

# 红外图像处理技术 之图像分割

陆哲明/郑阳明  
浙江大学航空航天学院

## 内容

- 14.1 图像分割引言
- 14.2 边缘检测
- 14.3 边缘跟踪 → 边缘
- 14.4 Hough变换 → 边缘
- 14.5 区域分割
- 14.6 区域增长 → 区域
- 14.7 角点检测 → 角点

## 14.1 图像分割引言

### • 14.1.1 图像处理

✓ 着重强调在图像之间进行变换以改善图像的视觉效果。

### • 14.1.2 图像分析的概念

✓ 主要是对图像中感兴趣的目标进行检测和测量，以获得它们的客观信息从而建立对图像的描述，即从图像中提取信息的技术。

✓ 图像分析系统的基本构成



### • 14.1.3 图像理解

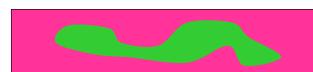
✓ 在图像分析的基础上，进一步研究图像中各目标的性质和它们之间的相互联系，并得出对原始客现场景的解释，从而指导和规划行动。

### • 14.1.4 区域

✓ 是指相互连通的、有一致属性的像素的集合。

✓ 是一个方便的、很好的图像高层描述符号，是对图像模型化和进行高层理解的基础。

✓ 为了辨识和分析目标，需要将它们分离提取出来，在此基础上才有可能对目标进一步分析。



### • 14.1.5 图像分割的概念

✓ 就是指把图像分解成构成它的区域并提取出感兴趣目标在图像中的位置和范围的技术和过程。

✓ 图像分割是由图像处理到图像分析的关键步骤：

➢ 是目标表达的基础，对特征测量有重要的影响。

➢ 将原始图像转化为更抽象更紧凑的形式。



✓ 图像分割的定义：图像分割是指将图像中具有特殊涵义的不同区域区分开来，这些区域是互相不交叉的，每一个区域都满足特定区域的一致性，即：

➢ 图像分割是将图像空间  $R$  划分为  $n$  个互不重叠的区域  $R_i$

$$R = \bigcup_{i=1}^n R_i, \quad R_i \cap R_j = \emptyset,$$

$$P(R_i) = \text{真}, \quad P(R_i \bigcup R_j) \neq \text{真}.$$

其中  $P(R_i)$  为作用于  $R_i$  中所有像素的相似性逻辑谓词。

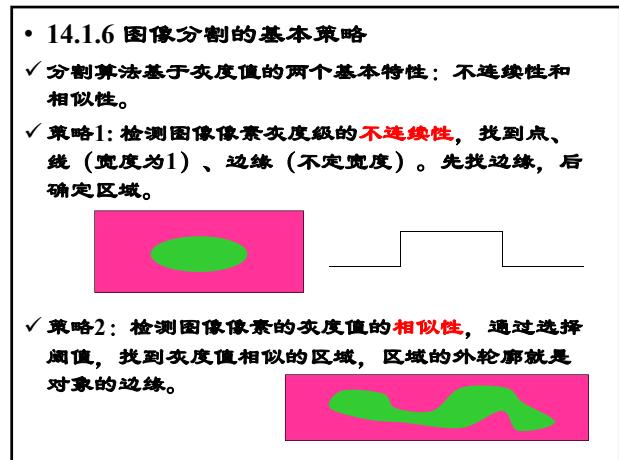
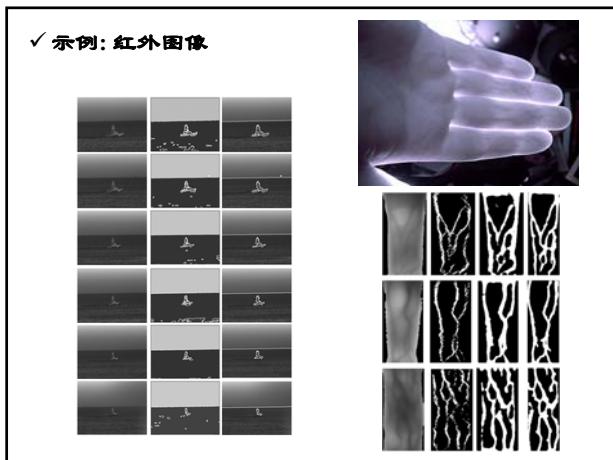
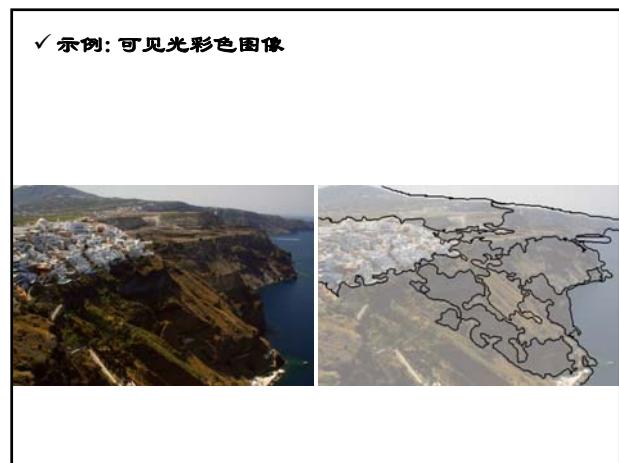
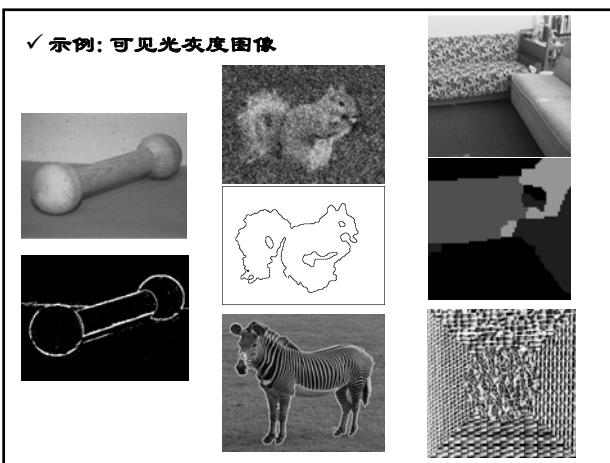
✓ 基本思路

