

图像分割

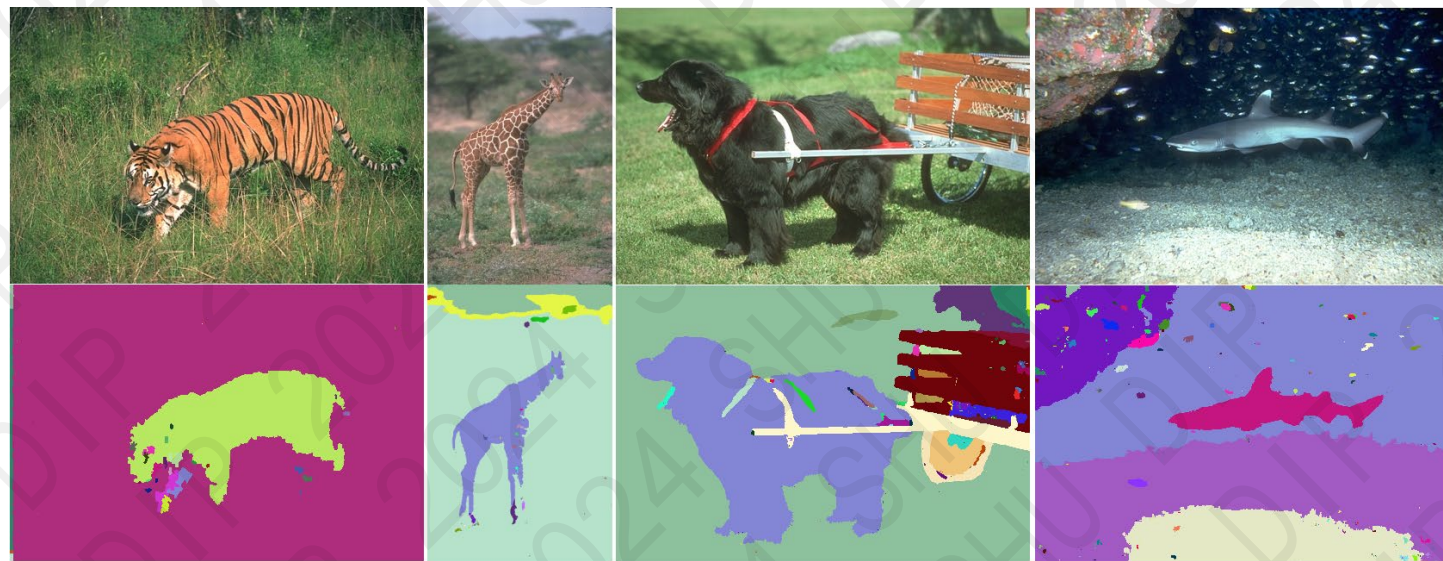
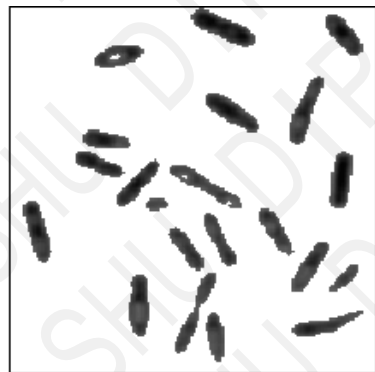
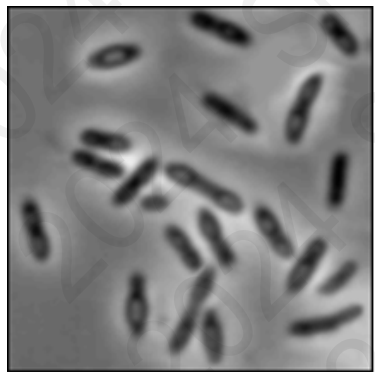
数字图像处理 第8讲

内容

- 图像分割概述
- 基于阈值的方法
- 基于边缘的方法
- 基于区域的方法

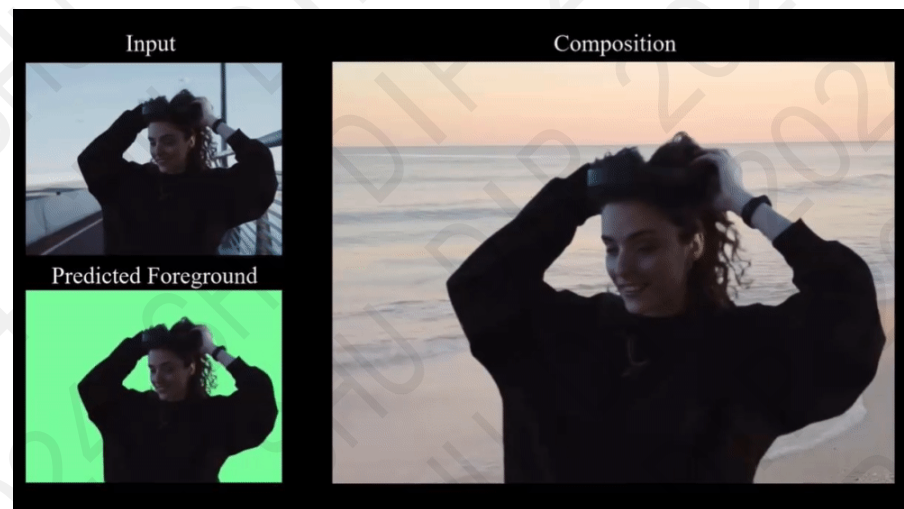
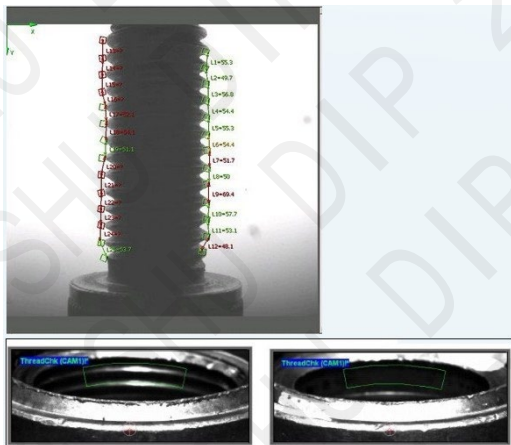
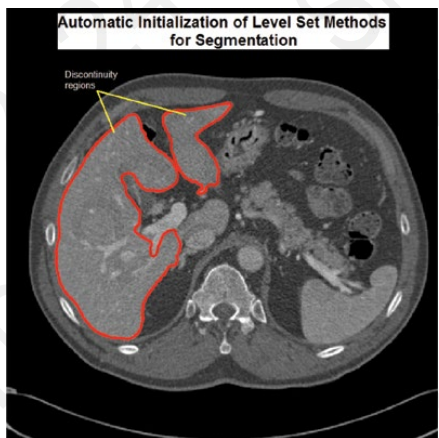
什么是图像分割

- 将图像划分成不相交的子区域
- 区域内具有某种意义上的一致性
 - 颜色、亮度、纹理、语义...



