第8章 图像分割

8.1 概述

- 图像分割的目的是理解图像的内容，提取出我们感兴趣的对象。
- 图像分割按照具体应用要求和具体图像内容将图像分割成一块块各具特性的区域。
- 图像分割是模式识别和图像分析的预处理阶段。
- 通常图像分割采用聚类方法

- 图像分割基于：
  (1) 区域内部的像素一般具有灰度相似性
  (2) 区域之间的边界上一般具有灰度不连续性

传统的图像分割技术：

- 基于像素灰度值的分割技术
- 基于边界的分割技术
- 基于区域的分割技术
- 基于纹理的分割技术

- 二值图像处理—数学形态学

➢图像的特征

- **幅度特性：** 区域(2N+1)×(2N+1)内的平均幅度：

$$\bar{F}(i,j)=\frac{1}{(2N+1)^2}\sum_{m=-N}^{N}\sum_{n=-N}^{N}F(i+m,j+n)$$

- **直方图特征：**

一阶直方图：p(b)=N(b)/M

一阶直方图的特征参数有：

平均值：$\bar{b}=\sum_{b=0}^{L-1}bP(b)$　　方差：$\sigma_b^2=\sum_{b=0}^{L-1}(b-\bar{b})^2P(b)$

歪斜度：$b_n=\frac{1}{\sigma_b^3}\sum_{b=0}^{L-1}(b-\bar{b})^3P(b)$　　峭度：$b_k=\frac{1}{\sigma_b^4}\sum_{b=0}^{L-1}(b-\bar{b})^4P(b)$

能量：$b_N=\sum_{b=0}^{L-1}P(b)^2$　　熵：$b_E=-\sum_{b=0}^{L-1}P(b)\log_2[P(b)]$

- 线条和角点的特征
- 灰度边缘特征：

边缘检测的方法：通常先对图像进行灰度边缘的增强处理，得出一个增强处理后的图像，然后设立门限，进行过门限操作来确定出明显边缘的像素位置。

门限应随着空间整体幅度的变化而变化

- 纹理特征：

(1)人工纹理：一般是有规则的
(2)自然纹理：往往是无规则的

人工纹理

自然纹理

8.2 基于像素灰度值的分割技术

灰度阈值分割法 ----------按幅度分割

灰度阈值分割法是最古老的分割技术

思想： 事先决定一个阈值，当一个像素的灰度值超过这个阈值，则这个像素属于所感兴趣的对象；反之则属于背景部分。

只能应用于图像中感兴趣对象的灰度值是均匀的，并且和背景的灰度值不同。

➢用阈值检测对象边缘

确定一个门限，进行水平和垂直方向扫描，然后执行下列步骤：

⑴对f(x,y)的每行进行检测，产生的图像$f_1(x,y)$的灰度如下：

$$f_1(x,y)=\begin{cases} L_E & f(x,y)和f(x,y-1)处在不同的灰度带上 \\ L_B & 其它 \end{cases}$$

⑵对f(x,y)的每列进行检测，产生的图像$f_2(x,y)$的灰度如下：

$$f_2(x,y)=\begin{cases} L_E & f(x,y)和f(x-1,y)处在不同的灰度带上 \\ L_B & 其它 \end{cases}$$

$$则边缘图像f(x,y)=\begin{cases} L_E & f_1(x,y)和f_2(x,y)中的任何一个等于L_E \\ L_B & 其它 \end{cases}$$

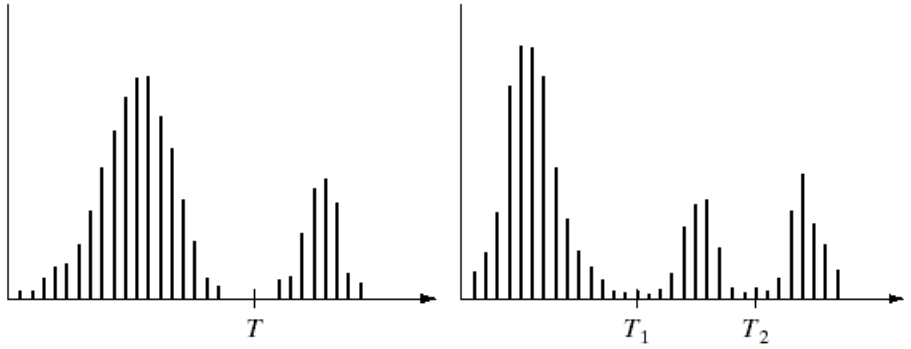
➢半阈值化方法：

仅把图像中的背景表示成最黑或最白的图像，称半阈值化。

$$g_T(x,y)=\begin{cases} f(x,y) & \text{若} f(x,y)\geq T \\ 0\text{或}1 & \text{其它} \end{cases}$$

通常阈值化方法与其它方法配合使用，以检测图像中的对象或其边缘的方法。

这种方法的关键是怎样选择阈值，一种简便的方法是检查图像的直方图，然后选择一个合适的阈值。



a b

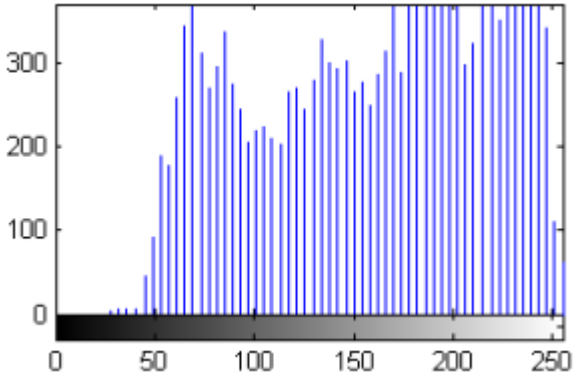
(a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

人工阈值

原始图像

T=155的二值化图像

T=210的二值化图像

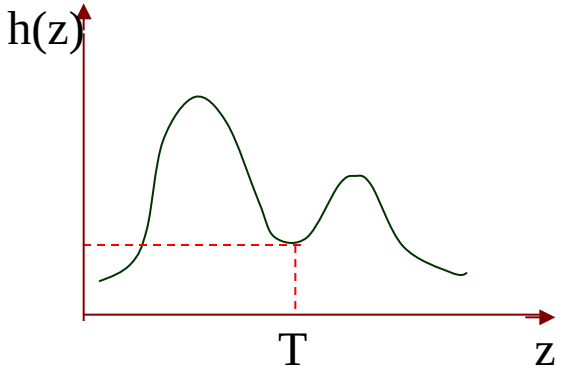


图像直方图

☞阈值的选择

⑴当已知被处理后的二值图像的灰度分布的百分比时，可用试探的方法选取阈值，直到阈值化后图像的灰度分布百分比刚好和已知数据一致。

⑵若被处理图像灰度直方图属于双峰型，则选取与两个峰中间谷底对应的灰度作为阈值(或门限)。



寻求谷底的方法：

- 方法1：首先计算图像的直方图，在直方图中找出两个局部极大值以及它们之间的极小值。
- 方法2：用一个二次曲线来拟合直方图的谷底部分
- 方法3：选取对象边界两侧点的灰度直方图的谷底作阈值
- 方法4：最小误差阈值化：使误分概率取最小值。

Otsu自动阈值选取方法（大津法）

- 使类间方差最大的一种自动阈值选取方法，适合于图像直方图是双峰情况
- Matlab工具箱提供的graythresh函数求取阈值采用的正是Otsu法

Otsu算法步骤：

1） 统计灰度级中每个像素在整幅图像中的个数。

2） 计算每个像素在整幅图像的概率分布。

3） 对灰度级进行遍历搜索，计算当前灰度值下前景背景类间概率。

4） 通过目标函数计算出类内与类间方差下对应的阈值。

- 对于图像f(x,y)，前景(即目标)和背景的分割阈值记作T，属于**前景的像素点数占整幅图像的比例**记为**É0**，其**平均灰度μ0**；**背景**像素点数占整幅图像的**比例**为**É1**，其**平均灰度**为**μ1**。图像的**总平均灰度记为μ，类间方差记**为**g**。
- 假设图像的前景较暗，并且图像的大小为M×N，图像中像素的灰度值小于阈值T的像素个数记作N0，像素灰度大于阈值T的像素个数记作N1，则有：

$$É0=N0/(M\times N) \quad (1)$$

$$É1=N1/(M\times N) \quad (2)$$

$$N0+N1=M\times N \quad (3)$$

$$É0+É1=1 \quad (4)$$

$$\mu=É0*\mu0+É1*\mu1 \quad (5)$$

$$g=É0*(\mu0-\mu)^2+É1*(\mu1-\mu)^2 \quad (6)$$

将式(5)代入式(6)，得到等价公式： $g=É0*É1*(\mu0-\mu1)^2$ (7)