➢ 从简到难，逐级分割。

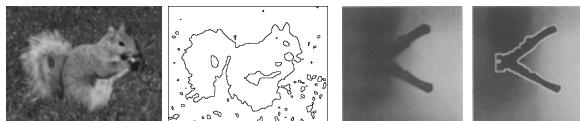
➢ 控制背景环境，降低分割难度。

➢ 把焦点放在增强感兴趣对象，缩小不相干图像成分的干扰上。





- 14.1.7 分割算法的大致分类（共五类）
  - 对图像特征空间做聚类的方法（软分割）
  - 基于区域的方法（区域分割、区域生长等）
  - 基于边缘的方法（边缘检测/主动边缘）
  - 综合考虑边缘和区域信息的混合分割方法
  - 基于函数优化的方法（如构造能量函数）



#### 14.1.8 今后主要的研究方向

- 提取有效的属性；
- 寻求更好的分割途径和分割质量评价体系；
- 分割自动化。

## 14.2 边界分割法(边缘检测)

### 14.2.1 概念

✓ **边缘(边界):**是指图像中像素灰度有阶跃变化或屋顶状变化的那些像素的集合。

✓ **边缘的性质:**边缘能勾划出目标物体轮廓，使观察者一目了然，包含了丰富的信息(如方向、形状等)，是图像识别中抽取的重要属性。

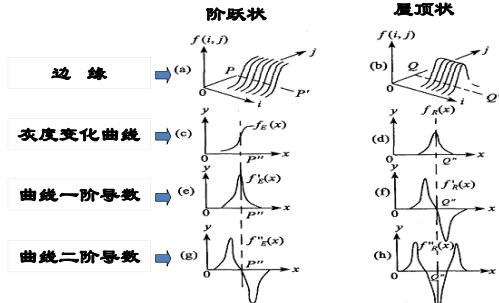
✓ **边缘分类:**阶跃状和屋顶状两种。

> 阶跃状边缘位于两边的像素灰度值有明显不同的地方；

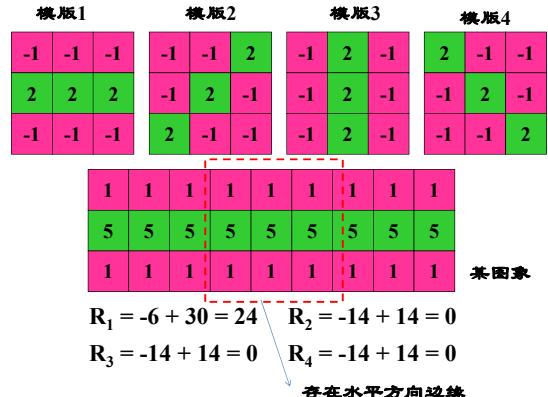
> 屋顶状边缘位于灰度值从增加到减少的转折处。



#### ➤ 阶跃状和屋顶状边缘局部灰度方向导数变化情况



#### ✓ 边缘检测的思路：计算局部微分算子(利用模板)



### 14.2.2 梯度算子

- 对阶跃状边缘，在边缘点处一阶导数有极值，因此可计算每个像素处的梯度来检测边缘点。
- 梯度的定义式及其简化计算
  - 函数  $f(x,y)$  在  $(x,y)$  处的梯度为一个向量：  
 $\nabla f = [\partial f / \partial x, \partial f / \partial y]$
  - 计算这个向量的大小为：  
 $\nabla f = \text{mag}(\nabla f) = [(\partial f / \partial x)^2 + (\partial f / \partial y)^2]^{1/2}$   
 近似为： $\nabla f \approx (\partial f / \partial x) + (\partial f / \partial y)$   
 或： $\nabla f \approx \text{Max}[(\partial f / \partial x), |\partial f / \partial y|]$
  - 梯度的方向角为：  
 $\alpha(x,y) = \tan[(\partial f / \partial y) / (\partial f / \partial x)]$
  - 高斯形式和模板形式：一阶差分

(x,y)	(x,y-1)
(x-1,y)	(x,y)

(x-1,y-1)	(x,y-1)
(x-1,y)	(x,y)
(x,y+1)	(x,y)

-1	1
1	
1	-2
	1

✓ 梯度的大小代表边缘的强度，梯度方向与边缘走向垂直。

#### ✓ 利用梯度检测边缘点

- 计算梯度图像  $\nabla f = [(\partial f / \partial x)^2 + (\partial f / \partial y)^2]^{1/2}$ ；
- 选取适当的阈值  $T$ ，对梯度图像进行二值化；
- 当  $\nabla f(x, y) \geq T$  时，令  $G(x, y) = 1$ ，则为阶跃状边缘点。否则，令  $G(x, y) = 0$ ，形成一幅边缘二值图像  $G(x, y)$ 。



$$G(x, y) = \begin{cases} 1, & |\nabla f(x, y)| \geq T \\ 0, & |\nabla f(x, y)| < T \end{cases}$$

-1	1	1	1
1	5	5	5
1	1	1	1

#### • 14.2.3 Roberts梯度算子

✓ 差分表达式如下：

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

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

✓ Roberts梯度算子对应的模板：

-1	
	1

	-1
1	

✓ Roberts梯度计算：

$$\text{按 } \nabla f \approx \left| \left( \frac{\partial f}{\partial x} \right) + \left( \frac{\partial f}{\partial y} \right) \right|$$

$$\text{或: } \nabla f \approx \text{Max} \left\{ \left| \frac{\partial f}{\partial x} \right|, \left| \frac{\partial f}{\partial y} \right| \right\}$$

计算Roberts梯度。

✓ Roberts梯度算子特点：

➢ Roberts算子进行边缘检测的同时去噪作用仍然小，但效果较梯度算子好。

#### • 14.2.4 Prewitt和Sobel算子

✓ Prewitt算子

➢ 算子思想：

□ 在检测边缘的同时减少噪声的影响，Prewitt从加大边缘检测算子的模板大小出发，由 $2 \times 2$ 扩大到 $3 \times 3$ 来计算差分算子。

