



北京航空航天大学

数字图像处理

图像分割的表示与描述

00:22:58

计算机学院



■ 内容回顾

■ 图像分割的表示与描述



- ✓ 图像分析的概念：图像中提取信息
- ✓ 图像分析系统的基本构成：
预处理、分割、特征提取、识别
- ✓ 图像分割的概念：分解、定位
- ✓ 图像分割的基本思路：
逐级、控制背景、缩小干扰
- ✓ 图像分割的基本策略：
利用不连续性和相似性，由点到线到边



- 1.边界分割法
- 2.边缘连接法
- 3.阈值分割法
- 4.面向区域的分割



- 边界分割法

- 点的检测
- 线的检测
- 边的检测



• 点的检测

—用空域的高通滤波器来检测孤立点

例：3×3的掩模，掩模在该图像中任何一点处的响应R由下式给出

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9 = \sum_{i=1}^9 w_i z_i$$

z_i 是与掩模系数 w_i 相关的像素的亮度

点检测： $|R| \geq T$ T是非负阈值

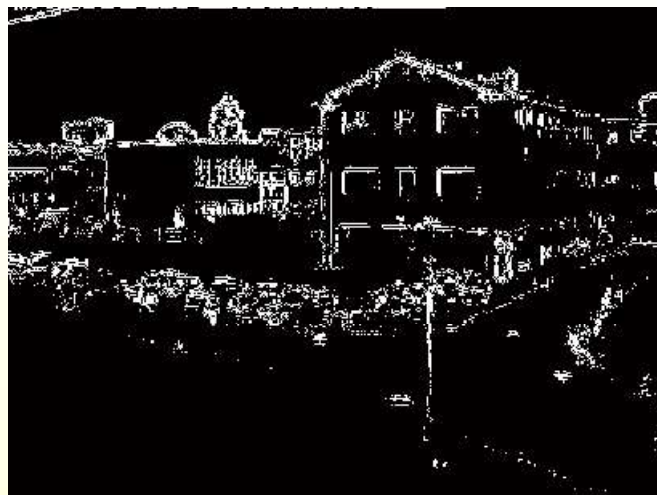


- 点的检测

- 当掩模的中心位于一个孤立点时，掩模的响应必须最强。即当R的值足够大时，说明该点的值与周围的点非常不同，是孤立点。通过阈值T来控制如 $T = 32、64、128$ 等，如 $|R| > T$ ，则便检测到一个孤立点
- 在亮度不变的区域中响应为零



北京航空航天大学



$T=16$

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



$T=32$

00:22:58

计算机学院



• 线的检测

—通过比较典型模板的计算值，确定一个点是否在某个方向的线上

-1	-1	-1
2	2	2
-1	-1	-1

最佳响应水平线

-1	-1	2
-1	2	-1
2	-1	-1

最佳响应+45度线

-1	2	-1
-1	2	-1
-1	2	-1

最佳响应垂直线

2	-1	-1
-1	2	-1
-1	-1	2

最佳响应-45度线

每个掩模的系数之和为零，表明亮度不变区域中来自掩模的响应为零。



北京航空航天大学

-1	-1	-1	-1	-1	2	-1	2	-1	2	-1	-1
2	2	2	-1	2	-1	-1	2	-1	-1	2	-1
-1	-1	-1	2	-1	-1	-1	2	-1	-1	-1	2



10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	10	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	10	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	10	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	10	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	10	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	10	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	10	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	10	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	10	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	10	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	10



00:22:58





• 边的检测

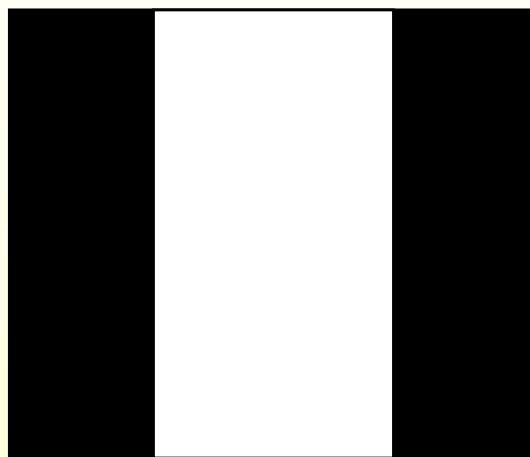
- **边界的定义**：一段边是两个具有相对不同灰度值特性的区域的边界线
- **适用于**：假定问题中的区域是非常类似的，两个区域之间的过渡，仅仅根据灰度的不连续性便可确定
- **不适用于**：当假定不成立时，阈值分割技术一般来说比边缘检测更加实用



- 边的检测

—基本思想：
计算局部微
分算子

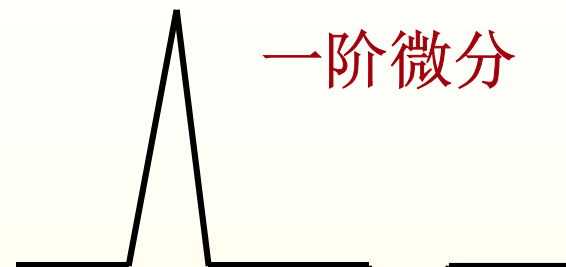
边界图像



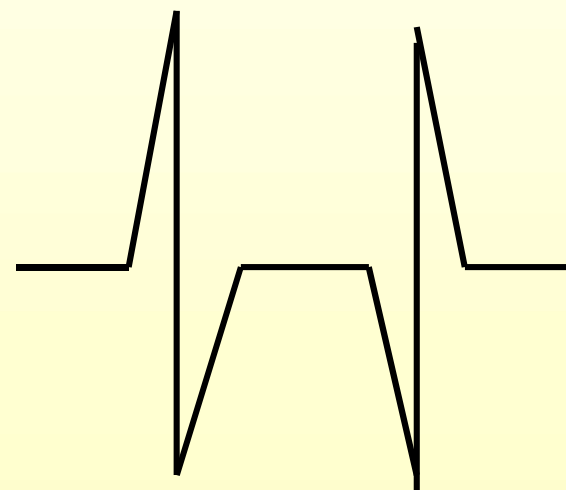
截面图



一阶微分



二阶微分





• 边的检测

— 一阶微分：用梯度算子来计算

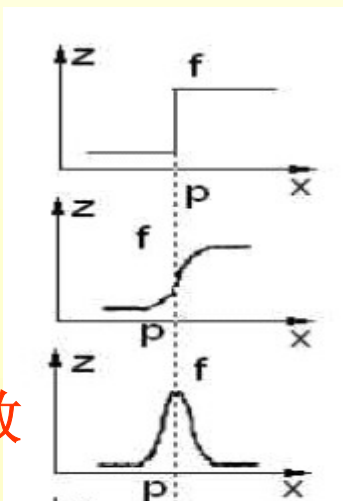
- ✓ 特点：对于由暗到亮的边缘，边的一阶微分为正值；对于由亮到暗的边缘，边的一阶微分为负值；常数部分为零；
- ✓ 用途：用于检测图像中边的存在。

图像

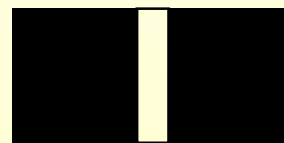


剖面

一阶导数

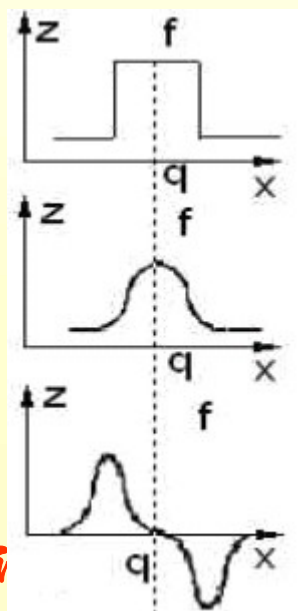


图像



剖面

一阶导数

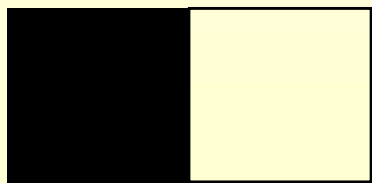




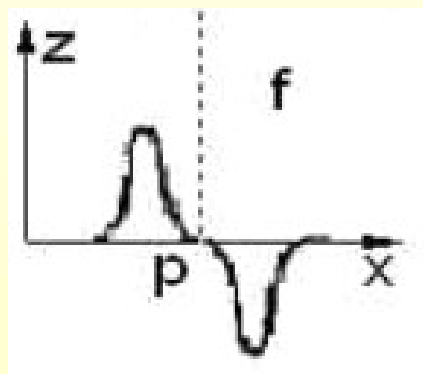
- 边的检测

- 二阶微分：通过拉普拉斯来计算

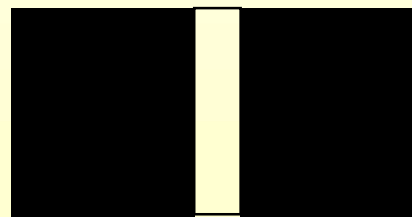
- 特点：二阶微分边缘与黑色一边相关的跃变点二阶微分为正，与亮色一边相关的跃变点二阶微分为负；常数部分为零。



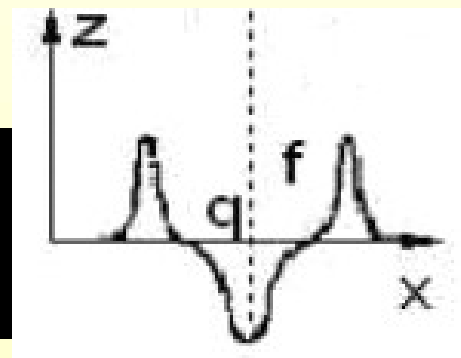
图像



二阶导数



图像



二阶导数



- 用途：
 - 1) 二次导数的符号，用于确定边上的像素是在亮的一边，还是暗的一边。
 - 2) 0跨越，确定边的准确位置



- 边缘连接法

- 由于噪音原因，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。
- 因此典型的边检测算法后面总要跟随着连接过程和其它边界检测过程，用来归整边像素，成为有意义的边。



- 边缘连接法

- 局部处理法：对已完成边界检测的图像进行；连接间断的边；用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。
- Hough变换：在找出边界点集之后，需要连接，形成完整的边界图形描述；对于边界上的n个点的点集，找出共线的点集和直线方程；进行扩展。

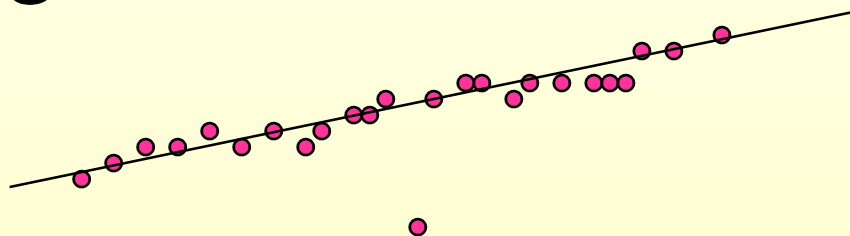


➤ 局部处理法：

$$|\nabla f(x,y) - \nabla f(x',y')| \leq T$$

$$|\alpha(x,y) - \alpha(x',y')| < A$$

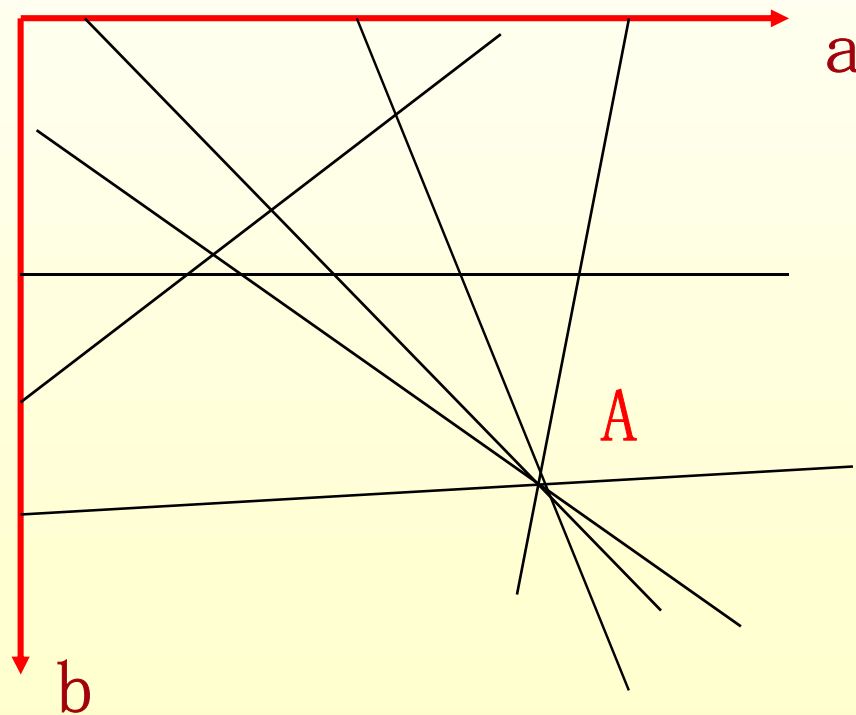
➤ Hough变换：



设给定图像中的 n 个点，要从中确定连在同1条直线上的点的子集。即已检测出1条直线上的若干个点，需要求它们所在的直线。



- Hough变换的基本思想:点 - 线对偶性
 - 图像空间中共线的点对应在参数空间里相交的线





• 阈值分割法

- 通过交互方式得到阈值：光标获得样点值，选取容忍度 R
- 通过直方图得到阈值：波谷—边界上的点的灰度值出现次数较少
- 通过边界特性选择阈值：只把区域边缘的像素绘入直方图，改善直方图的波峰形状（波峰的对称性、高度）
- 简单全局阈值分割：规定阈值 T ，直接比较
- 分割连通区域
- 基于多个变量的阈值



- 面向区域的分割
 - 基本概念
 - 通过像素集合的区域增长
 - 区域分裂与合并



- 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：

- 1) **完备性**：分割应将图像中的每个像素都分进某1个子区域中

- 2) **连通性**：每个 R_i 都是一个连通区域

- 3) **独立性**：各个子区域互不重叠

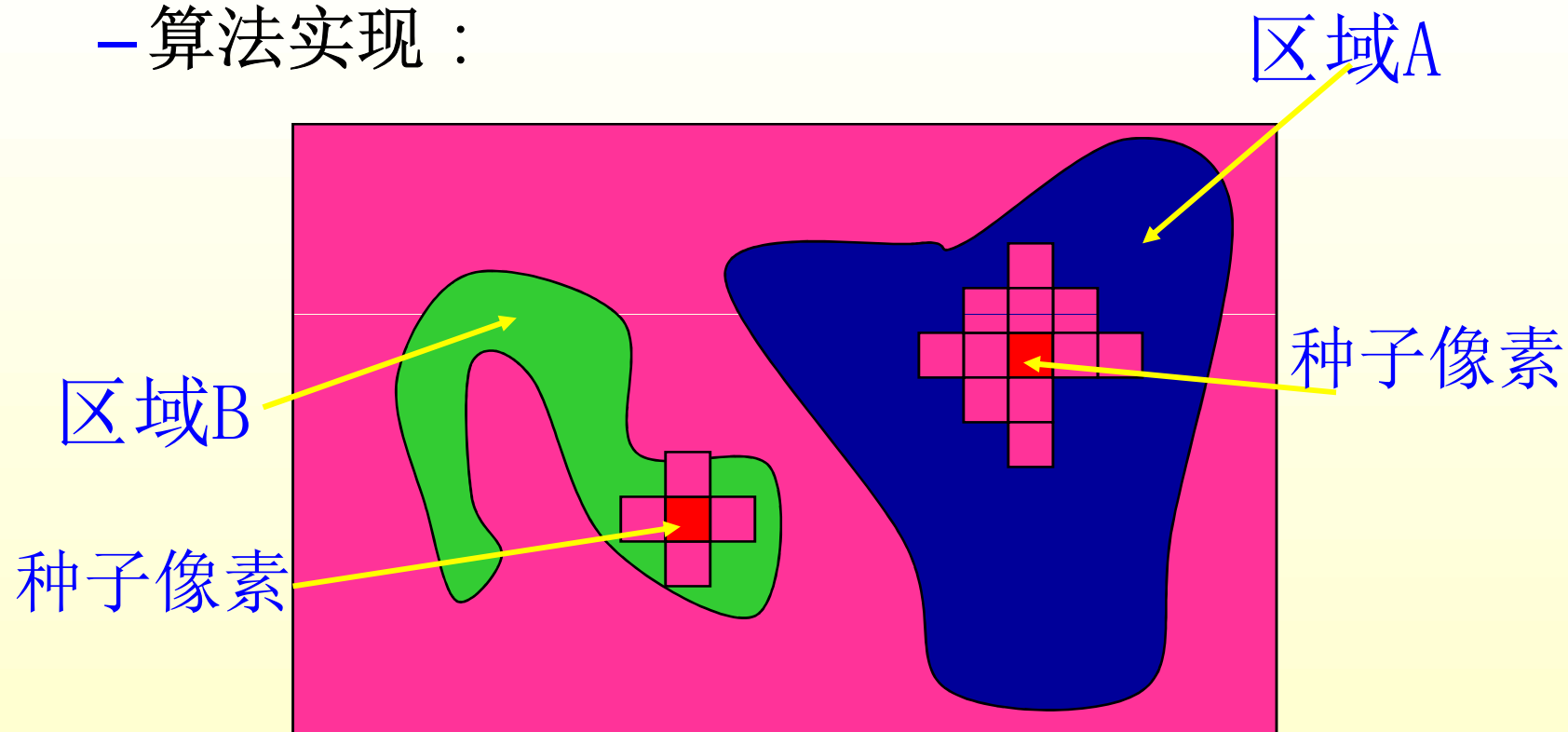
- 4) **单一性**：属于同一区域的像素应该具有某些相同特性

- 5) **互斥性**：属于不同区域的像素应该具有某些不同特性



- 通过像素集合的区域增长

—算法实现：



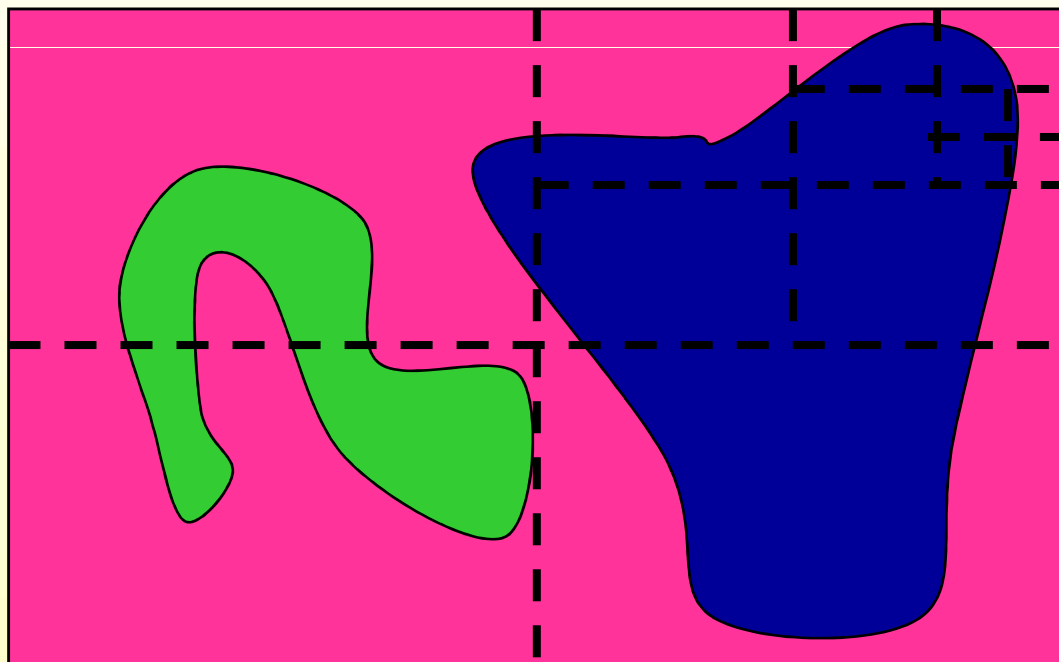
- 选种子
- 选描述符
- 判断增长结束条件



4—面向区域的分割

- 区域分裂与合并
— 算法实现：

- 1) 对于图像中灰度级不同的区域，均分为四个子区域。
- 2) 如果相邻的子区域所有像素的灰度级相同，则将其合并。
- 3) 反复进行上两步操作，直至不再有新的分裂与合并为止。





北京航空航天大学

数字图像处理

图像分割的表示与描述

00:22:58

计算机学院



- 表示与描述的基本概念
- 边界分割表示法设计
- 边界描述子
- 关系描述子



- 基本概念

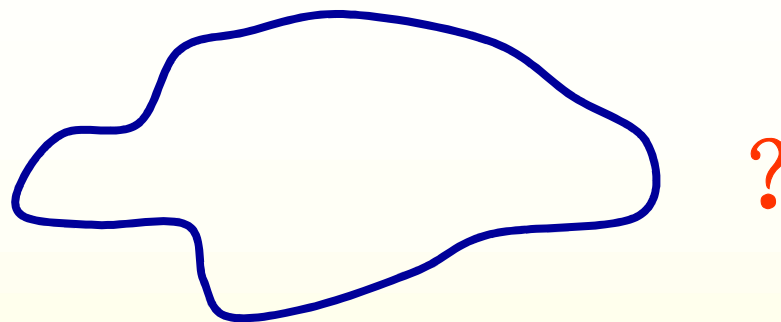
- 图像分割后，为了进一步处理，分割后的图像一般要进行形式化的表达和描述
- 表达侧重于数据结构，而描述侧重于区域特性以及不同区域之间的联系和差别
- 解决形式化表达问题一般有两种选择：
 - 1) 根据区域的外部特征来进行形式化表示
 - 2) 根据区域的内部特征（比较区域内部的像素值）来进行形式化表示



- 基本概念

- 选择表达方式，要本着使数据变得更有利于下一步的计算工作（节省存储空间、易于特征计算）。下一步工作是基于所选的表达方式描述这个区域，一般情况下：

- 1) 如果关注的焦点是形状特性，选择外部表示方式
- 2) 如果关注的焦点是反射率特性，如颜色、纹理时，选择内部表示方式。
- 3) 所选表示方式，应该对尺寸、变换、旋转等变量尽可能的不敏感