图像分割的作用

- 简化或改变图像的表现形式
- 使得图像更容易理解和分析
- 在各个领域有应用广泛



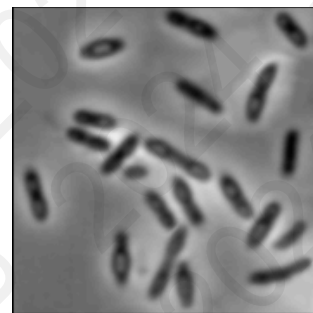
图像分割算法

- 和特定应用、图像不同性质相关
- 利用了图像的两个基本特性
 - 同一目标内有相似性
 - 不同目标间的边界上，有不连续性
- 常见分割方法的类型
 - 基于阈值的方法
 - 基于边缘的方法
 - 基于区域的方法



基于阈值的方法

- 适合不同对象(区域)在灰度值上存在较大差异的情况
- 通过设定灰度值阈值来分割
 - 以图像直方图来指导
 - 单阈值分割（二值化问题）
 - 多阈值分割

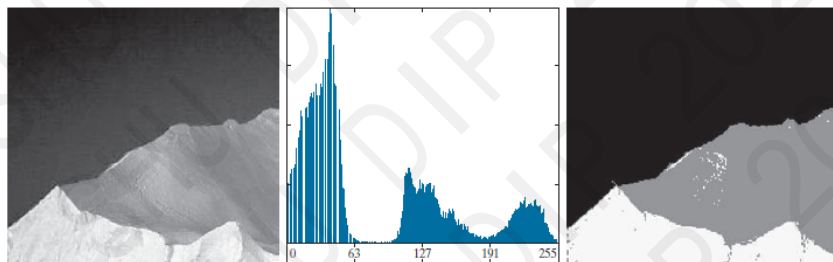


$T = 102$

细菌像素

背景

$$g(x, y) = \begin{cases} a, & f(x, y) > T_2 \\ b, & T_1 < f(x, y) \leq T_2 \\ c, & f(x, y) \leq T_1 \end{cases}$$

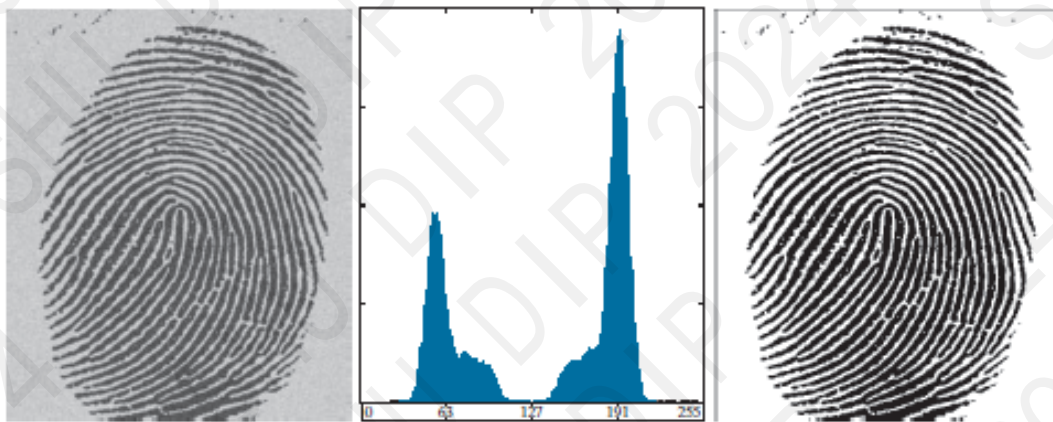
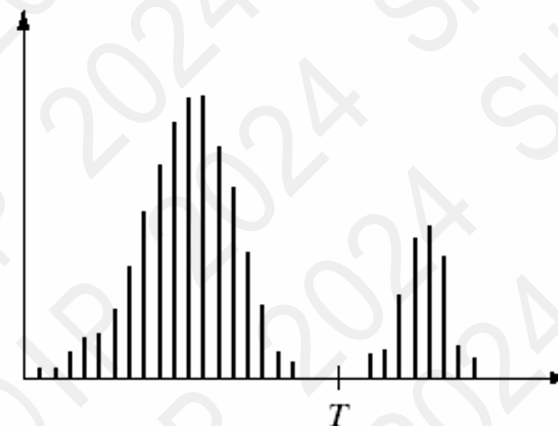


$$g(x, y) = \begin{cases} 255 & f(x, y) > T \\ 0 & f(x, y) \leq T \end{cases}$$

自动确定阈值的算法

- 全局阈值分割

- 初始阈值 T 为整幅图像的平均灰度值
- 产生两组像素集合： $\geq T$ 和 $< T$
- 分别计算各组的平均灰度值 $m1$ 和 $m2$
- 得到一个新的阈值： $T = (m1 + m2) / 2$
- 重新产生两组像素集合，重复前述步骤，直到 T 变化很小时停止