➢ Prewitt算子对应的模板：

$$g(x, y) = \max \{g_1(x, y), g_2(x, y)\}$$

$$g_p(x, y) = \sum_{k=-1}^1 \sum_{l=-1}^1 h_p(k, l) f(x+k, y+l) \quad h_1 \quad h_2$$

➢ Prewitt算子特点：  $p=1, 2$

□ Prewitt算子检测边缘点的同时能抑制噪声的影响。

-1	0	1	-1	-1	-1
-1	0	1	0	0	0
-1	0	1	1	1	1

✓ Sobel算子

➢ 算子思想：

□ Sobel在Prewitt算子的基础上，对4邻域采用带权的方法计算差分。

➢ Sobel算子对应的模板：

-1	0	1	-1	-2	-1
-2	0	2	0	0	0
-1	0	1	1	2	1

$$g(x, y) = \max \{g_1(x, y), g_2(x, y)\}$$

$$g_p(x, y) = \sum_{k=-1}^1 \sum_{l=-1}^1 h_p(k, l) f(x+k, y+l) \quad h_1 \quad h_2$$

➢ Sobel算子特点：

□ Sobel算子检测边缘点的同时能进一步抑制噪声的影响，但检测的边缘较宽。

#### • 14.2.5 方向算子

✓ 算法思想：利用一组模板分别计算在不同方向上的差分值，取其中最大的值作为边缘强度，而将与之对应的方向作为边缘方向。

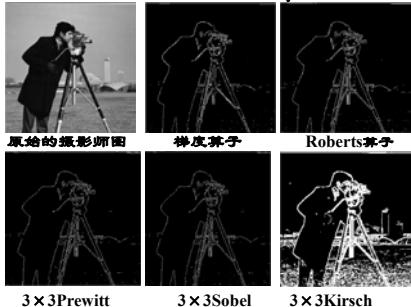
✓ Kirsch ( $3 \times 3$ ) 八方向模板：各方向间的夹角为 $45^\circ$ 。

-5	3	3	3	3	3	3	3
-5	0	3	-5	0	3	3	0
-5	3	3	-5	-5	3	-5	-5
3	3	-5	3	-5	-5	-5	3
3	0	-5	3	0	-5	3	0
3	3	-5	3	3	3	3	3
3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3

➢ 如果取最大值的绝对值为边缘强度，并用考虑最大值符号的方法来确定相应的边缘方向，则考虑到各模板的对称性只要有前四个模板即可。

✓ 算子梯度计算：设图像 $f$ ，模板为 $W_k (k=1, 2, \dots, 8)$ ，则边缘强度在 $(x, y)$ 处为(以·表示点乘): $E(x, y) = \text{Max}\{W_1 f, W_2 f, \dots, W_8 f\} = W_i f$ , 方向为 $i$ 模板方向。

✓ 下例中几种梯度算子检测出的边缘二值图表明 $3 \times 3$ 的算子比 $2 \times 2$ 的算子边缘检测能力强，且抗噪性能好。



#### • 14.2.6 Laplace算子

✓ Laplace算子：

➢ 二维函数 $f(x, y)$ 的拉普拉斯是一个二阶的微分定义为：

$$\nabla^2 f(x, y) = [\frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial y^2}]$$

➢ 可以用多种方式被表示为数学形式。对于一个 $3 \times 3$ 的区域，经验上被推荐最多的形式是：

$$\nabla^2 f(x, y) = 4f(x, y) - f(x+1, y) - f(x-1, y) - f(x, y+1) - f(x, y-1)$$

✓ Laplace算子对应的模板：

➢ 定义数学形式的拉普拉斯的基本要求是：作用于中心像素的系数是一个正数，而且其周围像素的系数为负数，系数之和必为0。

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

✓ Laplace算子特点分析：

- 优点：各向同性，对细线和孤立点检测效果好。
- 缺点：对噪声敏感，会产生双像素的边缘，不能检测出边的方向。
- 应用：拉普拉斯算子不直接用于边的检测，通常只起第二位的角色。

由于梯度算子和Laplace算子都对噪声敏感，因此一般在用它们检测边缘前要先对图像进行平滑。

• 14.2.7 马尔算子

✓ 问题：

- 在原始图像上进行边缘检测，由于噪声的影响，可能把噪声当边缘点检测出来了，而真正的边缘又没被检测出来。

✓ 马尔算子(Marr)思想：

- 由于拉普拉斯算子对噪声比较敏感，为了减少噪声影响，可先对待检测图进行平滑然后再用拉普拉斯算子检测边缘。

✓ 平滑函数：

- 由于在成像时，一个给定像素所对应的场景点，它的周围点对该点的光强呈正态分布，所以平滑函数应反映不同远近的周围点对给定像素具有不同的平滑作用，呈正态分布的平滑函数可如下定义：

$$h(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

➢ 用 $h(x, y)$ 对图像 $f(x, y)$ 的平滑可表示为：

$$g(x, y) = h(x, y) * f(x, y) \text{ *--卷积}$$

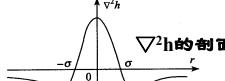
- 令 $r$ 是离原点的径向距离，即 $r^2 = x^2 + y^2$ ，对图像 $g(x, y)$ 采用拉普拉斯算子进行边缘检测，可得：

$$\begin{aligned} \nabla^2 g &= \nabla^2 [h(x, y) * f(x, y)] = \left(\frac{r^2 - \sigma^2}{\sigma^4}\right) \exp\left(-\frac{r^2}{2\sigma^2}\right) * f(x, y) \\ &= \nabla^2 h * f(x, y) \end{aligned}$$

- 利用二阶导数算子过零点的性质，可确定图像中阶梯状边缘的位置。

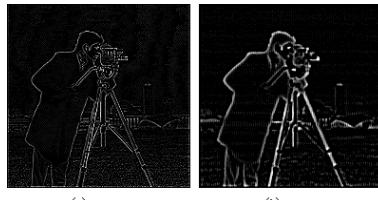
✓  $\nabla^2 h$ 高斯—拉普拉斯滤波算子：轴对称，各向同性，截面称为“墨西哥草帽”。

$$\nabla^2 h = \left(\frac{r^2 - \sigma^2}{\sigma^4}\right) \exp\left(-\frac{r^2}{2\sigma^2}\right)$$



- 这个函数在 $|r|=0$ 处有过零点，在 $|r|<\sigma$ 时为正，在 $|r|>\sigma$ 时为负。
- 这个算子定义域内的平均值为零，因此将它与图像卷积并不会改变图像的整体动态范围。
- 由于它相当光滑，将它与图像卷积会模糊图像，并且其模糊程度是正比于 $\sigma$ 的。
- 正因为 $\nabla^2 h$ 的平滑性质能减少噪声的影响，所以当边缘模糊或噪声较大时，利用 $\nabla^2 h$ 检测过零点能提供较可靠的边缘位置。
- 该算子中， $\sigma$ 选择很重要， $\sigma$ 小时位置精度高但边缘细节变化多，应注意马尔算子用于噪声较大的区域会产生高密度的过零点。

- ✓ 马尔算子用到的卷积模板一般较大(典型半径为8-32个像素)，不过这些模板可以分解为一维卷积来快速计算。数学上已证明，马尔算子是按零交叉检测阶梯边缘的最佳算子。但在实际图像中，高斯滤波的零交叉点不一定全部是边缘点，还需要进一步对其真伪进行检验。



(a)、(b)分别是用Laplace算子和Marr算子检测出的边缘二值化图像。

14.2.8 曲面拟合法

✓ 问题

- 解决基于差分检测图像边缘算子对噪声的敏感，检测一些噪声比较严重的图像边缘。

✓ 原理

- 用平面或高阶曲面来拟合图像中某一小区域的灰度表面，求这个拟合平面或曲面的外法线方向的微分或二阶微分检测边缘，可减少噪声影响。

✓ 四点拟合灰度表面法

➢  $p=ax+by+c$  来拟合空间四邻像素的灰度  $f(x,y)$ ,  $f(x,y+1)$ ,  $f(x+1,y)$ ,  $f(x+1,y+1)$

$$\frac{\partial p}{\partial x} = a, \frac{\partial p}{\partial y} = b$$

(x,y)	(x+1,y)
(x,y+1)	(x+1,y+1)

➢  $a$  和  $b$  对应的模板如下：

$$\frac{1}{2} \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}, \quad \frac{1}{2} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}$$

□  $a$  为两行像素平均值的差分,  $b$  为两列像素平均值的差分。其过程是求平均后再求差分, 因而对噪声有抑制作用。



Laplace算子 Marr算子 曲面拟合法

✓ 思考题

1. 提取下图边缘可采用什么方法?
2. 用Matlab图像工具包体验一下各边缘检测算子的效果。
3. 用c/c++在Matlab/VC中实现一个边缘检测算子, 比较其与Matlab提取的效果是否相同。



### 14.3 边缘跟踪(边缘连接)

- 14.3.0 引出: 不大可能总是检测出连续的边缘



### 14.3 边缘跟踪(边缘连接)

#### • 14.3.1 边缘跟踪概念

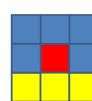
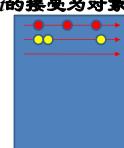
- ✓ 定义: 将检测的边缘点连接成线就是边缘跟踪。
- ✓ 过程: 提取可构成线特征的边缘; 将边缘连成线。
- ✓ 意义:
  - 线是图像分析中一个基本而重要的内容, 它是图像的一种中层符号描述, 它使图像的表达更简洁, 并可用来完成一定图像的识别任务。
  - 由于噪音的原因, 边界的特征很少能够被完整地描述, 在亮度不一致的地方会中断。
  - 因此典型的边缘检测算法后面总要跟着连接过程和其它边界检测过程, 用来归整边缘像素, 成为有意义的边。
- ✓ 连接法: 局部处理法; Hough变换

#### • 14.3.2 光栅跟踪

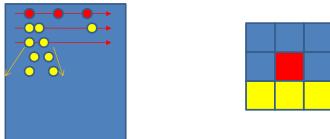
✓ 利用类似于电视光栅扫描技术, 结合阈值检测而实现的边缘跟踪方法。

##### ✓ 光栅跟踪步骤:

- 1. 确定一个比较高的阈值  $d$  作为 **检测阈值** 和一个比较低的阈值  $t$  作为 **跟踪阈值**。
- 2. 用检测阈值  $d$  对当前行像素进行检测, 凡超过  $d$  的点都接受为**对象点**, 并作为下一步**跟踪的起始点**。
- 3. 对当前点(起始点)的八邻点的下面三个点进行判断, 大于跟踪阈值  $t$  的接受为**对象点**, 反之去除。



- 4. 如果在下一行像素中，对应上一行已检测出的某一对象点，没有任何一个邻域像素被接受为对象点，那么，这一条曲线的跟踪便可结束。
- 5. 如果同时有两个，甚至三个邻域点均被接受为对象点，则说明曲线发生分支，跟踪将对各分支同时进行。如果若干分支曲线合併成一条曲线，则跟踪可集中于一条曲线上进行。
- 6. 一曲线跟踪结束后，采用类似上述步骤从第一行的其他起始点开始下一条曲线的跟踪。
- 7. 当所有起始点跟踪完毕，跟踪即结束。



✓ 例 输入图像 (可能是梯度图象)

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

$f \geq 7$ 二值化

1		1			
1				1	
				1	1
					1

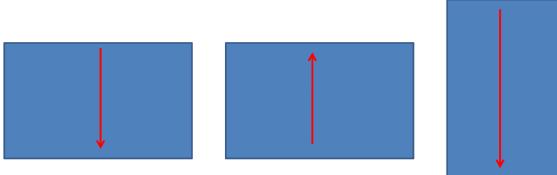
$d=7, t=4$ 光栅跟踪

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

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

✓ 注意：

- 光栅顺序跟踪和光扫描的方向有关。最好用其他方向再跟踪一次，例如逆序向上，两种方法结合起来能得到更好的方向跟踪。
- 若边缘和光栅扫描方向平行时效果不好。最好在垂直扫描方向跟踪一次，它相当于把图像转置90°后再顺序跟踪。



