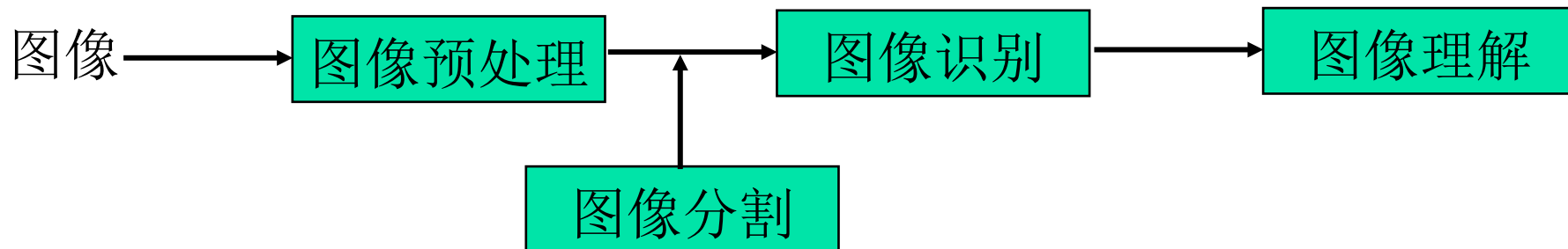




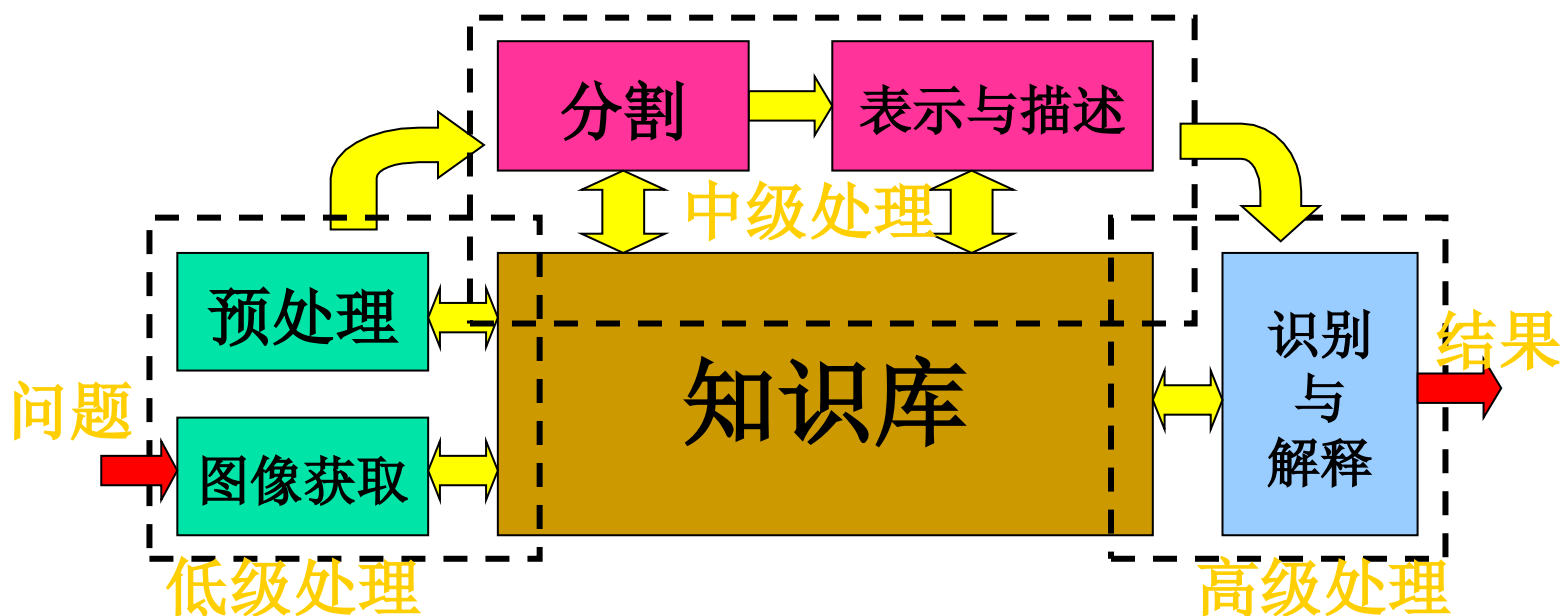
第5章 图像分割

5.1 图像分割的概念和分类

- 计算机处理图像的两个目的
 - 产生更适合人观察和识别的图像
 - 由计算机自动识别和理解图像
- 图像分割是图像识别和图像理解的基础



5.1 图像分割的概念和分类



在该系统中，图像的增强和恢复可以看作预处理，其输入、输出均是图像，它是传统的图像处理的内容。而图像分割、特征提取及结构分析等称为图像识别，其输入是图像，输出是描述或解释。



5.1 图像分割的概念和分类

- 图像分割的定义
 - 图像分割是把图像分割成**互不交叠的有意义区域**，以便进一步的分析，分开的区域一般是图像中我们感兴趣的目标
- 图像分割是基于目标或区域的特征进行的
 - 每个目标或区域由于某些特征的不同与其它区域区别开来，边缘、纹理、形状、颜色等都是重要的特征



5.1 图像分割的概念和分类

- 图像分割算法一般是**基于亮度值的不连续性和相似性**
 - **不连续性**是基于亮度的不连续变化分割图像，称为基于**边界**的技术，如边缘检测
 - **相似性**是指根据不同的准则将图像分割成相似的区域，称为基于**区域**的技术，如阈值分割、区域生长、区域分裂和合并