初始值为图像的
像素平均值

通过3次迭代自
动找到 $T=125$

Otsu 大津法

- 用一个阈值，将图像分成前景/背景的分类问题
- 最佳阈值的选择
 - 让两类各自的方差都足够小，从而避免两类交叠
- 最小化类内方差

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

- 方差的计算量太大
- 等价于：最大化类间方差（计算量更小）

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2$$

- 穷举搜索一个阈值让类间方差最大

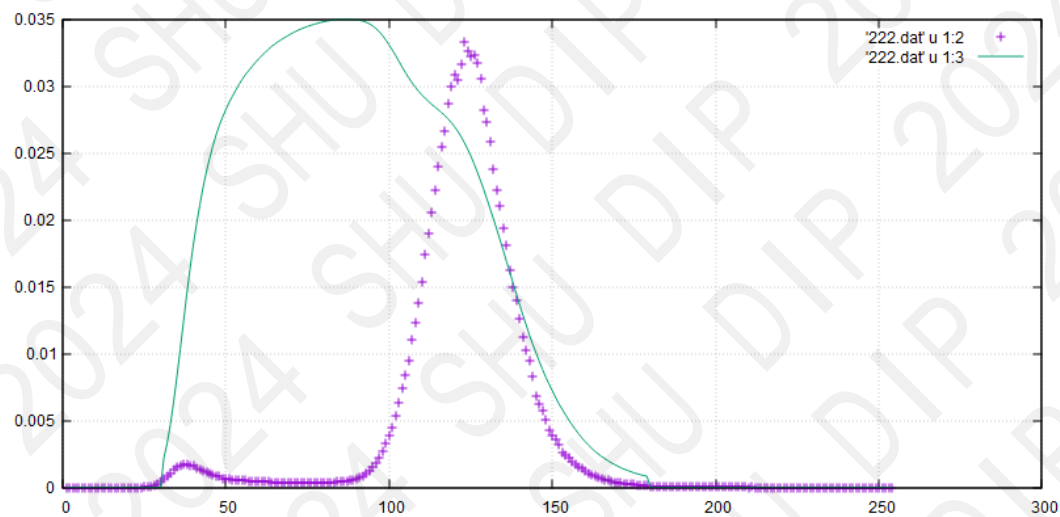
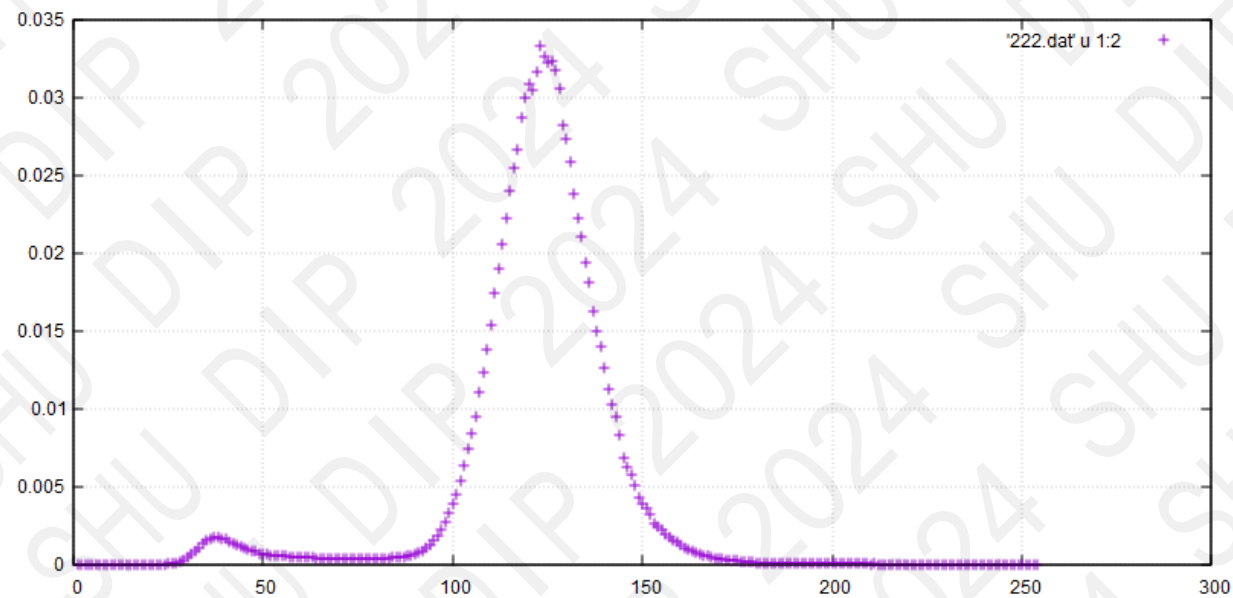
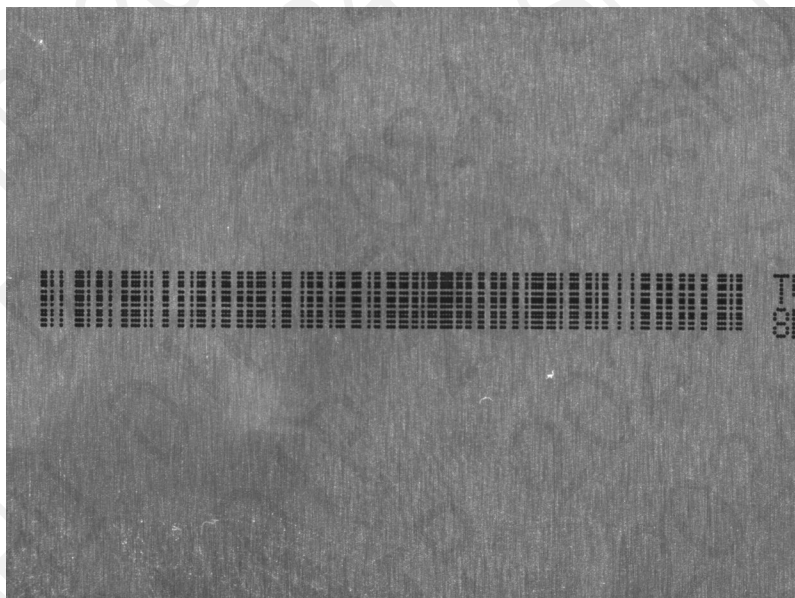
$$\omega_0(t) = \sum_{i=0}^{t-1} p(i)$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i)$$

$$\omega_0 + \omega_1 = 1$$

$$\mu_0(t) = \frac{\sum_{i=0}^{t-1} ip(i)}{\omega_0(t)}$$

$$\mu_1(t) = \frac{\sum_{i=t}^{L-1} ip(i)}{\omega_1(t)}$$



$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2$$

Otsu 大津法

• 算法流程

计算每个强度级的直方图和概率

遍历所有可能的阈值 $t = 1 \dots$ 最大强度

1. 更新 ω_i 和 μ_i

2. 计算 $\sigma_b^2(t)$

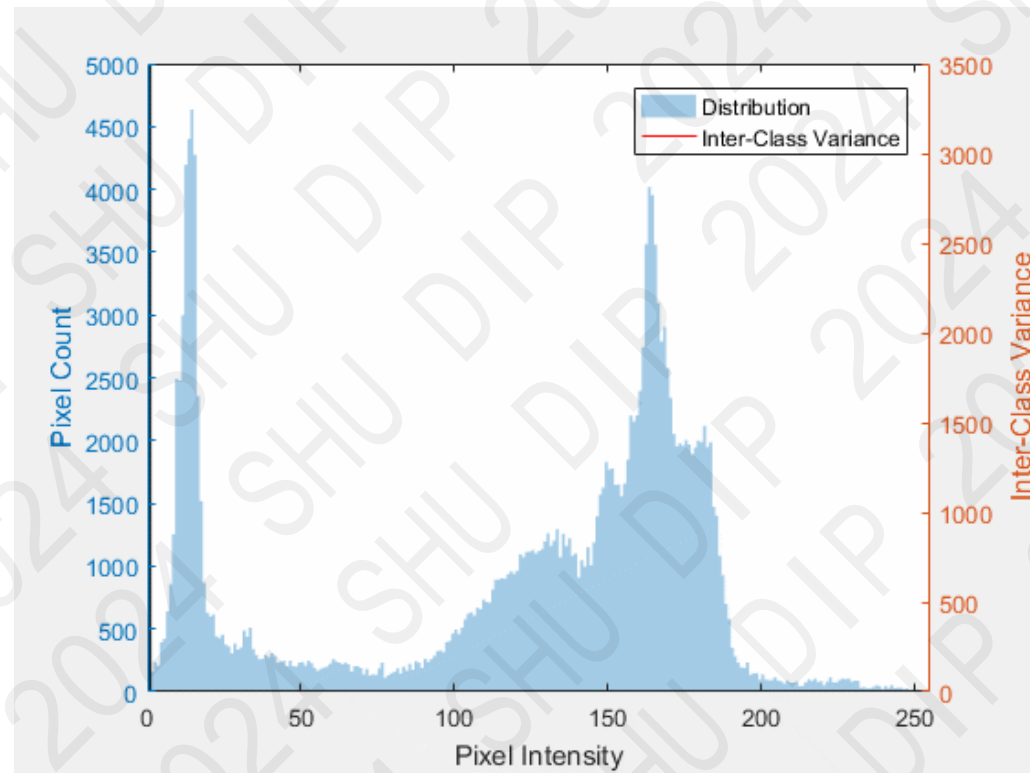
所需的阈值对应于最大的 $\sigma_b^2(t)$

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \quad \mu_0(t) = \frac{\sum_{i=0}^{t-1} ip(i)}{\omega_0(t)}$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i) \quad \mu_1(t) = \frac{\sum_{i=t}^{L-1} ip(i)}{\omega_1(t)}$$