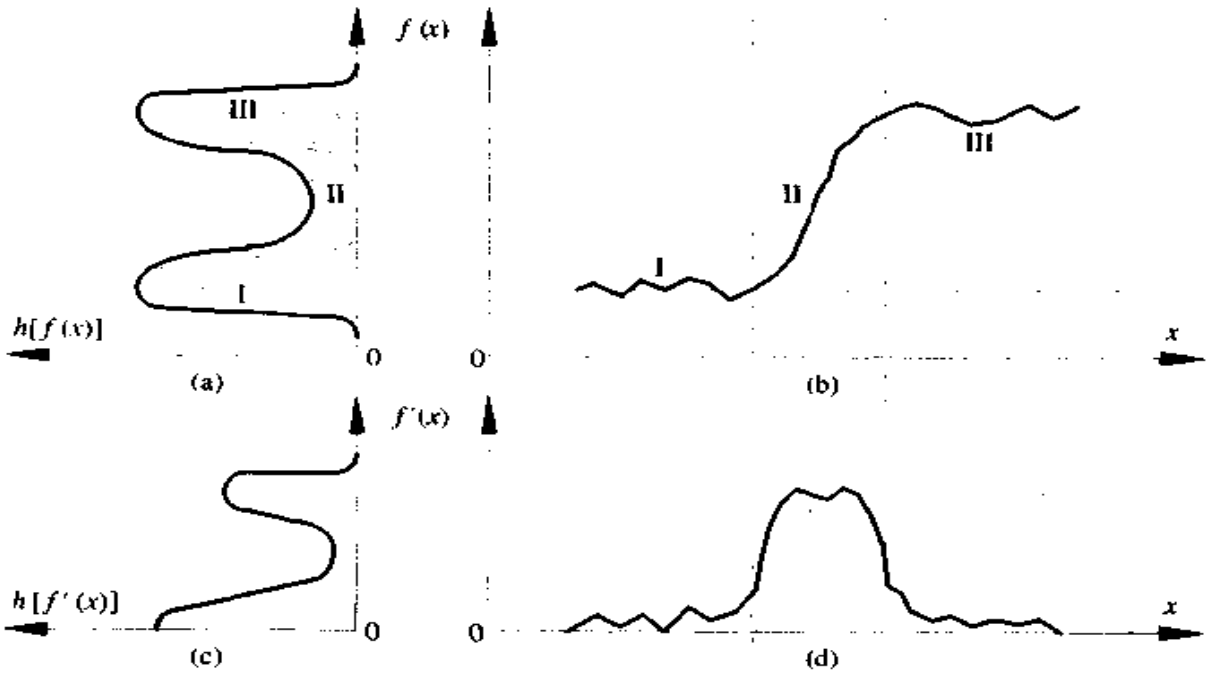
采用遍历的方法可以得到类间方差最大的阈值T。

原始图像

Otsu法二值化图像

- 方法5：直方图变换

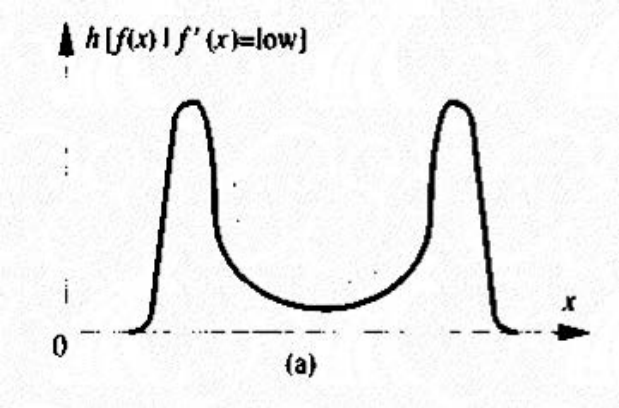
  - 基本思想：利用一些像素邻域的局部性质变换原来的直方图以得到一个新的直方图，使新的直方图谷更深或谷转变成峰



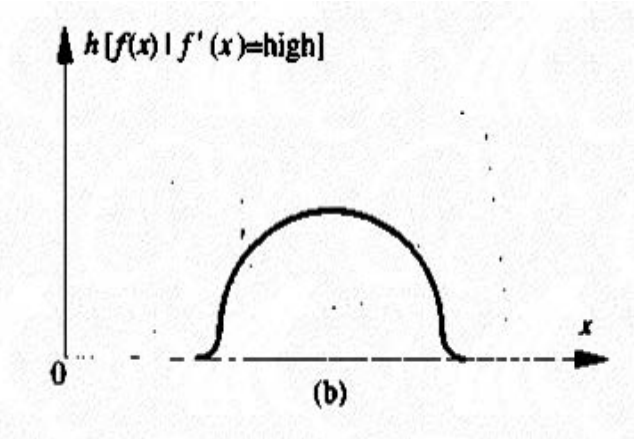
♦两类变换的直方图

a．具有低梯度值像素的直方图：通常计算加权的直方图

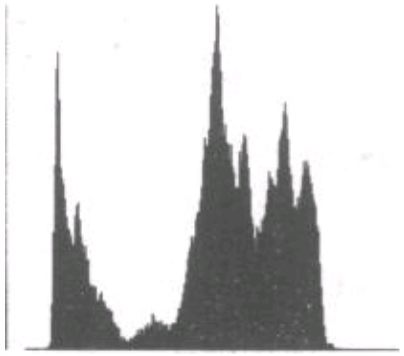
b．具有高梯度值像素的直方图：通常计算加权的直方图



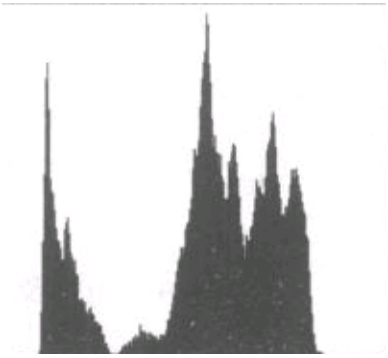
(a)



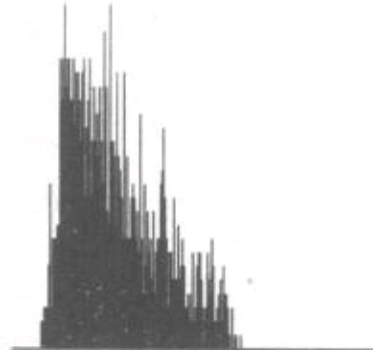
(b)



原始直方图



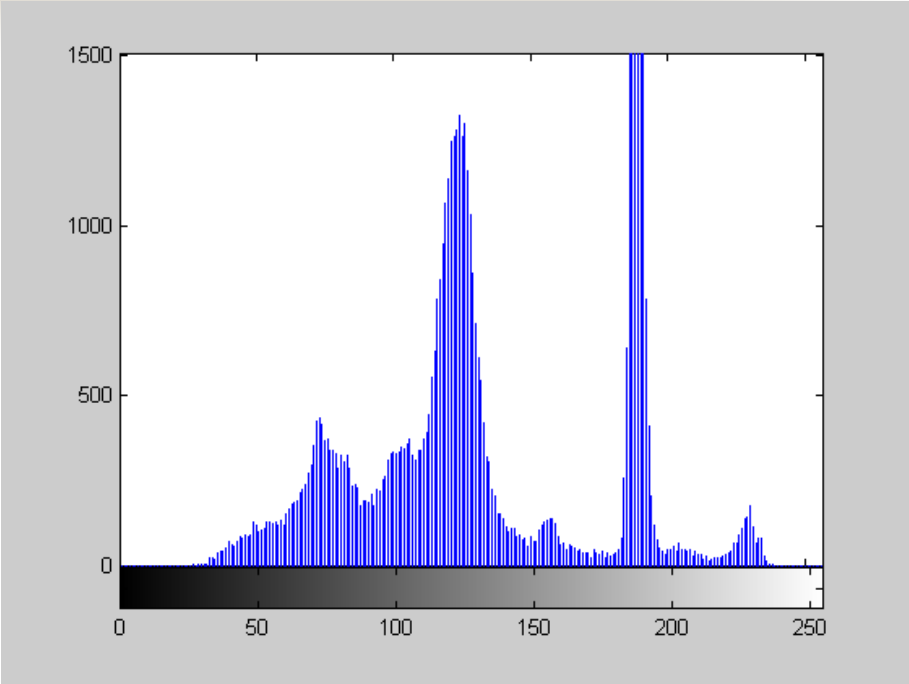
低梯度像素的直方图



高梯度像素的直方图

原始图像

分割结果
(T=170)



单一阈值方法也不适合于由许多不同纹理组成一块块区域的图像。

- 这种单一阈值分割方法一种拓展就是将图像分成一个个子区域，不同的子区域采用不同的阈值。

•方法6：依赖坐标的阈值选取

基本步骤：

（1）将整幅图像分成一系列互相之间有50%重叠的子图像

（2）作出每个子图像的直方图

（3）如果子图像的直方图是双峰，则确定一个阈值，否则就不进行处理

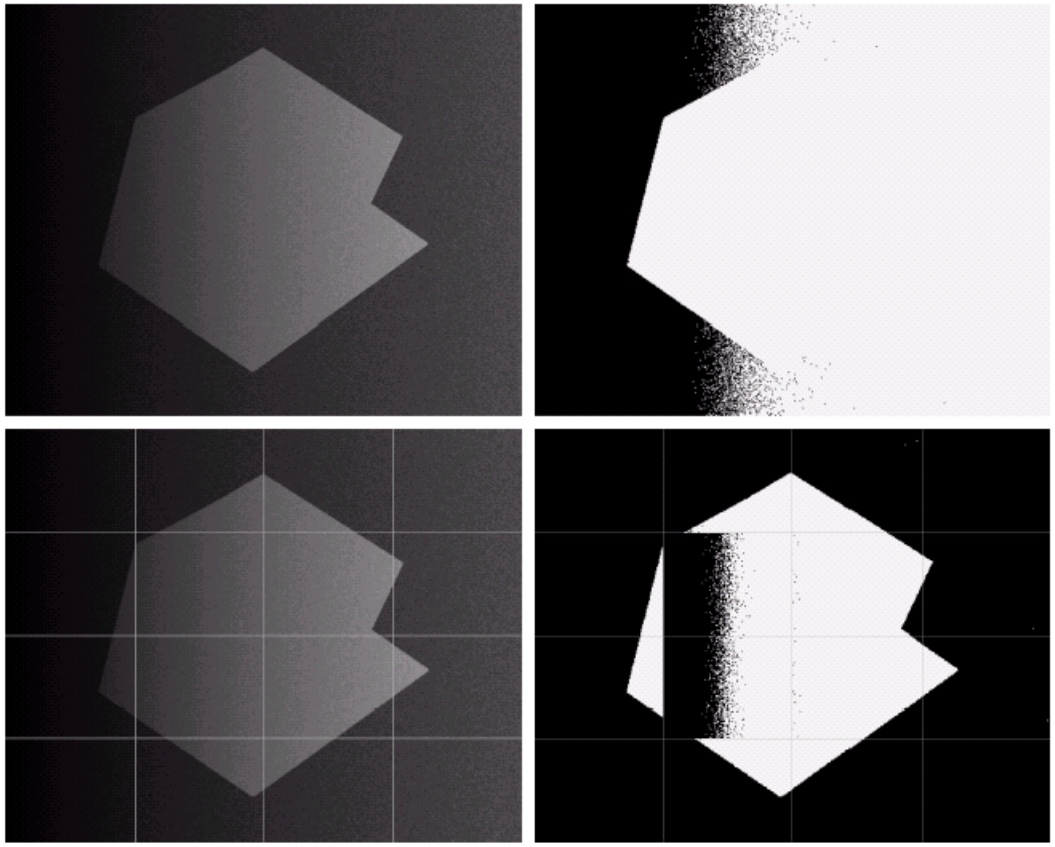
（4）对确定的阈值通过插值得到所有子图像的阈值

（5）根据各子图像的阈值再通过插值得到所有像素的阈值，然后对图像进行分割。

举例：

a b
c d

(a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.



