbf{p} \sim \mathbf{E}_0^{-1} \tilde{\mathbf{K}}_0^{-1} \tilde{\mathbf{x}}_0$$

$$\tilde{\mathbf{x}}_1 \sim \tilde{\mathbf{K}}_1 \mathbf{E}_1 \mathbf{p} = \tilde{\mathbf{K}}_1 \mathbf{E}_1 \mathbf{E}_0^{-1} \tilde{\mathbf{K}}_0^{-1} \tilde{\mathbf{x}}_0 = \tilde{\mathbf{P}}_1 \tilde{\mathbf{P}}_0^{-1} \tilde{\mathbf{x}}_0 = \mathbf{M}_{10} \tilde{\mathbf{x}}_0.$$



**Figure 2.12** A point is projected into two images: (a) relationship between the 3D point coordinate  $(X, Y, Z, 1)$  and the 2D projected point  $(x, y, 1, d)$ ; (b) planar homography induced by points all lying on a common plane  $\hat{\mathbf{n}}_0 \cdot \mathbf{p} + c_0 = 0$ .

## □ 数字图像基础

### ■ 成像过程

### ■ 数字化

✓ 图像传感流水线

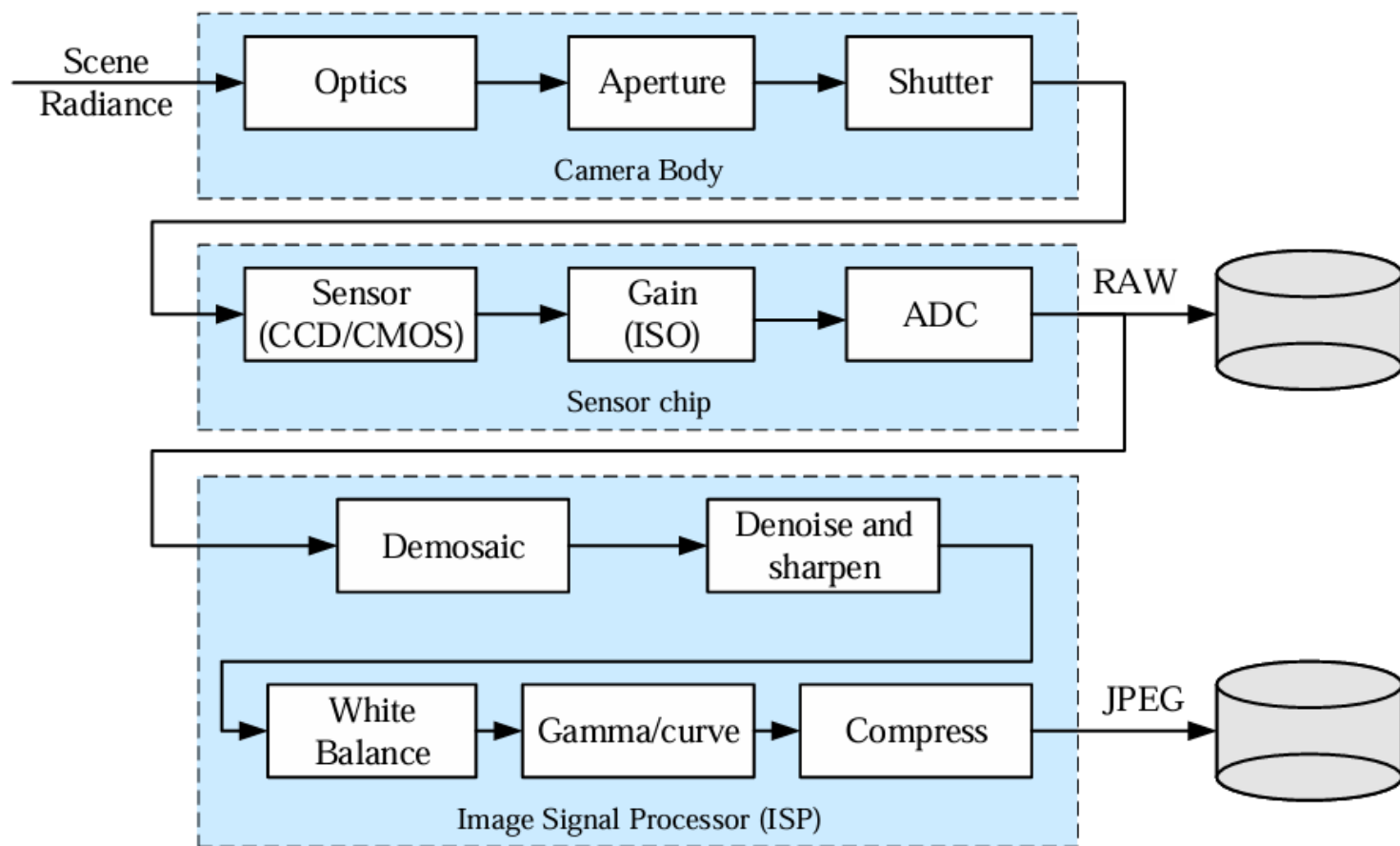
✓ 图像颜色空间

✓ 图像压缩

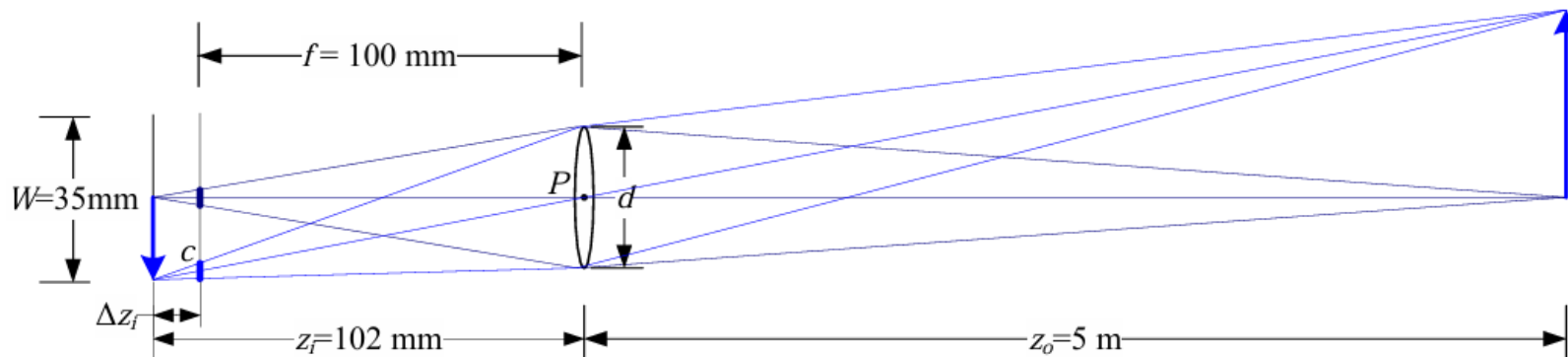
### ■ 数字图像基础

# 图像传感流水线

## □ 图像传感流水线



## 相机的光学过程

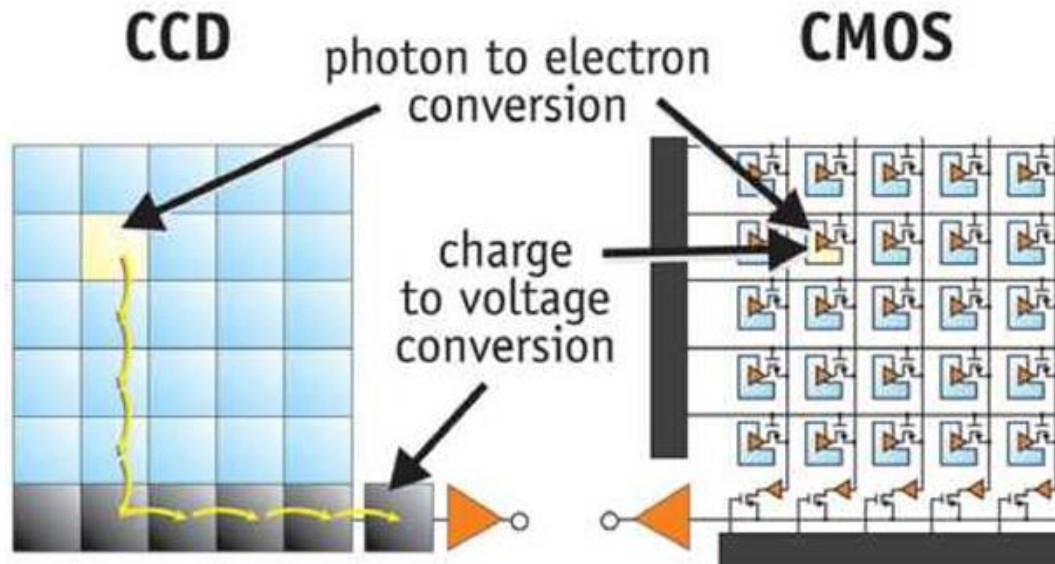


**Figure 2.19** A thin lens of focal length  $f$  focuses the light from a plane at a distance  $z_o$  in front of the lens onto a plane at a distance  $z_i$  behind the lens, where  $\frac{1}{z_o} + \frac{1}{z_i} = \frac{1}{f}$ . If the focal plane (vertical gray line next to  $c$ ) is moved forward, the images are no longer in focus and the circle of confusion  $c$  (small thick line segments) depends on the distance of the image plane motion  $\Delta z_i$  relative to the lens aperture diameter  $d$ . The field of view (f.o.v.) depends on the ratio between the sensor width  $W$  and the focal length  $f$  (or, more precisely, the focusing distance  $z_i$ , which is usually quite close to  $f$ ).

# Sensor

## □ CCD and CMOS

- CCD：通过光电效应将光信号转换为电荷，电荷逐行传输到读出电路进行信号处理；电荷传输过程中需要较高的电压，功耗较大；高精度成像领域，如天文摄影、科学仪器、高端工业检测
- CMOS：每个像素点都集成了光电二极管和信号处理电路，光信号直接转换为电信号；信号读取方式为并行读取，功耗较低；消费电子领域，如智能手机、数码相机、安防监控