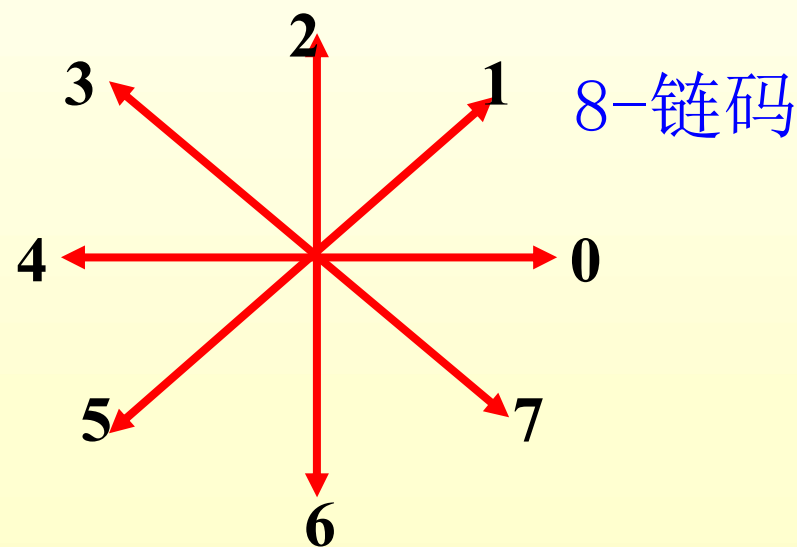
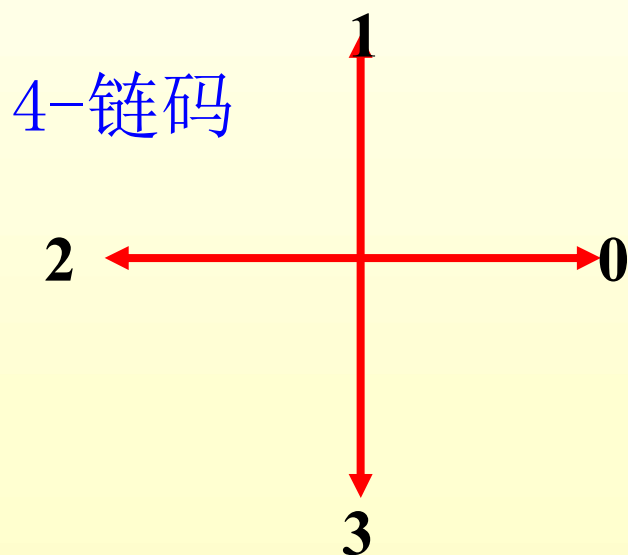
- 表示法设计
 - 链码
 - 多边形逼近
 - 外形特征
 - 边界分段
 - 区域骨架



• 链码

—定义：1) 链码是一种边界的编码表示法。

2) 用边界的方向作为编码依据。为简化边界的描述。一般描述的是边界点集。





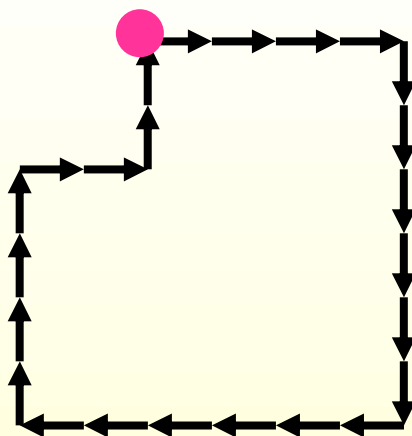
- 链码

- 算法：

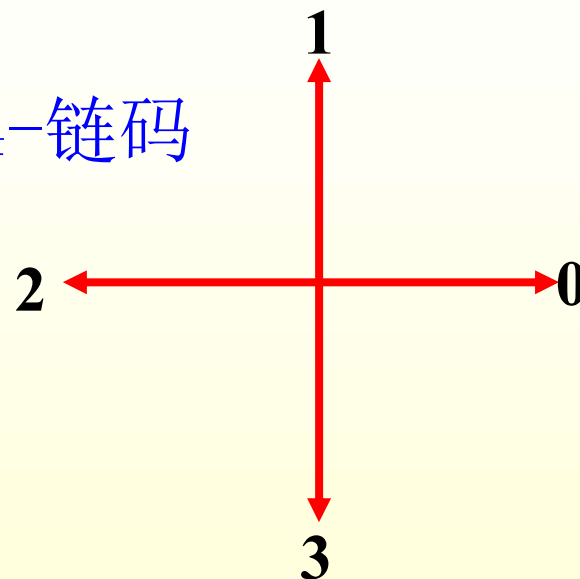
- 给每一个线段一个方向编码。
 - 有4-链码和8-链码两种编码方法。
 - 从起点开始，沿边界编码，至起点被重新碰到，结束一个对象的编码。



- 链码举例：



4-链码



4-链码：

000033333322222211110011



- 边界的起点需用（绝对）坐标表示，其余点都可只用连续方向（相对）来代表偏移量。
- 由于表示1个方向数比表示1个坐标值所需比特数少，而且对每1个点又只需1个方向数就可以代替2个坐标值，所以链码表达可以大大减少边界表示所需的数据量。



- 链码

- 问题1：

- 1) 链码相当长。

- 2) 噪音会产生不必要的链码：噪声等干扰会导致小的边界变化，而使链码发生与目标整体形状无关的较大变动。

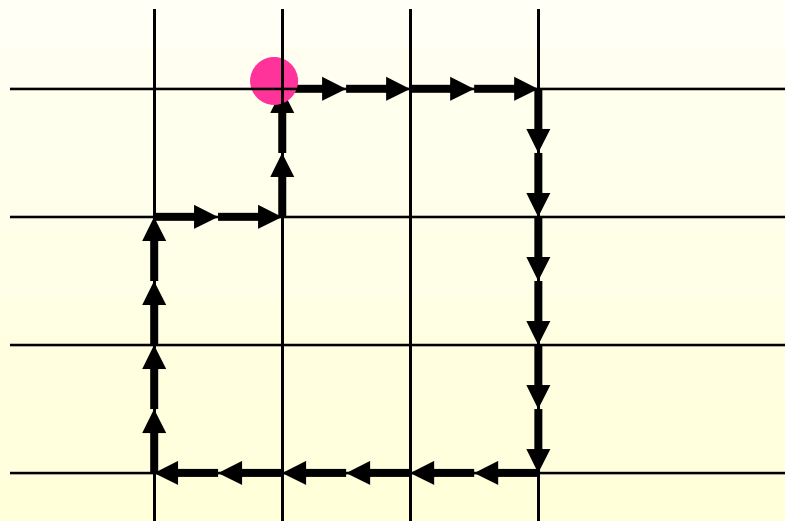
- 改进：

- 1) 加大网格空间。

- 2) 把与原边界点最接近的大网格点定为新的边界点。



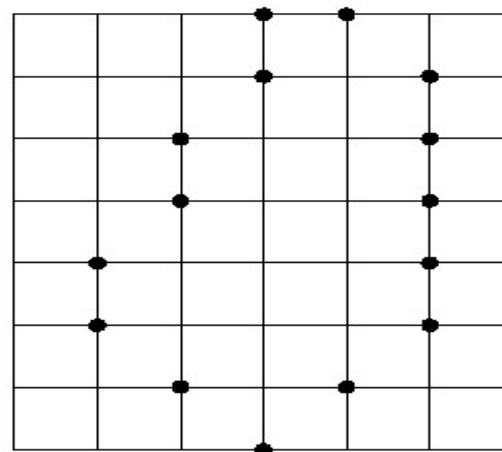
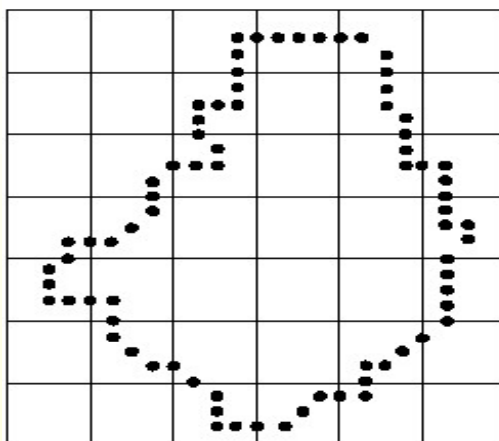
- 链码举例：



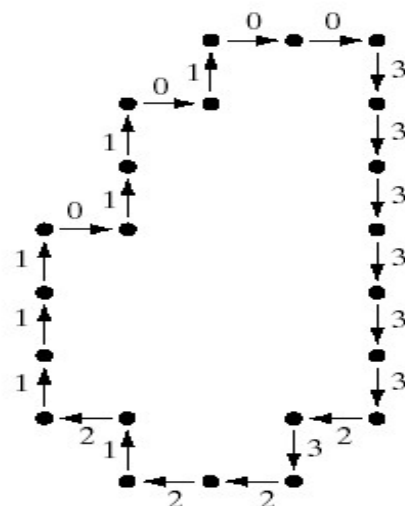
4-链码：003332221101



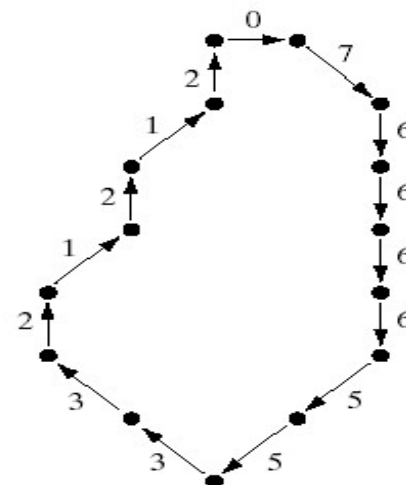
- 链码举例：



4-链码



8-链码





- 链码

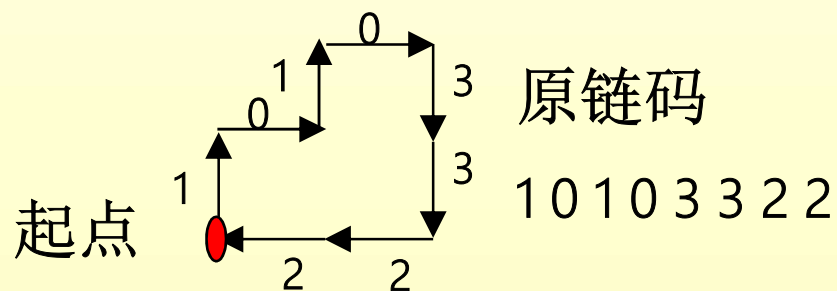
- 问题2：

由于起点的不同，造成编码的不同

- 改进：

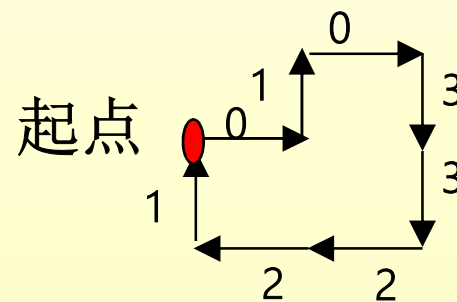
1) 从固定位置作为起点(最左最上)开始编码

2) 将这些方向数依1个方向循环以使它们所构成的自然数的值最小



归一化链码

01033221





- 链码

- 问题3：

- 由于角度的不同，造成编码的不同

- 改进：

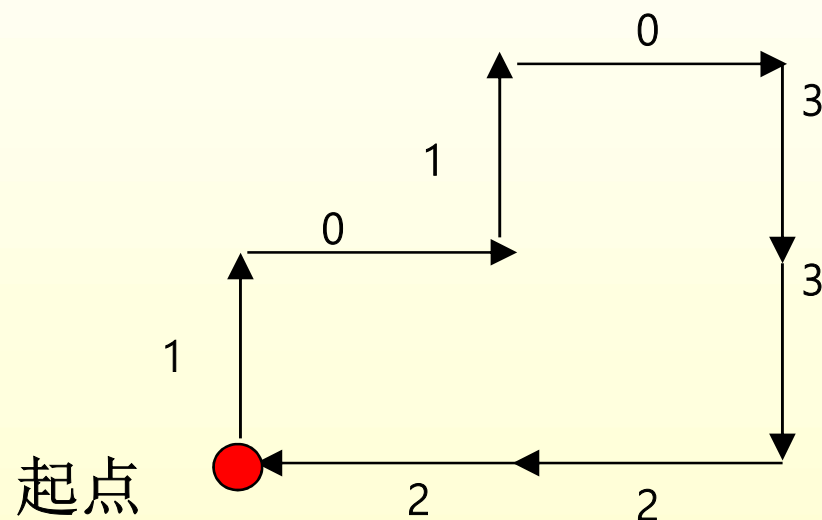
- 通过使用链码的首差代替码子本身的方式



- 链码

—循环首差链码：用相邻链码的差代替链码

例如：



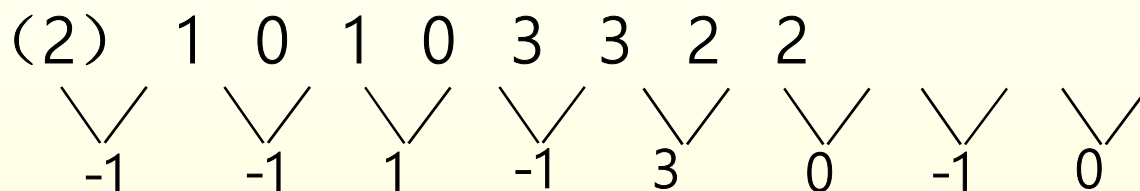
4-链码10103322



- 链码

4-链码10103322

循环首差为：33133030

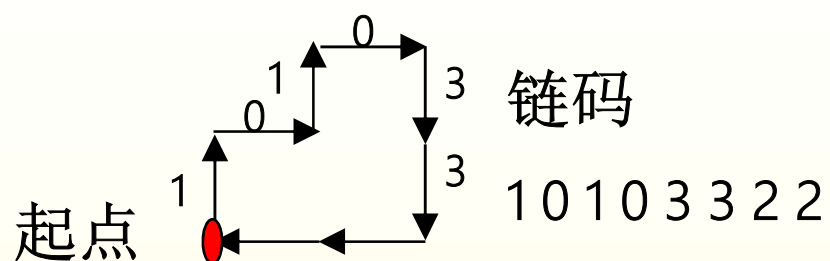


循环首差： $1 - 2 = -1(3)$ $3 - 0 = 3$

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

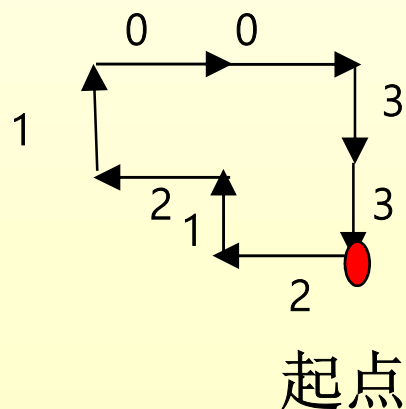
$1 - 0 = 1$ $2 - 3 = -1(3)$

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



循环首差为：33133030

↓ 逆时针旋转90度



循环首差为：33133030



- 链码

- 应用背景：

- 如果边界的本身对于旋转和比例修改来说是无变化的，使用链码才是正确的。一般来说这是不可能的，实际应用时还需要改进。
 - 用链码后，对象只要用1)起点坐标，2)周长（边界点数）3)链码，4)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

- 在数字图像中，如果多边形的线段数与边界上的点数相等，则多边形可以完全准确地表达边界。
- 基本思想：用最少的多边形线段，获取边界形状的本质。
- 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

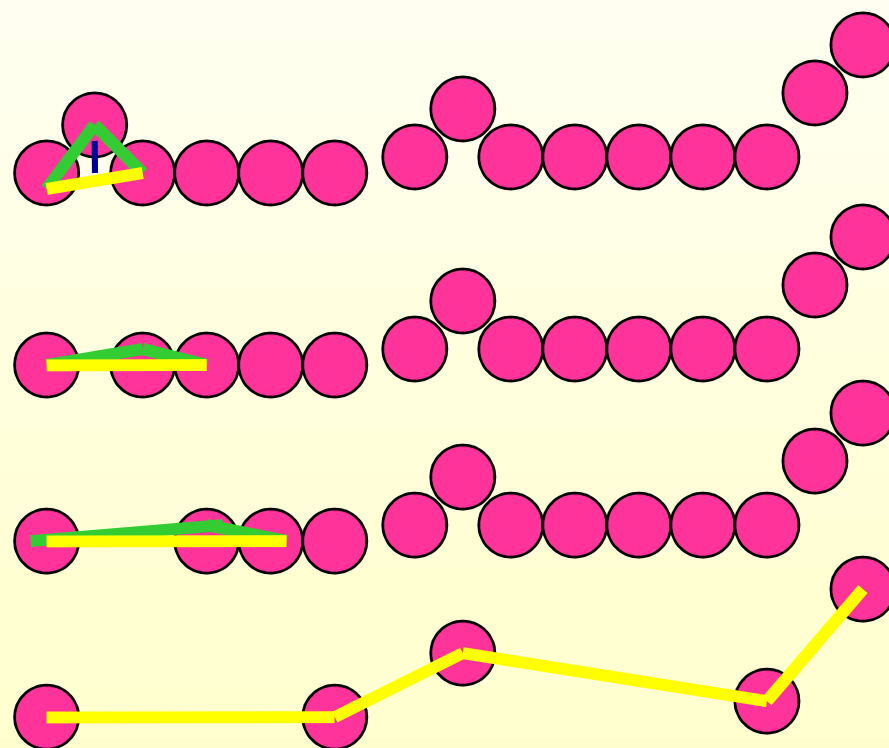
- 合成点算法：

- 1) 沿着边界选两个相邻的点对，计算首尾连接直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) 当程序的第一个起点被遇到，程序结束。



- 多边形逼近

— 点合成算法思想举例：





- 多边形逼近

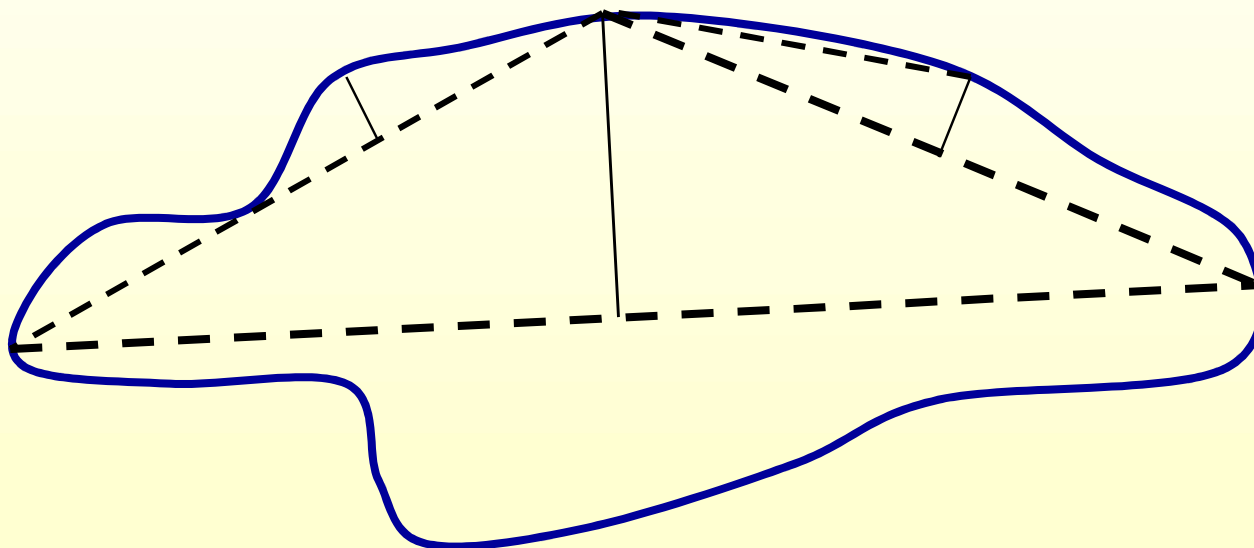
- 合成点算法的问题：

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

下面讲到的分裂法可用于缓解这个问题。



- 多边形逼近
 - 边分裂算法思想举例：





- 多边形逼近

- 分裂边算法：

- (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。



- 外形特征

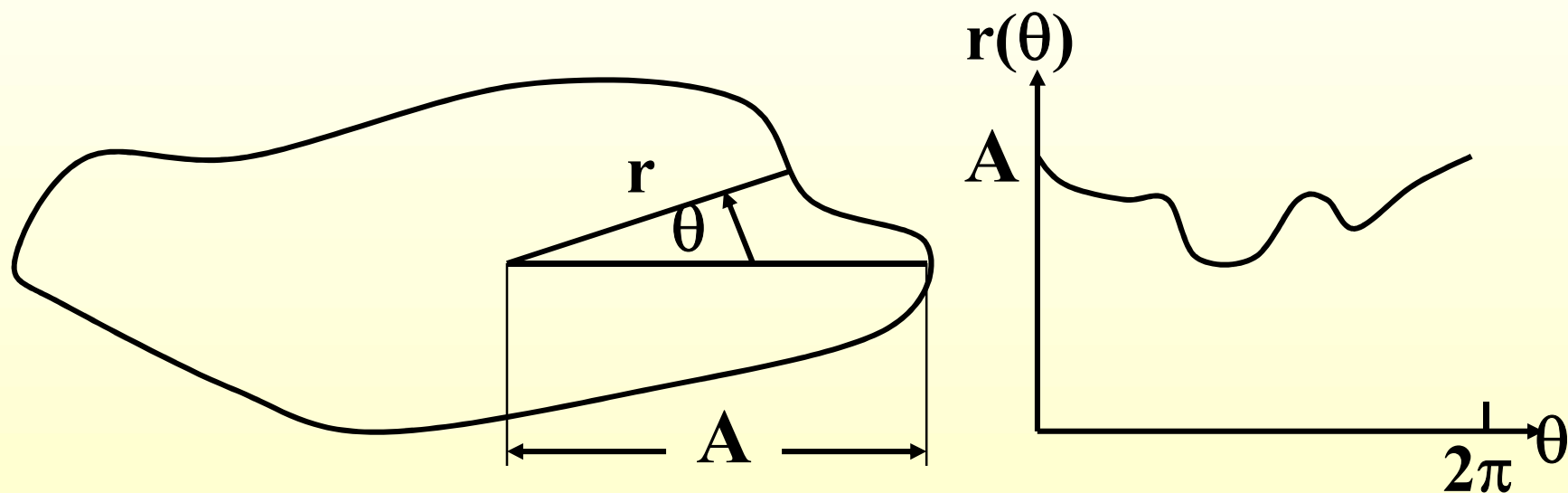
- 基本思想：

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。



- 外形特征

- 函数定义——质心角函数：边上的点到质心的距离 r ，作为夹角的 θ 的函数。

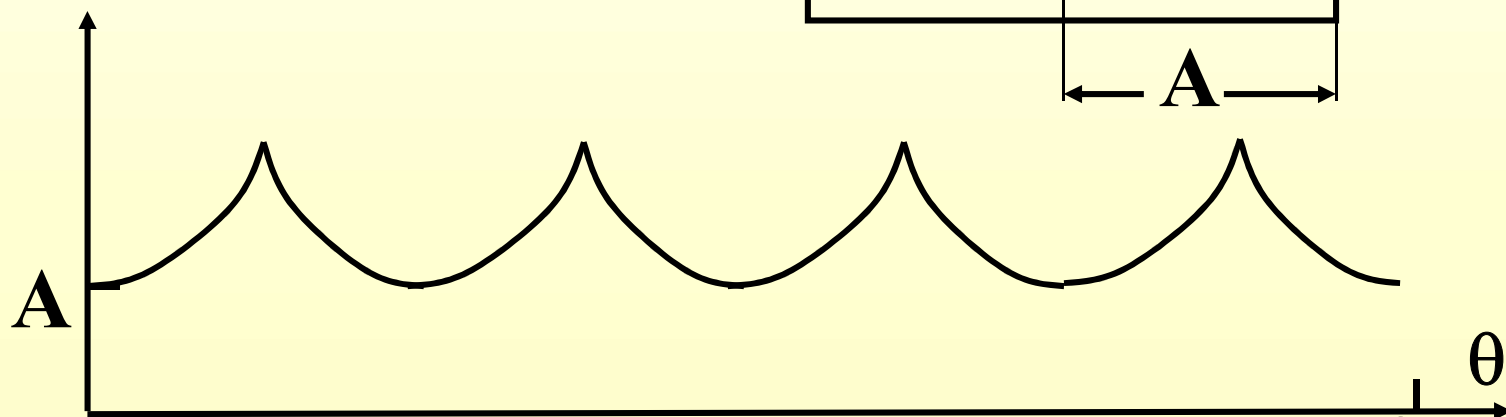




- 外形特征

— 举例：

$$r(\theta) = A * \sec(\theta)$$



00:22:58

计算机学院



- 外形特征

- 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. 选择从质心到主轴最远的点作为起点

- 对于比例变换：

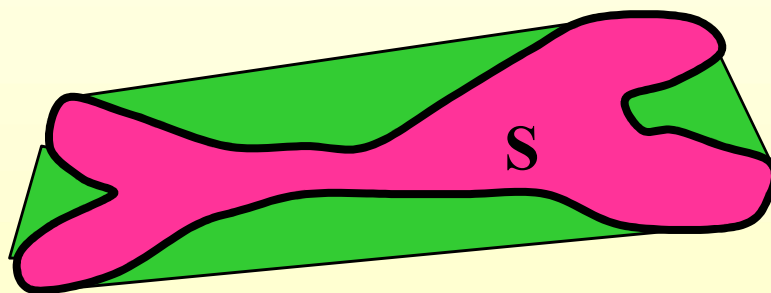
- 对函数进行正则化，使函数值总是分布在相同的值域里，比如说 $[0, 1]$



• 边界分段

—基本概念：

- 一个任意集合S（区域）的凸起外缘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的凸起补集D



S



D

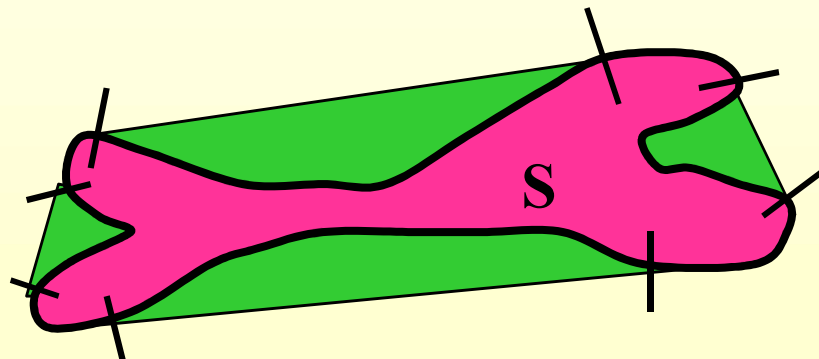
$$S + D = H$$



- 边界分段

- 分段算法：

- 给进入和离开凸起补集D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





- 边界分段

- 问题：

- 噪音的影响，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



• 区域骨架

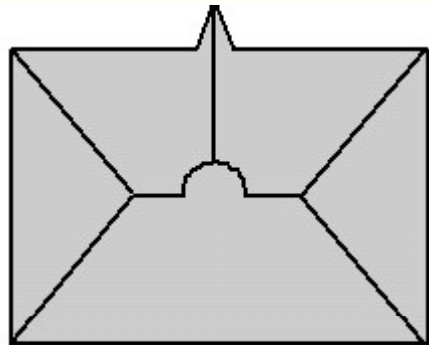
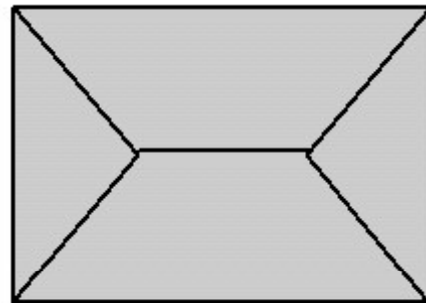
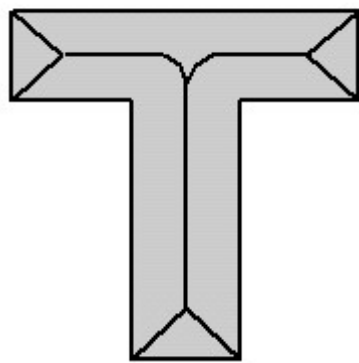
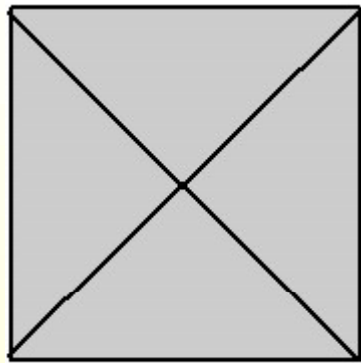
— 基本思想

- 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:Media Axis Transform）
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）

即有2个或以上的B中的点与p同时最近



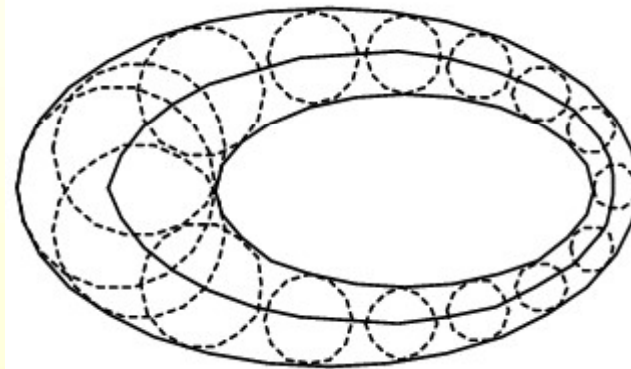
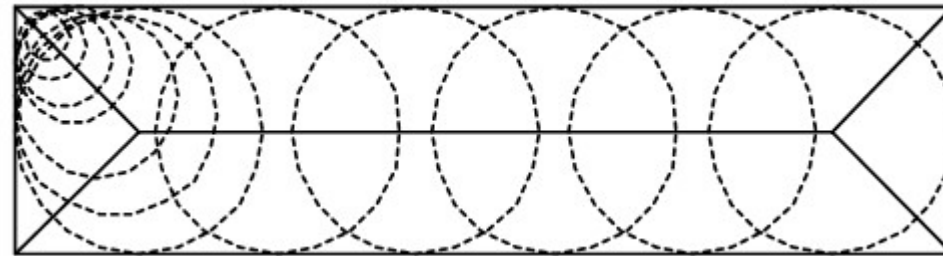
北京航空航天大学 表示法设计 - 区域骨架



骨架上的点P能在边界上找到至少2个点和P距离相等！



北京航空航天大学 表示法设计 - 区域骨架

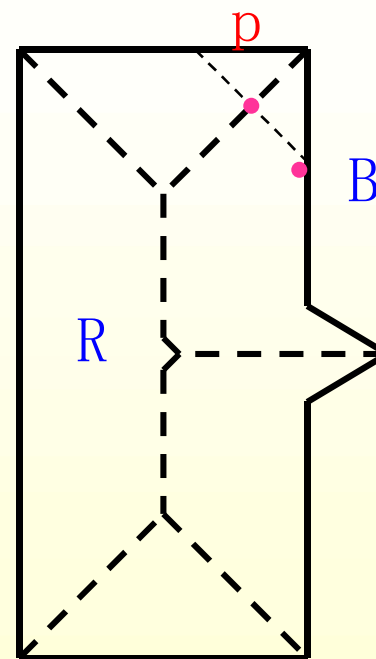


以每个骨架点为圆心，以前述最小距离为半径作圆周，
它们的包络就构成了区域的边界，填充圆周就得到区域



北京航空航天大学 — 表示法设计 — 区域骨架

- 区域骨架
 - 基本思想：需要计算所有边界点到所有区域内部点的距离
 - 问题：计算量大





- 区域骨架

- 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。实际中采用逐次消去边界点的迭代细化算法，它们不断删去边缘，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



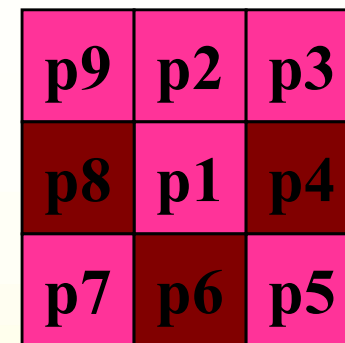
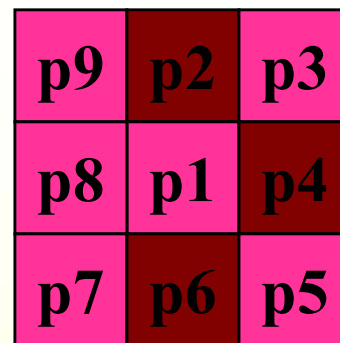
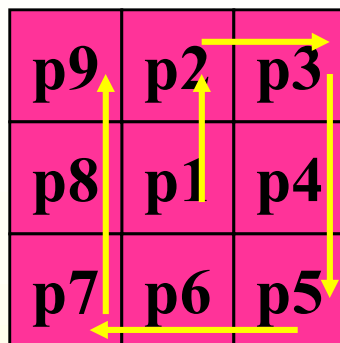
- 区域骨架

- 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



- 区域骨架
 - 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) $2 \leq N(p_1) \leq 6$ 其中 $N(p_1)$ 是点 p_1 的邻域中1的个数，

即：
$$N(p_1) = p_2 + p_3 + \dots + p_9$$

(b) $S(p_1) = 1$

其中 $S(p_1)$ 是按 p_2, p_3, \dots, p_9 顺序，0-1转换的个数

(c) $p_2 * p_4 * p_6 = 0$ （ p_2 、 p_4 、 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）



- 区域骨架

所有条件都满足，才打删除标记。删除并不立即进行，而是等到对所有边界点都打完标记后，再把作了标记的点一起删除。

— 举例：

$$N(p1) = 4$$

$$S(p1) = 3$$

$$p2 * p4 * p6 = 0$$

$$p4 * p6 * p8 = 0$$

第2个条件没满足不打标记

0	0	1
1	p1	0
1	0	1

p9	p2	p3
p8	p1	p4
p7	p6	p5

p9	p2	p3
p8	p1	p4
p7	p6	p5



- 区域骨架
 - 基本操作2

条件(a)、(b)与操作1相同

条件(c)、(d)改为：

$$c') \quad p_2^* p_4^* p_8 = 0$$

$$d') \quad p_2^* p_6^* p_8 = 0$$

p9	p2	p3
p8	p1	p4
p7	p6	p5

p9	p2	p3
p8	p1	p4
p7	p6	p5



- 区域骨架

- 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。



- 区域骨架

- 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 是线段端点，即骨架上的终点，显然不能删除；有7个点说明： p_1 过于深入区域内部，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。



- 区域骨架

- 算法分析：

p9	p2	p3
p8	p1	p4
p7	p6	p5

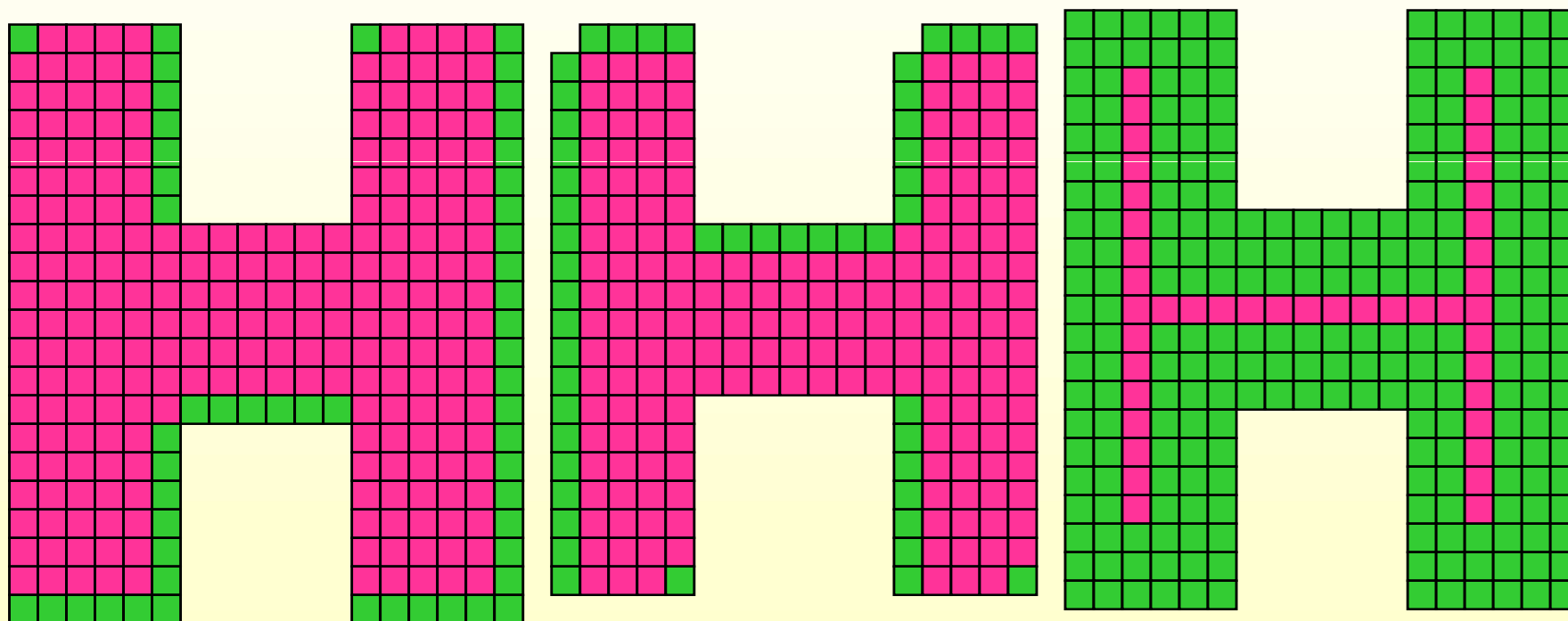
p9	p2	p3
p8	p1	p4
p7	p6	p5

(3) 当 $(p_4=0 \text{ or } p_6=0) \text{ or } (p_2=0 \text{ and } p_8=0)$ 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 $(p_4=0 \text{ and } p_6=0) \text{ or } (p_2=0 \text{ or } p_8=0)$ 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。



- 区域骨架
— 例:





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



- 简单描述子

- 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：边界上相隔最远的2点之间的距离。

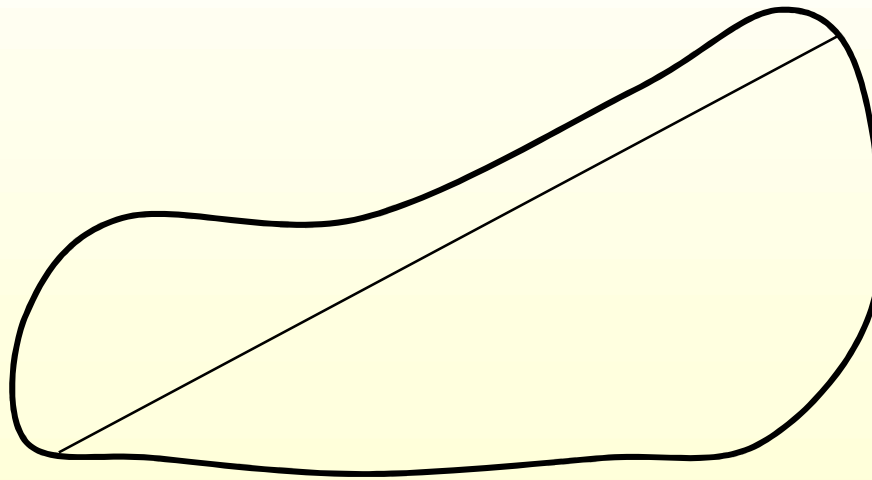
边界B的直径是：

$$\text{Diam}(B) = \max[D(p_i, p_j)]$$

D是欧氏距离或几何距离， p_i, p_j 是边界上的点。直径的长度和直径的两个端点连线（这条线被称为边界的主轴）的方向，是关于边界的有用的描述符。



- 简单描述子
 - 边界的直径举例

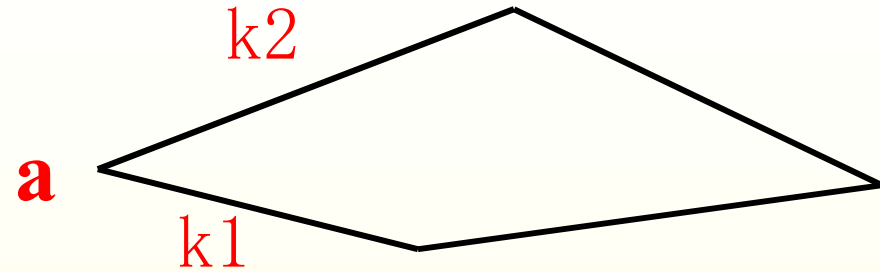




- 简单描述子

- 边界的曲率：

曲率被描述为斜率的变化率，它描述了边界上各点沿边界方向变化的情况。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子。



交点a处的曲率为 $dk = k1 - k2$

其中k1、k2 为相邻线段的斜率



- 简单描述子

- 边界的凸线段点：

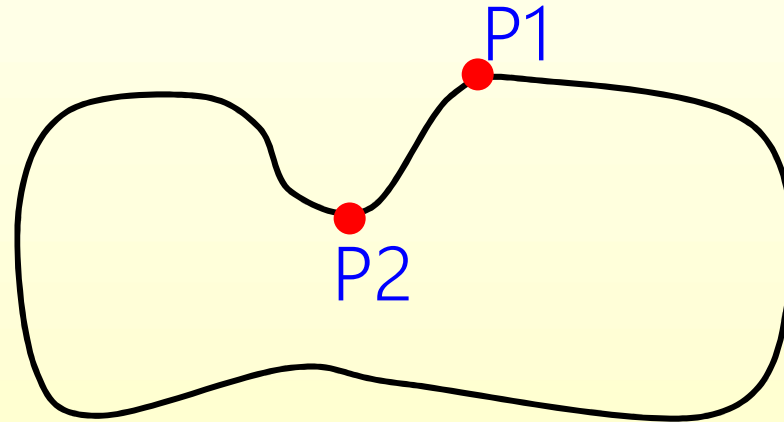
- 当顶点 p 上的斜率是非负时，称其为凸线段上的点

- 边界的凹线段点：

- 当顶点 p 上的斜率为负时，称其为凹线段上的点



- 简单描述子
 - 边界的凸线段点P1 :
 - 边界的凹线段点P2 :





- 形状数

形状数是基于链码的1种边界形状描述符

—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

循环首差： $1 - 2 = -1(3)$ $3 - 0 = 3$

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

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

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



北京航空航天大学3——边界描述子—形状数

- 形状数

- 形状数定义：

- 例如：4-链码 : 10103322

- 循环首差 : 33133|030

- 形状数 : 03033133

- 形状数是值最小的（链码的）差分码

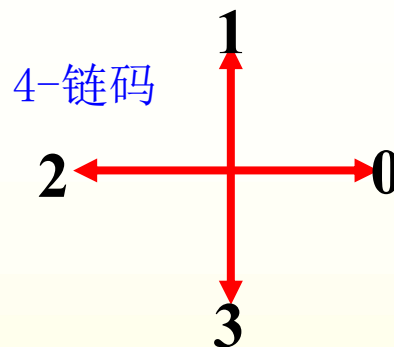
- 每个形状数都有1个对应的阶(order)，阶定义为形状数序列的长度（形状数中阿拉伯数字的个数），对于封闭边界序号是偶数。如order4、6、8。



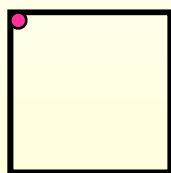
北京航空航天大学3—边界描述子—形状数

- 形状数

—形状数例如：

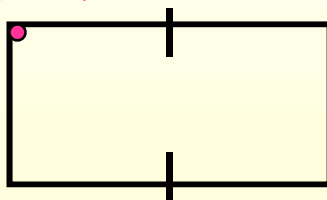


阶数4



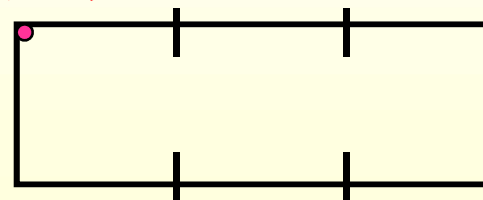
链码：0321
首差：3333
形状：3333

阶数6



链码：003221
首差：303303
形状：033033

阶数8



链码：00032221
首差：30033003
形状：00330033

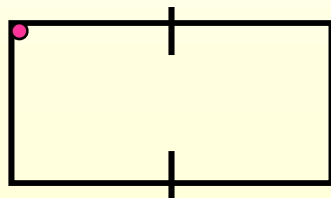


北京航空航天大学3—边界描述子—形状数

- 形状数

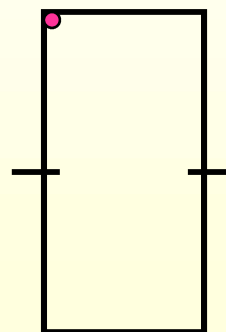
—形状数例如：

阶数6



链码：003221
首差：303303
形状：033033

阶数6



链码：033211
首差：330330
形状：033033

旋转不变性

00:22:58

计算机学院



北京航空航天大学3—边界描述子—形状数

- 形状数

- 问题：

虽然链码的最小首差不依赖于旋转，但一般情况下，边界的编码依赖于网格的方向。

- 改进：

规整化网格方向，具体方法如下：



- 形状数

- 几个基本概念：

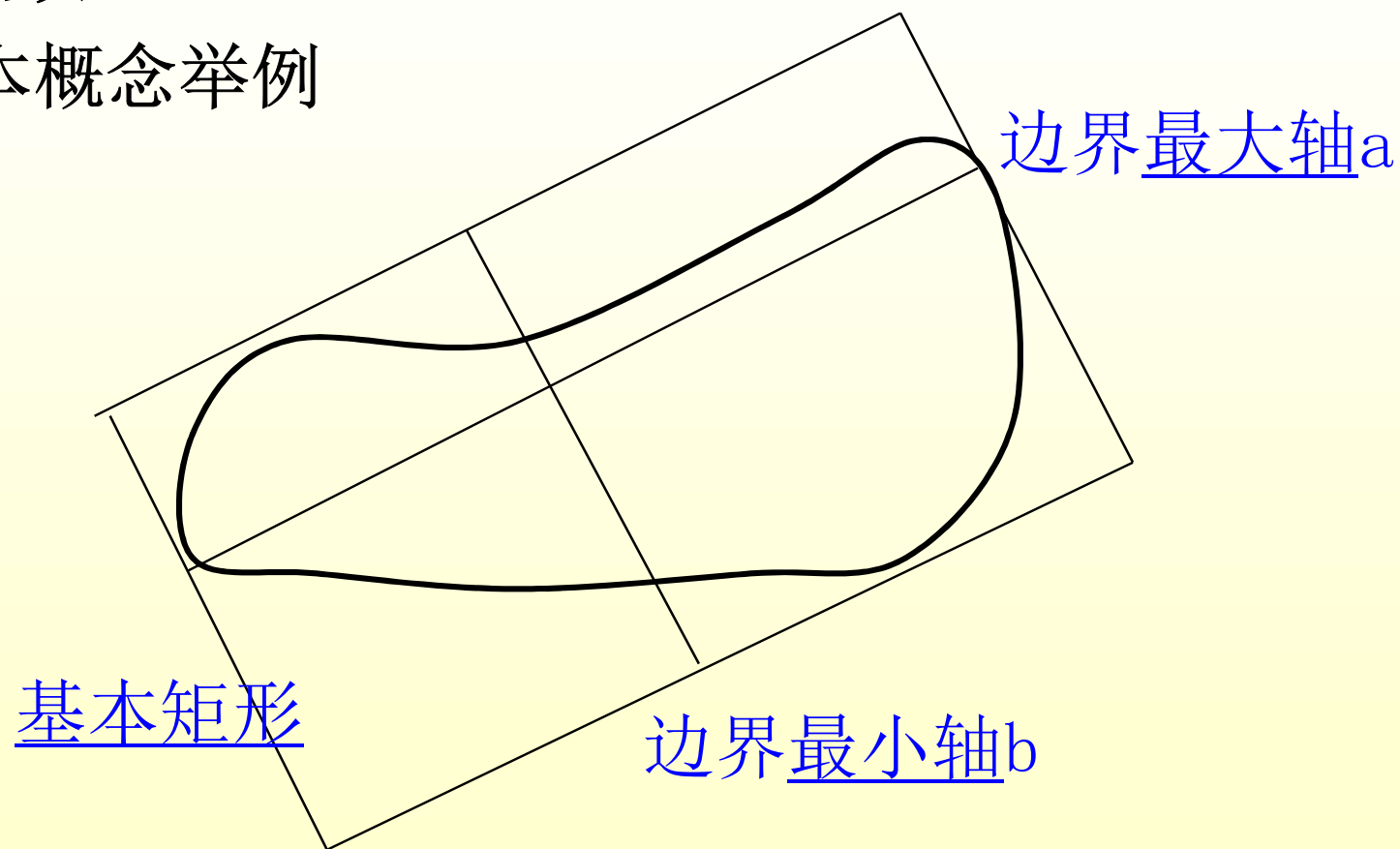
- 边界最大轴a:是连接距离最远的两个点的线段
 - 边界最小轴b:与最大轴垂直，且其长度确定的包围盒刚好包围边界。
 - 边界离心率c：最大轴长度与最小轴长度的比
- $$c = a / b$$
- 基本矩形: 包围边界的矩形。



北京航空航天大学3—边界描述子—形状数

- 形状数

- 基本概念举例





- 形状数

- 规整化网格方向算法的思想：

- 大多数情况下，将链码网格与基本矩形对齐，即可得到一个唯一的形状数。

- 对一个给定的形状序号，处理步骤如下：

- (1) 从所有满足给定阶数要求的矩形中选取一个矩形，它的离心率最接近于给定形状的基本矩形的离心率。



- 形状数

(2) 然后再用这个矩形构造网格。

例如: 如果 $n=12$, 所有序号为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, 我们建立一个 $2*4$ 的网格。

(3) 求出与边界最吻合的多边形, 例如将面积的50%以上包含在边界内的正方形划入内容。

(4) 再得到链码。

(5) 最后, 再得到循环首差。

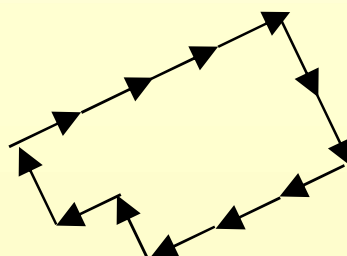
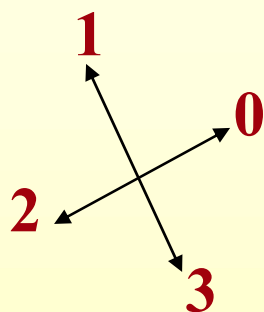
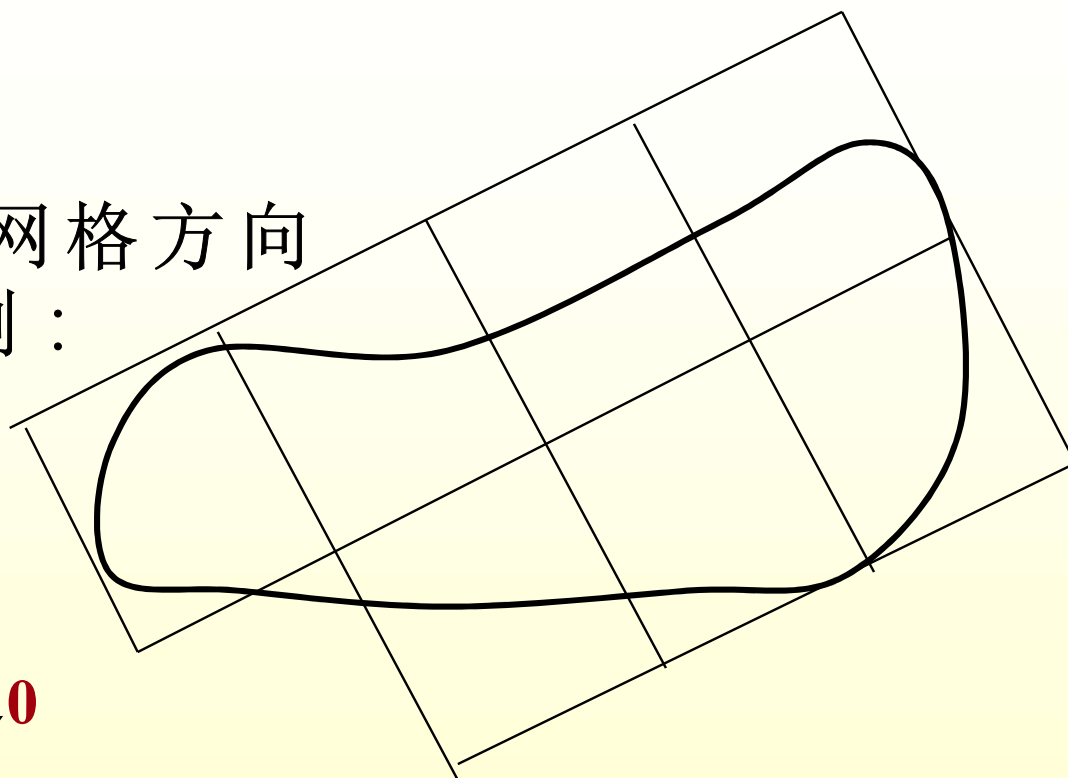
(6) 首差中的最小循环数即为形状数。



北京航空航天大学3—边界描述子—形状数

- 形状数

- 规整化网格方向
算法举例：



链码: 000033222121
首差: 300030300313
形状: 000303003133

00:22:58

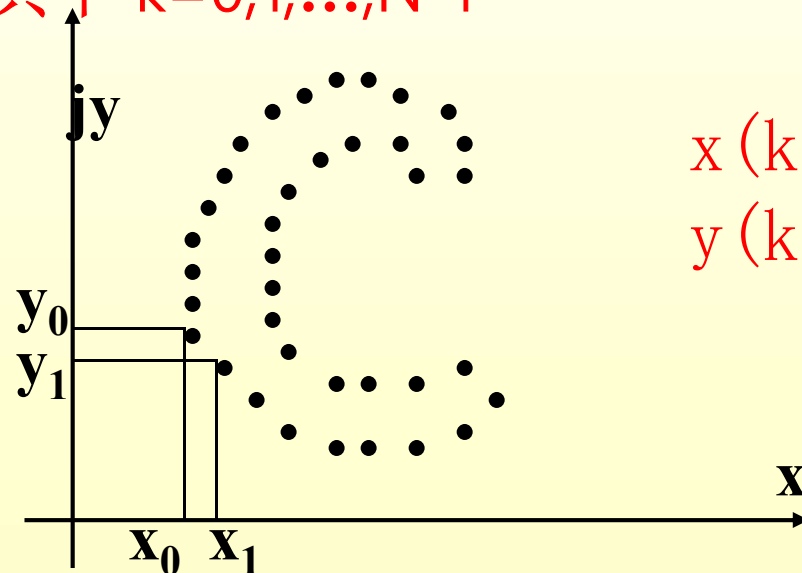


• 傅立叶描述子

—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： $s(k) = x(k) + jy(k)$

其中 $k=0,1,\dots,N-1$



$$x(k) = x_k$$

$$y(k) = y_k$$



- 傅立叶描述子

- 1) 基本思想：

- (2) 进行离散傅立叶变换

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

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

$a(u)$ 被称为边界的傅立叶描述子



- 傅立叶描述子

- 1) 基本思想：

- (3) 选取整数 $M \leq N-1$ ，进行逆傅立叶变换（重构）

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

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。

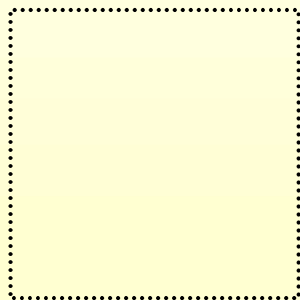


- 傅立叶描述符

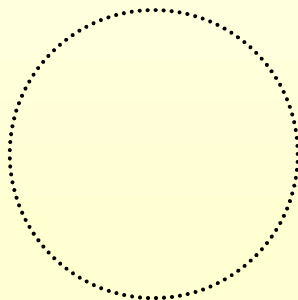
- 2) M的选取与描述符的关系

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。

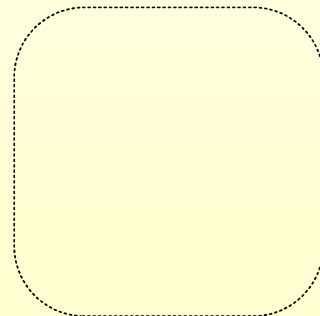
只用一些低频分量的傅立叶系数来近似描绘边界形状。



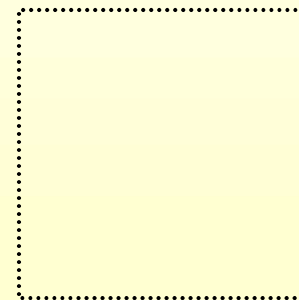
N=64



M=2



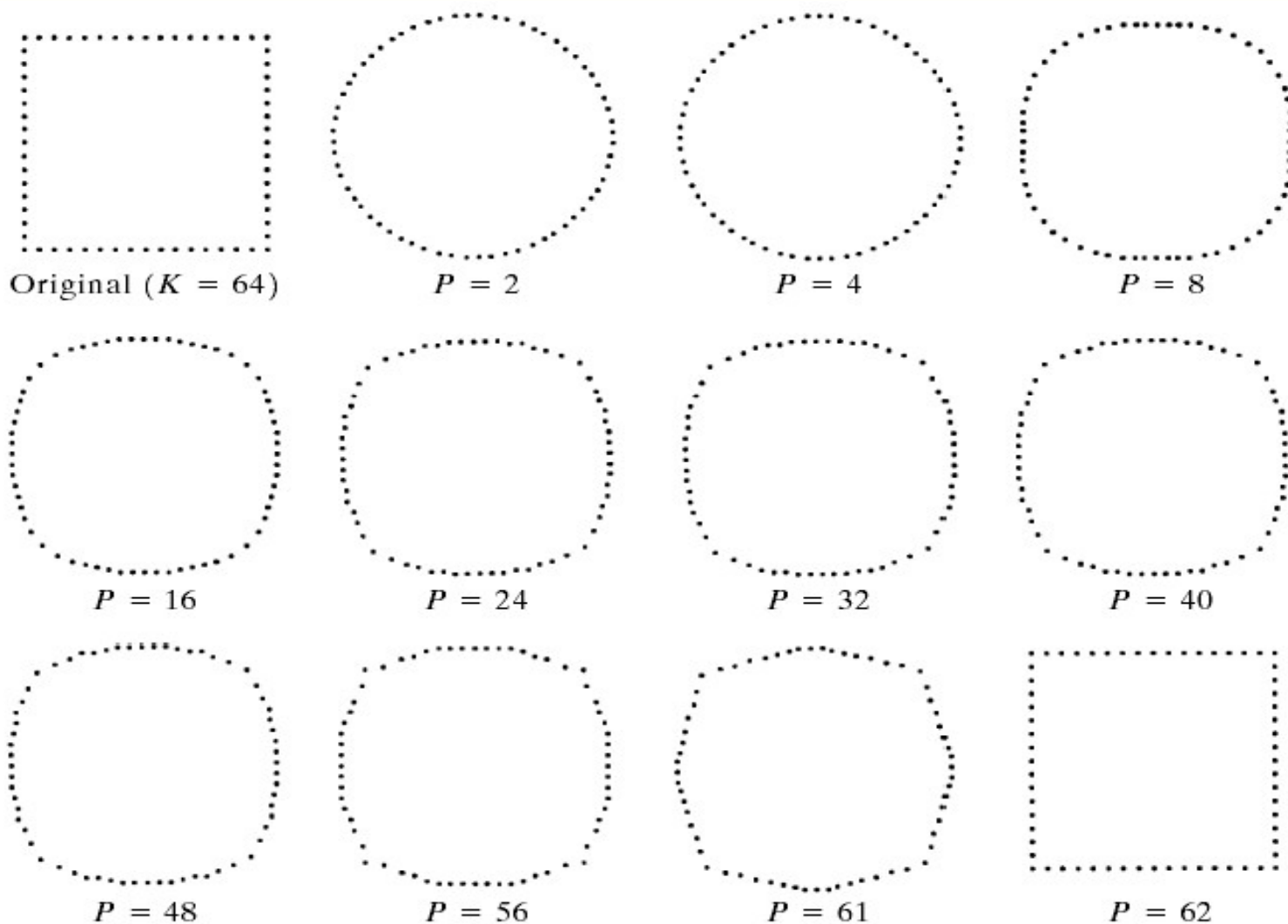
M=32



M=62



北京航空航天大学 — 边界描述子 — 傅立叶描述子



00:22:58

计算机学院



- 傅立叶描述符

- 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十分敏感。
 - (2) 以上几何变换均可以通过对描述子函数作简单变换来获得。



- 可通过矩来定量描述曲线而进一步描述整个边界。这种描述方法对边界的旋转不敏感。

f(r)的均值

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

f(r)对均值的n阶矩

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

μ_2 描述了曲线相对于均值的分布， μ_3 描述了曲线相对于均值的对称性。

平面上的矩类似



- 关系描述子
 - 基本思想
 - 阶梯关系编码
 - 骨架关系编码
 - 方向关系编码
 - 树结构关系编码

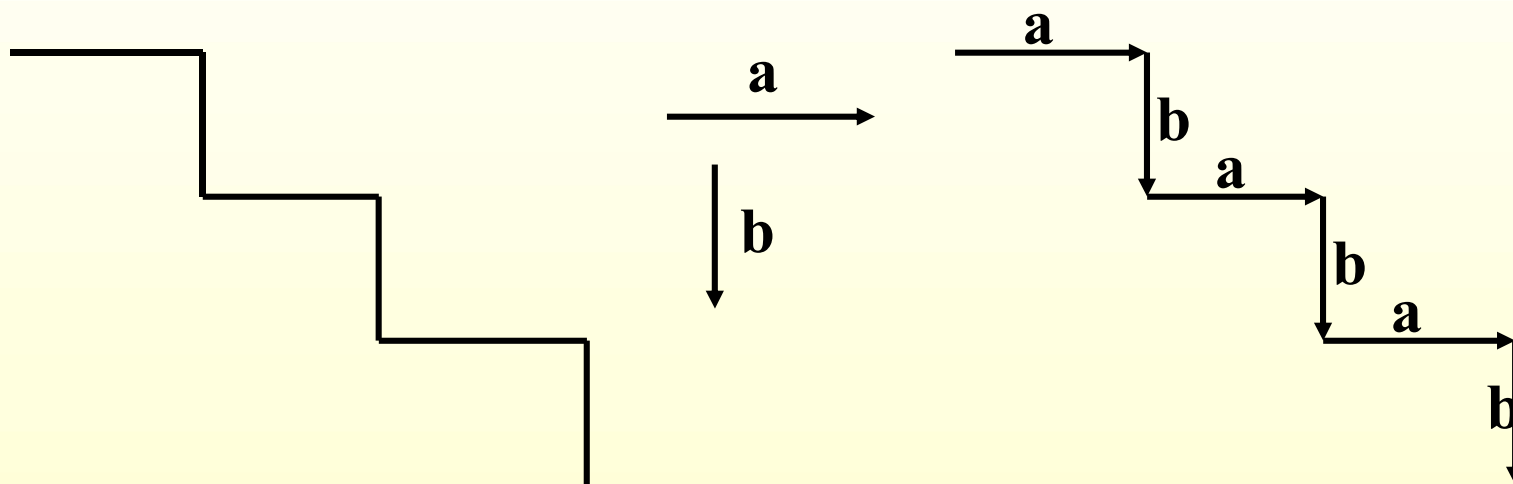


- 基本思想：
 - 通过挖掘各个成分之间的结构关系来描述边界
 - 图像中各个部分间的结构关系是二维的，而串是一维的，期望找到一种方法把二维关系转化为一维的串
 - 主导思想是考虑物体各个部分的连接线段



- 阶梯关系编码

— 对于如下阶梯形边界，定义两个基本元素a,b



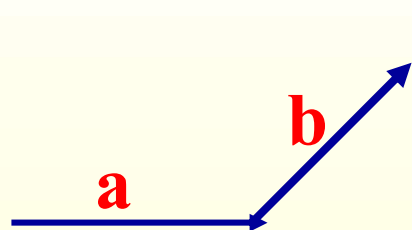


- 骨架关系编码

- 用有向线段来描述一个图像的各个部分（例如同构区域），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的方式描述的时候，最适于使用这种串来描述。

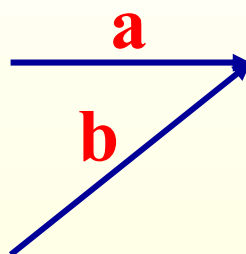


- 骨架关系编码

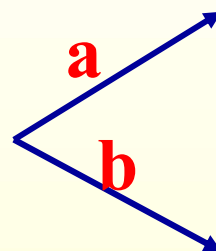


编码

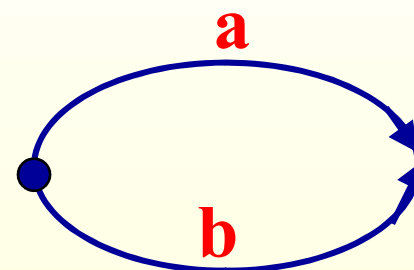
$$a + b$$



$$a - b$$



$$a \times b$$

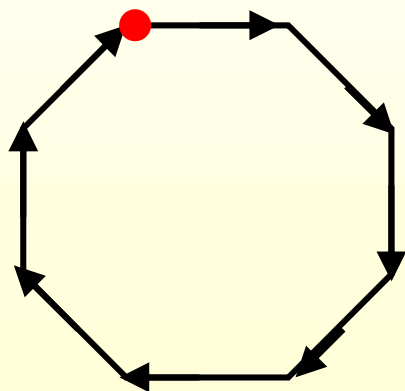


$$a * b$$

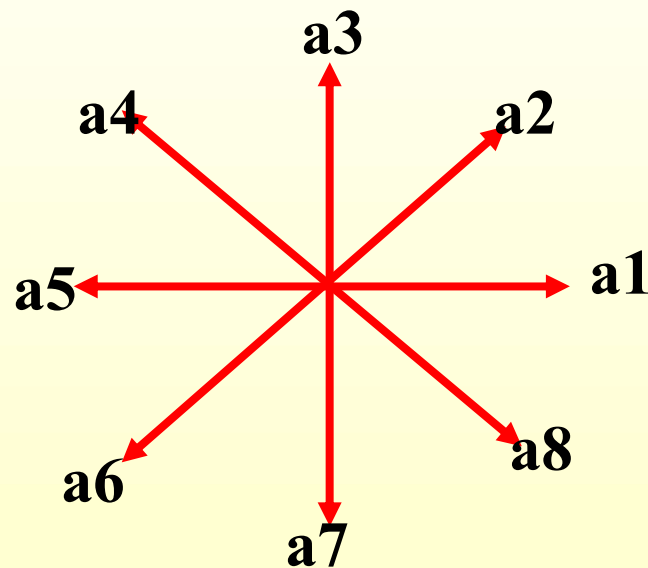


- 方向关系编码

—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



a1a8a7a6a5a4a3a2





- 树结构关系

— 树结构中每个节点的意义和节点之间的关系最为重要

举例：

