# 第9章目标表达和描述

- ▶ 9.1 概述
- ▶9.2 边界表达
- ▶ 9.3 区域表达
- ▶ 9.4 边界描述
- ▶ 9.5 区域描述
- ▶9.6 关系描述



## 9.1 概述

图像分割后,为了形式化表达和描述目标

- 表达: 直接具体地表示目标
  - ■解决形式化表达问题一般有两种选择:
    - 1)根据区域的外部特征来进行形式化表示
    - 2)根据区域的<u>内部特征</u>(如区域内部的像 素灰度值)来进行形式化表示



- 选择表达方式,本着使数据变得更有利于下一步的计算工作。一般情况下:
  - 1) 如果关注的是区域形状特性,选择外部表示方式
  - 2) 如果关注的是区域内部特性,如灰度、颜色、纹理时,选择<u>内部表示方式</u>。
  - 3)好的表达方法应具有节省存储空间、易于特征计算等优点。

- - 描述: 较抽象地表示目标
  - 好的描述应在尽可能区别不同目标的基础上, 对目标尺寸、平移、旋转等尽可能不敏感。
  - 分为:
    - 1) 边界的描述
    - 2) 区域的描述
    - 3) 边界和边界或区域和区域之间的关系描述

表达: 侧重于数据结构

描述: 侧重于区域特征以及不同区域间的联系

和差别



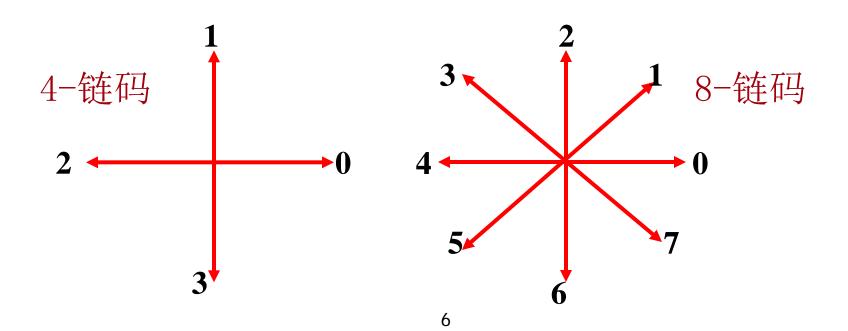
# 9.2 边界表达

- 链码
- 多边形逼近
- ■外形特征(标记)
- ■边界段

# 4

#### ■ 链码

- 定义: 1)链码是一种边界的编码表示法。
  - 2) <u>用边界的方向作为编码依据</u>。为简化边界的描述。<u>一般描述的是边界点集</u>。

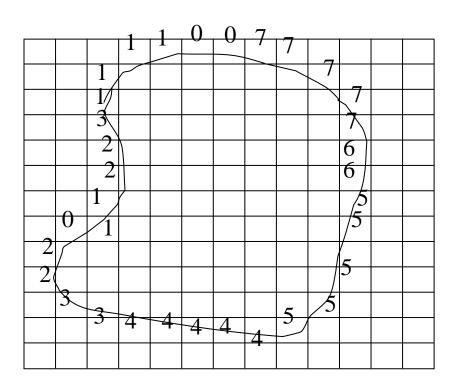




#### • 链码

- 算法:
  - 给每一个线段一个方向编码。
  - 有4-链码和8-链码两种编码方法。
  - 从起点开始,沿边界编码,至起点被重新 碰到,结束一个对象的编码。