• 14.3.3 全向跟踪

- ✓ 解决光栅跟踪依赖于扫描，漏掉平行于扫描方向曲线的缺点。

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

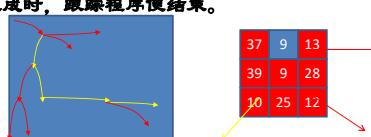
- ✓ 通过定义不同邻点的方法克服光栅跟踪依赖于扫描的缺点。选取的跟踪准则能够辨别远非紧靠着被跟踪对象点的像素，适当克服光栅跟踪会漏掉平行于扫描方向曲线的缺点。

✓ 全向跟踪的具体步骤：

- 按光栅扫描方式对图像进行扫描，用检测阀值找出一个作为开始跟踪的初选点(流动点)。
- 选取一个适当的、能进行全向跟踪的邻域定义(例如八邻域)和一个适当的跟踪准则(例如次度阀值、对比度和相对流动点的距离等)，对流动点进行跟踪。



- 在跟踪过程中，若遇到了分支点或者若干曲线的交点，则先取其中和流动点性质最接近的一个作为新的流动点，继续进行跟踪。而把其余诸点存储起来，以备下面继续跟踪。
- 如果在跟踪过程中又遇到了新的分支或交叉点，则重复上面的处理步骤。
- 当按照跟踪准则没有未被检测过的点可接受为对象点时，一个分支曲线的跟踪便结束。在一个分支曲线跟踪完毕以后，回到最近的一个分支点处，取出另一个性质最接近该分支点的像素作为新的流动点，重复上述跟踪程序。
- 当全部分支点处的全部待跟踪点均已跟踪完毕，便返回第一步，继续扫描，以选取新的初选点(流动点)(不应是已被接收为对象的点)。
- 当整幅图像扫描完成时，跟踪程序便结束。



✓ 分支问题解决：

- 通过一点(x, y)有几条曲线，可设立一准则φ，对每一分支计算一个φ值，应按大的φ值进行跟踪。

➤ 例：选取φ为平均灰度减去平均曲度率。

□ 设第一行流动点  $f(x,y)=5$ , 第二行  $f(x,y+1)=5$ , 第三行  $f(x+1,y+1)=7, f(x+1, y+2)=4$ , 形成一个分支点, 灰度顺序为5, 5, 7 和5, 5, 4两个分支。

□ 计算：

$$\phi(5,5,7)=(5+5+7)/3-0=5.67$$

$$\phi(5,5,4)=(5+5+4)/3-(\pi/4)/3=4.4$$

□ 结论：对“5,5,7”跟踪，暂时保留“5,5,4”

20	5	10
39	5	28
25	7	4

#### • 14.3.4 局部连接处理

##### ✓ 连接处理的时机和目的：

➢ 时机：对做过边界检测的图像进行

➢ 目的：连接间断的边

✓ 连接处理的原理：比较梯度算子的响应强度和梯度方向确定两个点是否同属一条边

➢ 通过比较梯度，确定两个点的连接性：对于点 $(x',y')$ ，判断其是否与邻域内的边界点 $(x,y)$ 相似，当：

$$|\nabla f(x,y) - \nabla f(x',y')| \leq T \text{ 其中 } T \text{ 是一个非负的阈值}$$

➢ 比较梯度向量的方向角：对于点 $(x',y')$ ，判断其是否与邻域内的边界点 $(x,y)$ 的方向角相似，当：

$$|\alpha(x,y) - \alpha(x',y')| \leq A \text{ 其中 } A \text{ 是一个角度阈值}$$

➢ 当梯度值和方向角都是相似的，则点 $(x',y')$ 与边界点 $(x,y)$ 是连接的

#### ✓ 局部连接算法描述：

➢ 设定 $A$ 、 $T$ 的阈值大小，确定邻域的大小

➢ 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。

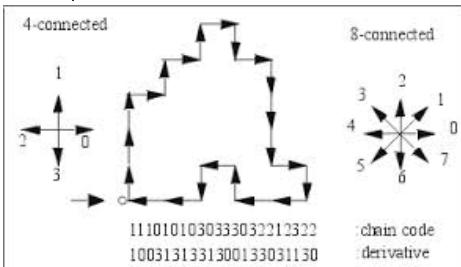
➢ 记录像素连接的情况，另开一个空间，给不同的边以不同的标记。

➢ 最后，删除孤立线段，连接断开的线段。



#### • 14.3.5 链码跟踪

• 链码（又称为freeman码）是用曲线起始点的坐标和边界点方向代码来描述曲线或边界的方法，常被用来在图像处理、计算机图形学、模式识别等领域中表示曲线和区域边界。它是一种边界的编码表示法，用边界方向作为编码依据，为简化边界的描述，一般描述的是边界点集。



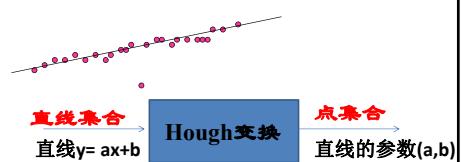
## 14.4 Hough变换

#### • 14.4.1 Hough变换含义

✓ Hough变换是一种线描述方法，即线到点的变换。

✓ Hough变换问题的提出

➢ 在找出边界点集之后，需要连接，形成完整的边界图形描述



➢ 例：找出黑白点状图像过原点的直线？(第一象限)

➢ 解答：

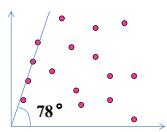
□ 若直线过原点，则 $y=mx$ ,  $m$ 为斜率。

□ 取 $\tan\theta_i = m_i = y/x_i$ 作为判据，计算图像中所有点的 $\theta_i$ 。

□ 在内存中为 $\theta_i$ 开辟一个存储区，共有91个地址，相当于 $0^\circ \sim 90^\circ$ 。

□ 统计每个 $\theta_i$ 地址中有多少个像素，其中 $\theta_i$ 点数最多的地址中的点就是可连成一条直线的点。

□ 以上可扩展到通过某点的直线检测。



0	0
1	1
...	...
$\theta_{78}$	78 4
...	...
90	0

#### • 14.4.2 Hough变换的起源

✓ 背景问题：对于 $n$ 个边缘点构成的点集，找出共线的点集和对应的直线方程。

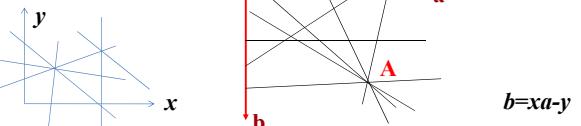
✓ 对于任意两点的直线方程： $y = ax + b$ ，构造一个参数 $ab$ 的平面，从而有如下结论：