8.3 基于边界的分割技术

边缘检测： 用一阶和二阶导数来求

➢ 微分算子：

- **梯度算子：**一阶导数算子
  - **Robert算子**
  - **Prewitt算子**
  - **Sobel算子**
  - **Canny算子**

$$d_x=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}\qquad d_y=\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}$$

Robert算子

$$d_x=\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix}\qquad d_y=\begin{bmatrix}-1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{bmatrix}$$

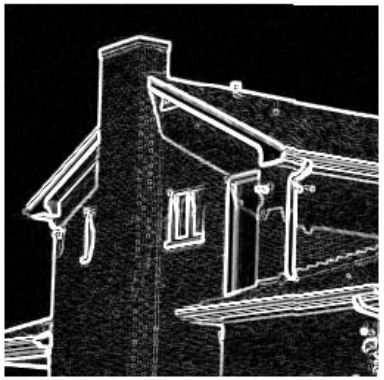
Prewitt算子

$$d_x=\begin{bmatrix}1 & 0 & -1\\ 2 & 0 & -2\\ 1 & 0 & -1\end{bmatrix}\qquad d_y=\begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix}$$

Sobel算子

几种常用梯度算子

利用Roberts算子进行边缘提取的结果



利用Sobel算子进行边缘提取的结果

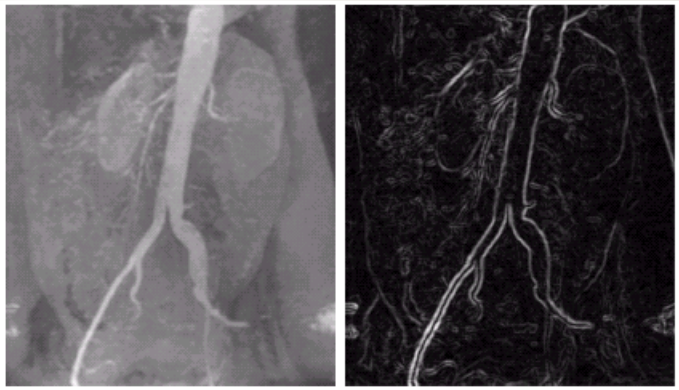
利用Prewitt算子进行边缘提取的结果

➢ 微分算子：

- 拉普拉斯算子：二阶导数算子

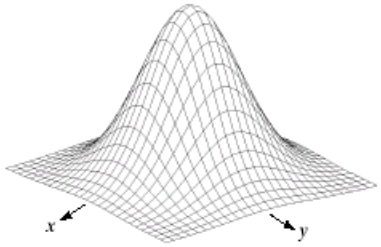
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \qquad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

利用拉普拉斯算子进行边缘提取的结果



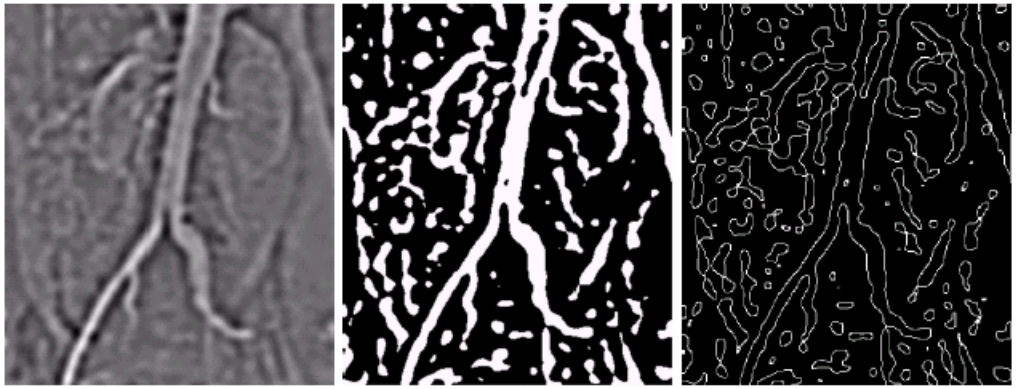
a b
c d
e f g

(a) Original image. (b) Sobel gradient (shown for comparison). (c) Spatial Gaussian smoothing function. (d) Laplacian mask. (e) LoG. (f) Thresholded LoG. (g) Zero crossings. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)



−1	−1	−1
−1	8	−1
−1	−1	−1

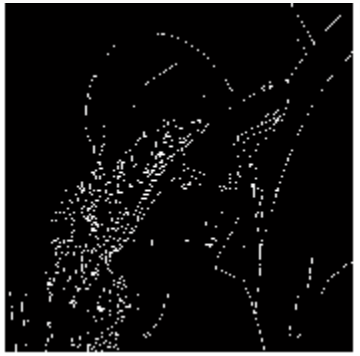
Log:高斯型的拉普拉斯算子，先对图像使用高斯函数平滑，再用拉普拉斯算子



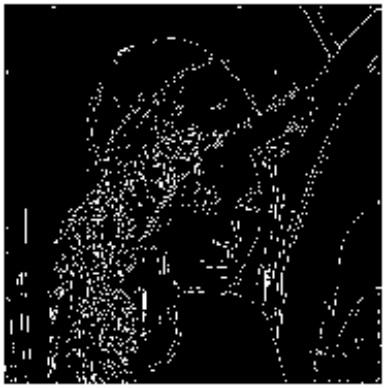
原图

Roberts算子

Sobel算子



Prewitt算子



LOG算子

Canny算子

➢ 边缘连接：

- 边界闭合：

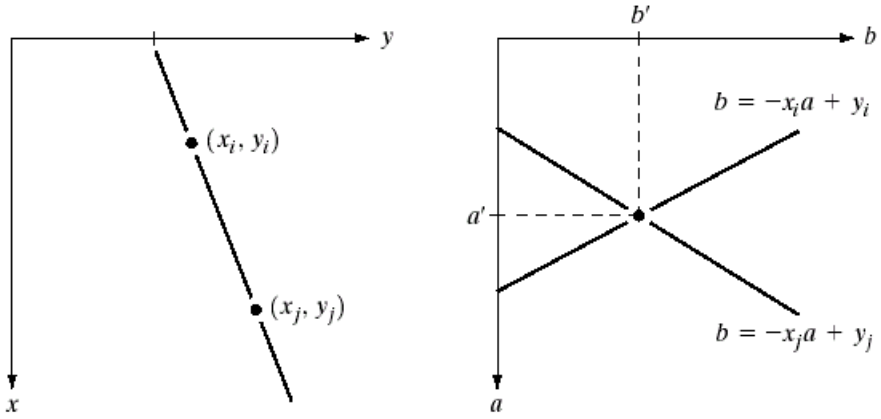
  利用梯度的幅度和方向进行边界闭合(填充小的间隙)

- 曲线拟合：　如果边缘点很稀疏，可用分段线性或高阶样条曲线来拟合这些点

- Hough变换：　检测已知形状的曲线

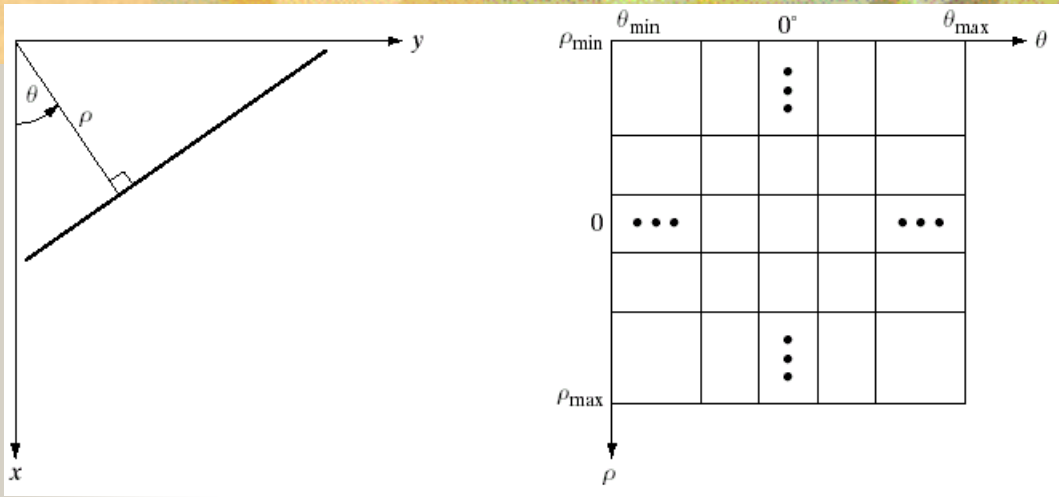
✦寻找直线

直线方程y=mx+b用极坐标表示r=xcosθ＋ysin θ，将r、θ域量化成许多小格,把图像平面中的点映射到参数空间,然后找出最大凝聚点。



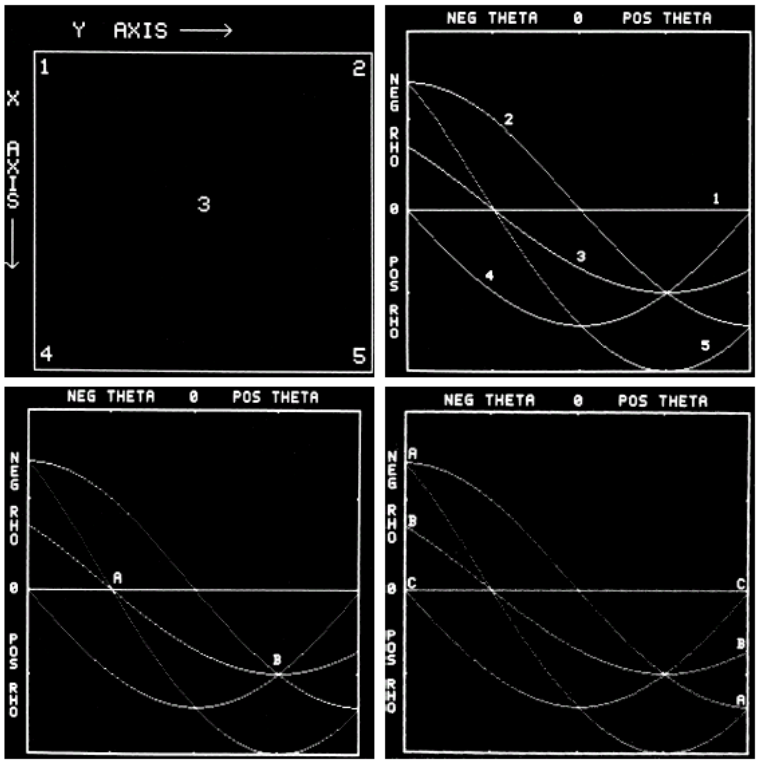
a b

(a) *xy*-plane. (b) Parameter space.