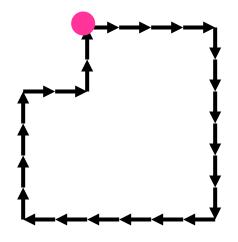




用8-链码表示边界



#### ■ 链码举例:



4-链码: 00003333332222211110011



#### ■ 链码

#### ■ 问题1:

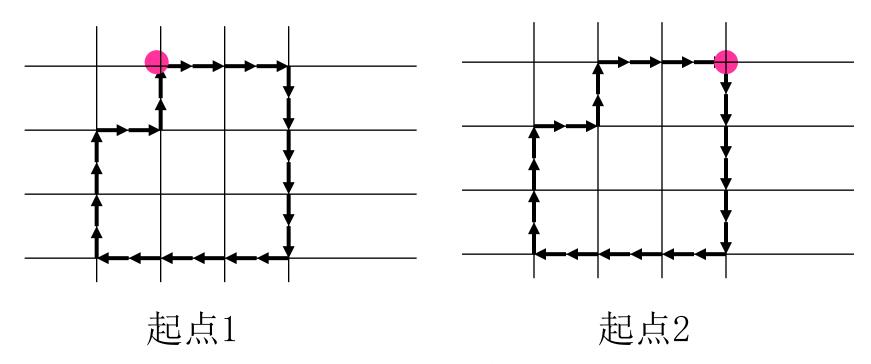
- 1) 链码相当长。
- 2) 噪声会产生不必要的链码。

#### ■ 改进1:

- 1)加大网格空间。
- 2) 依据原始边界与结果的接近程度,来确定新点的位置。



### • 链码举例:



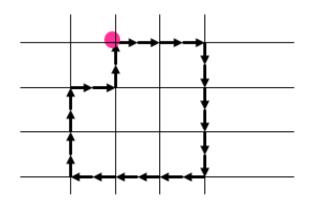
4-链码: 003332221101 4-链码: 333222110100



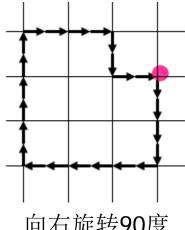
#### 链码

#### ■ 问题2:

- 1) 由于起点的不同,造成编码的不同
- 2) 由于目标旋转,造成编码的不同



4-链码: 003332221101



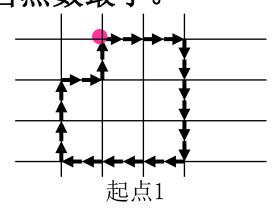
向右旋转90度

4-链码: 332221110030

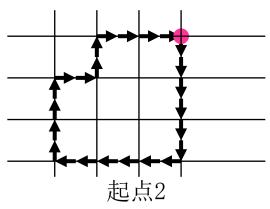
#### 链码

#### ■ 改进2:

1) 链码起点归一化: 依一个方向循环, 使所构成的自然数最小。



4-链码: 003332221101



4-链码: 333222110100

链码起点归一化: 003332221101

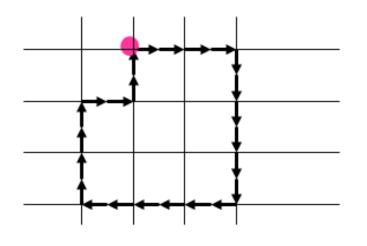
2) 链码旋转归一化:通过使用链码的首差(一阶差分)代替码本身的方式

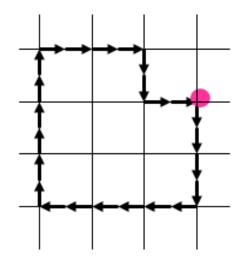
# 4

#### ■ 链码

■ 循环首差链码: 用相邻链码的差分代替链码 例如: 4-链码 10103322 循环首差为: 33133030







向右旋转90度

4-链码: 003332221101

4-链码: 332221110030

循环首差链码: 303003003031



#### ■ 链码

- 应用:
  - 用链码后,对象只要用1)起点坐标,2)链码,3)对象编号,就可以描述。
  - 链码一般用于一幅图像中有多个对象的情况。



#### ■ 多边形逼近

- 基本思想:用最少的多边形线段,获取边界形状的本质。
- 优点: 抗干扰性能好, 节省数据量
- 寻找最小基本多边形的方法一般有两种: 点合成法和边分裂法



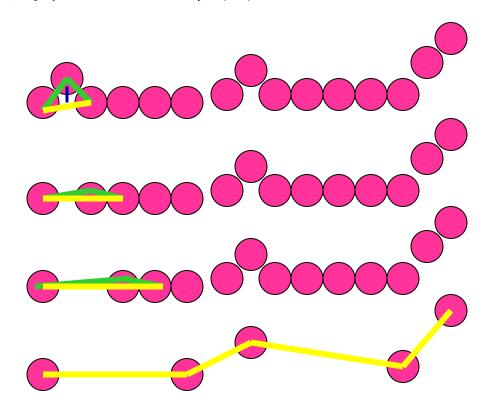
#### ■多边形逼近

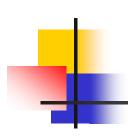
- 点合成算法:
  - 1)沿着边界选两个相邻的点对,计算<u>首尾连接直</u> 线段与原始折线段的误差。
  - 2)如果误差小于预先设置的阈值。去掉中间点, 选新点对与下一相邻点对,重复1);否则,存 储线段的参数,置误差为0,选被存储线段的终 点为起点,重复1)2)。
  - 3) 当第一个起点被遇到,结束。



### ■ 多边形逼近

■ 点合成算法思想举例:





■ 点合成算法的问题:

<u>顶点一般不对应于边界的拐点</u>(如拐角)。因为新的线段直到超过误差的阈值 才开始。

下面的分裂法可用于缓解这个问题

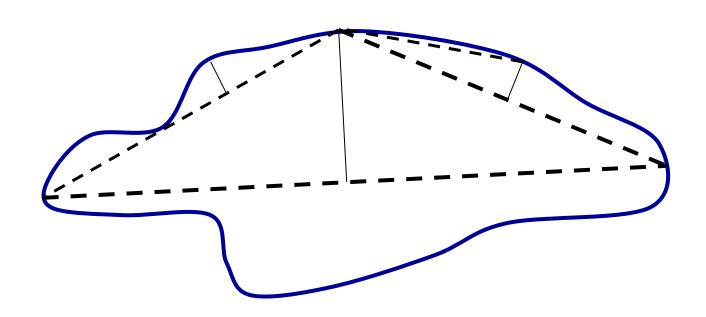


#### ■ 边分裂算法:

- (1) 连接边界线段的两个端点(如果是封闭边界,连接最远点);
- (2) 如果最大正交距离大于阈值,将边界分为两段,最大值点定位一个顶点。重复(1);
- (3) 如果没有超过阈值的正交距离,结束。



# ■ 边分裂算法思想举例:





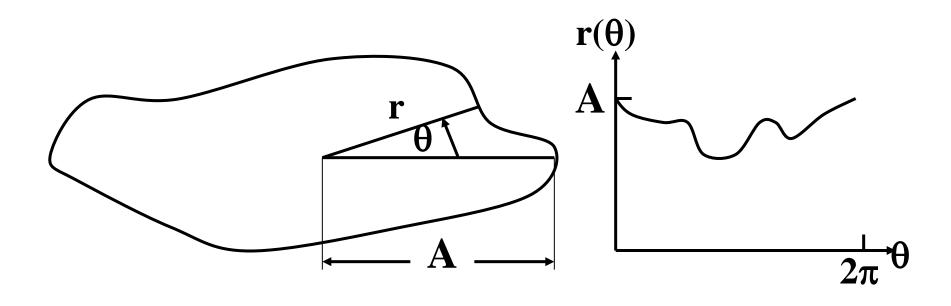
#### ■ 外形特征 (标记)

■ 基本思想:

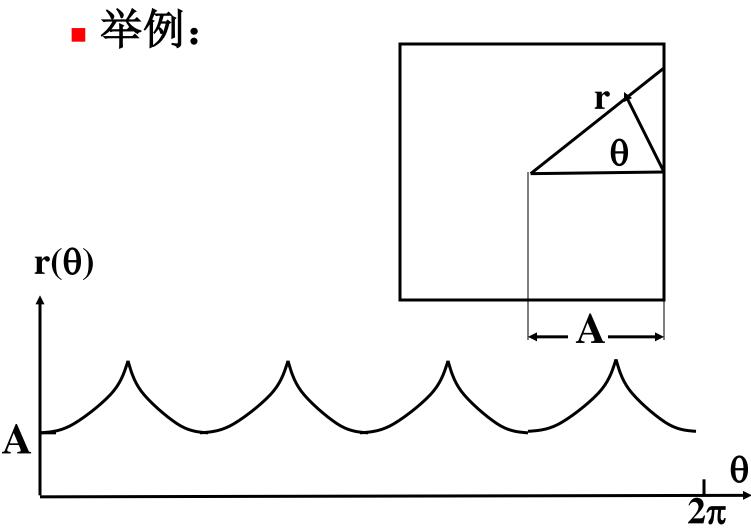
外形特征是一种用一维函数表达边界的 方法。基本思想是把边界的表示降到一维函 数

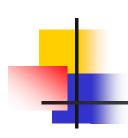


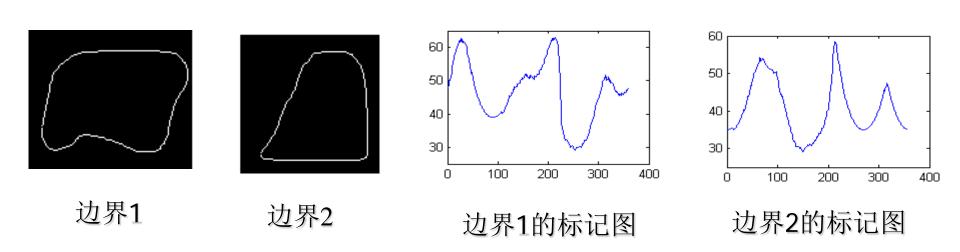
函数定义——质心角函数:边界上的点到质心的距离r,作为夹角θ的函数。



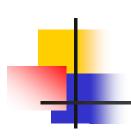








边界的标记图



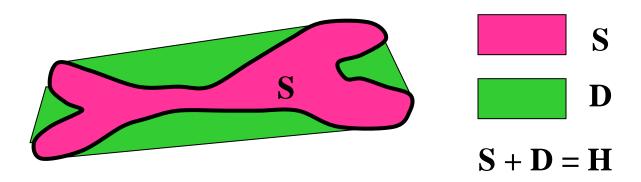
- 问题: 函数过分依赖于旋转和比例的变化
- 改进:
  - 对于旋转——两种改进:
    - a. 选择离质心最远的点作为起点
    - b. 选择到主轴最远的点作为起点
  - 对于比例变化:

对函数进行正则化,使函数值总是分布在相同的值域里,比如说[0,1]



#### ■边界分段

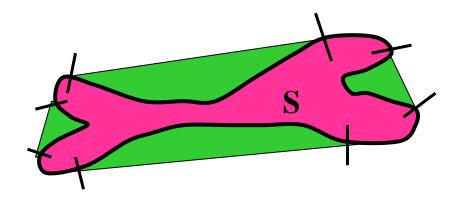
- 基本概念:
  - ■一个任意集合S(区域)的<u>凸包</u>H是:包含S 的最小凸起的集合
  - H-S的差的集合被称为集合S的凸残差D



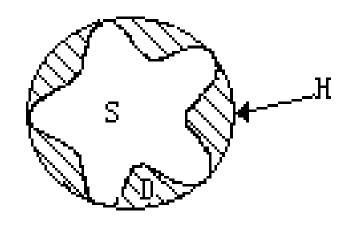


#### ■ 分段算法:

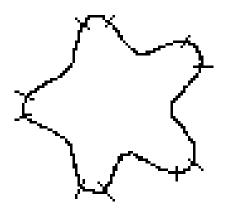
- 给凸包边界进入和离开区域的变换点打标 记来划分边界段。
- 优点:不依赖于方向和比例的变化







(a) 区域S, 其凸包H, 及其凸残差D



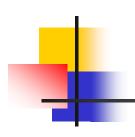
(b) 区域S的边界 分段结果

区域的边界分段



#### ■问题:

- •噪声的影响,导致出现零碎的划分。
- ■解决的方法:
  - 先平滑边界,或用多边形逼近边界,然 后再分段



# 9.3 区域表达

- 空间占有数组
- 四叉树
- 区域骨架



#### ■ 空间占有数组

- ■基本思想
  - 点f(x, y)在给定区域内就取1,否则取0
- 优点: 方便、简单、直观
- 缺点:逐点表达,需占用较大的空间



#### ■四叉树

- ■基本思想
  - 四叉树的树根对应整幅图,树叶对应各单个像素或具有相同特征的像素组成的方阵
  - 对方形图像,且像素点的个数是2的整数次幂时最适用
  - 优点: 容易生成得到,方便计算区域的多种特征
  - 缺点:确定级数,分辨率不易提高



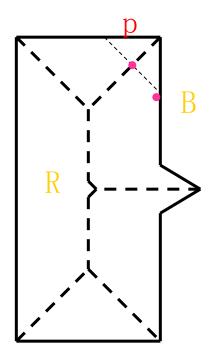
#### ■ 区域骨架

- ■基本思想
  - 表示一个平面区域结构形状的重要方法是把它削减成图形。这种削减可以通过细化(也称为抽骨架)算法,获取区域的骨架来实现
  - 中轴变换方法 (MAT)

设:R是一个区域,B为R的边界点,对于R中的点p, 找p在B上"最近"的邻居。如果p有多于一个的邻居,称它属于R的中轴(骨架)



■ 问题: 计算量大





### ■算法改进思想

- 在保证产生正确的骨架的同时,改进算法的效率。比较典型的是一类细化算法,通过不断删去边缘,但保证删除满足:
  - (1) 不移去端点
  - (2) 不破坏连通性
  - (3) 不引起区域的过度腐蚀



## ■一种细化二值区域的算法

- ■假设区域内的点值为1,背景值为0
- 这个方法由对给定区域的边界点连续进行 两个基本操作构成
- 这里边界点是指任何值为1且至少有一个8 邻域上的点为0的像素



<b>p9</b>	<b>p</b> 2	<b>p3</b>
<b>p8</b>	<b>p1</b>	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p</b> 5

<b>p9</b>	p2	р3
<b>p8</b>	p1	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p5</b>

<b>p9</b>	p2	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p5</b>

#### ■ 基本操作1

对于满足以下四个条件的边界点打标记准备删除:

- (a)  $2 \le N(p_1) \le 6$  其中 $N(p_1)$ 是点 $p_1$ 的邻域中1的个数,即:  $N(p_1) = p_2 + p_3 + ... + p_9$
- (b)  $S(p_1) = 1$ 其中 $S(p_1)$ 是按 $p_2, p_3, ..., p_9$ 顺序,<u>0-1转换</u>的个数
- (c)  $p_2 * p_4 * p_6 = 0$  ( $p_2 \times p_4 \times p_6$ 至少有一个0)
- (d)  $p_4 * p_6 * p_8 = 0$  ( $p_4 \ p_6 \ p_8$ 至少有一个0)



所有条件都满足,才打删除标记。删除并不立即 进行,而是对所有边界点都打完标记后,再将标 记的点一起删除。

#### ■ 举例:

$$N(p1) = 4$$
  
 $S(p1) = 3$   
 $p2*p4*p6 = 0$   
 $p4*p6*p8 = 0$   
 $0$   
 $0$   
 $1$   
 $p1$   
 $0$   
 $0$   
 $1$   
 $0$   
 $0$   
 $1$   
 $0$   
 $1$   
 $0$   
 $1$   
 $0$   
 $1$   
 $0$   
 $1$   
 $0$   
 $1$   
 $0$ 

<b>p9</b>	<b>p2</b>	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p5</b>

<b>p9</b>	p2	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p7</b>	<b>p6</b>	<b>p5</b>

第2个条件没满足不打标记

#### ■ 基本操作2

条件(a)、(b)与操作1相同

条件(c)、(d)改为:

c') 
$$p_2 * p_4 * p_8 = 0$$

d') 
$$p_2 p_6 p_8 = 0$$

<b>p9</b>	p2	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p5</b>

<b>p9</b>	p2	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p</b> 7	<b>p6</b>	<b>p</b> 5



#### ■细化算法

细化算法的一轮操作包括:

- ■按操作1,给边界点打标记——删除点
- ■按操作2,给边界点打标记——删除点
- 这个基本过程反复进行,直至没有点可以 删除为止。此时算法终止。

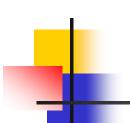


### ■ 算法分析:

(1)条件a)的分析: 当轮廓点 $p_1$ 的8邻域上有1个或7个值为1的点时,不满足条件a。

有1个点说明:  $p_1$ 是骨架上的终点,显然不能删除 有7个点说明: 如果删除 $p_1$ 会引起区域的腐蚀

(2)条件b)的分析: 当 $p_1$ 在宽度为1的笔划上时,不满足条件b。因而该条件保证了骨架的连续性。



<b>p9</b>	<b>p2</b>	<b>p3</b>
<b>p8</b>	p1	<b>p4</b>
<b>p7</b>	<b>p6</b>	<b>p5</b>

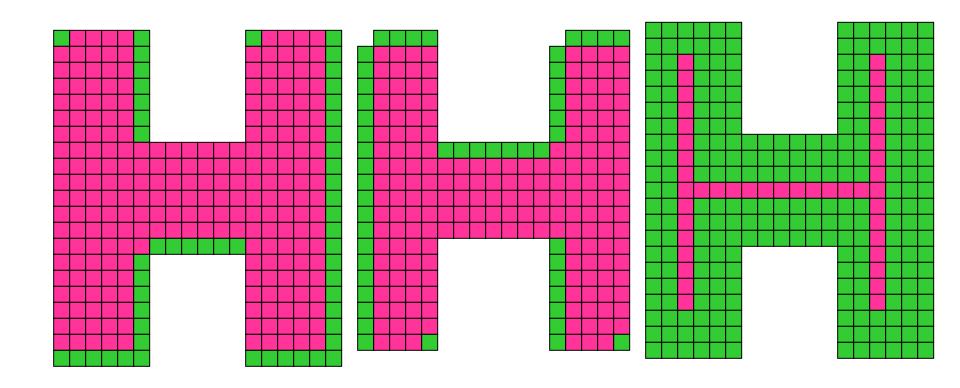
### ■ 算法分析:

(3) 当( $p_4$ =0 or  $p_6$ =0)or( $p_2$ =0 and  $p_8$ =0)时,条件c,d同时满足。满足这个条件的点可能是右边、下边、左上角的边界点。任何一种情况下, $p_1$ 都不是骨架的一部分,应被删除。

当(p4=0 and p6=0)or(p2=0 or p8=0)时, 条件c',d'同时满足。满足这个条件的点可能是左边、 上边、右下角的边界点,应被删除。



# ■ 例:





# 9.4 边界描述

- 边界描述
  - ■简单描述
  - ■形状数
  - ■傅立叶描述符
  - 矩量

# 4

## 简单描述

■ 边界的周长:

是最简单的描述符之一。沿轮廓线计算像素的个数,给出了一个长度的近似估计

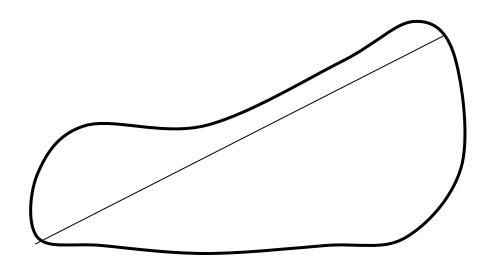
■ 边界的直径(主轴): 边界B的直径是:

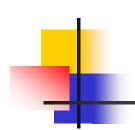
 $Diam(B) = max[D(p_i, p_i)]$ 

D是欧氏距离或街区等距离,p<sub>i</sub>, p<sub>j</sub>是边界上的点。 直径的长度和直径的两个端点连线(这条线被称为<u>边</u> <u>界的主轴</u>)的方向,是关于边界的有用的描述符。

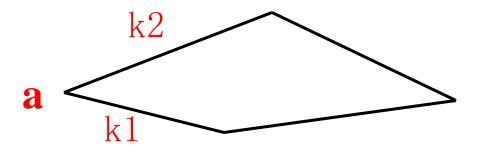


# ■ <u>边界的直径</u>举例





### ■ 边界的曲率:



曲率被描述为斜率的变化率。近似:用相邻边界线段(描述为直线)的斜率差作为在边界线交点处的曲率描述。

交点a处的曲率为 dk = k1 - k2 其中k1、k2 为相邻线段的斜率



#### ■ 边界的凸线段点:

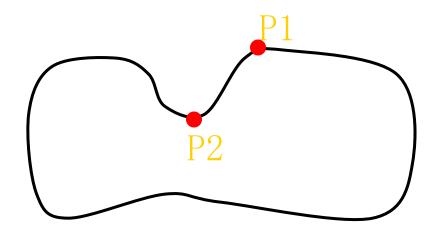
当顶点p上的曲率是非负时,称其为凸线 段上的点

## ■ 边界的凹线段点:

当顶点p上的曲率为负时,称其为凹线段上的点



- 边界的凸线段点P1:
- 边界的凹线段点P2:



## 形状数

- 形状数定义: 最小循环首差链码。
  - 循环首差链码: 用相邻链码的差代替链码

例如: 4-链码 10103322 循环首差为: 33133030

$$0 - 1 = -1(3)$$
  $3 - 3 = 0$ 

$$1 - 0 = 1$$
  $2 - 3 = -1(3)$ 

$$0 - 1 = -1(3)$$
  $2 - 2 = 0$ 



#### ■ 形状数定义:

例如: 4-链码 : 10103322

循环首差 : 33133 030

形状数 : 03033133

- <u>形状数的阶</u>定义:序列的长度n,即形状数中 阿拉伯数字的个数。
- 对于封闭边界的形状数阶是偶数。如order4、 6、8。



## ■形状数示例:

**阶**4

於8

链码: 0321

首差: 3333

形状: 3333

链码: 003221

首差: 303303

形状: 033033

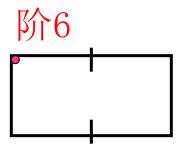
链码: 00032221

首差: 30033003

形状: 00330033



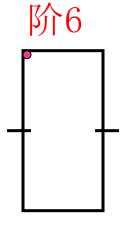
#### ■ 形状数示例:



链码: 003221

首差: 303303

形状: 033033



旋转90度,不同起点

链码: 033211

首差: 330330

形状: 033033



### ■形状数

■ 问题:

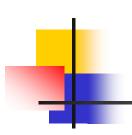
虽然链码的首差具有旋转不变性,但不具有尺度不变性。

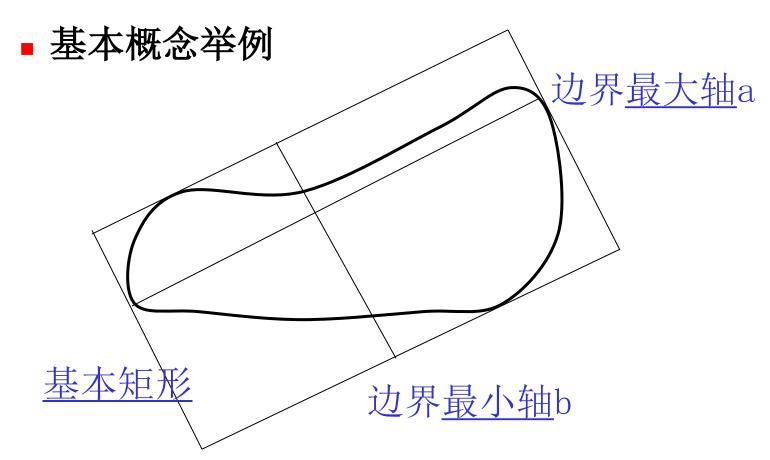
■ 改进:

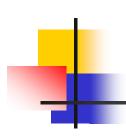
规整化网格,具体方法如下:



- 几个基本概念:
  - · 边界最大轴a:是连接距离最远的两个点的线段
  - 边界<u>最小轴</u>b:与最大轴垂直,且其长度确定的 包围盒刚好包围边界。
  - 边界<u>离心率</u>c: 最大轴长度与最小轴长度的比 c = a / b
  - ■基本矩形:包围边界的矩形。







#### ■ 规整化网格算法的思想:

大多数情况下,将链码网格与基本矩形对 齐,得到一个唯一的形状数。

对一个给定的形状数的阶,处理步骤如下:

(1) 找出一个阶为n的矩形,<u>它的离心率最接近</u> 于给定形状的基本矩形的离心率。



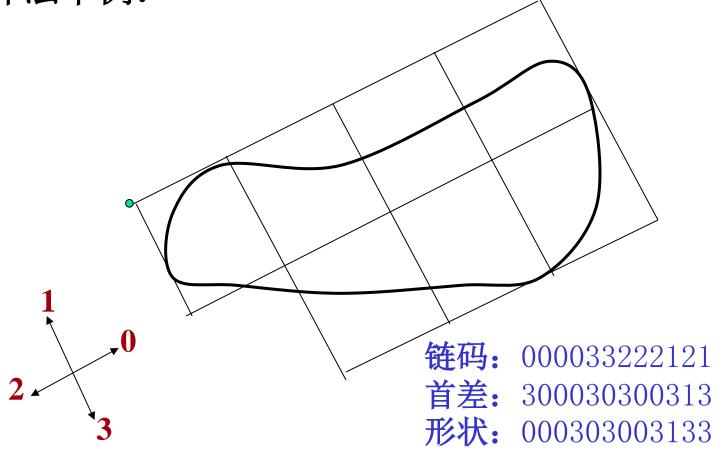
(2) 用这个矩形构造网格。

例如:如果n=12,所有阶为12的矩形(即周长为12)为2\*4,3\*3,1\*5。如果2\*4矩形的离心率最接近于给定边界的基本矩形的离心率,建立一个2\*4的网格。

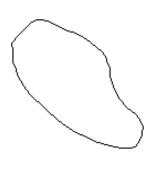
- (3) 求出与边界最吻合的多边形,再得到链码。
- (4) 最后,得到循环首差。
- (5) 首差中的最小循环数即为形状数。

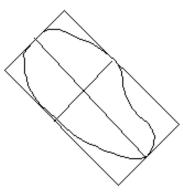


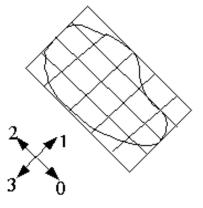
规整化网格方向 算法举例:

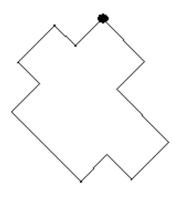












(a) 原边界

(b) 选取长短比最接近原边界的矩形以及相应坐标轴

(c) 将矩形进行等间 隔划分。四方向如下 面的坐标所示。

(d) 得到与边界最吻合的多边形。起始点用黑点标出。

链码: 003003323222121101

首差: 303103031300313031

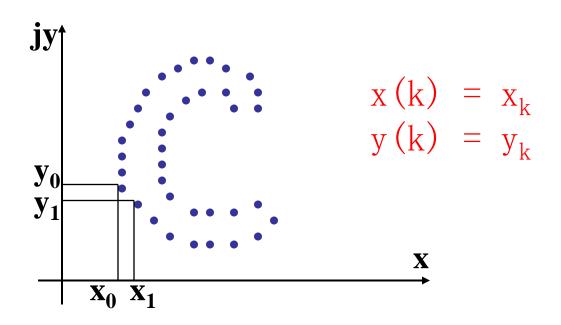
形状: 003130313031030313

#### 获取形状数的步骤

# ■傅立叶描述符

- 1) 基本思想:
  - (1) 对于XY平面上的每个边界点,将其坐标用复数表示为:

$$s(k) = x(k) + jy(k)$$
  $k=0,1,...,N-1$ 

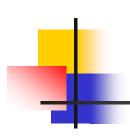


#### (2) 进行离散傅立叶变换

$$a(u) = 1/N \sum_{k=0}^{N-1} s(k) exp(-j2\pi uk/N)$$
  $u=0,1,...,N-1$ 

$$s(k) = \sum_{i=0}^{N-1} a(u) \exp(j2\pi u k/N) \qquad k=0,1,...,N-1$$

a(u) 称为边界的傅立叶描述符



(3) 选取整数 M≤N-1, 只利用前M个系数进行傅立叶 反变换(重构)

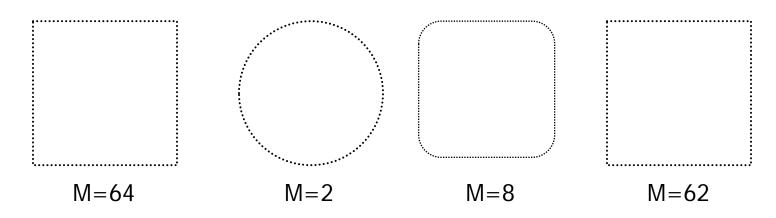
$$s'(k) = \sum_{u=0}^{M-1} a(u) \exp(j2\pi u k/N)$$
  $k=0,1,...,N-1$ 