5.1 图像分割的概念和分类

▶ 图像分割的目的

- 把图像分解成构成它的部件和对象；
- 有选择性地定位感兴趣对象在图像中的位置和范



5.1 图像分割的概念和分类

- 从简到难，逐级分割
 - 控制背景环境，降低分割难度
 - 注意力集中在感兴趣的对象，缩小不相干图像成分的干扰。



提取轮廓



车牌定位



车牌识别

5.1 图像分割的概念和分类

图像分割的基本策略：

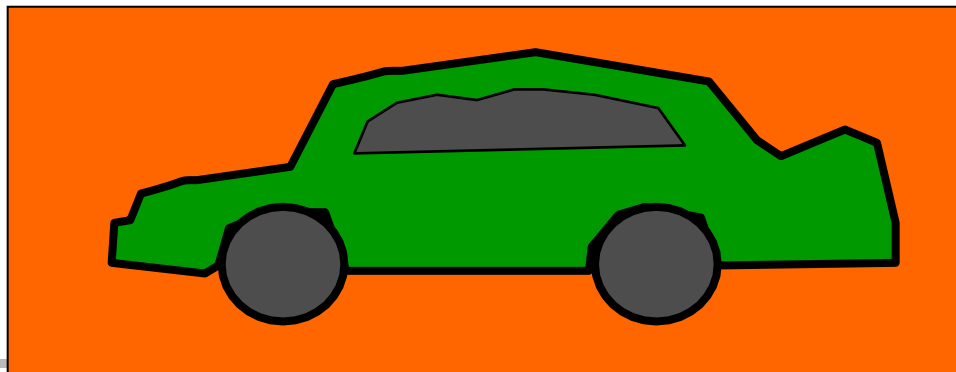
- 把像素按灰度划分到各个物体对应的区域中去；
- 先确定边缘像素点，然后将它们连接起来构成所需的边界；
- 确定存在于区域间的边界；

区域：像素的连通集

连通准则：

4-连通

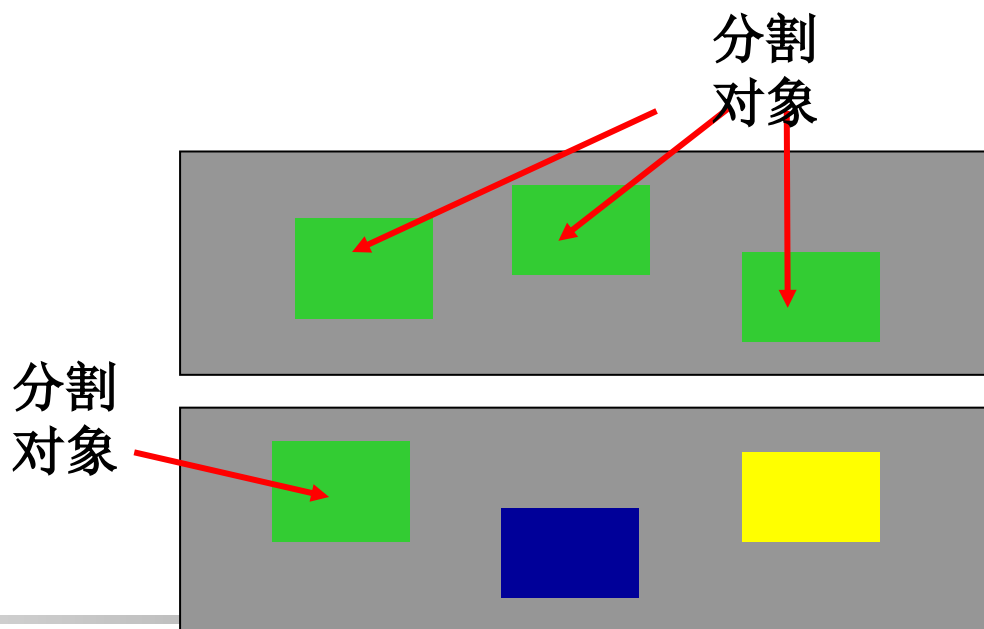
8-连通



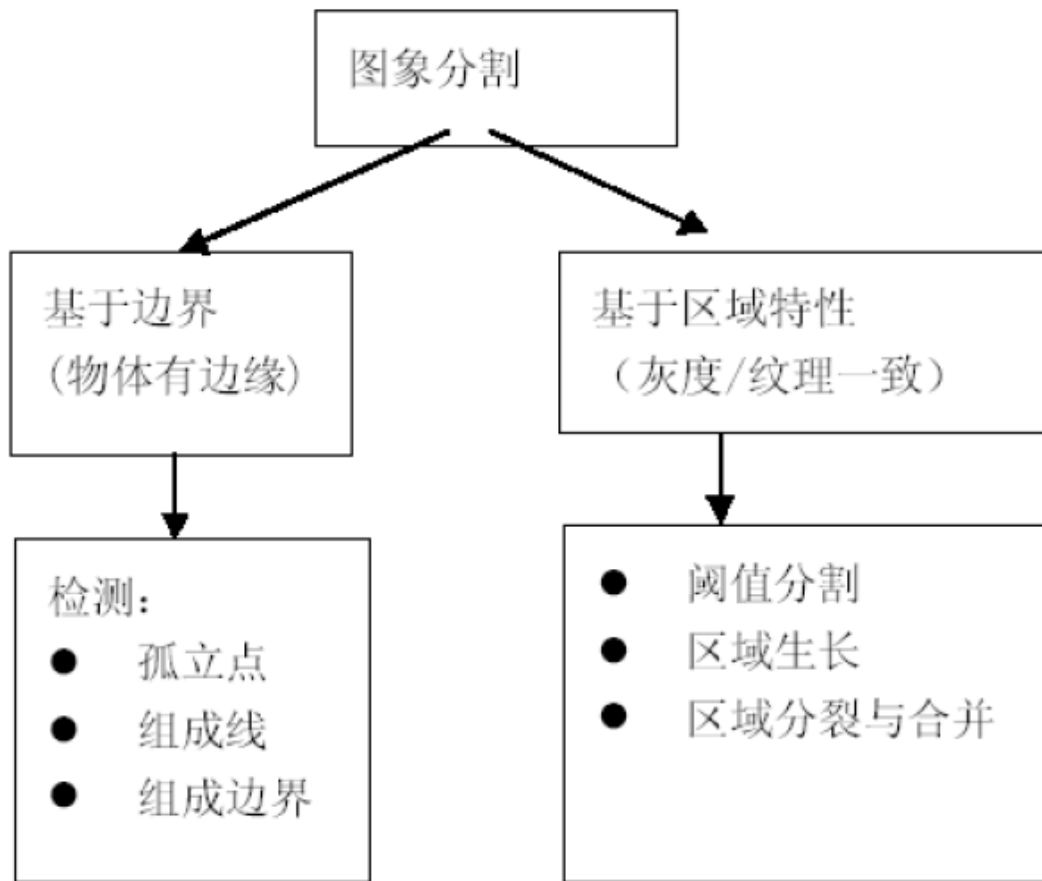
5.1 图像分割的概念和分类

图像分割的方法

- 1) **基于边缘的分割方法**:先提取区域边界,再确定边界限定的区域。
- 2) **区域分割**:确定每个像素的归属区域,从而形成一个区域图。
- 3) **区域生长**:将属性接近的连通像素聚集成区域。
- 4) **分裂—合并分割**:综合利用前两种方法,既存在图像的划分,又有图像的合并。



5.1 图像分割的概念和分类



5.1 图像分割的概念和分类

- 间断检测是基于图像像素灰度值的不连续性进行图像分割
 - 点检测
 - 线检测
 - 边界检测
- 寻找间断最一般的方法是模板检测

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9$$

$$= \sum_{i=1}^9 w_i z_i$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

5.1 图像分割的概念和分类

■ 间断检测：点检测

- 使用空域高通滤波器来检测孤立点，如果 $|R| \geq T$ ，则认为在模板中心的位置检测到一个点， T 是非负门限
- 孤立点：该点的灰度级与其背景的差异相当大，并且它所在的位置是一个均匀的或者近似均匀的区域
- 基本思想：如果一个孤立点与它周围的点很不相同，则很容易被上述模板检测到。在灰度级为常数的区域，模板响应 R 为 0

-1	-1	-1
-1	8	-1
-1	-1	-1

5.1 图像分割的概念和分类

间断检测：点检测

- $R = (-1 * 10 * 8 + 8 * 100) / 9 = 720 / 9 = 80$
- 可以设置阈值 $T = 50$
- 若 $R = 0$ ，则说明检测点与周围点像素值相同
- 若 $R > T$ ，则说明检测点与周围点像素值非常的不同，为孤立点

10	10	10
10	100	10
10	10	10

图像

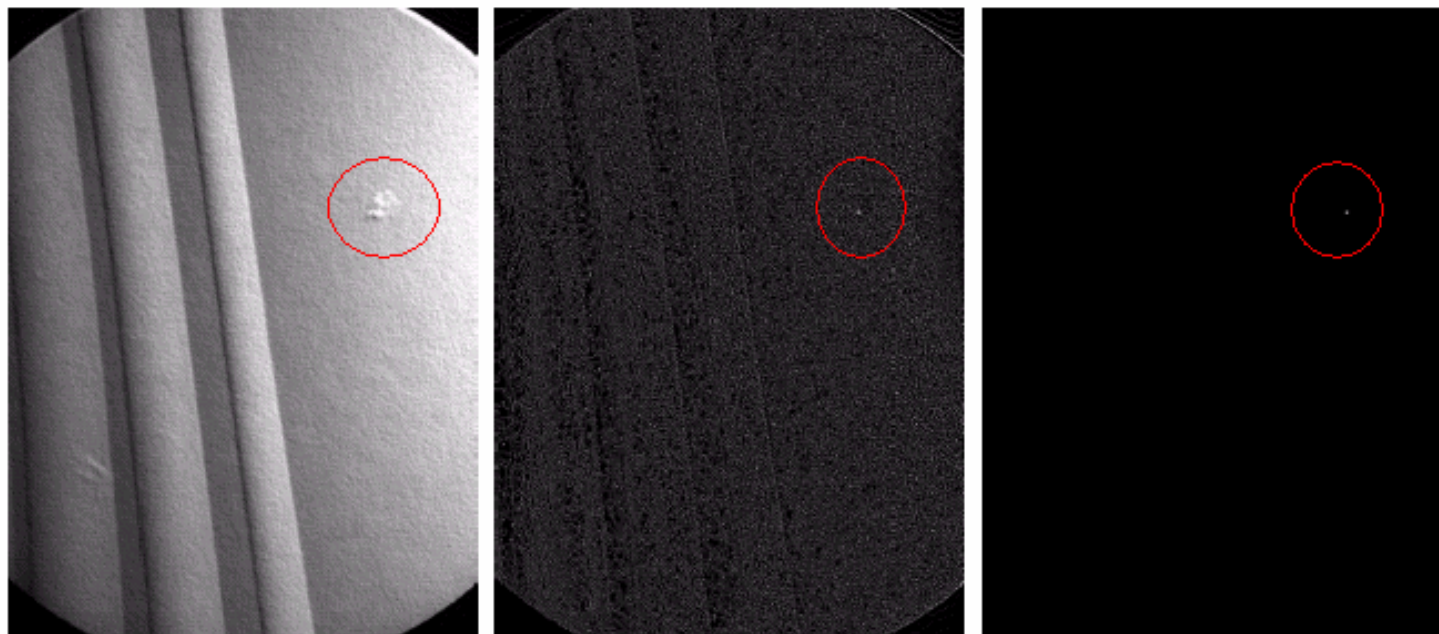
-1	-1	-1
-1	8	-1
-1	-1	-1

模板

5.1 图像分割的概念和分类

- 间断检测：点检测

-1	-1	-1
-1	8	-1
-1	-1	-1



a
b c d

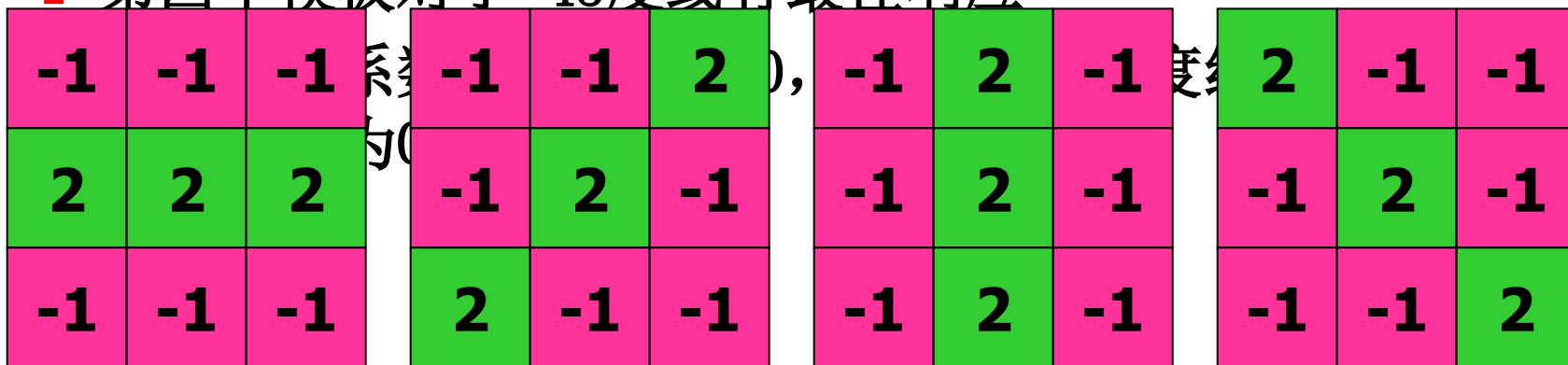
FIGURE 10.2

(a) Point detection mask.
(b) X-ray image of a turbine blade with a porosity.
(c) Result of point detection.
(d) Result of using Eq. (10.1-2).
(Original image courtesy of X-TEK Systems Ltd.)

5.1 图像分割的概念和分类

■ 间断检测：线检测

- 第一个模板对水平方向的线条（单像素宽）有更强的响应
- 第二个模板对于45度方向线有最佳响应
- 第三个模板对垂直线有最佳响应
- 第四个模板对于-45度线有最佳响应



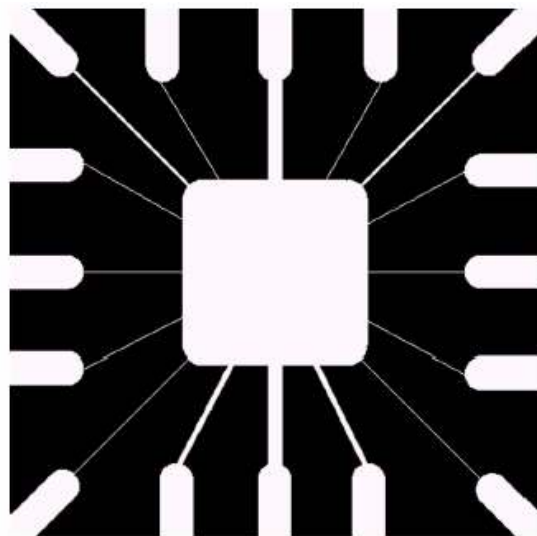
水平模板

45度模板

垂直模板

-45度模板

5.1 图像分割的概念和分类



a
b c

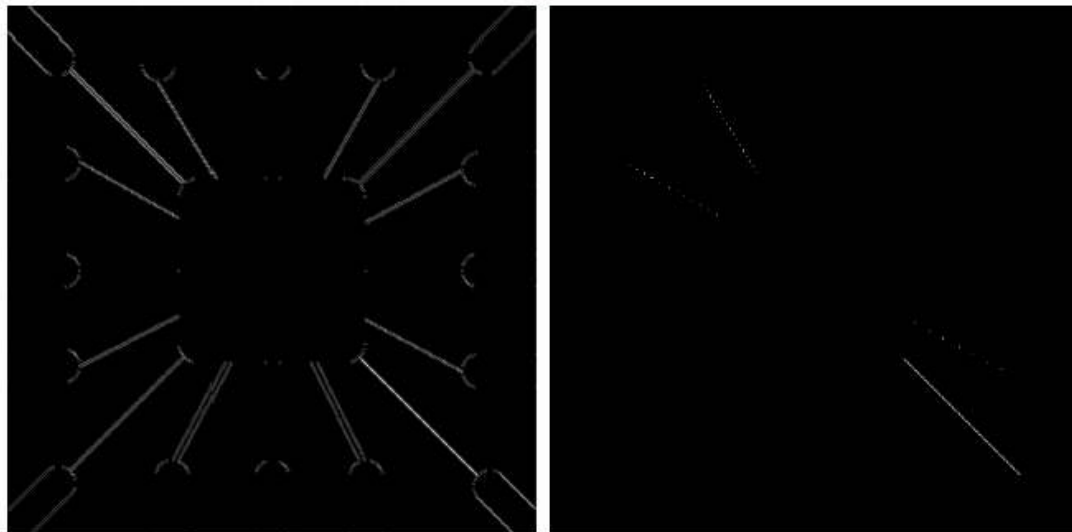
FIGURE 10.4

Illustration of line detection.

(a) Binary wire-bond mask.

(b) Absolute value of result after processing with -45° line detector.

(c) Result of thresholding image (b).



图一为原图像，图二为使用负**45**度检测器处理后的结果（取绝对值），图三为满足阈值条件的所有点，阈值条件为大于等于原图像中的最大值



5.2 基于边界的边缘检测

■ 边缘检测

- 边缘是位于两个区域的边界线上的相连像素的集合
- 边缘可以通过计算局部微分算子来检测
 - 一阶导数：通过梯度来计算
 - 二阶导数：通过拉普拉斯算子来计算

5.2 基于边界的边缘检测

■ 边缘检测

■ 拉普拉斯算子

图像函数 $f(x,y)$ 的拉普拉斯变换为

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$

$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_5 + z_6 + z_7 + z_8 + z_9)$$



5.2 基于边界的边缘检测

■ 边缘检测

- 拉普拉斯算子一般不以其原始形式用于边缘检测的原因是
 - 拉普拉斯算子对噪声非常敏感
 - 拉普拉斯算子的幅值产生双边缘
 - 不能检测边缘的方向
- 拉普拉斯算子在分割中的作用
 - 利用它的零交叉性质进行边缘定位
 - 确定一个像素在边缘暗的一边还是亮的一边