a b

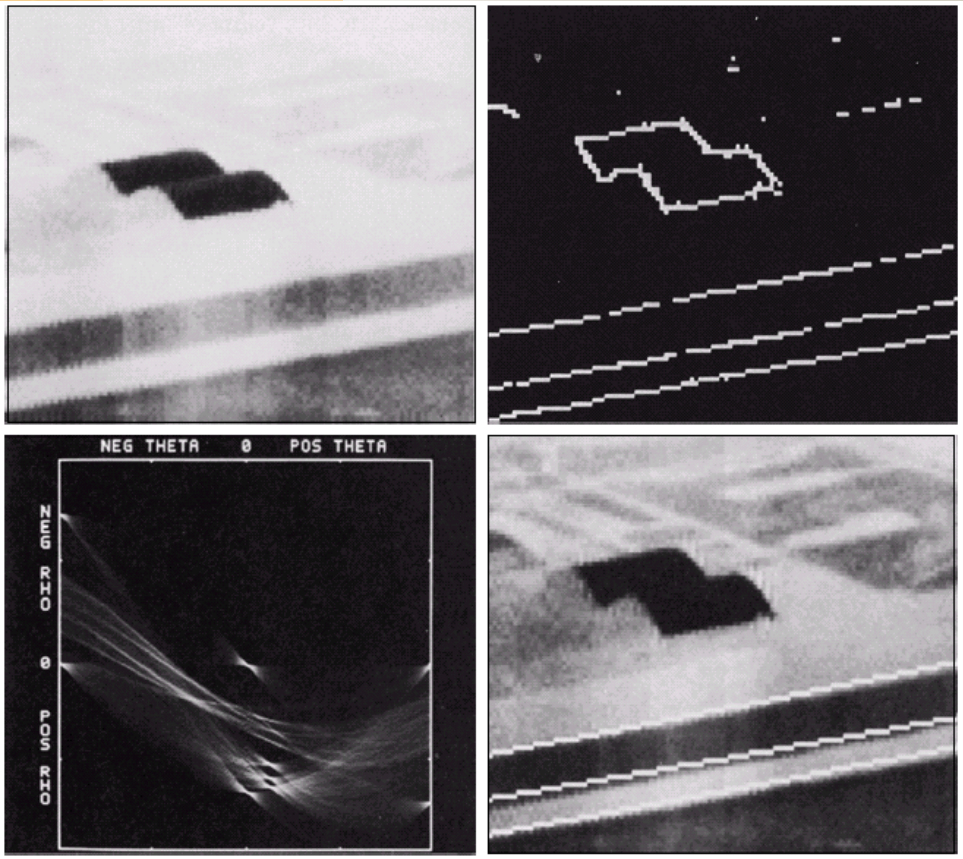
(a) Normal representation of a line.
(b) Subdivision of the $\rho\theta$-plane into cells.



a b
c d

Illustration of the Hough transform. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

•Hough变换：



a b
c d

(a) Infrared image. (b) Thresholded gradient image. (c) Hough transform. (d) Linked pixels. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

✦寻找园 $(x-a)^2+(y-b)^2=R^2$

✦寻找椭圆 $(x-x_0)^2/a^2+(y-y_0)^2/b^2=1$

➢跟踪

先对图像点进行边缘检测运算;再作跟踪运算

•跟踪的方法:

☞轮廓跟踪

设图像是黑色对象和白色背景的二值图像,找出对象的边缘轮廓

(1)步骤:靠近边缘任取一个起点,按照如下规律进行跟踪

(a)每次只前进一步,步距为一个像素

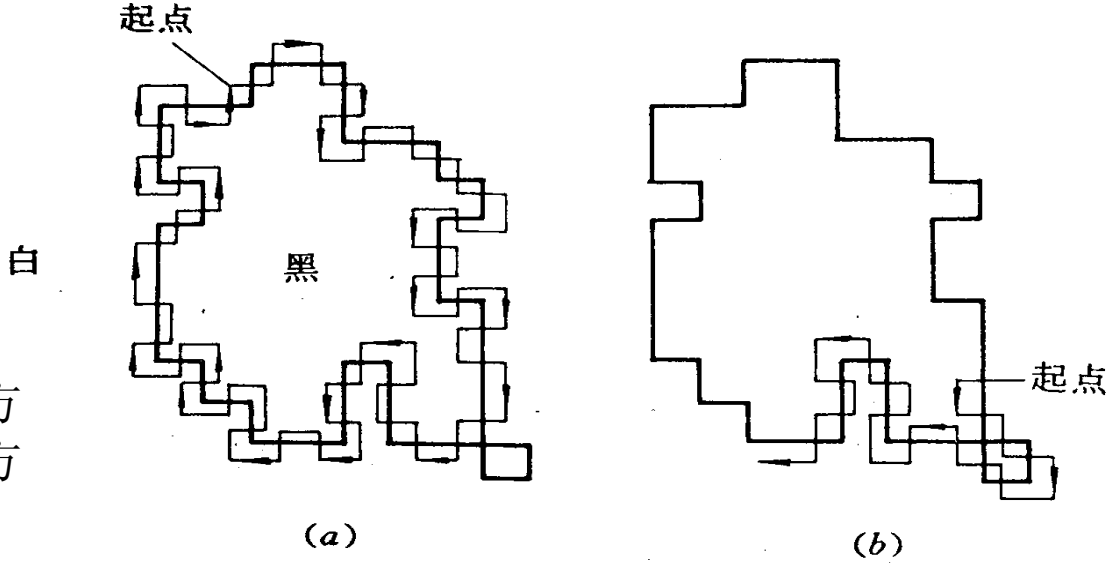
(b)当跨步由白区进入黑区时,以后各步向左转,直到穿出黑区为止

(c)当跨步由黑区进入白区时,以后各步向右转,直到穿出白区为止

(d)如此反复,直到回到起始点。

(2)举例:

用轮廓跟踪方法确定目标边界



(a)图 起始点在左上方

(b)图 起始点在右下方

(3) 轮廓跟踪方法的缺点

♦对象的某些小凸部可能被迂回过去（解决方法：用多选起始点并取不同方向进行跟踪，取相同轨迹）

♦“爬虫”掉进陷阱（解决方法：当发现在重复已走过的路径时-----即掉进陷阱，便退回，重新选择起始点和爬行方向进行轮廓跟踪）

☞光栅跟踪

利用光栅扫描技术，结合门限检测的边缘跟踪方法

✦步骤:

(a)确定一个较高的检测门限d，使得高于d的图像点必是对象点

(b)用d对图像的第一行像素进行检测，凡大于d的点作为对象点，并作为下一步跟踪的起点

(c)选取一个较低跟踪门限t

(d)确定对邻域点的定义

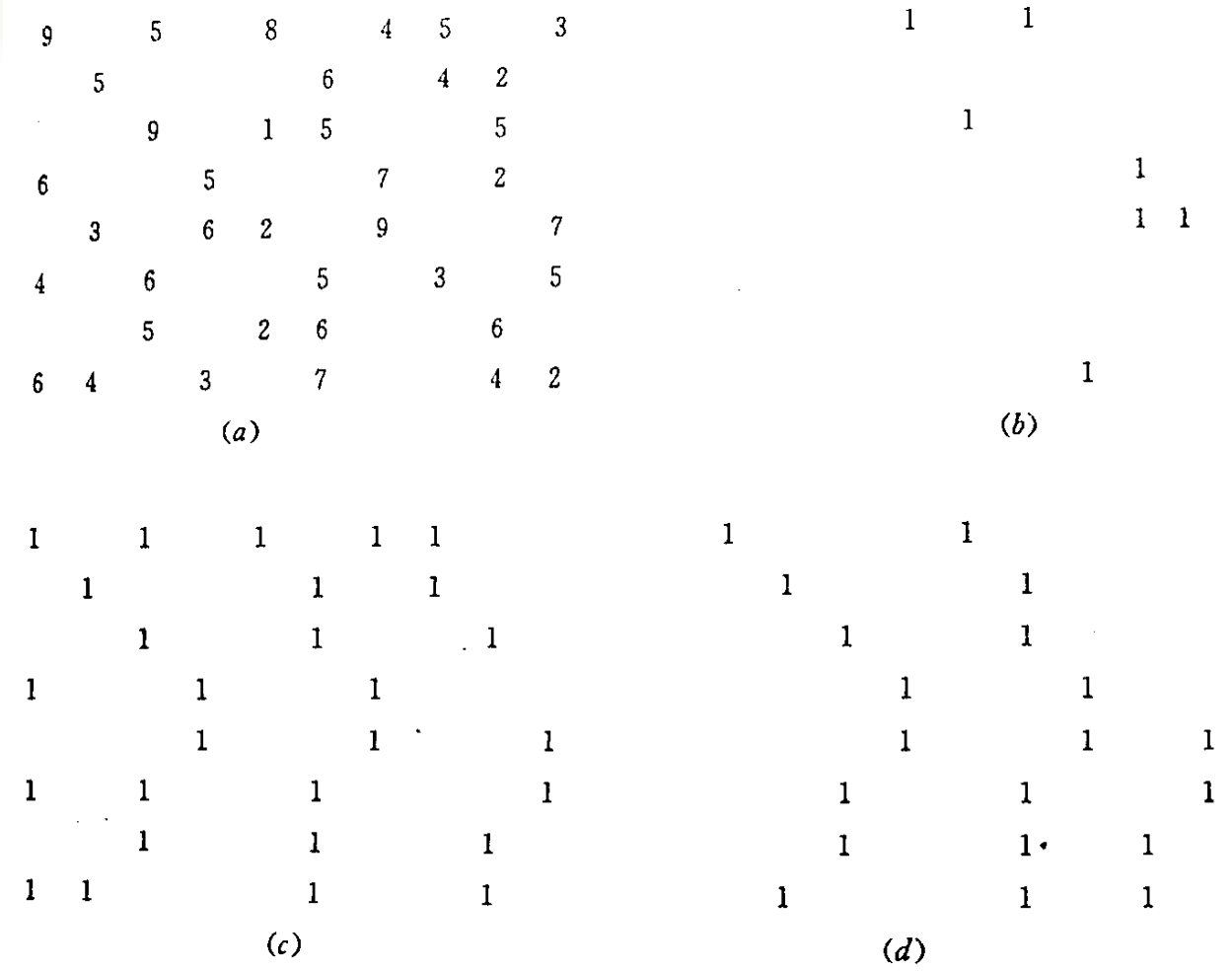
(e)扫视下一行像素，凡和上一行已检测出的对象点相邻接的像素点，其灰度差小于t的都为对象点

(f)如果上一行已检测出的某一对像点没有任何一个邻域被接收为对象点，则这一条曲线的跟踪结束

(g)对于未被接收为对象点的其它各行的像素，再次用检测门限进行检测，并以新检出的点为起始点，重新使用跟踪门限程序，以检测出不是从第一行开始的曲线

(h)当扫描完最后一行时，跟踪便结束

举例：



(a)

(b)

(c)

(d)

(a) 输入图象　(b) 取门限 7 对(a)进行门限化处理的结果

(c) 门限取 4 时，门限化结果　(d) 检测门限 $d = 7$，跟踪门限 $t = 4$ 进行跟踪的结果

✦光栅跟踪的缺点

(a) 跟踪结果将严重依赖于光栅扫描的方向性

(b) 如果曲线很接近于光栅扫描方向，则第二行的曲线点必被丢失

☞全向跟踪：

改变邻点定义和跟踪准则的光栅跟踪

✦通过定义不同的邻点来使跟踪方向不局限于逐行或列；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

✦全向跟踪的具体步骤：

(a)按光栅扫描方式对图像进行扫描,用检测门限找出一个作为开始跟踪的对象点,也叫流动点

(b)选一个适当的邻域定义、跟踪准则,对流动点进行跟踪：

发生分支点或若干曲线的交点,则选取一个和流动点性质最接近的像素作为新的流动点，继续跟踪；

在一个分支点跟踪完毕后，回到最近的一个分支点，选取另一个和该分支点性质最接近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，以选取新的流动点。

(c)当整幅图像扫描完成时，跟踪程序便结束。

8.4 基于区域的分割方法

➢区域生长

❖ 什么是区域

一般用以下性质来定义区域：

- 区域之间不能重叠，也就是说一个像素只能有一个标注。
- 在区域Ri中每一个像素点必须遵从某种规则P(Ri)。例如P(Ri)为真，当区域Ri中所有像素具有相似的灰度(相似性在一定的范围内)。
- 两个不同的区域Ri和Rj具有的规则不同。

❖ 区域生长法

最简单的区域生长法是将像素聚类：从一个种子像素点出发，按照某种连通方式和规则P来检查周围邻近的像素点，如果具有和种子像素点相似的性质，就说明它们属于同一区域。

这种算法有点类似于计算机图形学中的多边形种子填充算法。

➢区域生长

☞**基本思想：**将具有相似性质的像素集合起来构成区域

☞**具体步骤：**

(1)选取或确定一组能正确代表所需区域的种子像素

(2)确定在生长过程中能将相邻像素包括进来的准则

(3)制定让生长停止的条件或规则

☞**选取种子像素的方法**

(1)借助具体问题的特点

(2)借助生长所用准则对每个像素进行相应计算

(3)与所用图像数据的种类有关

如：可以利用直方图，为每一个区域寻找一个种子像素，找到具有图像直方图中峰值的像素点作为种子像素。

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

(a)

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

(b)

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

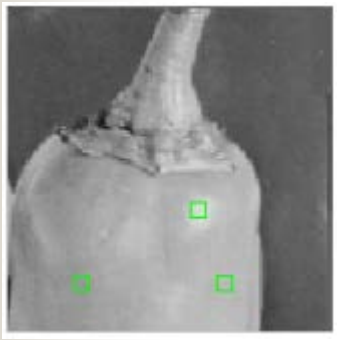
(c)

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

(d)

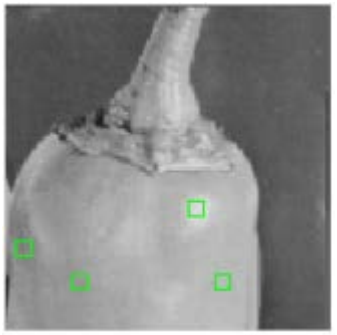
生长准则：选取的像素与区域内灰度之差<2

- 区域生长类似于光栅跟踪或全向跟踪
- 当对象点是一些纹理特性不同的区域时，选择可以鉴别纹理特性不同的平均灰度值作为区域相似的标准



原始图像及
种子点位置

三个种子点区域
生长结果



原始图像及
种子点位置

四个种子点区域
生长结果

区域生长效果

➢区域分裂合并：

原理

将图像分割成越来越小的区域直至每个区域中的像素点具有相似的性质，再将具有相似的性质相邻区域合并。

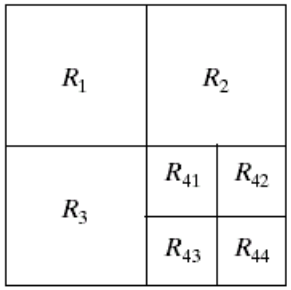
方法的优点:不需要种子像素

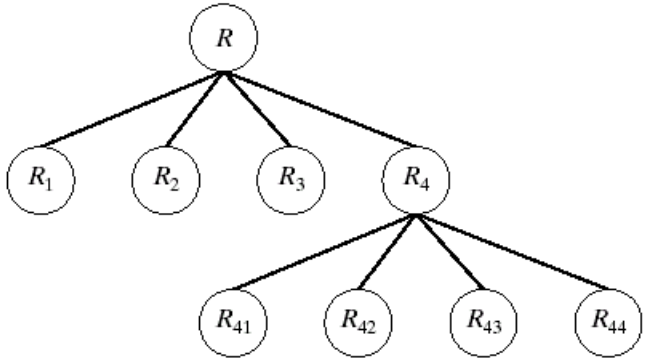
缺点:使分割后的区域具有不连续的边界。

☞ 利用四叉树表达方法的迭代分裂合并算法：

a b

(a) Partitioned image.
(b) Corresponding quadtree.



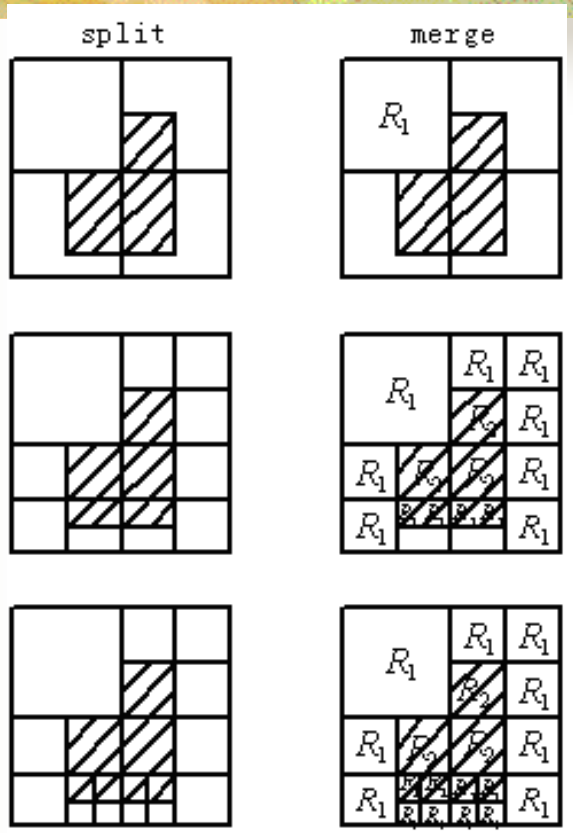


☞算法步骤：

（1）对任一个区域R_i ，如果$P(R_i)$=FALSE就将其分裂成不重叠的4等份

（2）对相邻的2个区域R_i和R_j，如果$P(R_i \cup R_j)$=TRUE，就将它们合并

（3）如果进一步的分裂或合并都不可能，则结束。

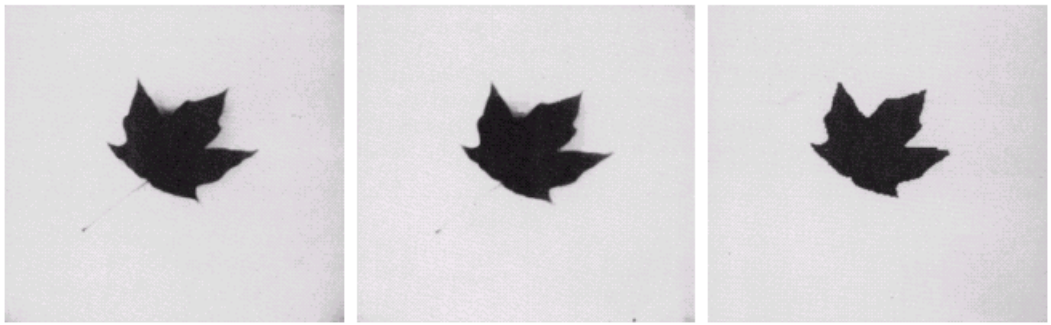


- 通常在一个区域中所要考虑的参数不只一个；
- 例如考虑一个区域中灰度的均值和方差等。 如果它的四个分块中的均值和方差相差不大的情况下， 则可以说一个区域是单调均匀的；
- 同样可以采用这种方式合并具有相同性质的区域。

简单的区域分割与合并算法过程

a b c

(a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).