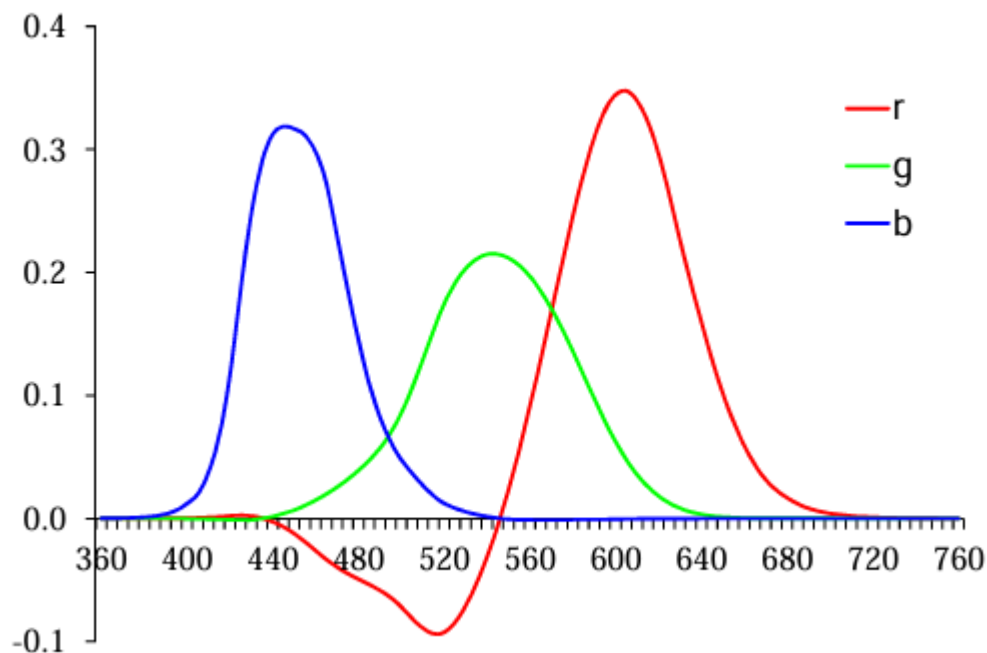


# Color Space



## □ RGB颜色空间

- R: 700.0nm; G: 546.1nm; B: 435.8nm
- RGB三原色的存在是人类视觉系统通过三种视锥细胞感知RGB三种颜色的结果

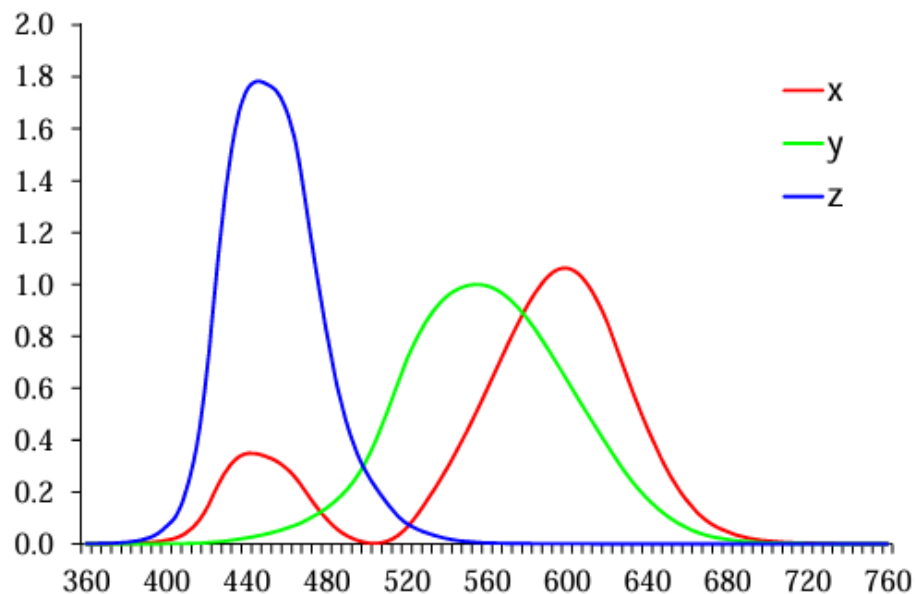


# Color Space

## □ XYZ颜色空间

- RGB表示所有波长有负数

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$





# Color Space

## □ YCbCr颜色空间

### ■ 广泛应用于视频压缩领域

$$Y = 0.299R + 0.587G + 0.114B$$

$$Cb = -0.169R - 0.331G + 0.500B$$

$$Cr = 0.500R - 0.419G - 0.081B$$

BT.601  
标清视频

$$Y = 0.2126R + 0.7152G + 0.0722B$$

$$Cb = -0.1146R - 0.3854G + 0.5000B$$

$$Cr = 0.5000R - 0.4542G - 0.0458B$$

BT.709  
高清视频

# Color filter arrays

## □ Bayer RGB pattern

- The reason that there are twice as many green filters as red and blue is because the luminance signal is mostly determined by green

G	R	G	R
B	G	B	G
G	R	G	R
B	G	B	G

rGb	Rgb	rGb	Rgb
rgB	rGb	rgB	rGb
rGb	Rgb	rGb	Rgb
rgB	rGb	rgB	rGb

# Compression

- All color video and image compression algorithms start by converting the signal into YCbCr (or some closely related variant), so that they can compress the luminance signal with higher fidelity than the chrominance signal
- 预测、变换、量化、熵编码、滤波



**Figure 2.34** *Image compressed with JPEG at three quality settings. Note how the amount of block artifact and high-frequency aliasing (“mosquito noise”) increases from left to right.*

## □ 数字图像基础

- 成像过程
- 数字化
- 数字图像基础
  - ✓ 空间分辨率和比特深度
  - ✓ 像素的邻域与距离
  - ✓ 距离变换

# 空间分辨率和比特深度

## □ 数字图像

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ f(1,0) & f(1,1) & \cdots & f(1,M-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{bmatrix}$$

图像（水平）尺寸  $M$ :  $M = 2^m$

图像（垂直）尺寸  $N$ :  $N = 2^n$

像素灰度级数  $G$  ( $k$ -bit):  $G = 2^k$

图像所需的位数  $b$ :  $b = M \times N \times k$

# 空间分辨率和比特深度

## □ 数字图像

$$b = M \times N \times k$$

存储1幅 $32 \times 32$ , 16个灰度级的图需要 4 096 bit

存储1幅 $128 \times 128$ , 64个灰度级的图需要 98 304 bit

存储1幅 $512 \times 512$ , 256个灰度级的图需要 2 097 152 bit

## □ 常见的分辨率 (P: Progressive, 代表逐行扫描)

- 720P:  $1280 \times 720$  (HD, 高清)
- 1080P:  $1920 \times 1080$  (full HD, 全高清)
- 2K:  $2560 \times 1440$ ,  $2048 \times 1536$ ,  $2560 \times 1600$
- 4K:  $4096 \times 2160$ ,  $4096 \times 3112$ ,  $3656 \times 2664$ ,  $3840 \times 2160$

# 空间分辨率和比特深度

## □ 图像空间分辨率变化所产生的效果

(515x512~16x16)