间断检测

■ 边缘检测

■ 高斯型拉普拉斯算子

- 高斯函数的目的是对图像进行平滑
- 拉普拉斯算子的目的是提供一幅用零交叉确定边缘位置的图像
- 图像的平滑处理减少了噪声的影响

5.2 基于边界的边缘检测

■ 边缘检测

- **Laplacian**
算子和平滑
Gaussian
滤波器进行
结合来进行
边缘检测

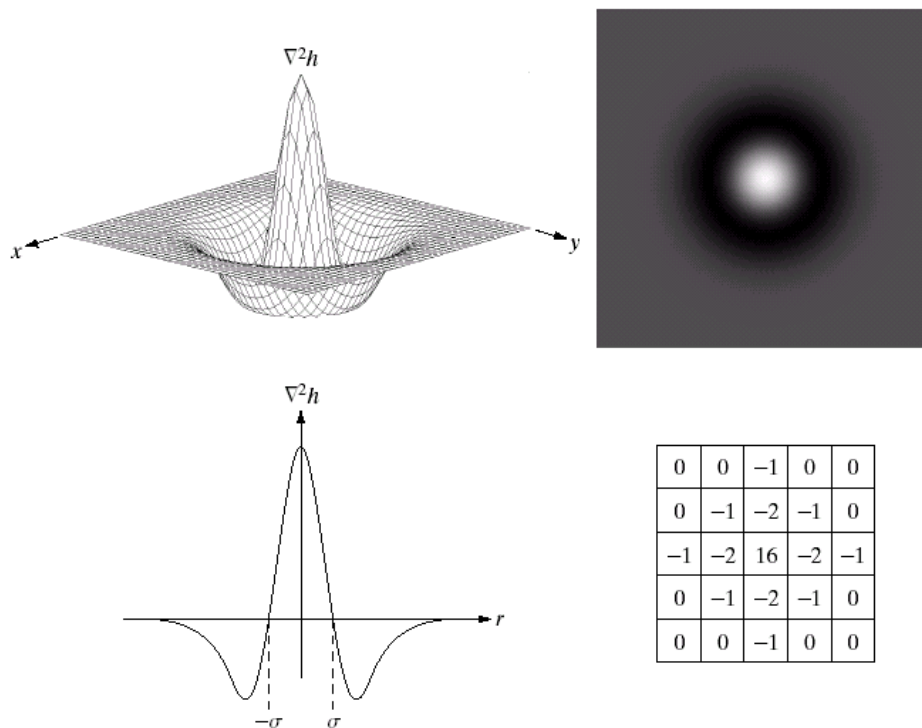
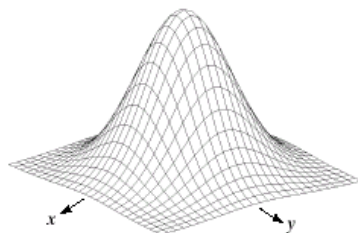


FIGURE 10.14
Laplacian of a Gaussian (LoG).
(a) 3-D plot.
(b) Image (black is negative, gray is the zero plane, and white is positive).
(c) Cross section showing zero crossings.
(d) 5×5 mask approximation to the shape of (a).

5.2 基于边界的边缘检测

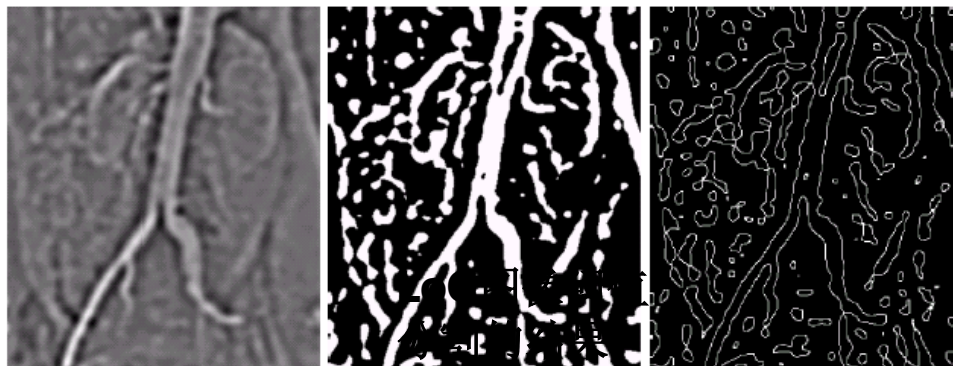


Sobel算子检测结果



-1	-1	-1
-1	8	-1
-1	-1	-1

LoG检测结果



零交叉点



5.2 基于边界的边缘检测

■ 边缘检测

■ 拉普拉斯算子和Sobel算子比较

■ 缺点

- 边缘由许多闭合环的零交叉点决定
- 零交叉点的计算比较复杂

■ 优点

- 零交叉点图像中的边缘比梯度边缘细
- 抑制噪声的能力和抗干扰能力比梯度算子强

■ 结论：梯度算子用的更多



5.2 基于边界的边缘检测

- 边缘检测

- 边缘检测的**Matlab**函数

- **BW = edge(I,'sobel')**
 - **BW = edge(I,'prewitt')**
 - **BW = edge(I,'roberts')**
 - **BW = edge(I,'log')**
 - **BW = edge(I,'zerocross',thresh,h)**

5.2 基于边界的边缘检测

■ Canny算子

- 1986年，Canny 提出了边缘检测算子应满足以下3个判断准则：信噪比准则，定位精确度准则，单边缘响应准则，并推导出了Canny算子。
- 实现步骤有四个子过程：
 - 第一步首先用二维高斯函数的一阶导数对图像进行平滑
 - 第二步用 2×2 邻域一阶偏导的有限方差来计算平滑后的数据阵列 $I(x,y)$ 的梯度幅值和梯度方向
 - 第三步，为了精确定位边缘，必须细化梯度幅值图像 $M(i, j)$ 中的屋脊带，只保留幅值局部变化最大的点，这一过程称为非极大值抑制
 - 最后在第四步，对经过非极大值抑制的数据阵列 $N(i, j)$ 分别使用高、低2个阈值 T_h 和 T_l 分割图像，得到两个阈值边缘图像



间断检测

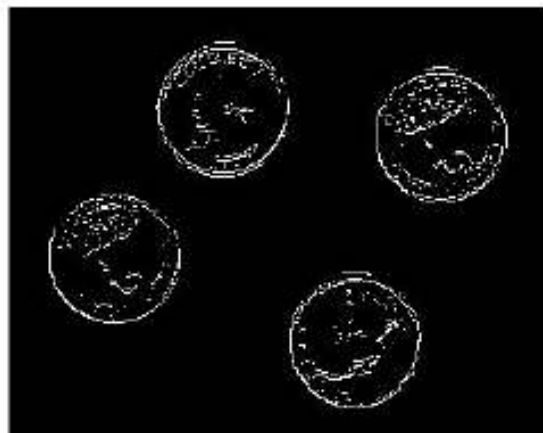
■ Canny算子的优缺点

- 该算法有较好的抑制噪声的能力，可以较完整的检测出边缘。
- 但比传统边缘微分算子复杂，运算速度慢。另外，**Canny** 算子的双阈值是根据全局特征信息来决定的，这导致了一方面无法消除局部噪声干扰，另一方面又会丢失灰度值变化缓慢的局部边缘。
- 可以通过改进双阈值的选取算法提高**Canny** 算子的边缘检测性能。