➢子区合并法：

以小区域内的灰度分布相似性进行区域合并

☞具体过程：

（1）将图像分割成M×N个小区域

（2）从第一个子区开始，分别计算子区和相邻子区的直方图，然后作相似性判定。

若符合准则，相邻子区并入当前子区，形成下一轮判定合并时的当前子区；若不符合准则，就打上“未分割标记”

继续新的一轮判定，直到没有可归并的子区时，一个区域分割完毕。

（3）搜索图像全域，对凡具有“未分割标记”的子区重复上述步骤

☞常用的相似性判定：

$$\begin{cases} N = \max\left|h_1(F) - h_2(F)\right| \\ N < T_1 \end{cases}$$

$$\begin{cases} N = \sum\left|h_1(F) - h_2(F)\right| \\ N < T_2 \end{cases}$$

原始图像

分裂最小子区域
大小16×16

分裂最小子区域
大小8×8

分裂最小子区域
大小2×2

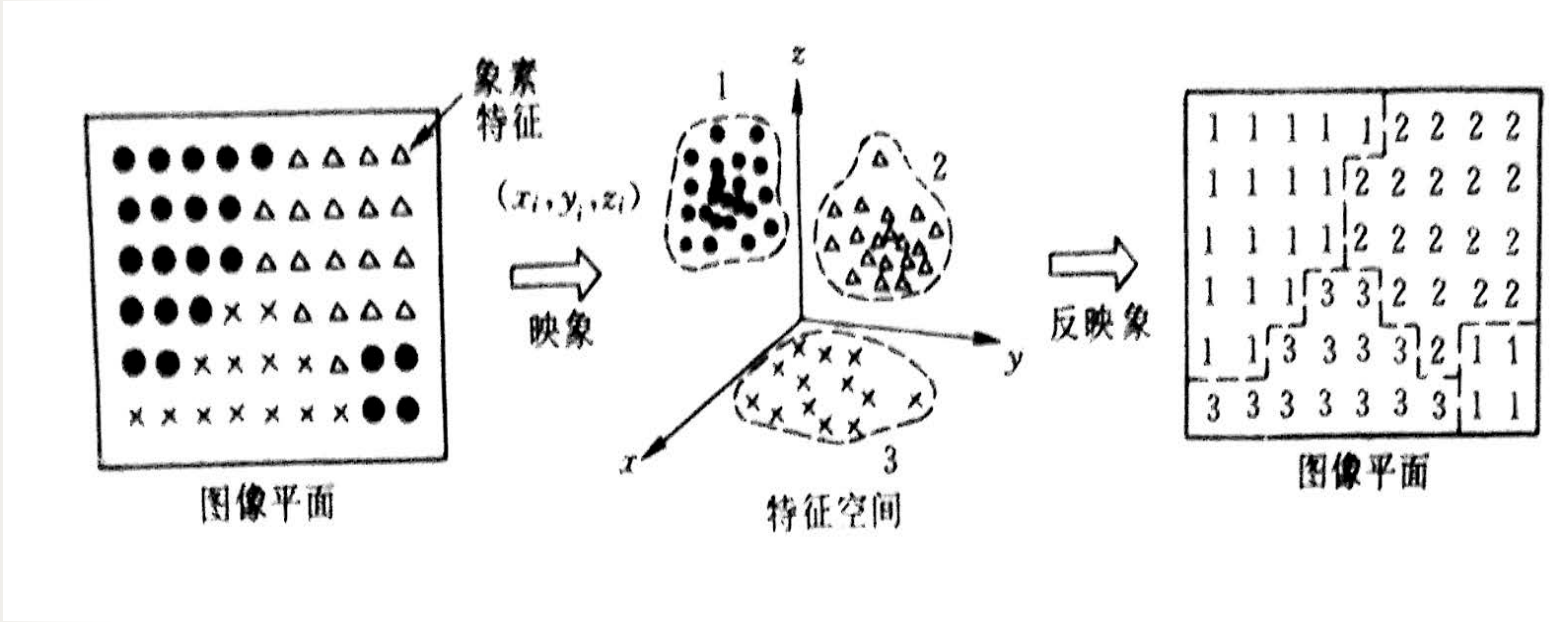
图像子区域合并效果

➢特征空间分类法：

☞基本思想：

将二维图像映射到特征空间，根据某种分类方法分析特征空间的像素并分类，再对一个类像素群中的各像素冠以同一编号，最后反变换到二维平面。

☞举例： 在特征空间中考虑群聚的区域分割



8.5 基于纹理的分割方法

什么是纹理

纹理是图像中一个重要而又难于描述的特征，至今还没有精确的纹理定义。

纹理图像在局部区域内呈现了不规则性，而在整体上表现出某种规律性。

纹理的组成

① 一是组成图像纹理的基元

② 另一个是这些基元之间的空间分布关系。

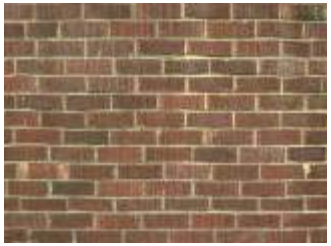
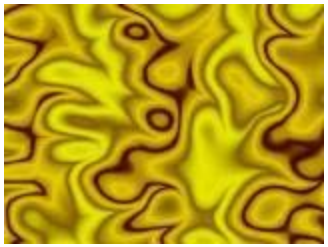
纹理基元的空间排列可能是随机的，也可能是相互之间互相依赖，这种依赖性可能是有结构的，也可能是按某种概率分布排列的，也可能是某种函数形式的。

纹理的描述

图像纹理可以定性用许多词汇来描述，如粗糙、精细、光滑、方向性、规则性和粒度等等。

但是要将这些语义描述转化为数学模型不是一件容易的事。

一般来说图像纹理由纹理中相邻像素之间的灰度变化及纹理基元模板来描述。

分析和测量纹理的算法(两类)

① 统计分析方法

从图像有关属性的统计分析出发

② 结构分析方法

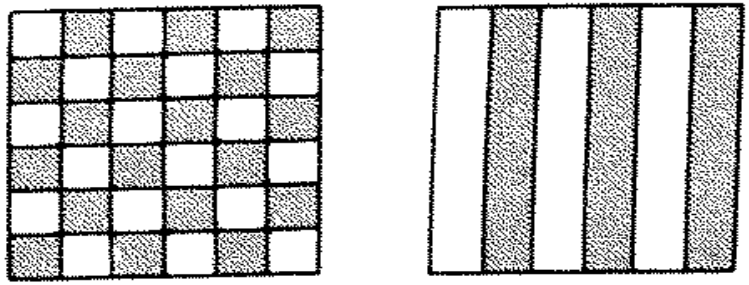
找出纹理基元，再从结构组成上探索纹理的规律，或直接探求纹理构成的结构规律。

不仅仅需要考虑相邻两个像素之间的灰度变化，还要考虑它们之间的空间关系。

在标注一个像素点的纹理特征时很可能是多维数据，如距离、方向、灰度变化等等。

➢纹理特征：

☞直方图特征：纹理区域的灰度直方图或灰度的平均值和方差等



具有相同灰度直方图的纹理

直方图是研究单个像素的灰度统计分布特性，但不能很好地反映出像素之间空间相关性的规律。

图像纹理的一个重要特征是局部区域中灰度的空间分布特性和像素位置之间的空间相关性；

因此希望能找出两个像素的联合分布的统计形式。

$L_c = \{1 \quad 2 \quad \cdots \quad N_c\}$：水平空间定义域

$L_r = \{1 \quad 2 \quad \cdots \quad N_r\}$：垂直空间定义域

$G = \{1 \quad 2 \quad \cdots \quad N_g\}$：灰度值的集合

图像I 为映射：

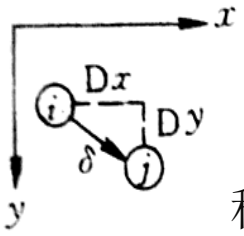
$$I : L_c \times L_r \mapsto G$$

灰度共生矩阵为概率矩阵：$[P_{ij}] : i, j \in G$

其中P_{ij}是距离为d 的两个像素，一个像素的灰度值为i，另一个像素灰度值为j 的情况在整幅图像中出现的频率。

☞灰度共生矩阵：

从灰度为i的点离开某个固定位置关系$\sigma = (Dx, Dy)$的点上的灰度j的概率。



移位$\sigma = (Dx, Dy)$

- 灰度共生矩阵表示空间灰度值依赖性的概率，灰度共生矩阵是对称的；
- 不仅和两个像素之间的距离有关，还与两个像素之间的空间角度有关。

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)
(4,1)	(4,2)	(4,3)	(4,4)

$L_y = \{1,2,3,4\}$

$L_x = \{1,2,3,4\}$

4×4图像的位置坐标

上图水平方向距离为1的像素对

$$R_H = \{((k,l),(m,n)) \in (L_y \times L_x) \times (L_y \times L_x) \mid k-m=0, \mid l-n \mid = 1\}$$

$$= \{((1,1),(1,2)),((1,2),(1,1)),((1,2),(1,3)),((1,3),(1,2)),((1,3),(1,4)),$$
$$((1,4),(1,3)),((2,1),(2,2)),((2,2),(2,1)),((2,2),(2,3)),((2,3),(2,2)),$$
$$((2,3),(2,4)),((2,4),(2,3)),((3,1),(3,2)),((3,2),(3,1)),((3,2),(3,3)),$$
$$((2,3),(2,4)),((2,4),(2,3)),((3,1),(3,2)),((3,2),(3,1)),((3,2),(3,3)),$$
$$((3,3),(3,2)),((3,3),(3,4)),((3,4),(3,3)),((4,1),(4,2)),((4,2),(4,1)),$$
$$((4,2),(4,3)),((4,3),(4,2)),((4,3),(4,4)),((4,4),(4,3))\}$$

如果角度以45度为间隔，P_{ij}的形式如下

$$P(i,j,d,0°) = \#\{((k,l),(m,n)) \in (L_r \times L_c) \times (L_r \times L_c) \mid k-m=0, \mid l-n \mid = d, I(k,l)=i, I(m,n)=j\}$$

$$P(i,j,d,45°) = \#\{((k,l),(m,n)) \in (L_r \times L_c) \times (L_r \times L_c) \mid (k-m=d, l-n=-d) or (k-m=-d, l-n=d), I(k,l)=i, I(m,n)=j\}$$

$$P(i,j,d,90°) = \#\{((k,l),(m,n)) \in (L_r \times L_c) \times (L_r \times L_c) \mid \mid k-m \mid = d, l-n=0, I(k,l)=i, I(m,n)=j\}$$

$$P(i,j,d,135°) = \#\{((k,l),(m,n)) \in (L_r \times L_c) \times (L_r \times L_c) \mid (k-m=d, l-n=d) or (k-m=-d, l-n=-d), I(k,l)=i, I(m,n)=j\}$$

其中符号#表示集合中元素的个数。上述公式中距离的尺度为

$$d((k,l),(m,n)) = \max\{\mid k-m \mid, \mid l-n \mid\}$$

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

一个4×4图像

$0°\quad \boldsymbol{P}_H = \begin{pmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix}$

$90°\quad \boldsymbol{P}_v = \begin{pmatrix} 6 & 0 & 2 & 0 \\ 0 & 4 & 2 & 0 \\ 2 & 2 & 2 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}$

$135°\quad \boldsymbol{P}_{LD} = \begin{pmatrix} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}$