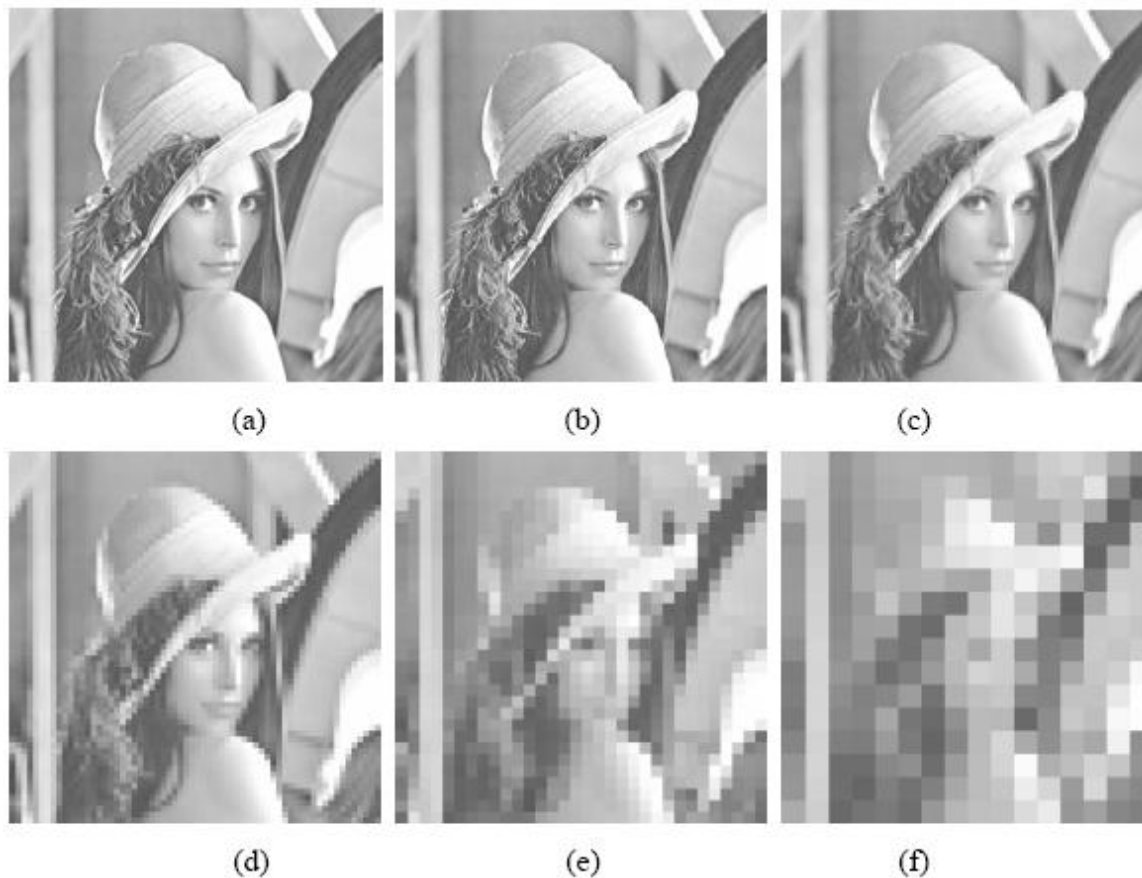


图 2.4.2 图像空间分辨率变化所产生的效果

# 空间分辨率和比特深度

## □ 图像幅度分辨率变化所产生的效果

(256, 64, 16, 8, 4, 2)