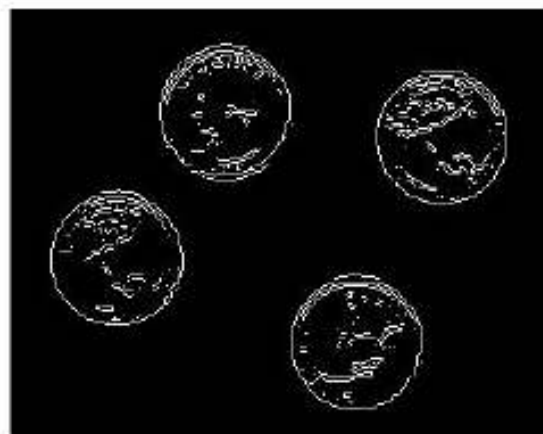
5.2 基于边界的边缘检测



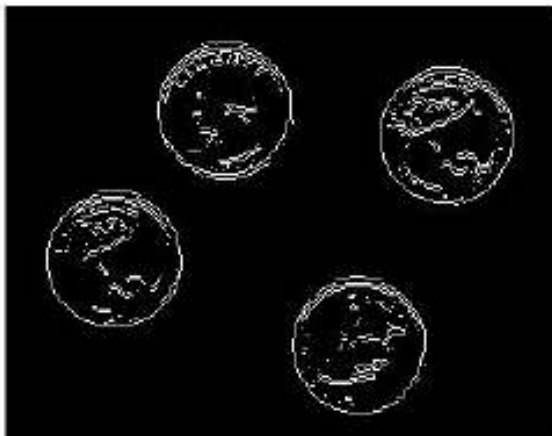
(a) 原始图像



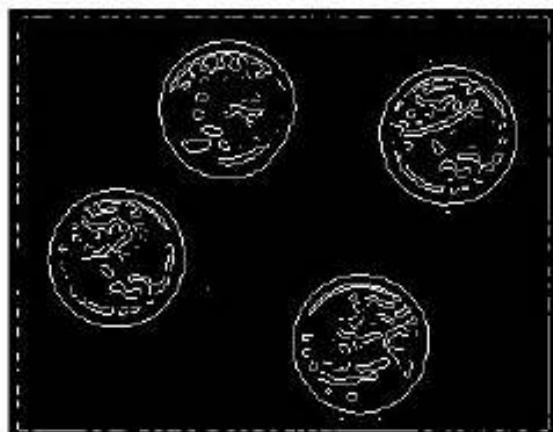
(b) Roberts 检测结果



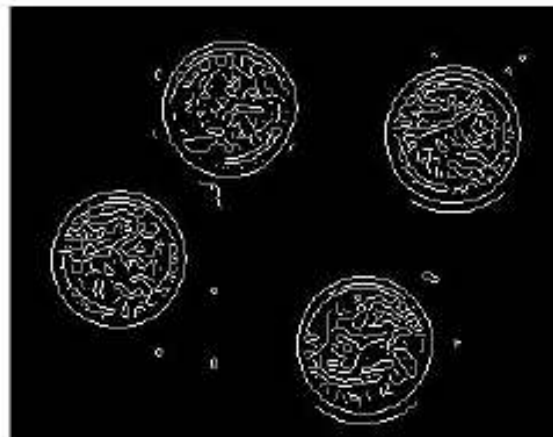
(c) Prewitt 检测结果



(d) Sobel 检测结果

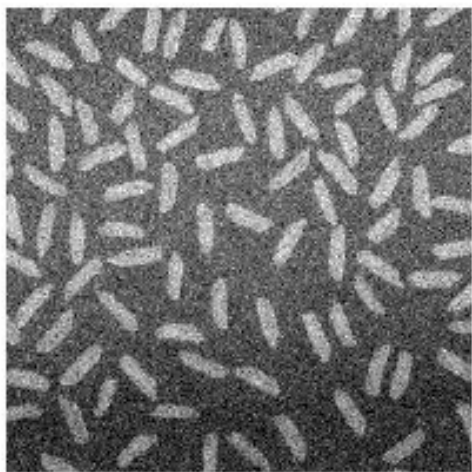


(e) LoG 检测结果

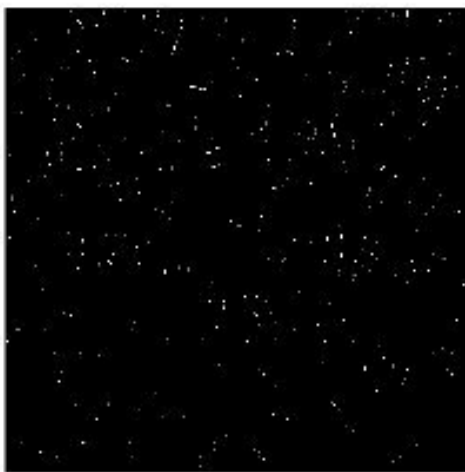


(f) Canny 检测结果

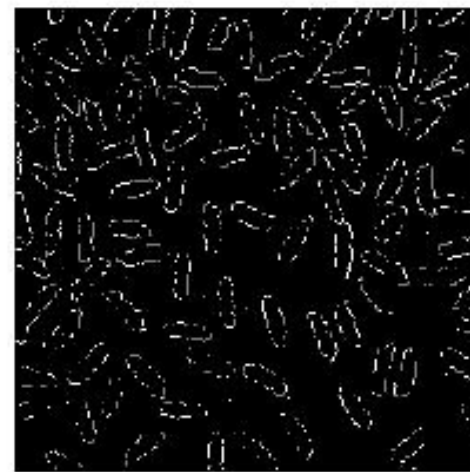
5.2 基于边界的边缘检测



(a) 高斯噪声图像



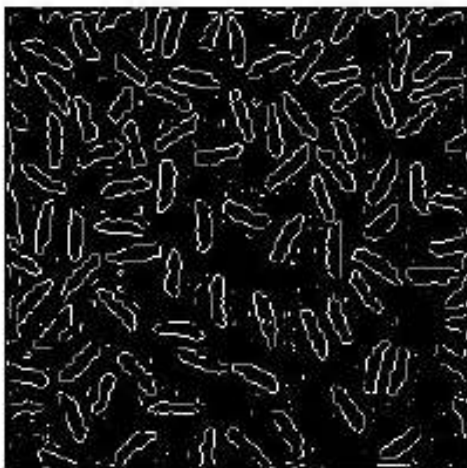
(b) Roberts 检测结果



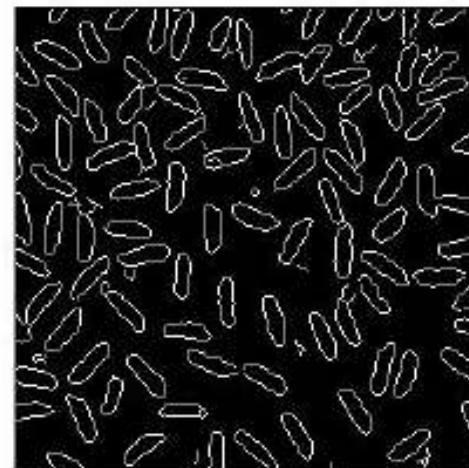
(c) Prewitt 检测结果



(d) Sobel 检测结果



(e) LOG 检测结果



(f) Canny 检测结果



5.2 基于边界的边缘检测

■ 边缘跟踪

- 由于噪声、不均匀照明等原因产生边缘间断，使得到的一组像素很少能完整地描绘一条边缘
- 典型的做法是在使用边缘检测算法之后，使用连接过程将边缘像素组合成有意义的边缘
- 局部处理

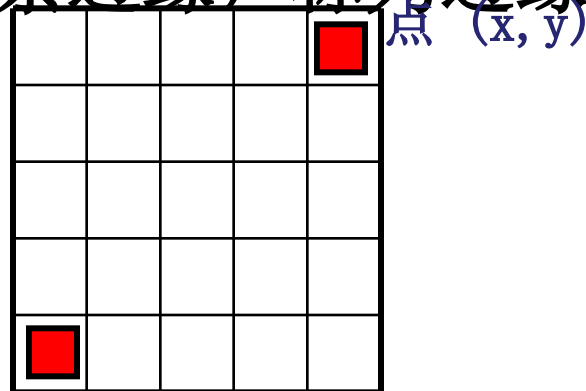
边缘连接和边界检测

■ 局部处理

- 分析图像中每个边缘点 (x, y) 的一个邻域内的像素，根据某种准则将所有相似点进行连接，由满足该准则的像素连接组成的一条边缘，称为边缘连接，又称为边缘跟踪。

■ 连接原则

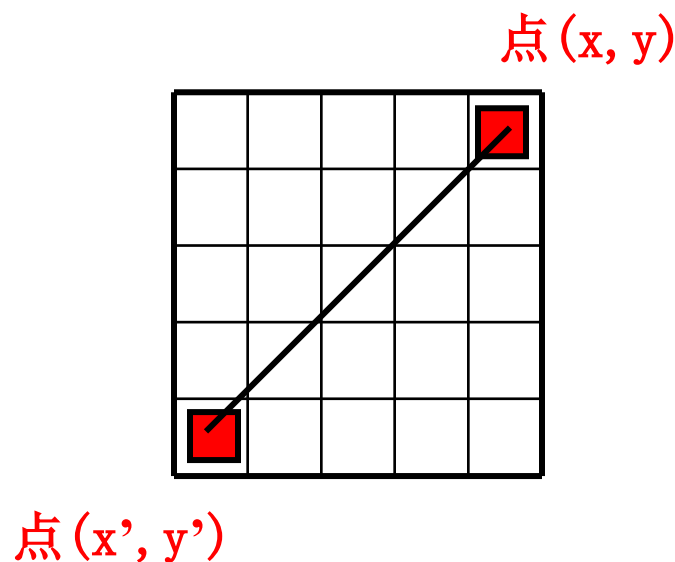
- 比较两个边缘点梯度算子的响应强度和梯度方向来确定两个点是否属于一条边



5.2 基于边界的边缘检测

连接算法步骤

- 1) 设定A、T的阈值大小，确定邻域的大小；
- 2) 对图像上每一个像素的邻域点进行分析，判断是否需要连接；
- 3) 记录像素连接的情况，另开一个空间，给不同的边以不同的标记；
- 4) 删除孤立线段，连接断开的线段。