$45°\quad \boldsymbol{P}_{RD} = \begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 2 & 2 & 0 \\ 0 & 2 & 4 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

左边图像相邻像素角度为0、90、135、45度,距离为1的灰度共生矩阵

基于灰度共生矩阵的纹理参数：

二阶矩

$$f_1 = \sum_{i,j} P_{ij}^2$$

- 二阶矩是图像灰度分布均匀性的度量。
- 二阶矩是灰度共生矩阵元素值平方和，所以也称为能量。
- 纹理较粗，此时二阶矩值f1较大，可以理解为粗纹理含有较多的能量；反之，二阶矩值f1较小，即细纹理含有较少的能量。

熵

$$f_2 = -\sum_{i,j} P_{ij} \log P_{ij}$$

- 熵值是图像所具有的信息量的度量
- 若图像没有任何纹理，则灰度共生矩阵几乎为零，则熵f_2接近为零；
- 若图像充满细纹理，则P_{ij}的值近似相等则该图像的熵f_2最大
- 若图像中分布较少的纹理，P_{ij}的数值差别较大，则图像的熵f_2较小

对比度

$$f_3 = \sum_{i,j} |i-j|^k \; P_{ij}^l$$

- 图像的对比度可以理解为图像的清晰度，即纹理清晰程度。在图像中，纹理的沟纹越深，则其对比度f_3越大，图像的视觉效果越是清晰。

相关

$$f_4 = \left(\sum_{i,j} ijP_{ij} - u_x u_y\right) / \sigma_x \sigma_y$$

- 用来衡量灰度共生矩阵的元素在行的方向或列的方向的相似程度。

例如，某图像具有水平方向的纹理占主导地位，则图像在0度的灰度共生矩阵的相关值往往大于90、135、45度的灰度共生矩阵的相关值。

- 上述4个统计参数为应用灰度共生矩阵进行纹理分析的主要参数，可以组合起来，成为纹理分析的特征参数使用。

基于灰度共生矩阵的纹理参数：

（角二阶矩）$q_1 = \sum_{i=1}^{n}\sum_{j=1}^{n}\{P_\delta(i,j)\}^2$　　$q_2 = \sum_{k=0}^{n-1} k\{\sum_{i=1}^{n}\sum_{j=1}^{n} P_\delta(i,j)\}_{|i-j|=k}$（对比度）

$$q_3 = -\sum_{i=1}^{n}\sum_{j=1}^{n} P_\delta(i,j)\lg P_\delta(i,j)$$（熵）

（相关性）　$$q_4 = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} \mathrm{i}\cdot\mathrm{j}\ P_\delta(i,j) - \mu_x\mu_y}{\sigma x \sigma y}$$

式中：$\mu_x = \sum_{i=1}^{n} i \sum_{j=1}^{n} P_\delta(i,j)$　　$\mu_y = \sum_{j=1}^{n} j \sum_{i=1}^{n} P_\delta(i,j)$

$\sigma_x^2 = \frac{1}{n}\sum_{i=1}^{n}(i-\mu_x)^2 \cdot \sum_{j=1}^{n} P_\delta(i,j)$　　$\sigma_y^2 = \frac{1}{n}\sum_{i=1}^{n}(j-\mu_y)^2 \cdot \sum_{j=1}^{n} P_\delta(i,j)$

纹理分析的自相关函数方法

自相关函数的定义

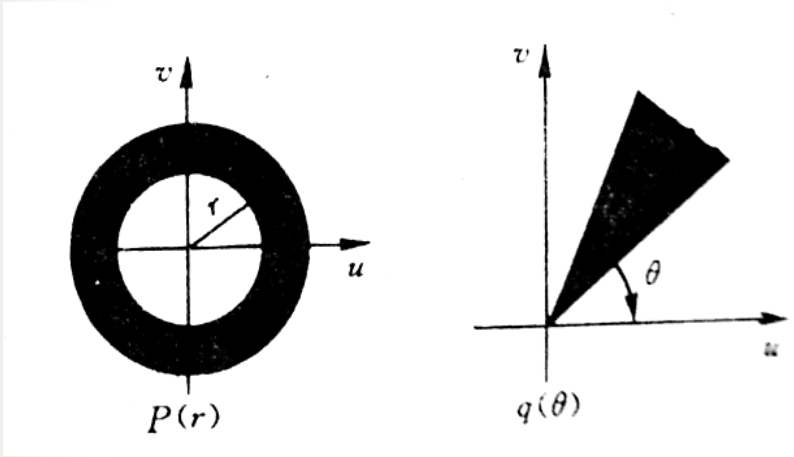
若有一幅图像$f(i, j)$, $i, j=0, 1, \ldots, N-1$, 它的自相关函数为：

$$\rho(x, y) = \frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i, j) f(i+x, j+y)}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f^2(i, j)}$$

如果图像中灰度基元的面积比较大，则自相关函数随距离的增大，下降速度比较慢

如果灰度基元中灰度呈周期变化，则自相关函数的升降也呈周期性变化。

☞**傅立叶特征：**对图像作傅立叶变换，从其频率成分的分布来求得纹理特征的方法。



$$P(r)=2\sum_{\theta=0}^{\pi}P(r,\theta)$$

$$q(\theta)=\sum_{r=0}^{W}P(r,\theta)$$

- **纹理区域的分割：**

  对于具有若干不同纹理的图像提取纹理区域。

- **纹理边缘检测：**

  对于对象和背景是由不同纹理组成的图像检测边缘。

8.6 数学形态学

--------二值图像处理

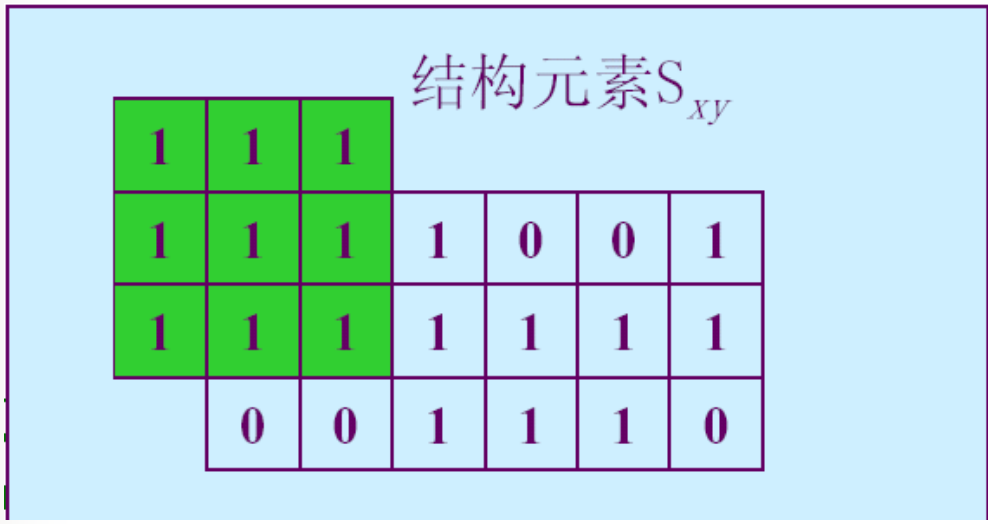
形态学（Morphology）原是对于动植物调查时采取的某种形式的研究。数学形态学（Mathematical Morphology）是分析几何形状和结构的数学方法，它建立在集合代数的基础上，是用集合论方法定量描述集合结构的学科。1985年之后，数学形态学逐渐成为分析图像几何特征的工具。

数学形态学包括一组基本的形态学运算子：腐蚀（Erosion）、膨胀（Dilation）、开（Opening）、闭（Closing）等。运用这些算子及其组合来进行图像形状和结构的分析及处理。

通常选择图像中感兴趣的目标图像区域像素集合来进行形态学变换。

结构元素S

1	1	1
1	1	1
1	1	1



图像B

基本运算

1. 集合关系

设B和S为R^2的子集，B为物体区域，S为某种结构单元，则S结构单元对B的关系有三类：

1) S包含于B，$S \subseteq B$

2) S不包含于B，$S \cap B \neq \Phi$

3) S击不中(MISS)B，$S \cap B = \Phi$

2. 平移

$S \subset R^2$，S原点平移到（x,y)记为S_{xy}

3. 膨胀

S为结构单元，广义的膨胀定义为

$$B \oplus S = \left\{ x, y \middle| S_{xy} \cap B \neq \Phi \right\}$$

当S为3×3结构元时，广义膨胀就为一般意义上的膨胀。

1. 一般意思上的膨胀是将与物体边界接触的背景像素合并到物体中的过程。
2. 如果物体是个圆，进行一次膨胀后，它的直径会增大两个像素。
3. 如果两个物体在某处用少于三个像素分开，膨胀后这两个物体就合并成为一个物体了。