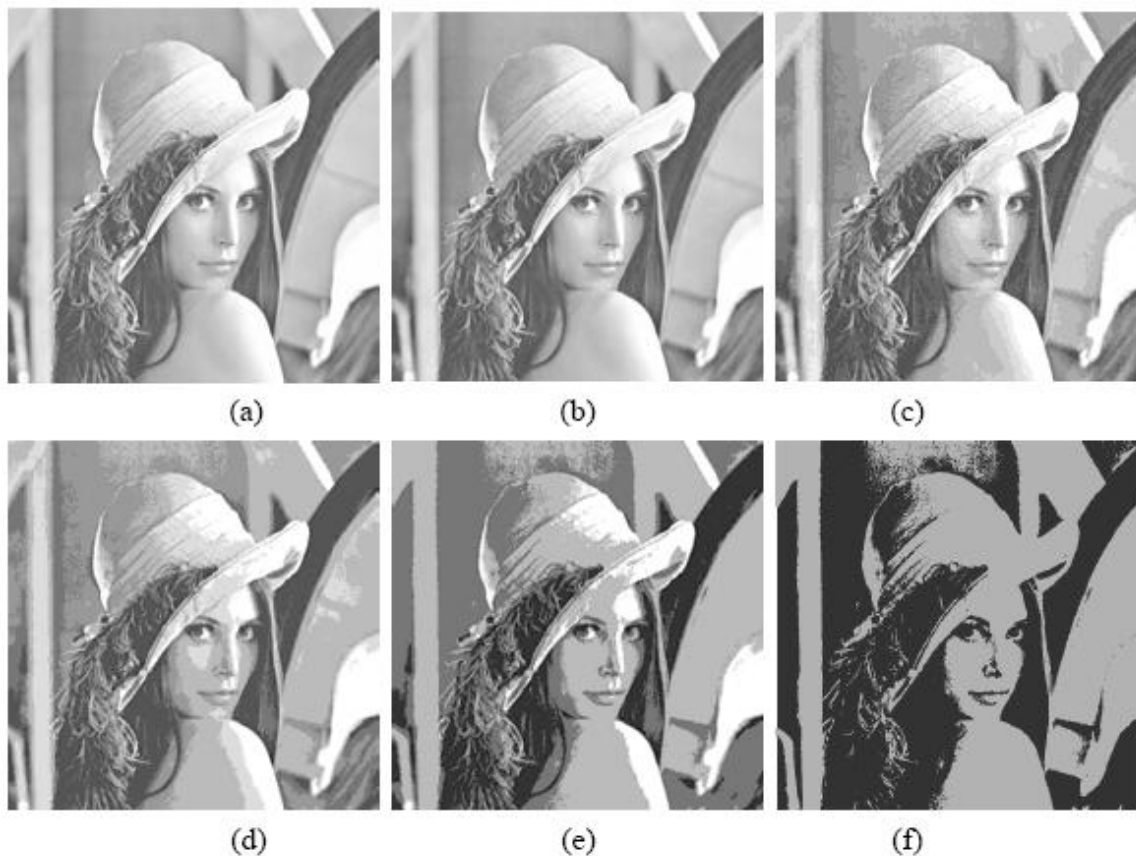


图 2.4.3 图像幅度分辨率变化所产生的效果



# 空间分辨率和比特深度

## □ 空间和幅度分辨率同时变化所产生的效果

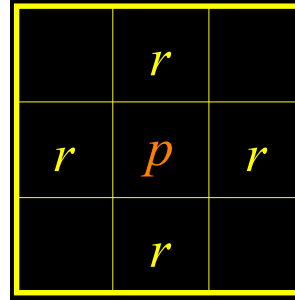


256x256,128;  
181x181,64;  
128x128,32;  
90x90,16;  
64x64,8;  
45x45,4

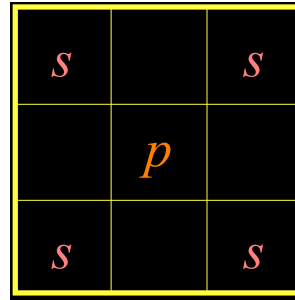
图 2.4.4 图像空间和幅度分辨率同时变化所产生的效果

# 像素的邻域

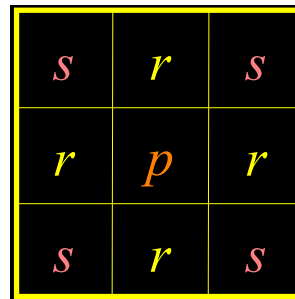
□ 4-邻域—— $N_4(p)$



□ 对角邻域—— $N_D(p)$



□ 8-邻域—— $N_8(p)$





# 像素间的邻接，连接和连通

## □ 连接和连通

(adjacency, 邻接) **vs.** (connectivity, 连接)

邻接仅考虑像素间的空间关系

两个像素是否连接：

(1) 是否接触（邻接）

(2) 灰度值是否满足某个特定的相似准则（同在一个灰度值集合中取值）

# 像素间的邻接，连接和连通

## □ 3种连接

### ■ 4-连接

- ✓ 2个像素  $p$  和  $r$  在  $V$  中取值且  $r$  在  $N_4(p)$  中

### ■ 8-连接

- ✓ 2个像素  $p$  和  $r$  在  $V$  中取值且  $r$  在  $N_8(p)$  中

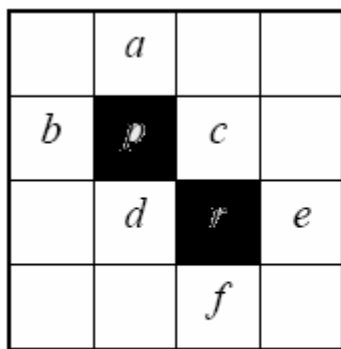
0	1	0
0	1	0
0	0	1

0	1	0
0	1	0
0	0	1

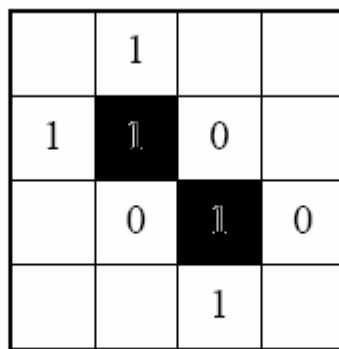
# 像素间的邻接，连接和连通

## □ 3种连接

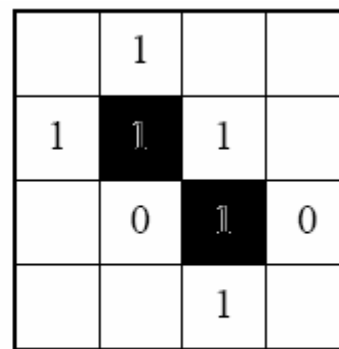
- m-连接（混合连接）：2个像素  $p$  和  $r$  在  $V$  中取值且满足下列条件之一
  - ✓  $r$  在  $N_4(p)$  中
  - ✓  $r$  在  $N_D(p)$  中且集合  $N_4(p) \cap N_4(r)$  不包含  $V$  中取值的像素



(a)



(b)



(c)

# 像素间的邻接，连接和连通

□ 混合连接的应用：消除8-连接可能产生的歧义性

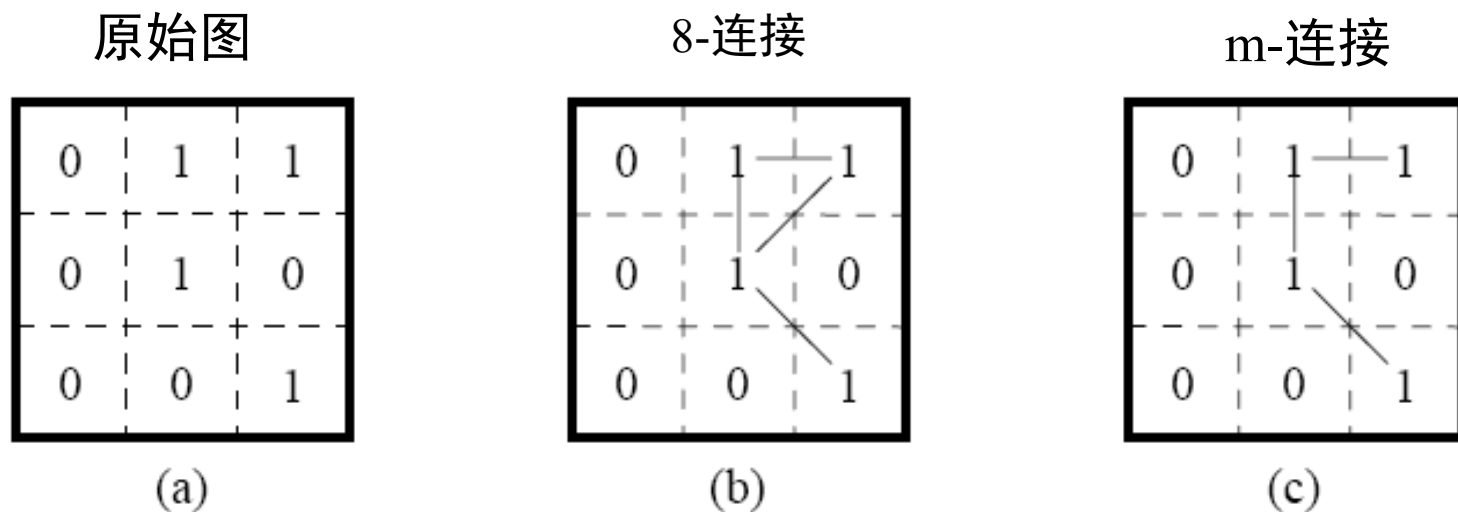


图 3.1.3 像素间的混合连接

# 像素间的邻接，连接和连通

## □ 连通

- 连接是连通的一种特例，两个像素之间有通路则称为连通

## □ 通路

- 由一系列依次连接的像素组成

从具有坐标 $(x, y)$ 的像素 $p$ 到具有坐标 $(s, t)$ 的像素 $q$ 的一条通路由一系列具有坐标 $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ 的独立像素组成。这里 $(x_0, y_0) = (x, y)$ ,  $(x_n, y_n) = (s, t)$ , 且 $(x_i, y_i)$ 与 $(x_{i-1}, y_{i-1})$ 连接, 其中 $1 \leq i \leq n$ ,  $n$ 为通路长度

4-连通, 8-连通  $\Rightarrow$  4-通路, 8-通路

# 像素间的邻接，连接和连通

## □ 像素集合的连接和连通

- 对2个图像子集  $S$  和  $T$  来说，如果  $S$  中的一个或一些像素与  $T$  中的一个或一些像素连接，则可以说2个图像子集  $S$  和  $T$  是连接的
- 完全在一个图像子集中的像素组成的通路上的像素集合构成该图像子集中的一个**连通组元**
- 如果  $S$  中只有1个连通组元，即  $S$  中所有像素都互相连通，则称  $S$  是一个**连通集**



# 像素间的距离

## □ 距离度量

### ■ 距离度量函数的定义

✓ 给定3个像素 $\mathbf{x}$ ,  $\mathbf{y}$ ,  $\mathbf{z}$ , 坐标分别为 $(x_1, x_2)$ ,  $(y_1, y_2)$ ,  $(z_1, z_2)$ , 如果下列条件满足, 则  $D$  是距离度量函数:

① 非负性:  $D(\mathbf{x}, \mathbf{y}) \geq 0$  ( $D(\mathbf{x}, \mathbf{y}) = 0$ , 当且仅当  $\mathbf{x} = \mathbf{y}$ )

② 对称性:  $D(\mathbf{x}, \mathbf{y}) = D(\mathbf{y}, \mathbf{x})$

③ 三角不等式:  $D(\mathbf{x}, \mathbf{z}) \leq D(\mathbf{x}, \mathbf{y}) + D(\mathbf{y}, \mathbf{z})$

■  $L_p$  范数 (norm) 及其诱导出的距离度量: 对于向量  $\mathbf{u} = [u_1, u_2, \dots, u_k]^T$

✓  $L_p$  范数:  $\|\mathbf{u}\|_p = (\sum_{i=1}^k |u_i|^p)^{\frac{1}{p}}$

✓  $L_p$  范数诱导的距离度量:  $D(\mathbf{u}, \mathbf{v}) = \|\mathbf{u} - \mathbf{v}\|_p = (\sum_{i=1}^k |u_i - v_i|^p)^{\frac{1}{p}}$

# 像素间的距离

## □ 距离量度函数

### ■ 欧氏 (Euclidean) 距离

✓ 等价于 $L_2$ 距离

$$D_e(\mathbf{x}, \mathbf{y}) = [(x_1 - y_1)^2 + (x_2 - y_2)^2]^{\frac{1}{2}}$$

### ■ 城区 (city-block) 距离

✓ 等价于 $L_1$ 距离

$$D_4(\mathbf{x}, \mathbf{y}) = |x_1 - y_1| + |x_2 - y_2|$$

### ■ 棋盘 (chessboard) 距离

✓ 等价于 $L_\infty$ 距离

$$D_8(\mathbf{x}, \mathbf{y}) = \max(|x_1 - y_1|, |x_2 - y_2|)$$

# 像素间的距离

## □ 距离量度函数

### 等距离轮廓图案

			3			
	2.8	2.2	2	2.2	2.8	
	2.2	1.4	1	1.4	2.2	
3	2	1	0	1	2	3
	2.2	1.4	1	1.4	2.2	
	2.8	2.2	2	2.2	2.8	
			3			

(a)

$D_E$

			3			
		3	2	3		
	3	2	1	2	3	
3	2	1	0	1	2	3
	3	2	1	2	3	
		3	2	3		
			3			

(b)

$D_4$

3	3	3	3	3	3	3
3	2	2	2	2	2	3
3	2	1	1	1	2	3
3	2	1	0	1	2	3
3	2	1	1	1	2	3
3	2	2	2	2	2	3
3	3	3	3	3	3	3

(c)

$D_8$

# 像素间的距离

## □ 用距离定义邻域

■ 考虑在空间点  $(x_p, y_p)$  的像素  $p$

✓ 4-邻域—— $N_4(p)$

$$N_4(p) = \{r \mid D_4(p, r) = 1\}$$

✓ 8-邻域—— $N_8(p)$

$$N_8(p) = \{r \mid D_8(p, r) = 1\}$$

# 距离变换

- 距离变换是一种特殊的变换，它把二值图像转化为灰度图像，其本身是一个全局概念，但可以借助对局部距离的计算而化整为零地进行
  - 定义和性质
  - 局部距离的计算
  - 离散距离变换的实现
  - 3-D距离变换

# 定义和性质

- 距离变换计算区域中的每个点与最接近的区域外的点之间距离，把二值图象变换为灰度图象
- 给定一个点集 $P$ 、一个子集 $B$ 以及满足测度条件的距离函数 $d(., .)$ ，对 $P$ 的距离变换中赋予点 $p \in P$ 的值为：

$$DT(p) = \min_{q \in B} \{d(p, q)\}$$

- 距离图 (map) 可用矩阵 $[DT(p)]$ 来表示

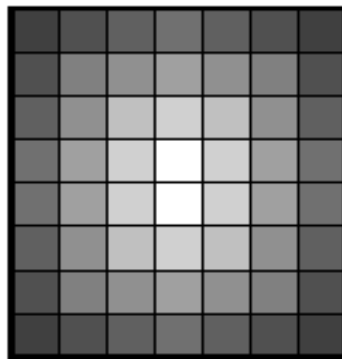


图 3.4.1 用灰度图表示的距离图

# 局部距离的计算

## □ 全局的操作，所以计算量会很大

性质：给定一个离散集合  $P$  和它的一个子集  $B$ ，用  $d$  表示计算距离图的离散距离函数。那么，对任何点  $p \in P^\circ$  (即  $p \in P-B$ )，存在  $p$  的一个邻域点  $q$  (即  $q \in N(p)$ )，使得在  $p$  的离散距离变换值  $DT(p)$  满足  $DT(p) = DT(q) + d(p, q)$

进一步，因为  $p$  和  $q$  互为邻接点，从  $p$  移动到  $q$  的长度为  $l(p, q) = d(p, q)$ 。这样，对任意点  $p \notin B$ ，离散距离变换可由  $DT(p) = \min\{DT(q') + l(p, q'), q' \in N(p)\}$  来刻画

# 局部距离的计算

## □ 用于局部距离扩展的模板

- 模板基于4-邻域定义且被用来扩展 $d_4$ 距离
- 模板基于8-邻域且被用来扩展 $d_8$ 距离或 $d_{a,b}$ 距离 ( $a=1, b=1$ )
- 模板基于16-邻域且被用来扩展 $d_{a,b,c}$ 距离

$\infty$	$a$	$\infty$
$a$	0	$a$
$\infty$	$a$	$\infty$

(a)

$b$	$a$	$b$
$a$	0	$a$
$b$	$a$	$b$

(b)

$\infty$	$c$	$\infty$	$c$	$\infty$
$c$	$b$	$a$	$b$	$c$
$\infty$	$a$	0	$a$	$\infty$
$c$	$b$	$a$	$b$	$c$
$\infty$	$c$	$\infty$	$c$	$\infty$

(c)

图 3.4.2 用于计算距离变换的模板



# 局部距离的计算

## □ 初始化距离图

$$DT^{(0)}(p) = \begin{cases} 0 & \text{当 } p \in B \\ \infty & \text{当 } p \notin B \end{cases}$$

## □ 用下面规则将距离值从像素 $q = (x_p + k, y_p + l)$ 传播到 $p$

$$DT^{(t)}(p) = \min_{k,l} \{DT^{(t-1)}(q) + M(k, l); q = (x_p + k, y_p + l)\}$$

## □ 更新过程持续进行到距离图不再变化而停止

# 离散距离变换的实现

## □ 串行实现

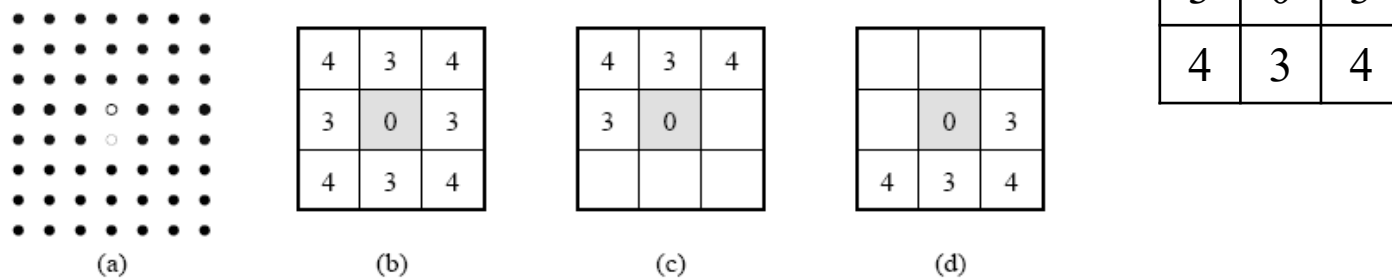


图 3.4.3 串行计算距离变换

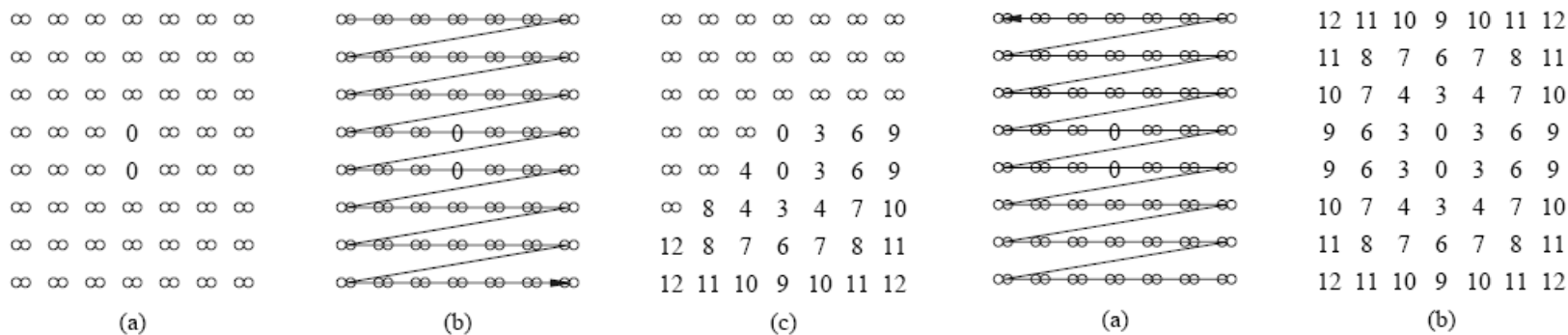


图 3.4.4 前向扫描

图 3.4.5 反向扫描

# 离散距离变换的实现

## □ 并行实现

$$DT^{(t)}(p) = \min_{k,l} \{DT^{(t-1)}(q) + M(k,l); q = (x_p + k, y_p + l)\}$$

∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	4	3	4	∞	∞
∞	∞	3	0	3	∞	∞
∞	∞	3	0	3	∞	∞
∞	∞	4	3	4	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞

(a)

∞	∞	∞	∞	∞	∞	∞
∞	8	7	6	7	8	∞
∞	7	4	3	4	7	∞
∞	6	3	0	3	6	∞
∞	6	3	0	3	6	∞
∞	7	4	3	4	7	∞
∞	8	7	6	7	8	∞
∞	∞	∞	∞	∞	∞	∞

(b)

12	11	10	9	10	11	12
11	8	7	6	7	8	11
10	7	4	3	4	7	10
9	6	3	0	3	6	9
9	6	3	0	3	6	9
10	7	4	3	4	7	10
11	8	7	6	7	8	11
12	11	10	9	10	11	12

(c)

图 3.4.6 并行计算过程中的各个步骤所得到的距离图

# 3-D距离变换

## □ 3-D距离

$$d_E(p, q) = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2 + (z_p - z_q)^2}$$

$$d_4(p, q) = |x_p - x_q| + |y_p - y_q| + |z_p - z_q|$$

$$d_8(p, q) = \max\{|x_p - x_q|, |y_p - y_q|, |z_p - z_q|\}$$

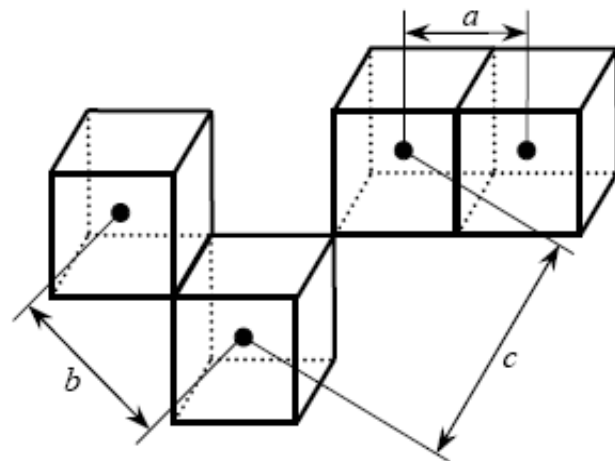


图 3.4.8 3-D 体素间的 3 种距离

# 3-D图象中的连通和拓扑

## □ 邻域和连通

### ■ 邻域的通用定义

设  $\mathbf{x} = (x_0, \dots, x_n)$  为图象网格上的一个单元  $\mathbf{x}$  的  $V_1^r$  邻域定义为:

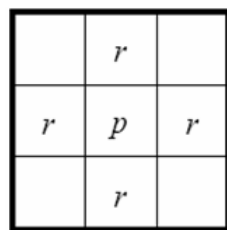
$$V_1^r = \{\mathbf{y} \mid D_1(\mathbf{x}, \mathbf{y}) \leq r\}$$

$\mathbf{x}$  的  $V_\infty^r$  邻域定义为:  $V_\infty^r = \{\mathbf{y} \mid D_\infty(\mathbf{x}, \mathbf{y}) \leq r\}$

$$N_4(\mathbf{x}) = V_1^1(\mathbf{x}) \quad N_8(\mathbf{x}) = V_\infty^1(\mathbf{x})$$

# 3-D图象中的连通和拓扑

## □ 2-D像素邻域



(a)

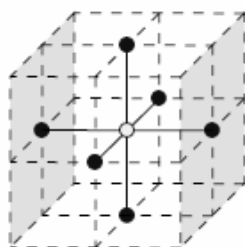


(b)

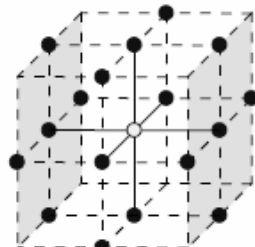
$$N_4(\mathbf{x}) = V_1^1(\mathbf{x})$$

$$N_8(\mathbf{x}) = V_\infty^1(\mathbf{x})$$

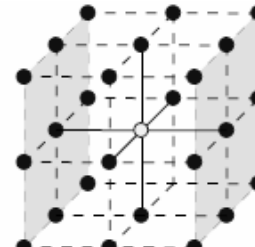
## □ 3-D体素邻域



(a)



(b)



(c)

$$N_6(\mathbf{x}) = V_1^1(\mathbf{x})$$

$$N_{18}(\mathbf{x}) = V_\infty^1(\mathbf{x}) \cap V_1^2(\mathbf{x})$$

$$N_{26}(\mathbf{x}) = V_\infty^1(\mathbf{x})$$

# 3-D图象中的连通和拓扑

## □ 像素点连通

- 存在一系列依次连接的像素组成的通路

## □ 连通组元

- 完全在一个图象子集中的像素组成的通路上的像素集合构成该图象子集中的一个连通组元

## □ 连通集

- 如果  $S$  中只有1个连通组元，即  $S$  中所有像素都互相连通，则称  $S$  是一个连通集

# 3-D距离变换

## □ 3-D距离变换的模板

- 前向扫描：从图象第一层的左上角向最后一层的右下角进行

$$\begin{array}{ccc} c & b & c \\ b & a & b \\ c & b & c \end{array} \quad \begin{array}{ccc} b & a & b \\ a & 0 & 0 \\ 0 & 0 & 0 \end{array} \quad \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}$$

图 3.4.9 3-D 前向扫描模板

- 反向扫描：从最后一层的右下角向第一层左上角进行