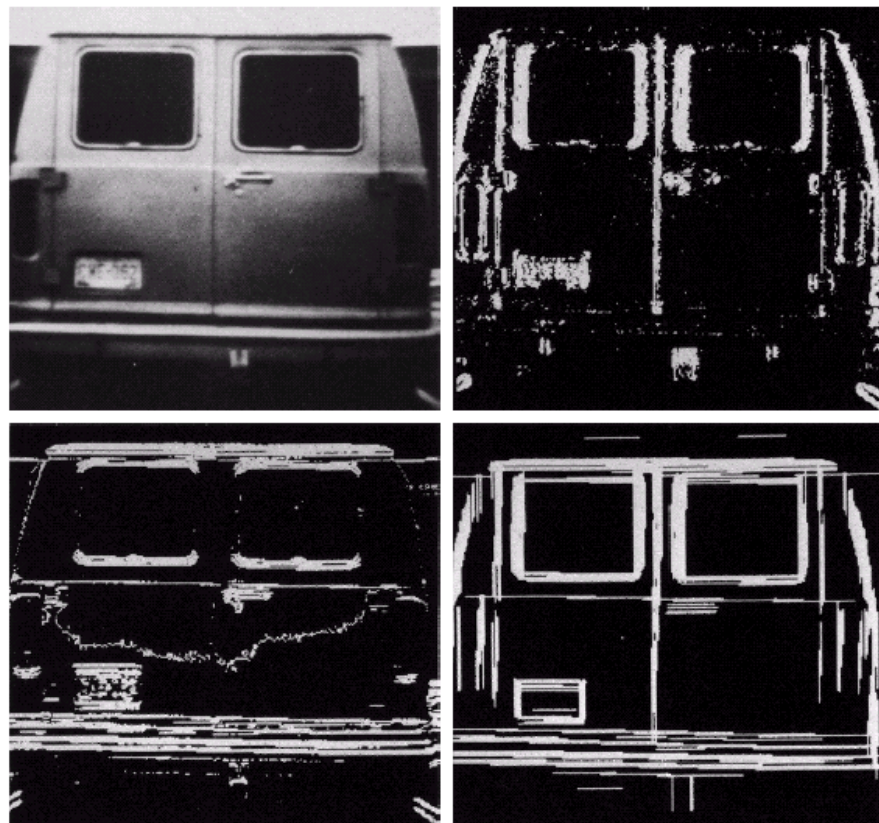
5.2 基于边界的边缘检测

■ 局部处理

- 如果梯度算子的响应强度和梯度方向都是相似的，则边缘点 (x, y) 和 (x', y') 是连接的

a b
c d

FIGURE 10.16
(a) Input image.
(b) G_y component of the gradient.
(c) G_x component of the gradient.
(d) Result of edge linking. (Courtesy of Perceptics Corporation.)





5.2 基于边界的边缘检测

- 边缘跟踪的方法很多，常见的有光栅跟踪和轮廓跟踪
- 光栅扫描跟踪
 - 光栅扫描跟踪是一种采用电视光栅行扫描顺序对遇到的像素进行分析，从而确定是否为边缘的跟踪方法。
 - 光栅跟踪方法的基本思想：
 1. 利用检测准则确定和接受对象点，
 2. 根据被接受的对象点和跟踪准则确定并接受新的对象点，
 3. 将所有标记为1且相邻的对象点联接起来就得到了检测到的细曲线。

5.2 基于边界的边缘检测

- 使用光栅跟踪方法，需要遵循下面的三个准则
 - 参数准则：需要事先确定检测阈值 d 、跟踪阈值 t ，且要求 $d > t$ ；
 - 检测准则：对图像逐行扫描，将每一行中灰度值大于或等于检测阈值 d 的所有点（称为接受对象点）记为1；
 - 跟踪准则：设置位于第 i 行的点 (i, j) 为已接受的对象点，如果位于第 $i + 1$ 行上的相邻点 $(i + 1, j - 1)$ 、 $(i + 1, j)$ 和 $(i + 1, j + 1)$ 的灰度值大于或等于跟踪阈值 t ，就将其接受为新的对象点，并记为1。

5.2 基于边界的边缘检测

光栅扫描跟踪具体步骤

- (1) 确定一个较大的阈值 d 为检测阈值，把高于该阈值的像素作为对象点。
- (2) 选择一个较低的阈值 t 作为跟踪阈值，且要求 $t < d$ ，该阈值可以根据不同准则来选择；可选择灰度差、梯度方向、对比度等作为跟踪阈值。
- (3) 从第一行起用检测阈值 d 逐行对图像进行扫描，依次将灰度值大于或等于检测阈值 d 的点的位置记为1。
- (4) 确定跟踪邻域，如下图中选取的 $(i+1, j-1)$ 、 $(i+1, j)$ 、 $(i+1, j+1)$ 。
- (5) 从第二行起逐行扫描图像，若图像中的 (i, j) 点为对象点，则在第 $i+1$ 行上找该点跟踪邻域中灰度差小于或等于跟踪阈值 t 的邻点，并确定为新的对象点，将相应位置记为1。
- (6) 对于已检测出来的某个对象点，进行跟踪结束、分支和合并的处理。如果某个对象点（由于步骤(3)的原因产生的对象点）在上一行的对应邻域中没有对象点，则说明一条新的曲线可开始。
- (7) 重复(5)、(6)这两个步骤，直至图像中最末一行被扫描完为止。

5.2 基于边界的边缘检测

9		5		8		4	5		3
	5				6		4	2	
		9		1	5			5	
6			5			7		2	
	3		6	2		9			7
4		6			5		3		5
		5		2	6			6	
6	4		3		7			4	2

(a)

1				1					
		1							
						1			
						1			1
						1			

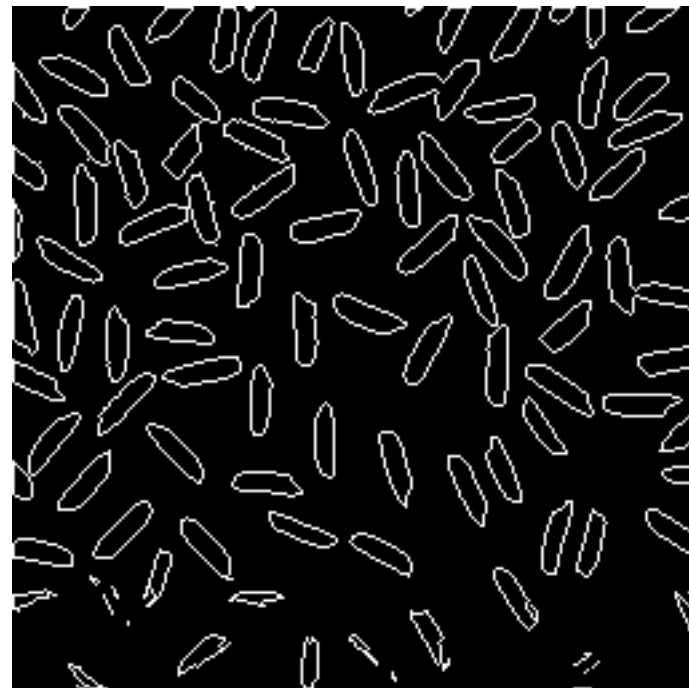
(b)

1				1					
	1				1				
		1			1				
			1			1			
			1			1			1
		1			1			1	
		1			1			1	
		1			1			1	
	1				1			1	

(c)

-  第一次跟踪后接受对象点
-  第二次跟踪后接受对象点
-  第三次跟踪后接受对象点
-  第四次跟踪后接受对象点
-  第五次跟踪后接受对象点

5.2 基于边界的边缘检测



(b) 光栅扫描法

图 采用光栅扫描法对米粒进行边缘跟踪

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/058037074137007000>