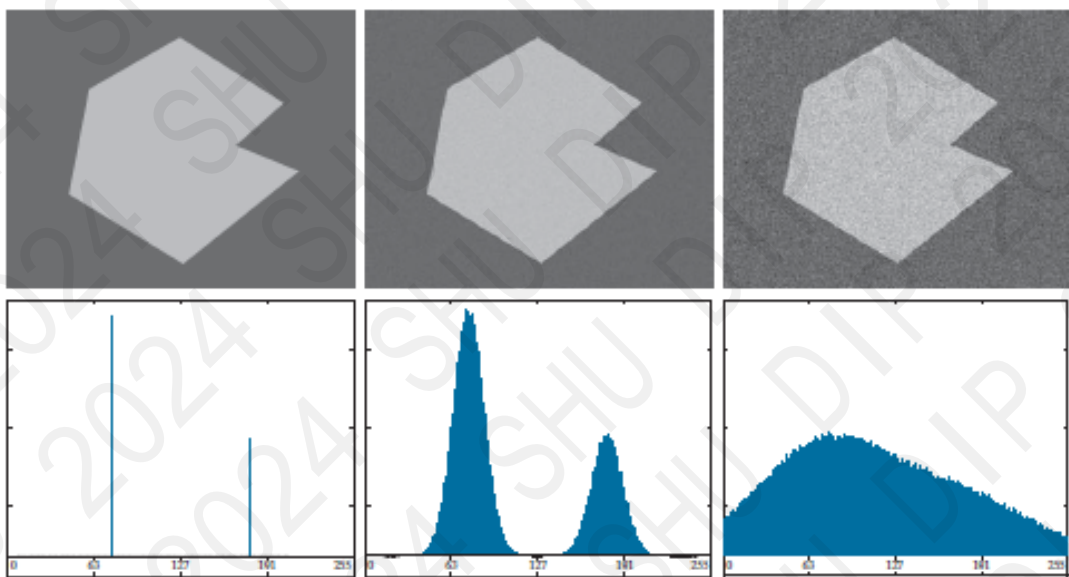
$$\omega_0 + \omega_1 = 1$$

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2$$

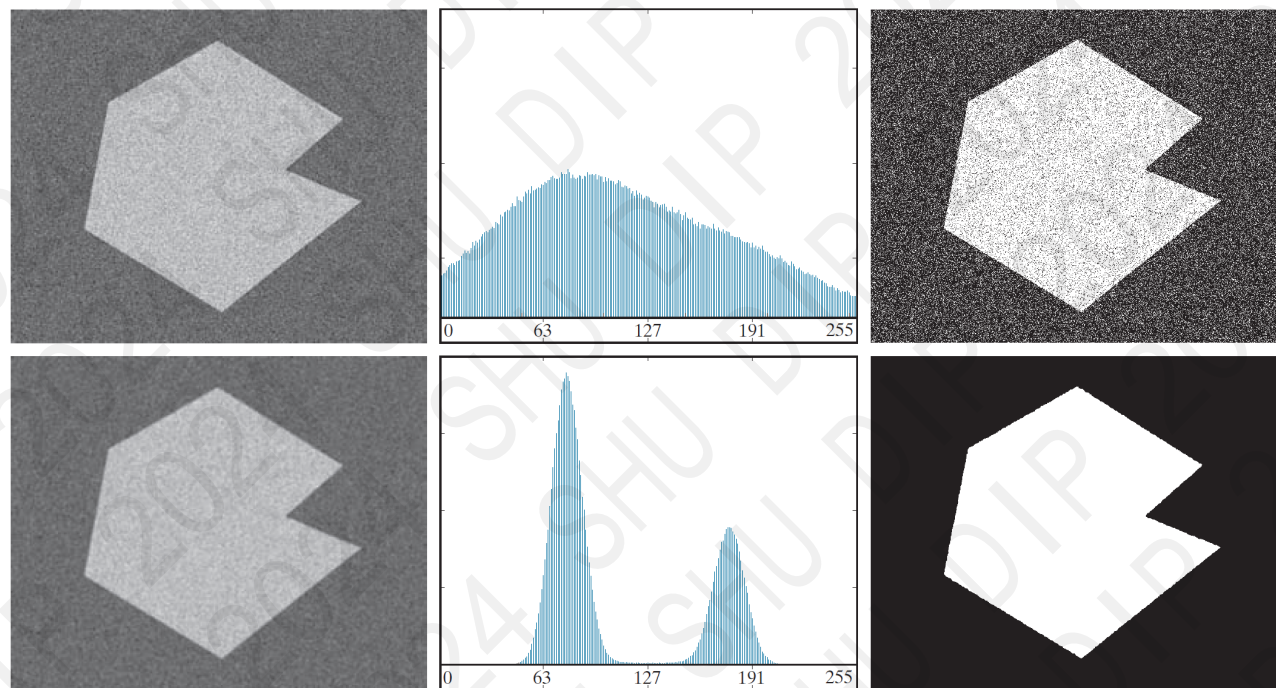


图像噪声对阈值方法的影响

- 噪声对灰度直方图的形态造成较大影响
- 解决方法：对噪声先进行平滑

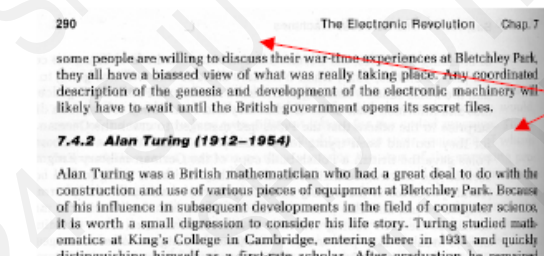
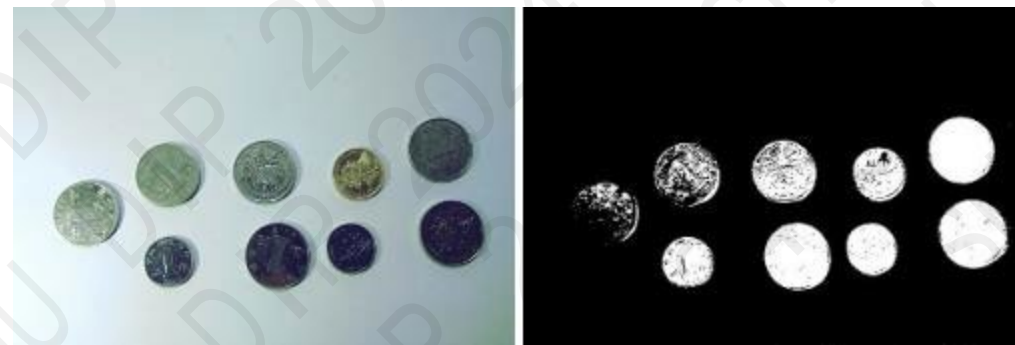


不同标准差噪声影响下的直方图

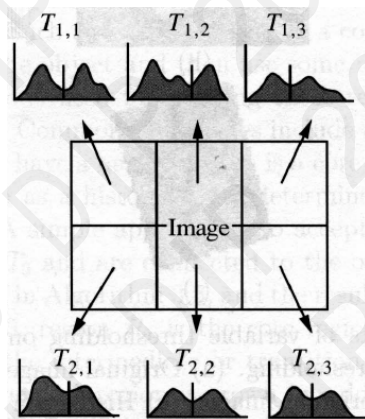
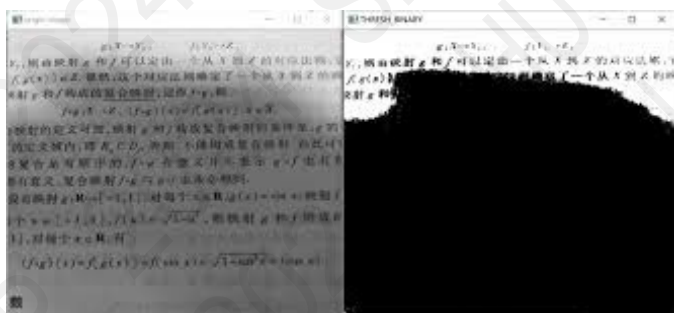


光照不均对阈值方法的影响

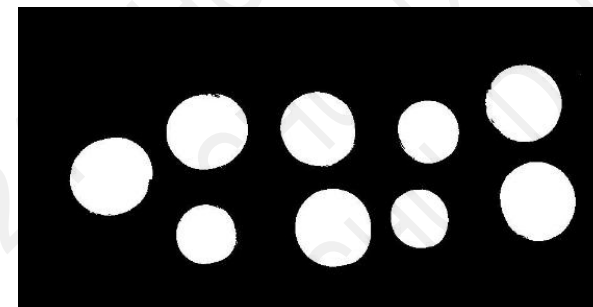
- 阈值对不均匀的光照敏感
- 图像分区域，不同的阈值



光照不均匀，全局阈值不好取



T=208 d=9	T=207 d=6	T=201 d=7	T=184 d=10	T=163 d=12
T=224 d=9	T=215 d=5	T=204 d=6	T=188 d=10	T=121 d=94
T=192 d=91	T=175 d=82	T=156 d=93	T=140 d=88	T=142 d=101
T=209 d=72	T=152 d=138	T=139	T=121	
T=242 d=9	T=223 d=12	T=201 d=12	T=176 d=12	T=152 d=15

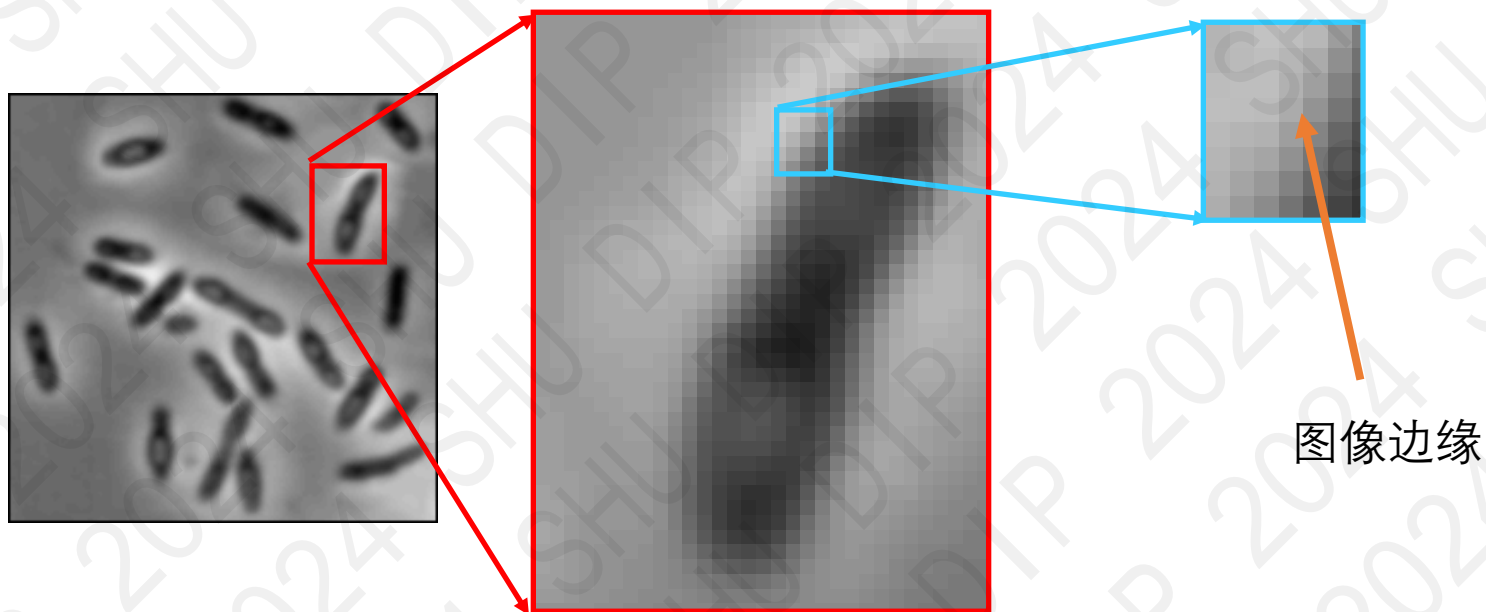


内容

- 图像分割概述
- 基于阈值的方法
- 基于边缘的方法
- 基于区域的方法

图像边缘 Edge

- 不同目标的边界上，存在不连续性
- 图像边缘
 - 像素值发生较大变化的地方



检测图像边缘

- 边缘模型



- 一阶差分

$$\frac{\partial f(x,y)}{\partial x} \approx f(x+1, y) - f(x, y)$$

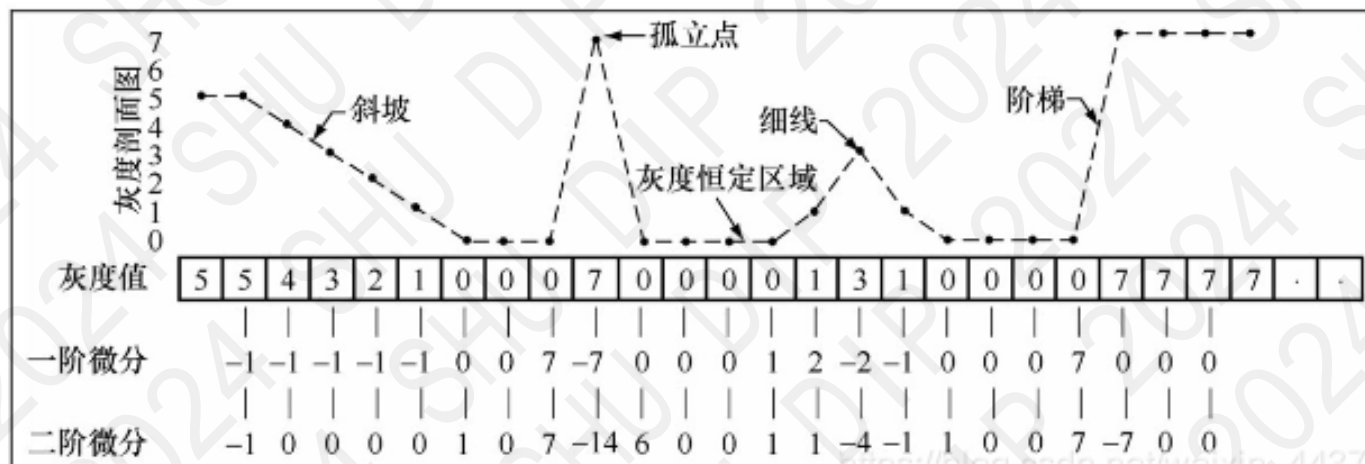
$$\frac{\partial f(x,y)}{\partial y} \approx f(x, y+1) - f(x, y)$$

- 二阶差分

$$\frac{\partial^2 f(x,y)}{\partial^2 x} \approx f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f(x,y)}{\partial^2 y} \approx f(x, y+1) + f(x, y-1) - 2f(x, y)$$

图像的一维剖面



边缘检测 – 一阶导数

- 梯度的计算模板

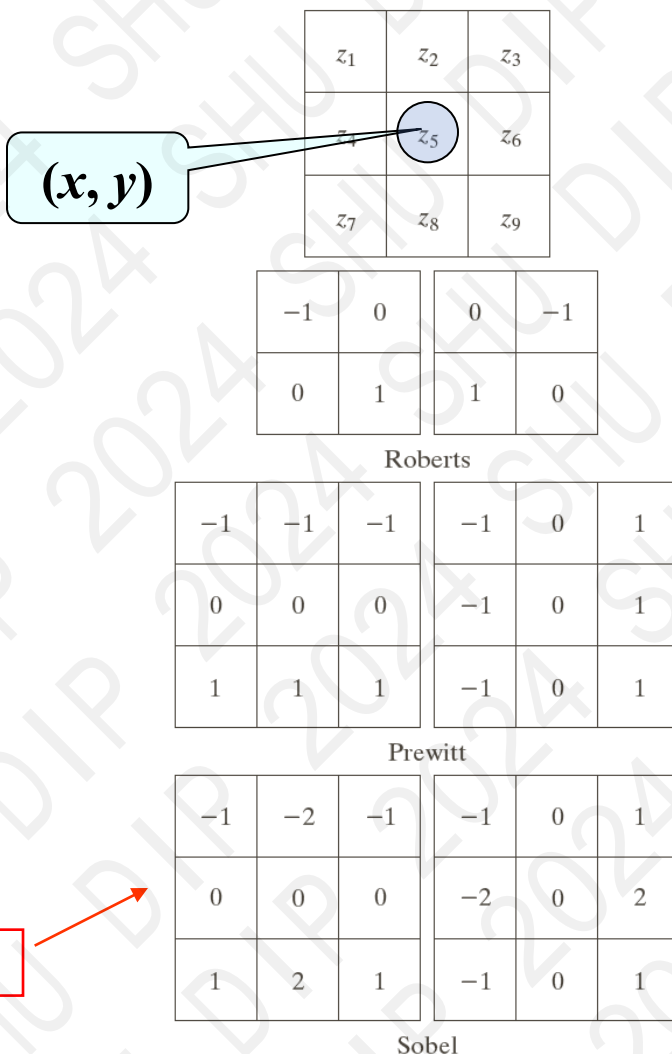


$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$$

$$g = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}$$

$$g = \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right|$$

噪声抑制更佳



水平垂直差分法

$$\begin{cases} g_x = z_8 - z_5 \\ g_y = z_6 - z_5 \end{cases}$$

Robert Gradient

$$\begin{cases} g_x = z_9 - z_5 \\ g_y = z_8 - z_6 \end{cases}$$

Prewitt 算子:

$$\begin{cases} g_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \\ g_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \end{cases}$$

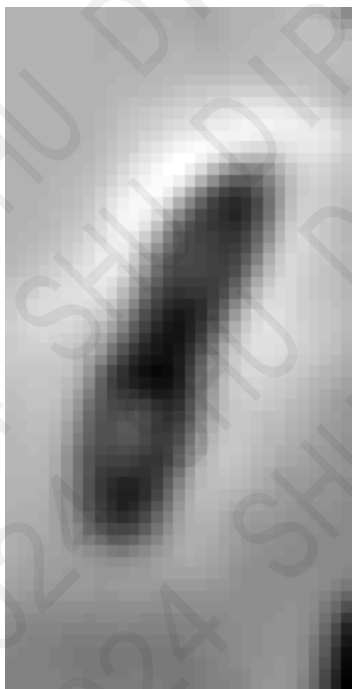
Sobel 算子:

$$\begin{cases} g_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \\ g_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \end{cases}$$

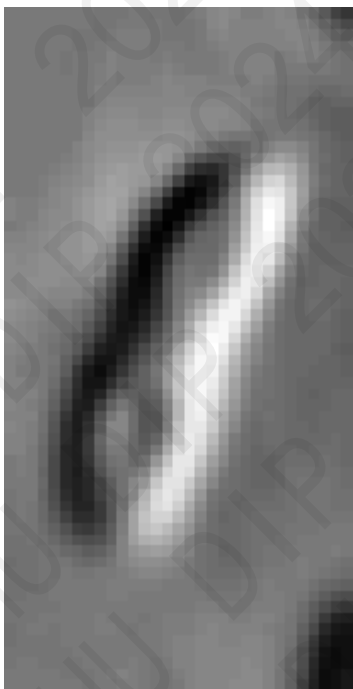
边缘检测 — 一阶导数

$$|\nabla P| = \sqrt{\left(\frac{\partial P}{\partial x}\right)^2 + \left(\frac{\partial P}{\partial y}\right)^2}$$

梯度幅值



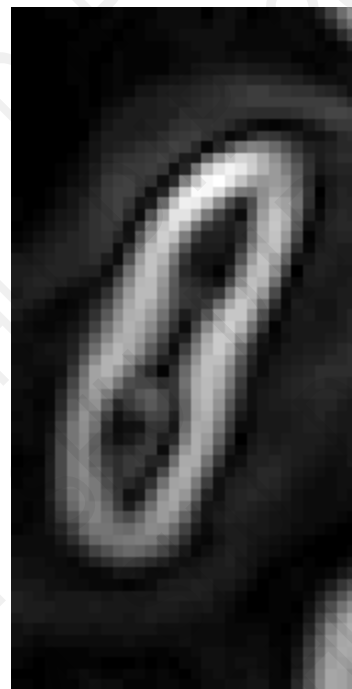
原图 P



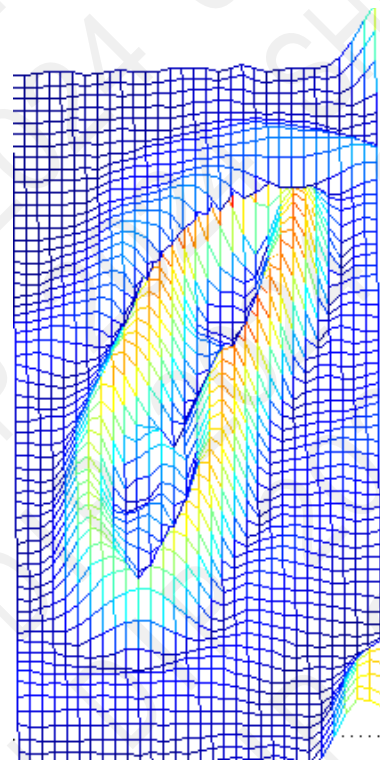
$\frac{\partial P}{\partial x}$



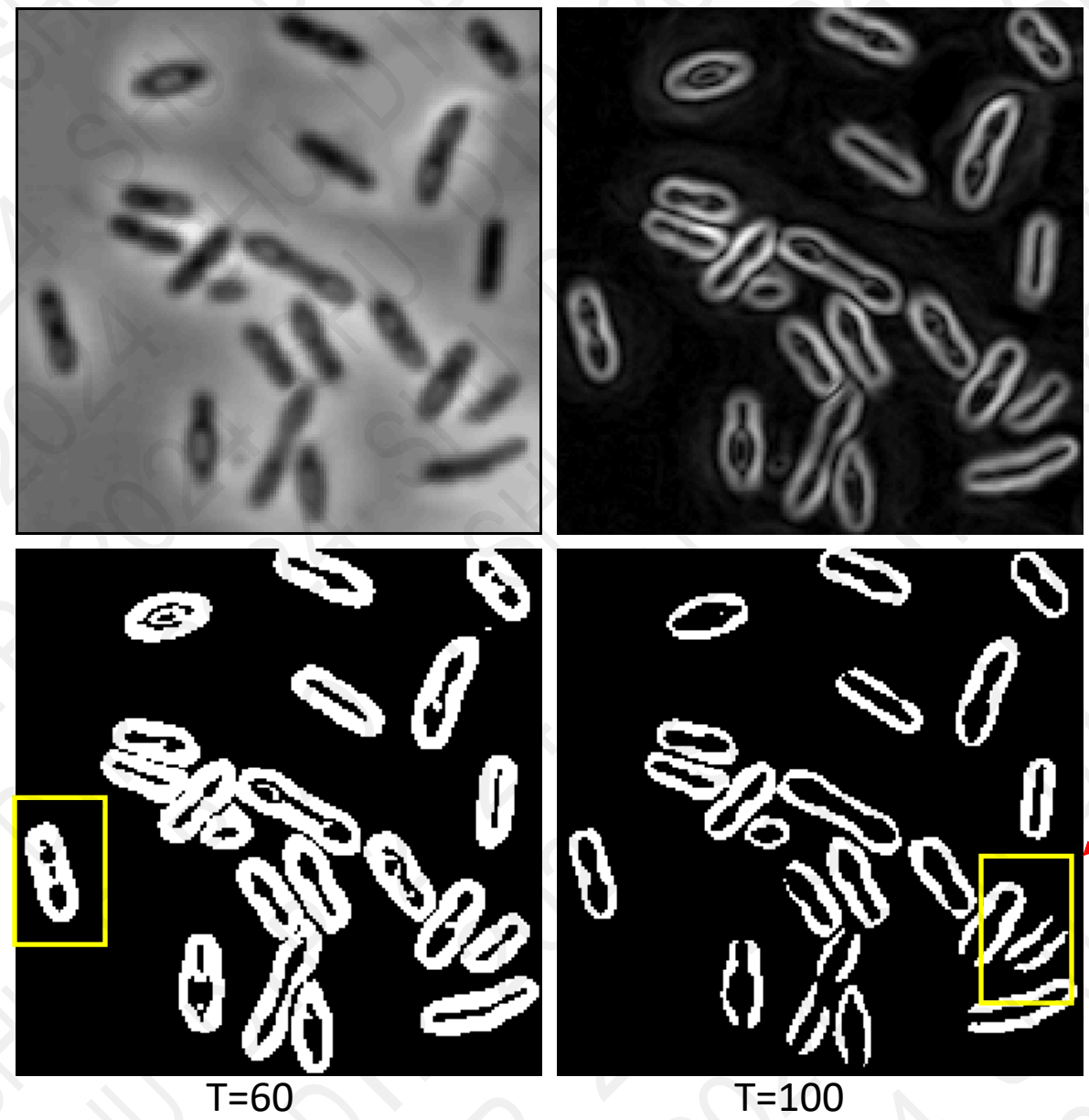
$\frac{\partial P}{\partial y}$



$|\nabla P|$



$|\nabla P|$



内容

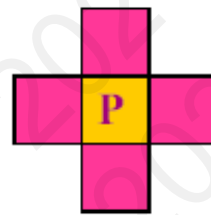
- 图像分割概述
- 基于阈值的方法
- 基于边缘的方法
- 基于区域的方法
 - 区域生长
 - 分裂与聚合

区域生长 Region Growing

- 边缘不适用时
- 从区域一致性出发
- 基本思想
 - 事先定义相似性准则
 - 满足准则的子区域聚合成更大区域
- 相似性可以定义在特征上
 - 灰度值、颜色、纹理、形状、语义

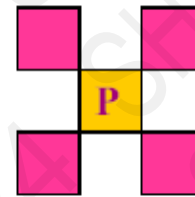


区域生长



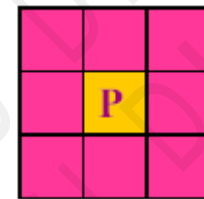
4-邻域

$$N_4(p)$$



对角邻域

$$N_D(p)$$



8-邻域

$$N_8(p)$$

• 基本步骤

- 找一些像素点作为种子
- 设定相似准则，寻找种子邻域里满足规则的像素
- 将相似像素加入，作为新的种子，重复上述过程
- 满足一定条件停止：比如区域不再变化

$$\begin{bmatrix} 4 & 3 & 7 & 3 & 3 \\ 1 & 7 & (8) & 7 & 5 \\ 0 & 5 & 6 & 1 & 3 \\ 2 & 2 & 6 & 0 & 4 \\ 1 & 2 & 1 & 3 & 1 \end{bmatrix}$$

(a) 原图像灰度矩阵生长点

$$\begin{bmatrix} 4 & 3 & (7) & 3 & 3 \\ 1 & (7) & (8) & (7) & 5 \\ 0 & 5 & 6 & 1 & 3 \\ 2 & 2 & 6 & 0 & 4 \\ 1 & 2 & 1 & 3 & 1 \end{bmatrix}$$

(b) 第一次区域生长结果

$$\begin{bmatrix} 4 & 3 & (7) & 3 & 3 \\ 1 & (7) & (8) & (7) & 5 \\ 0 & 5 & (6) & 1 & 3 \\ 2 & 2 & 6 & 0 & 4 \\ 1 & 2 & 1 & 3 & 1 \end{bmatrix}$$

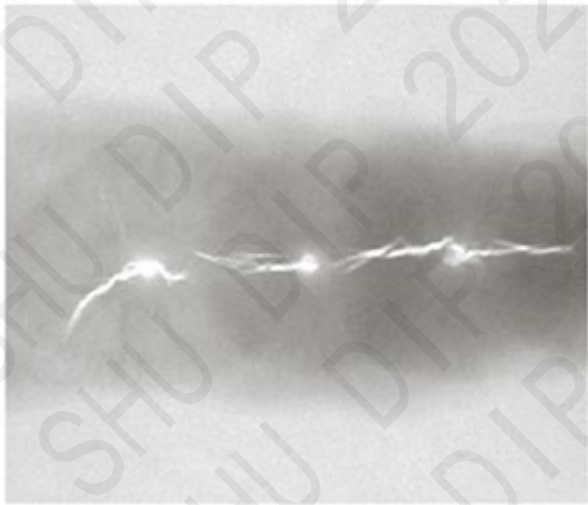
(c) 第二次区域生长结果

$$\begin{bmatrix} 4 & 3 & (7) & 3 & 3 \\ 1 & (7) & (8) & (7) & 5 \\ 0 & (5) & (6) & 1 & 3 \\ 2 & 2 & (6) & 0 & 4 \\ 1 & 2 & 1 & 3 & 1 \end{bmatrix}$$

(d) 第三次区域生长结果

8邻域，与种子灰度值之差小于等于1

区域生长举例



(a) 缺陷焊缝的X光图像



(b) 初始的种子点



(c) 区域生长的结果

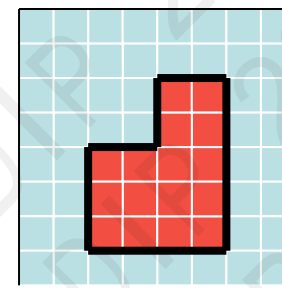
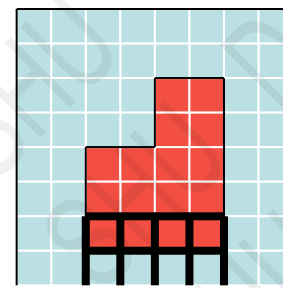
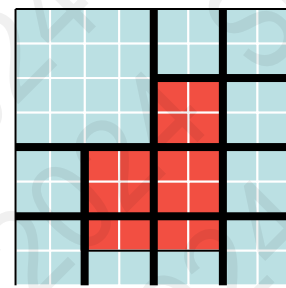
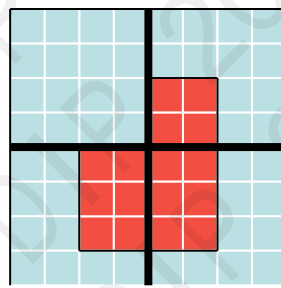
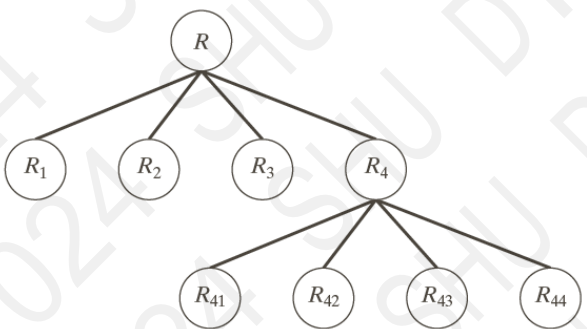
内容

- 图像分割概述
- 基于阈值的方法
- 基于边缘的方法
- 基于区域的方法
 - 区域生长
 - 分裂与聚合

分裂与聚合

- 从整幅图像出发，按照预定准则，不断分裂和聚合
- 算法步骤
 - 设定某个准则，比如区域方差小于10
 - 如该准则在当前区域不满足，则将其区域分裂
 - 如不能进一步分裂，满足准则的相邻区域聚合
 - 如无法进一步聚合，则算法终止

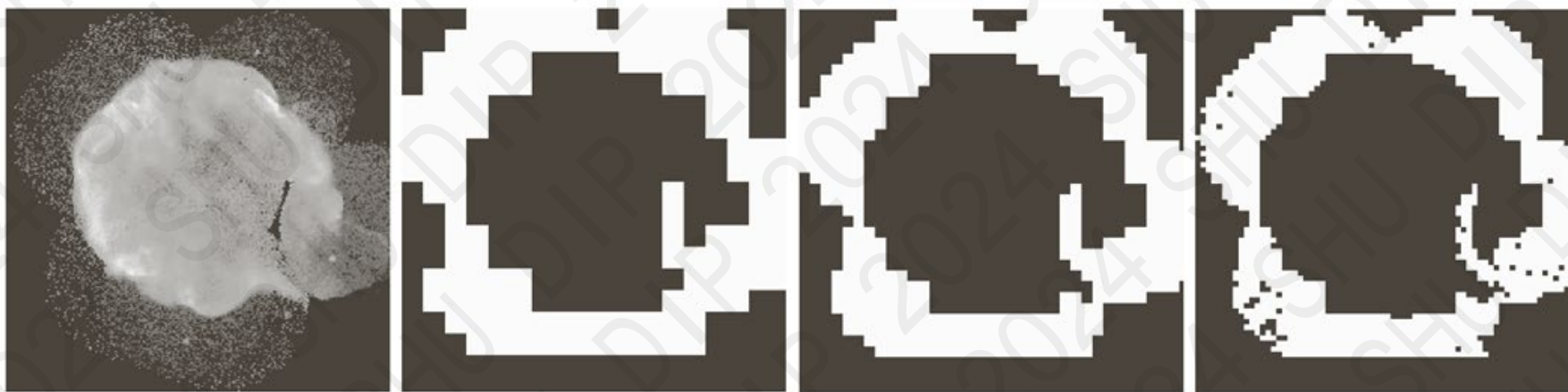
R_1	R_2	
R_3	R_{41}	R_{42}
	R_{43}	R_{44}



分裂与聚合

- 分割出图片中的环状行星带
- 行星带所在区域的像素均值和方差有其特性

$$Q = \begin{cases} TRUE & \sigma > a, \quad 0 < m < b \\ FALSE & other \end{cases}$$



原图

最小四象限尺寸为 32x32, 16x16, 8x8 分割结果

怎样才是一个好的分割

- 任务相关
- 没有一个通用的方法
- 语义级的分割
 - Semantic Object
 - 分割依赖于低层视觉，同时又依赖于高层视觉
 - 只有对图像内容的彻底理解，才能产生完美的分割
- 迄今仍是充满挑战的任务