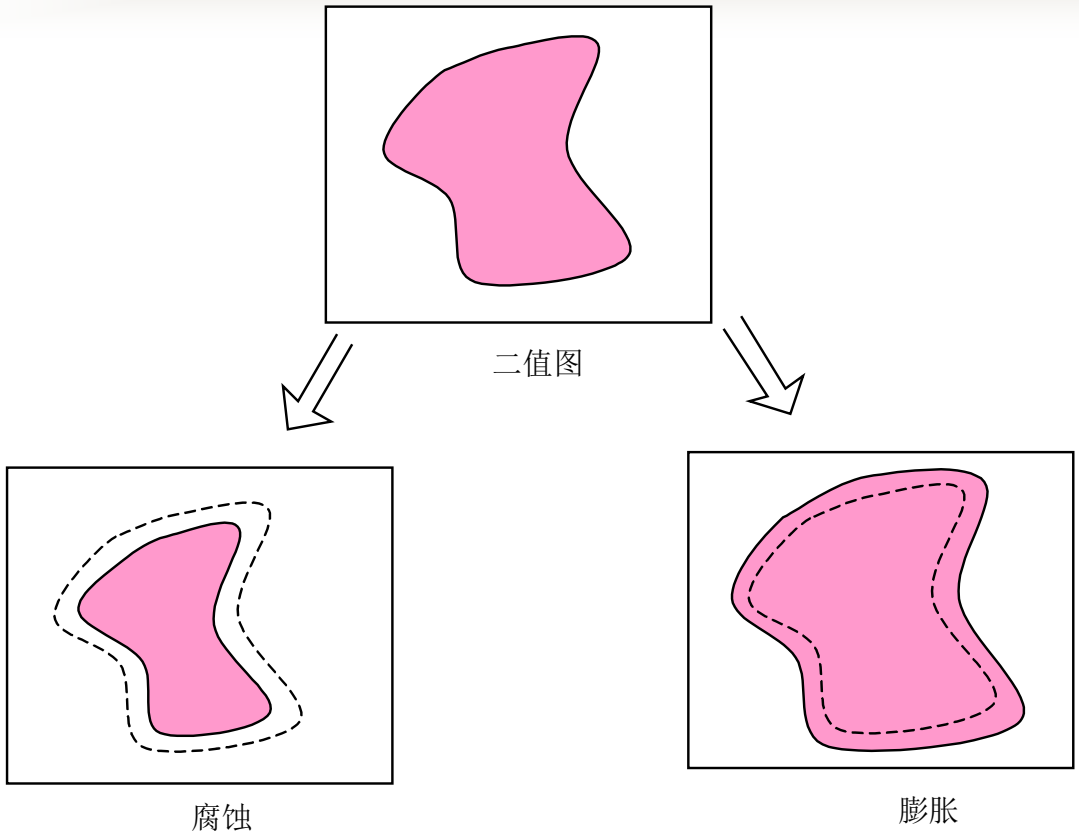
4. 腐蚀

S为结构单元，广义的腐蚀运算定义为

$$B \otimes S = \left\{ x, y \middle| S_{xy} \subseteq B \right\}$$

当S为3×3结构元时，广义腐蚀就为一般意义上的腐蚀。

1. 简单的腐蚀运算是将一个物体沿边界减小的过程，在物体的周边减少一个像素。
2. 如果物体是一个圆，则进行一次腐蚀运算后，它的直径减少2。



腐蚀和膨胀示意图

(a) 原图像

(b) 膨胀后图像

(c) 腐蚀后图像

5. 开和闭运算

✦腐蚀运算后再进行膨胀运算的组合运算称为开运算（Opening）。

$$B \circ S = (B \otimes S) \oplus S$$

开运算的效果：

1）删除小物体；

2）将物体拆分为小物体；

3）平滑大物体边界而不明显改变它们的面积；

✦膨胀运算后再进行腐蚀运算的组合运算称为闭运算（Closing）。

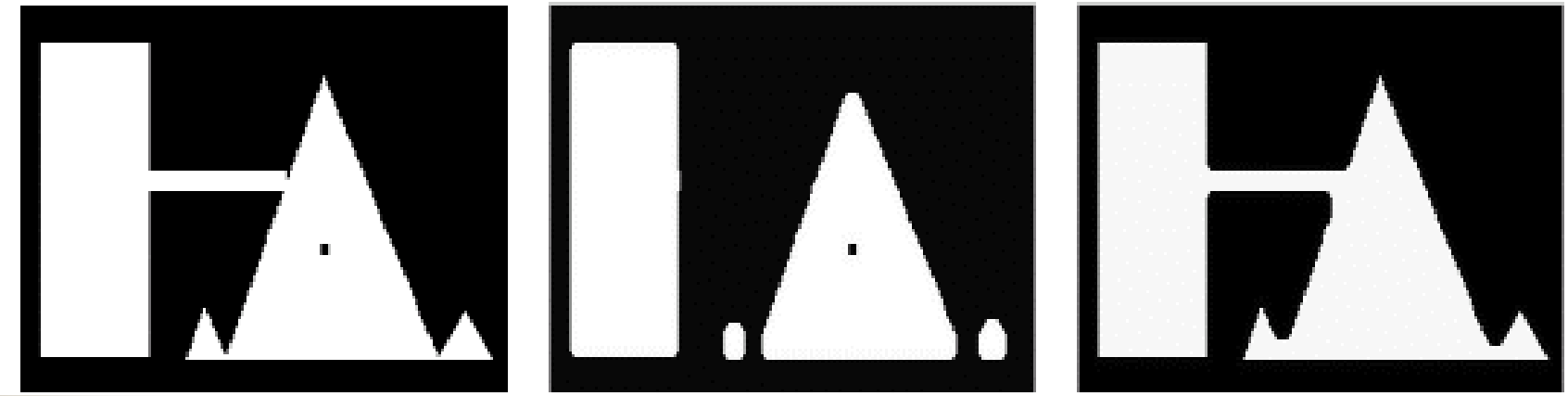
$$B \bullet S = (B \oplus S) \otimes S$$

闭运算的效果：

1）填充物体的小洞；

2）连接相近的物体；

3）平滑物体的边界而不明显改变它们的面积。



(a) 原图像　　(b) 开运算结果　　(c) 闭运算结果

腐蚀和膨胀的衍生运算

腐蚀的反复进行会导致物体消失，而膨胀的反复进行的结果是所有物体都合并到一起了。
改变这些过程来产生一些别的效果以适应实际的应用。

1. 收缩

保持单个像素的物体不变的腐蚀运算称为收缩(Shrinking)。

2. 细化

修改腐蚀计算过程来保持物体不被分开。首先进行有条件的常规腐蚀过程，只是将要删除的像素打上标记而并不真正删除；然后逐步访问打上标记的像素，如果删除该标记像素不会分开物体，就删除它，否则就保留它。以上过程就是细化(Thinning)。

宽细化的结果是把曲线型物体变成一个像素的线型图。

（a）Hello World 图像

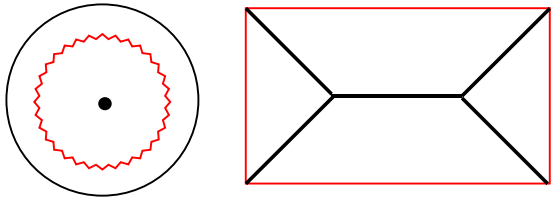
（b）细化一次的结果

（c）多次细化结果

图像的细化

3. 骨骼化

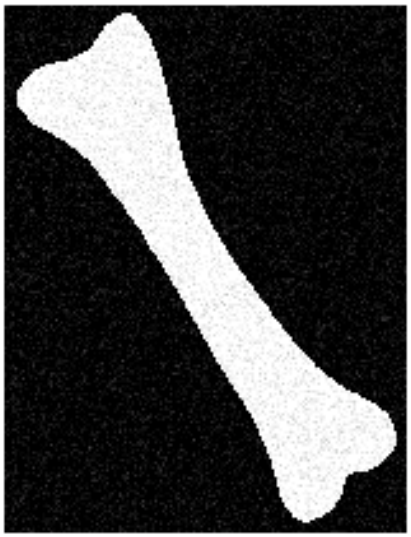
和细化相关的一个算子是骨骼化，也被称为中轴变换或火烧草场算法。中轴是和边界上至少两点相切的圆的圆心的轨迹。可以用火烧草场来说明，设物体区域上铺满了草料，火从物体边界同时均匀地烧起，最后草场全部烧光火熄灭的地方就是它的骨骼或骨架。



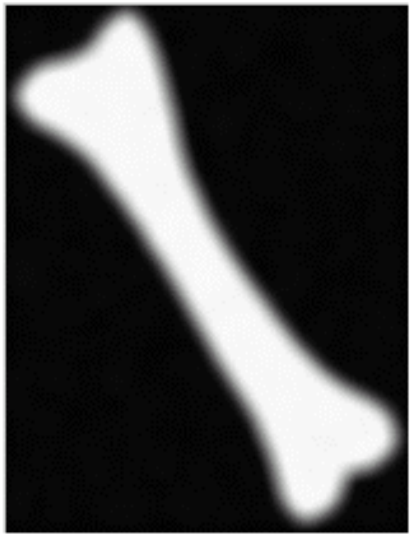
中轴变换的火烧草场算法示意图

4. 修剪

在很多情况下，细化或骨骼化过程会留下很多短刺，这些是有两三个像素点的分支。这些短刺是由于边界上的单个像素的摆动引起的。短刺可以用3×3的算子来移去端点，然后重新建立删去的分支。

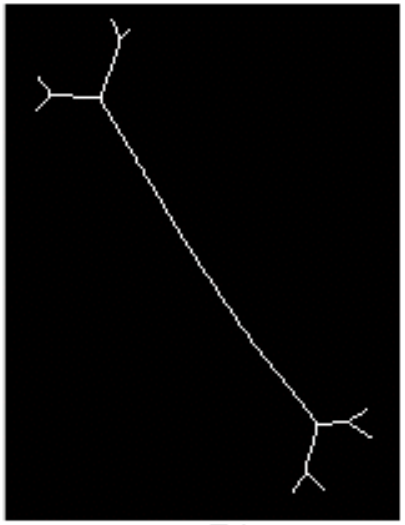


(a) 带噪声的骨骼图像

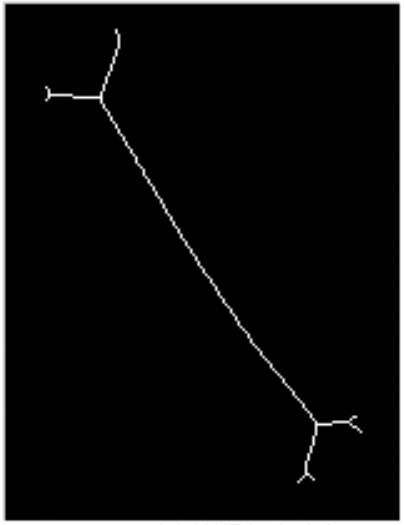


(b) 用 10×10 高斯模板滤波

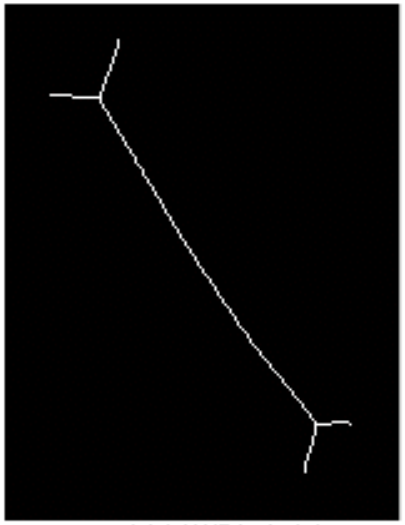
(c) 二值图像



(d) 骨架



(e) 四次消除尖刺



(f) 八次消除尖刺

骨架提取

5. 加厚

不把相近物体合并的膨胀过程称为加厚(Thickening)。和细化过程一样，它也可以分两步完成。和它互补的操作是对背景进行细化，任何一种腐蚀类的操作都伴随着膨胀类的操作作用与互补的图像区域上。