这时,对应于边界的点数没有改变,但在重构每一个点所需要的计算项大大减少了。如果边界点数很大,M一般选为2的指数次方的整数。



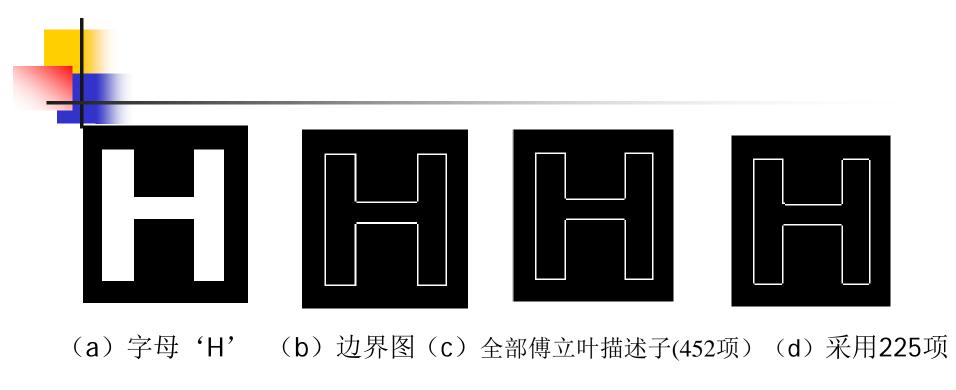
#### ■ 2) M的选取与描述符的关系

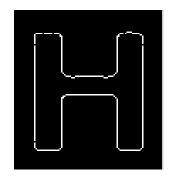
在上述方法中,相当于对于u > M-1的部分舍去不予计算。由于傅立叶变换中高频部分对应于图像的细节描述,因此M取得越小,细节部分丢失得越多。



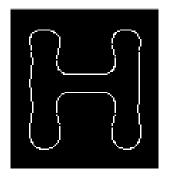
傅立叶变换中低频部分对应总体形状 利用前M个系数,用较少的数据量表达边界的基本形状

66

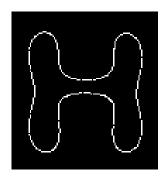




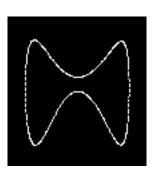
(e) 采用45项



(f) 采用27项



(g) 采用18项



(h) 采用9项

边界的傅立叶描述子及重构



#### ■ 3) 优点

- (1) 使用复数作为描述符,对于旋转、平移、放缩等操作和起始点的选取不敏感。
- (2) 旋转、平移、放缩等几何变换均可以通过对描述 子函数作简单变换来获得。

变换/变化	边界点序列	傅里叶变换系数序列
平移( $\Delta x, \Delta y$ )	$s_{t}(k) = s(k) + \Delta xy$	$S_{t}(w) = S(w) + \Delta x y \bullet \delta(w)$
旋转 $(\theta)$	$s_{\rm r}(k) = s(k) \exp(\mathrm{j}\theta)$	$S_{\rm r}(w) = S(w) \exp(\mathrm{j}\theta)$
尺度( $C$ )	$s_{c}(k) = C \bullet s(k)$	$S_{c}(w) = C \bullet S(w)$
起点 $(k_0)$	$s_{p}(k) = s(k - k_0)$	$S_{p}(w) = S(w) \exp(-j2\pi k_0 w/N)$

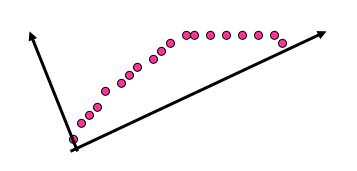


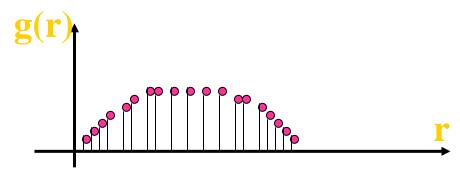
#### 矩量

■ 基本思想:

将描述形状的任务减少至描述一个一维函数, 边界段和标记的形状可以用矩量来量化描述

- 矩量的定义:
  - 把边界当作函数g(r),将g(r)的线下面积看成一个直方图







#### ■ 矩量的定义:

$$\mu_n(r) = \sum_{i=1}^{L} (r_i - m)^n g(r_i)$$

$$i=1$$

$$m = \sum_{i=1}^{L} r_i g(r_i) (均值)$$

$$i=1$$

这里L是边界上点的数目,  $\mu_n(r)$ 是边界的矩量

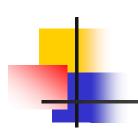
 $\mu_2(r)$ : 描述了边界曲线相对于均值的分布

μ<sub>3</sub>(r): 描述了边界曲线相对于均值的对称性



#### ■ 矩量的优点:

- ■直接实现
- 附带了一种关于边界形状的"物理"解释
- 对于旋转的不敏感性
- 为了使大小比例不敏感,可以通过伸缩r的 范围将大小正则化。



# 9.5 区域描述

- ■简单描述
- 拓扑描述符
- ■形状描述符
- 纹理描述符



#### ■ 简单描述:

- 区域面积A: 对属于区域的像素计数 简单、准确
- 区域重心:

$$\bar{x} = \frac{1}{A} \sum_{(x,y) \in R} x$$
$$\bar{y} = \frac{1}{A} \sum_{(x,y) \in R} y$$

区域灰度:目标灰度的最大值、最小值、中值、 平均值、方差、高阶矩等

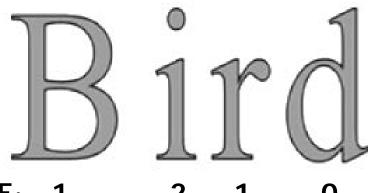


#### ■ 拓扑描述符:

- 欧拉数E是一个区域的拓扑描述符
- E=C-H

C: 区域内的连通组元

H: 区域内的孔数



字母的欧拉数E: -1

74

#### ■ 形状描述符:

- 形状参数: F=||B||<sup>2</sup>/4πA B是区域周长,A是面积 描述区域的紧凑性,对尺度变化、旋转不敏感
- 偏心率:

描述区域的紧凑性

- (1)常用的简单方法: 边界长轴长度与短轴长度的比值
- (2) 较好的方法: 利用整个区域的所有像素, 提高抗干扰能力
- 球状性:  $S=r_i/r_c$   $r_i$ : 区域内切圆半径  $r_c$ : 区域外接圆半径 不受区域平移、旋转、尺度变化的影响
- 圆形性: C=μ<sub>R</sub>/σ<sub>R</sub>

 $\mu_{R}$ . 从区域重心到边界点的平均距离;

σ<sub>R</sub>: 从区域重心到边界点的距离的均方差

75



## • 纹理描述符:

• (1) 统计法: 直方图的矩描述纹理 矩的定义:

$$\mu_n(r) = \sum_{i=1}^{L} (r_i - m)^n g(r_i)$$

$$i = 1$$

$$m = \sum_{i=1}^{L} r_i g(r_i) \quad (均値)$$

$$i = 1$$

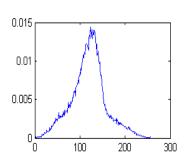
这里L是边界上点的数目, μ<sub>n</sub>(r)是边界的矩量

- (2) 结构法: 纹理基元有规律的重复
- (3) 频谱法:

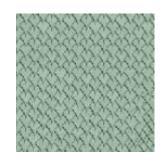


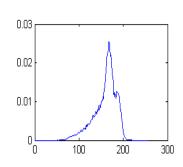
## (1) 统计法:





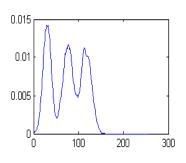
木纹





周期纹理





砖块

#### 纹理图像及其直方图



- 图像平面的区域不变矩: 区域的矩
- f(x,y)的p+q阶矩定义为:

$$m_{pq} = \sum_{x} \sum_{y} x^{p} y^{q} f(x,y)$$
  $m_{pq} = f(x,y) -- \text{Min}$ 

•f(x,y)的p+q阶中心矩定义为:

$$\mu_{pq} = \sum_{x} \sum_{y} (x - \bar{x})^{p} (y - \bar{y})^{q} f(x, y)$$

$$\pm \bar{x} = \frac{m_{10}}{m_{00}} \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

矩对平移、旋转、尺度具有不变性

## 归一化的中心矩:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad p, q = 0, 1, 2, \dots$$

$$\gamma = \frac{p+q}{2} + 1$$
  $p, q = 2,3,4,...$ 

对平移、旋转、镜面以及尺度变换的不变性

# 7个不变矩

$$\phi_{1} = \eta_{20} + \eta_{02}$$

$$\phi_{2} = (\eta_{20} - \eta_{02})^{2} + 4\eta_{11}^{2}$$

$$\phi_{3} = (\eta_{30} - 3\eta_{12})^{2} + (3\eta_{21} - \eta_{03})^{2}$$

$$\phi_{4} = (\eta_{30} + \eta_{12})^{2} + (\eta_{21} + \eta_{03})^{2}$$

$$\phi_{5} = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^{2} - 3(\eta_{21} + \eta_{03})^{2}]$$

$$+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^{2} - (\eta_{21} + \eta_{03})^{2}]$$

$$\phi_{6} = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^{2} - (\eta_{21} + \eta_{03})^{2}] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_{7} = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^{2} - 3(\eta_{21} + \eta_{03})^{2}]$$

$$+ (3\eta_{21} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^{2} - (\eta_{21} + \eta_{03})^{2}]$$









(a) lena图

(b) 旋转-4° (c) 垂直镜像 (d) 缩小二分之一

#### lena图及其几何变换图

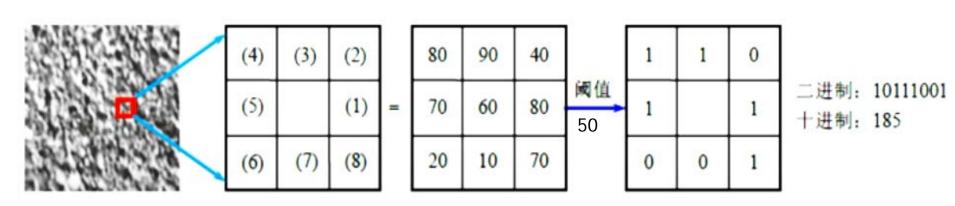
不变矩  log	Ф1	Ф2	Ф3	Ф4	Ф5	Ф6	Ф7
原图	6.621	18.802	27.382	25.206	54.294	34.822	51.502
旋转 4°	6.6209	18.802	27.382	25.206	54.308	34.822	51.502
垂直镜像	6.621	18.802	27.382	25.206	54.294	34.822	51.598
缩小一半	6.621	18.801	27.396	25.206	54.179	34.823	51.511

从表中可以看出, 在图像经过旋转、镜像以及尺度变换之后, 这 七个不变矩的值只有十分小的变化,可以看作是基本保持不变



#### (2) 结构法:

- 局部二值模式: LBP (纹理分析算子)
- 定义:对一个像素的3×3邻域的像素按顺序阈值化, 并将结果看作一个二进制数(也可转为十进制),并 作为中心像素的标号
- 有256个(2<sup>8</sup>)不同标号,统计图像每个像素标号得 到的直方图可进一步用作整个区域的纹理描述符



#### 局部二值模式分类

◆均匀模式:将一个邻域中的像素按顺序循环考虑,如果包含最多两个从0到1或从1到0的过渡,则这个二值模式是均匀的

均匀模式对应纹理

◆否则为非均匀模式 非均匀模式对应噪声

例: 11111001 (2个过渡) 为均匀模式

10111001 (4个过渡) 为非均匀模式

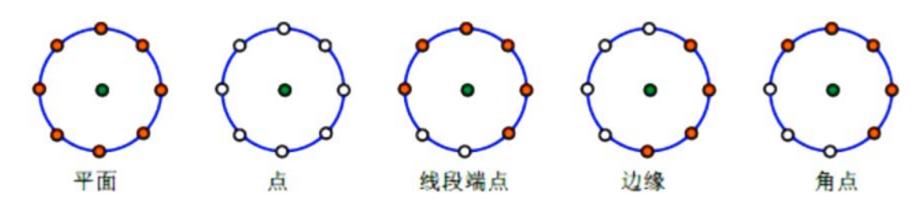
256个模式中,58个模式为均匀模式,所有非均匀模式共同使用同一个标号,所以一共59个标号



#### ■ 局部二值模式LBP:

根据LBP的标号可以获得不同的局部基元,分别对 应不同的局部纹理

示例: (以中间点为中心的8邻域,空心圆点代表1,实心圆点代表0)



用LBP标号获得的局部基元



#### (3) 频谱法:

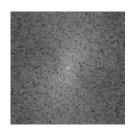
用傅里叶频谱的频率特性来描述周期或近乎周期的 2D图像模式的方向性

- ◆借助傅里叶频谱中突起的峰值确定纹理模式的主方向
- ◆这些峰在频域平面的位置来确定模式的基本周期









(a) 鹅卵石

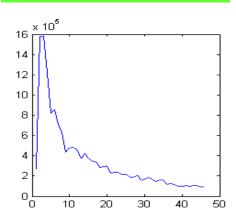
(b) 沙石 (c) 鹅卵石频谱图 (d) 沙石频谱图

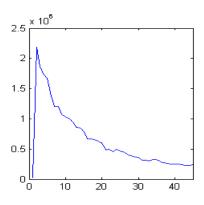
纹理图像及其频谱图

在极坐标系中,频谱用函数  $S(r,\theta)$ 表示给定方向 $\theta$ ,得  $S_{\theta}(r)$  给定频率r,得  $S_{r}(\theta)$ 

$$S(r) = \sum_{\theta=0}^{\pi} S_{\theta}(r)$$

$$S(\theta) = \sum_{r=1}^{R} S_r(\theta)$$

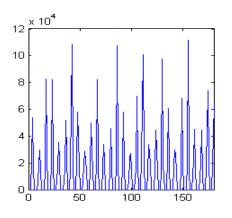




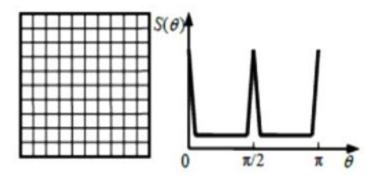
(a) 鹅卵石图像的S(r) (b) 沙石图像的S(r)