➢  $xy$ 平面任选一条直线，对应在参数 $ab$ 平面上都有一个点。

➢ 过 $xy$ 平面点 $(x_0, y_0)$ 的所有直线，构成参数 $ab$ 平面的一条直线。

➢ 如果点 $(x_1, y_1)$ 与点 $(x_2, y_2)$ 共线，那么这两点在参数 $ab$ 平面上的直线将有一个交点。

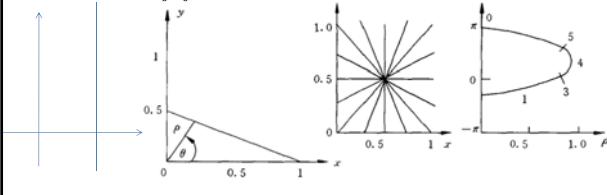
➢ 在参数 $ab$ 平面上相交直线最多的点，对应的 $xy$ 平面上的直线就是我们的解。



### • 14.4.3 Hough变换算法实现

- ✓ 由于垂直直线,  $\theta$ 为无穷大, 我们改用极坐标形式:  

$$x \cos \theta + y \sin \theta = \rho$$
- ✓ 参数平面为  $\theta, \rho$ , 对应不是直线而是正弦曲线
- > 在直角坐标系中过任一点  $(x_0, y_0)$  的直线系, 在极坐标系中所对应的点  $(\rho, \theta)$  构成一条正弦曲线。
- > 反之, 在极坐标系中位于这条正弦曲线上的点, 对应直角坐标系中过点  $(x_0, y_0)$  的一条直线。

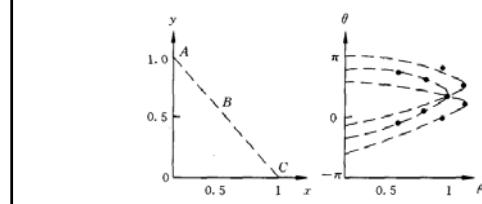


> 设平面上有若干点, 过每点的直线系分别对应于极坐标上的一条正弦曲线。若这些正弦曲线有共同的交点  $(\rho', \theta')$ , 则这些点共线, 且对应的直线方程为:

$$\rho' = x \cos \theta' + y \sin \theta'$$

这就是Hough变换检测直线的原理。

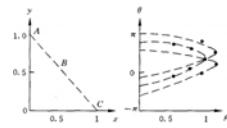
- ✓ 使用交点累加器, 或交点统计直方图, 找出相交线段最多的参数空间的点, 然后找出该点对应的xy平面的直线。



### ✓ Hough变换检测直线的算法步骤

- > 在  $\rho, \theta$  的极值范围内对其进行  $m, n$  等分, 设一个二维数组的下标与  $\rho, \theta$  的取值对应;
- > 对图像上的所有边缘点坐标  $(x, y)$  作Hough变换:求每个点在  $\theta_j (j=0, 1, \dots, n)$  时的  $\rho$  值, 并找到最近的  $\rho_j$  值, 判断  $(\rho_j, \theta_j)$  与哪个数组元素对应, 则让该数组元素值加1;
- > 比较数组元素值的大小, 最大值所对应的  $(\rho_j, \theta_j)$  就是这些共线点对应的直线方程的参数。
- > 共线方程为:  $\rho_j = x \cos \theta_j + y \sin \theta_j$
- ✓ 算法特性
- > 对  $\rho, \theta$  量化过粗, 直线参数就不精确, 过细则计算量增加。因此, 对  $\rho, \theta$  量化要兼顾参数量化精度和计算量。
- > Hough变换的抗噪性能强, 能将断开的边缘连接起来。
- > Hough变换也可用来检测曲线。

	$p_1$	$p_2$	$p_3$	$p_4$
$\theta_1$	13	0	5	2
$\theta_2$	1	1	2	3
$\theta_3$	4	4	7	9
$\theta_4$	12	0	0	0



### • 14.4.4 Hough变换的扩展

- ✓ Hough变换不只对直线, 也可以用于圆, 椭圆等。
- ✓ 例如
- > 有一曲线方程为:  $Ax^2 + By^2 = C$ , 在此式中  $x, y$  是变量,  $A, B, C$  是系数。
- > 把上式写成如下形式:  $x^2A + y^2B = C$ , 把  $A, B, C$  看成变量, 把  $x^2, y^2$  看成系数。
- >  $(x, y)$  域中的任何一点将对应于变换域中的一个曲面。
- >  $(x, y)$  域中椭圆上的  $n$  个点将对应于变换域中  $n$  个有共同交点的  $n$  个曲面。
- ✓ Hough变换检测圆:
- > 圆的方程:  $(x - c_1)^2 + (y - c_2)^2 = c_3^2$
- > 需要三个参数的参数空间。

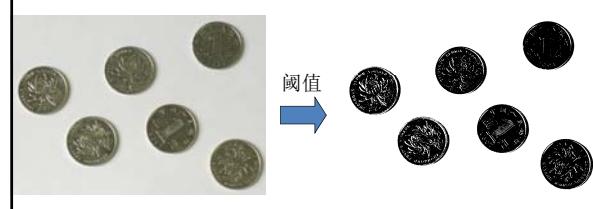
## 14.5 区域分割

### • 14.5.1 区域分割思想

- ✓ 一幅图像中属于同一区域的像素应具有相同或相似的属性, 不同区域的像素属性不同。
- ✓ 因此图像的分割就要寻求具有代表性的属性, 利用这类属性进行划分, 使具有相同属性的像归属同一区域, 不同属性的像素归属不同区域。

### • 14.5.2 阈值分割法

- ✓ 考虑角度: 前景与背景分割, 一个暗一个亮, 可采用阈值分割
- ✓ 问题: 如何求最优分割阈值?



✓ 阈值分割法的基本思想：

- 确定一个合适的阈值  $T$
- (阈值选定的好坏是此方法成败的关键)。
- 将大于等于阈值的像素作为物体或背景，生成一个二值图像。

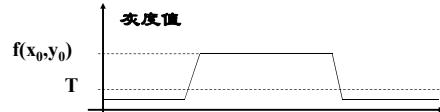
If  $f(x,y) \geq T$  set 255

Else set 0

0	0	255
0	255	255
255	255	255

✓ 阈值分割法的特点：

- 适用于物体与背景有较强对比的情况，重要的是背景或物体的灰度比较单一。（可通过先求背景，然后求反得到物体）
- 这种方法总可以得到封闭且连通区域的边界。

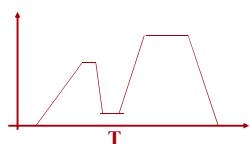


✓ 通过直方图得到阈值

- 基本思想：边界上的点的灰度值出现次数较少。

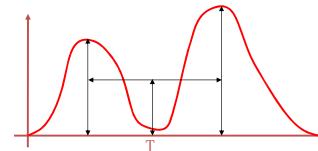


- 最简单的方法：取对应直方图谷底的灰度值为阈值  $T$ 。



- 缺点：会受到噪音的干扰，最小值不是预期的阈值，而偏高期望的值。

- 改进：取两个峰值之间某个固定位置，如中间位置上。由于峰值代表的是区域内外的典型值，一般情况下，比选谷底更可靠，可排除噪音的干扰。
- 对噪音的处理：对直方图进行平滑处理，如最小二乘法等不过点插值。



➢ 最大熵算法

$$\text{熵: } H = \sum_{s \in [0,255]} p(s) \log \frac{1}{p(s)}$$

- 物理含义：分布的“均匀性”

» 越均匀，熵越大

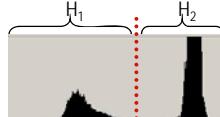
» 越不均匀，熵越小

➢ 图像假设：前景/背景亮度不同

◆ 直方图如同时包含前景和背景，则其熵将变小

◆ 如只包含前景或背景，熵将变大

➢ 把直方图分解成两部分，分别计算熵，两部分熵的和最大时为最佳分割阈值



$$\hat{T} = \arg \max_{T \in [0,255]} \{H_1 + H_2\}$$

➢ 实际应用：指纹图像处理



```
I = imread('Inputimage.bmp');
J = entropy(I);
```

✓ 通过边界特性选择阈值

➢ 基本思想：

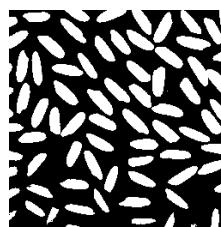
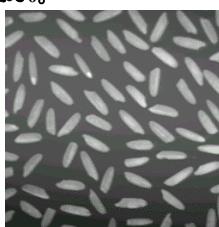
- 如果直方图的各个波峰很高、很窄、对称，且被很深的波谷分开时，有利于选择阈值。
- 为了改善直方图的波峰形状，我们只把区域边缘的像素绘入直方图，而不考虑区域中间的像素。
- 用微分算子处理图像，使图像只剩下边界中心两边的值。
- 方法优点：
- 在前景和背景所占区域面积差别很大时，不会造一个灰度级的波峰过高，而另一个过低。
- 边缘上的点在区域内还是区域外的概率是相等的，因此可以增加波峰的对称性。
- 基于梯度算子选择的像素，可以增加波峰的高度。

➢ 算法的实现：

- 对图像进行梯度计算，得到梯度图像。
- 得到梯度值最大的那一部分（比如10%）的像素直方图。
- 通过直方图的谷底，得到阈值T。

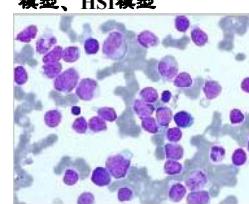
✓ 简单全局阈值分割

- 基本思想：用前述方法获得阈值T，并产生一个二值图，区分出前景对象和背景。
- 算法实现：
- 规定一个阈值T，逐行扫描图像。
- 凡灰度级大于T的，颜色置为255；凡灰度级小于T的，颜色置为0。



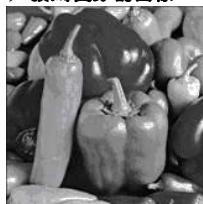
✓ 基于多个变量的阈值

- 基本思想：把前面的方法扩展到多维空间，则寻找波谷的过程，变为寻找点簇的过程。
- 算法实现：各维分量波谷之间进行逻辑与运算，从波谷重合的点，得到实际的阈值T。
- 应用场合：有多个分量的颜色模型，如RGB模型、CMYK模型、HSI模型



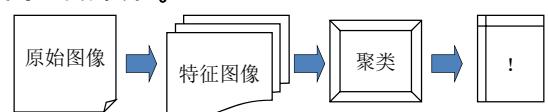
• 14.5.3 复杂图像的区域分割法

- ✓ 基本思想：在全灰度范围内搜索出最佳的阈值组合
- ✓ 方法步骤：
- 自动平滑直方图
- 确定区域类数
- 自动搜索多阈值值
- 找阈值分割图像



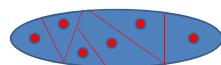
• 14.5.4 特征空间聚类法(模式识别方法)

- ✓ 根据特征进行模式分类是指根据提取的特征值将一组目标划分到各类中的技术。利用特征空间聚类的方法进行图像分割可看成是对阈值分割概念的推广。它将图像空间中的像素用对应的特征空间点表示，根据它们在特征空间的聚类对特征空间进行分割，然后将它们映射回原图像空间，得到分割的结果。
- ✓ 聚类方法也是一种全局的方法，比基于边缘检测的方法更抗噪声。



✓ K—均值聚类方法：

- 一般的阈值分割可看成是以像素的灰度为特征，灰度直方图代表特征空间，用阈值将灰度直方图特征空间划分，把得到的特征类映射回图像空间，不同灰度的像素构成不同的区域。除像素灰度外，其他图像特征也可用于聚类。
- K—均值算法是一种~~迭代算法~~，每迭代一次，类中心就刷新一次，经过多次迭代，使类中心趋于稳定为止。
- K—均值算法可以总结为下述几步：
- (1)任意选K个初始类中心 $Z_1, Z_2, \dots, Z_k$ ；
- (2)使用最小距离判别法将所有样本进行分类；  
若 $|x-Z_i| < |x-Z_j|, j \neq i$ , 则 $x \in i$ 类；
- (3)计算新的聚类中心 $Z'_1, Z'_2, \dots, Z'_k$ ；  
(计算各类均值，并以此作为新的类均值)
- (4)若 $Z'_i = Z_i$ 则终止算法，输出结果；否则，返回步骤(2)，继续进行。



➢ 使用RGB颜色作为特征，K—均值聚类法



✓ 迭代自组织分析算法 (ISODATA)

- 设定聚类参数；
- 任选C个初始聚类中心，用最小距离判别法将所有样本分给C类；
- 计算类内方差，类间距离等参数；
- 确定类别的“分裂”与“合并”产生新的聚类中心；
- 直到新的“聚类中心”分类满足所设定聚类参数，分类结束。



## 14.6 区域增长

### • 14.6.1 基本概念

- ✓ 目标：将区域R划分为若干个子区域 $R_1, R_2, \dots, R_n$ ，这些子区域满足5个条件：
- 完备性： $\bigcup_{i=1}^n R_i = R$
- 连通性：每个 $R_i$ 都是一个连通区域
- 独立性：对于任意 $i \neq j$ ,  $R_i \cap R_j = \emptyset$
- 单一性：每个区域内的灰度级(性质)相近或相等，  
 $P(R_i) = \text{TRUE}, i = 1, 2, \dots, n$
- 互斥性：任两个区域的灰度级(性质)明显不同，  
 $P(R_i \cup R_j) = \text{FALSE}, i \neq j$

✓ 区域增长：

- 把图像分割成特征相同的若干小区域。比较相邻小区域特征的相似性，若它们足够相似，则作为同一区域合并，以此方式将特征相似的小区域不断合并，直到不能合并为止，最后形成特征不同的各区域。
- 相对于图像灰度阈值分割技术而言，考虑了区域增长图像像素空间的连通性。

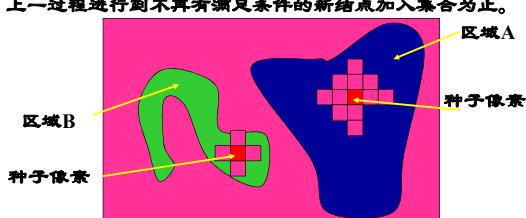
✓ 分类：

- 通过像素集合的区域增长
- 通过区域的分离与合并。

### • 14.6.2 通过像素集合的区域增长

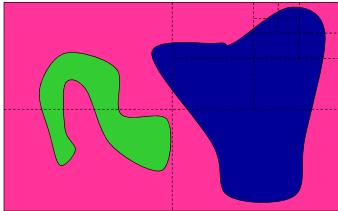
#### ✓ 算法实现：

- 根据图像的不同应用选择一个或一组种子，它或者是最亮或最暗的点，或者是位于点簇中心的点。
- 选择一个描述符（条件）
- 从该种子开始向外扩张，首先把种子像素加入集合，然后不断将与集合中各个像素连通、且满足描述符的像素加入集合
- 上一过程进行到不再有满足条件的新结点加入集合为止。



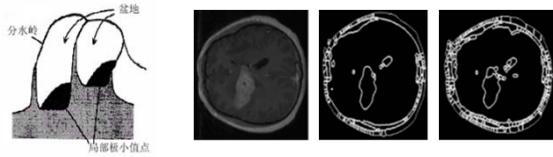
#### • 14.6.3 区域分裂与合并：

- ✓ 把图像分割成小区域，比较相邻的小区域的相似性，相似则合并
- ✓ 算法实现：
  - 对于图像中次度级不同的区域，均分为四个子区域。
  - 如果相邻的子区域所有像素的次度级相同，则将其合并。
  - 反复进行上两步操作，直至不再有新的分裂与合并为止。



#### • 14.6.4 分水岭方法(watershed)

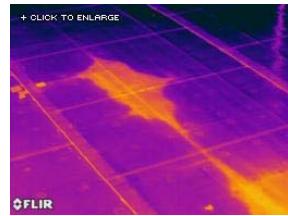
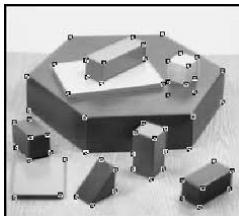
- 它将梯度幅值图像看成一幅地形图，而梯度幅值对应海拔高度，图像中不同梯度值的区域就对应于山峰和山谷间盆地。
- 该规则在各个局部极小值点的位置打一个洞，然后将地形图逐渐浸入一个湖中，全局极小值点的盆地先进水。
- 水位逐渐升高漫过盆地，当相邻两个盆地的水即将合拢时，这时在两个盆地间建坝拦截。
- 此过程将图像划分为许多个山谷盆地，分水岭就是分隔这些盆地的堤坝。



## 14.7 角点检测

#### • 14.7.1 基本概念

— 角点：两条边的交点



#### • 14.7.2 常用方法

- Harris算子
- SUSAN算子
  - SUSAN圆形模板与物体的5种几何位置关系
  - 对于图像中非纹理区域的任一点，在以它为中心的模板窗中存在一块亮度与其相同的区域(USAN)。
  - USAN区域包含了图像结构的重要信息，当模板中心像素点位于区域内部时，USAN的面积最大，当该像素点位于区域边界时，则面积为最大的一半，当该像素点为角点时，USAN区域面积约为最大的1/4。
  - SUSAN根据不同位置时USAN区域的面积来考察当前像素点为区域内部点、边缘点或角点。