纹理图像的频谱特征

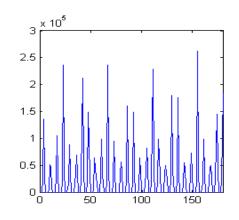




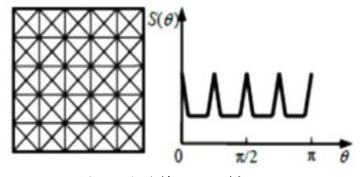
(c) 鹅卵石图像的  $S(\theta)$ 



(e) 纹理图像1及其  $S(\theta)$ 



(d) 沙石图像的  $S(\theta)$ 

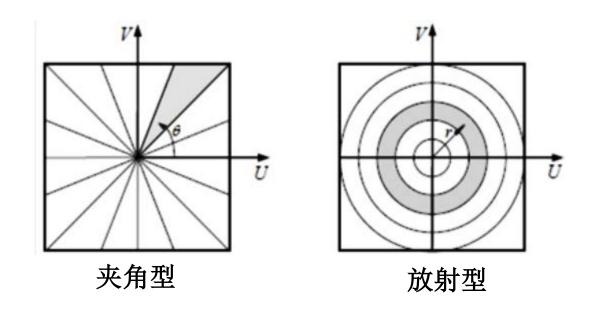


(f) 纹理图像2及其 $S(\theta)$ 

#### 纹理图像的频谱特征



#### 纹理模式特征:将傅里叶空间分块,在分块计算能量



傅里叶空间分块



# 9.6 关系描述

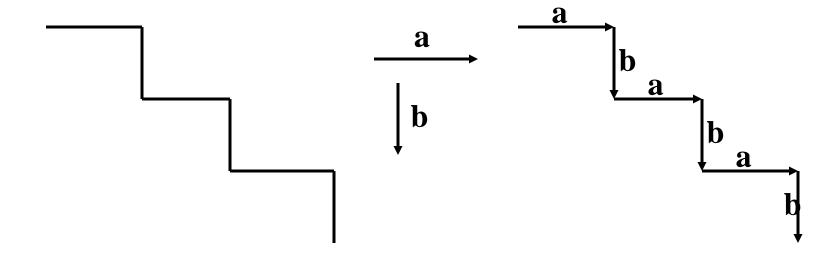
- ■基本思想
- 阶梯关系编码
- 骨架关系编码
- ■方向关系编码
- 内角关系编码
- 树结构关系编码



#### ■ 基本思想:

- 通过挖掘各个成分之间的结构关系来描述边界
- 图像中各个部分间的结构关系是二维的,而串是一维的,期望找到一种方法把二维关系转化为一维的串
- 主导思想是考虑物体各个部分的连接线段



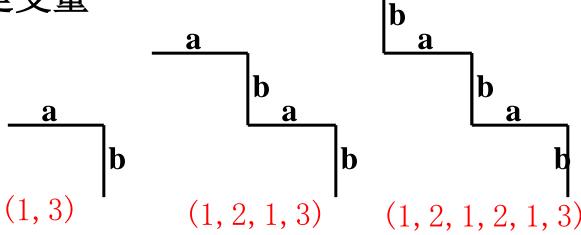




- 阶梯结构关系
  - 定义如下产生规则:
    - (1) S -> aA
    - (2) A -> bS
    - (3) A -> b

其中S、A是变量

举例:



a

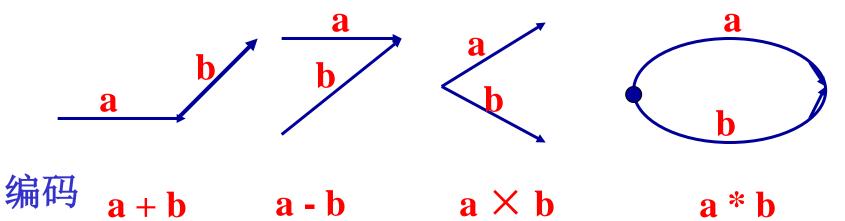


## ■骨架关系编码

- 用有向线段来描述一个图像的各个部分,这个线段是通过头尾连接等方法得到的。线段之间的不同运算代表了区域的不同组合。
- 当图像的连通性可以通过首尾相接或其它连 续方式描述时,最适于使用这种串来描述。



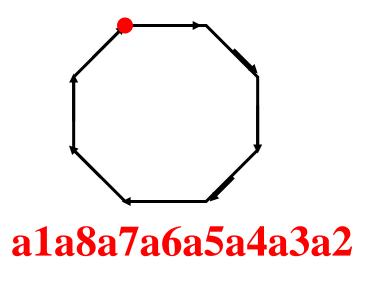
# ■骨架关系编码

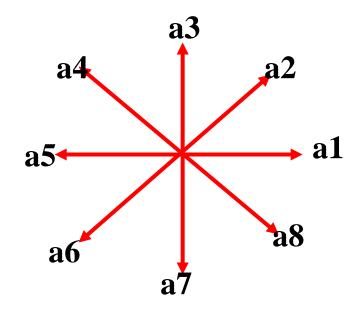




## → 方向关系编码

跟踪对象的边界,将跟踪得到的线段按照方向 或长度来编码





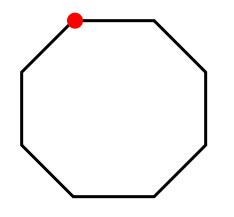


#### 内角关系编码

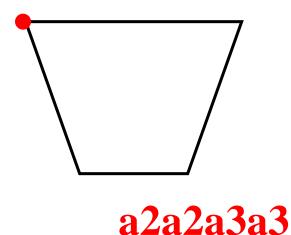
■ 根据角度范围不同,编码为8个符号

即: a1:0-45; a2:45-90;a3:90-135;...; a8:315-360

举例:



a3a3a3a3a3a3a3





## 材结构关系

树结构中每个结点的意义和结点之间的关系 最为重要

#### 举例:

