

# 第四章 距离变换

## 第四章 距离变换

- √4.1 距离
- √4.2 实体间的距离
- √4.3 求距方法与类型
- √4.4 地图代数的距离变换
- √4.5 形态变换
- √4.6 小结



## 4.1 距离

### √4.1.1 距离

### √4.1.2 距离类型

## 4.1.1 距离

### 一、距离

§ 尺度的定义是：假定 $D$ 是一个集， $J$ 是一个把 $D \times D$ 变进1维实数域的变换，如果对于所有的 $x \in D$ 、 $y \in D$ 、 $z \in D$ 满足如下条件：

Ø①  $J(x, y) \geq 0$  当且仅当 $x=y$ 时等号成立；

Ø②  $J(x, y) = J(y, x)$ ；

Ø③  $J(x, y) + J(y, z) \geq J(x, z)$ ；

§ 则称 $J$ 为 $D$ 的一个量度， $J(x, y)$ 是在这个尺度下 $x$ 与 $y$ 的距离， $D$ 是以 $J$ 为尺度的尺度空间（距离空间）。

## 4.1.1 距离

### ✓二、距离变换

### ✓三、点a到点集B的欧氏距离

点a到点集B的欧氏距离 $de(a, B)$ 为

$$de(a, B) = \min \{ de(a, b) \mid b \in B \}$$

其中， $de(a, b)$ 为a、b两点间欧氏距离，二、三维为

$$d_e(a, b) = \left[ \sum_{i=1}^n (a_i - b_i)^2 \right]^{1/2} \quad n = 2, 3$$

## 4.1.2 距离类型

### √ 一、曼哈顿距离

$$\text{Ø } d_M(1, 2) = |x_2 - x_1| + |y_2 - y_1|$$

### √ 二、棋盘距离

$$\text{Ø } d_C(1, 2) = \max\{|x_2 - x_1|, |y_2 - y_1|\}$$

### √ 三、时间距离

$$\text{Ø } d_T(1, 2) = |x_2 - x_1|$$

$$\text{Ø } d_T(1, 2) = |y_2 - y_1|$$

## 4.1.2 距离类型

### 四、栅格路径距离

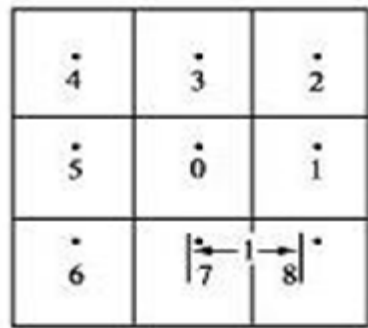


图4-1 栅格平面上相邻栅格关系

图4-1描述了一像元与其相邻像元的关系，中央栅格中心到周围栅格中心的距离

一种是边相邻栅格距离，如0~1、0~3、0~5、0~7均为1（假设栅格为正方形，栅格尺寸为1）；

一种为最相邻（或称点相邻）栅格距离，如0~2、0~4、0~6、0~8均为 $\sqrt{2}$ 。

当  $|y_2 - y_1| > |x_2 - x_1|$  时

$$d_R(1, 2) = |x_2 - x_1| + (|y_2 - y_1| - |x_2 - x_1|) \cdot \sqrt{2}$$

否则，当  $|y_2 - y_1| \leq |x_2 - x_1|$  时

$$d_R(1, 2) = |y_2 - y_1| + (|x_2 - x_1| - |y_2 - y_1|) \cdot \sqrt{2}$$

## 4.1.2 距离类型

### √ 五、欧几里得距离

$$d_E(1, 2) = [(x_2 - x_1)^2 + (y_2 - y_1)^2]^{1/2}$$

### √ 六、大地线距离

### √ 七、条件距离



## 4.2 实体间的距离

### √ 4.2.1 点到线状实体的距离

- ∅ 点  $a$  与线  $B$  的距离应定义为  $d(a, B) = d(a, b)$ ,  $b \in B$
- ∅ 且  $d(a, b) \leq d(a, b_i)$ ,  $b_i \in B$

### √ 4.2.2 点到面状实体的距离

- ∅  $d(a, B) = d(a, b)$ ,  $b \in B$
- ∅ 且  $d(a, b) \leq d(a, b_i)$ ,  $b_i \in B$

### √ 4.2.3 等距点的轨迹、相应距离的若干概念

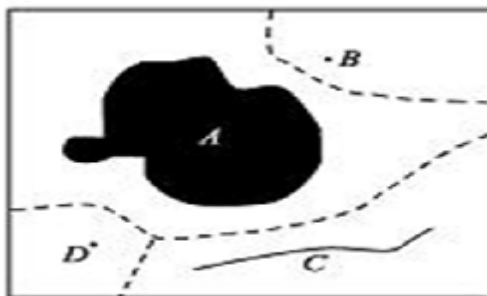


图4-2 等距点轨迹

## 4.3 求距方法与类型

### √ 4.3.1 求距方法

§ 矢量方法:

§ 栅格方法:

### √ 4.3.2 求距类型

§ ①**外距变换**: 求任一空间点或全部空间点到最近实体的距离问题, 这时, 实体内任一点(栅格)到实体的距离为0, 与实体边相邻的栅格距实体距离为1, 其次为1.414, 2, ...。

§ ②**内距变换**: 求实体上全部点到周围空间的最近距离, 也即到非实体空间距离。

§ ③**条件距离变换**: 这类变换要满足一定条件, 例如 $f(x, y) = c$ ,  $c$ 为常数, 其他点在距离变换中成为障碍。所谓障碍, 即这些点本身无须算出距离, 也不能传递距离, 距离增长或减少时只能绕过它。

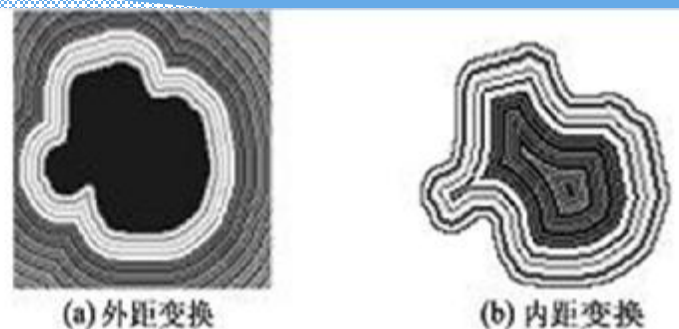


图4-3 欧氏内距、外距变换

## 4.4 地图代数的距离变换

- ✓ 4.4.1 栅格平面上的若干基础概念
- ✓ 4.4.2 传统栅格距离变换
- ✓ 4.4.3 栅格路径距离变换
- ✓ 4.4.4 栅格欧氏距离变换
- ✓ 4.4.5 出租车距离变换及时间距离变换
- ✓ 4.4.6 距离变换特性
- ✓ 4.4.7 加权距离变换

## 4.4.1 栅格平面上的若干基础概念

- ✓ **两定点 $a$ 、 $b$ 间在栅格下的距离**由其所属两栅格间的距离表示或定义。
- ✓ **两栅格间的距离**由其所属两栅格中心间的距离表示或定义。
- ✓ 将指定两定点 $a$ 、 $b$ 所属栅格中心间的、并由相通栅格中心间线段组成的有序线段集称为**栅格路径**。这里的相通，即是指该线段集与全部障碍栅格的区域无交点。称其相应累积长度为**栅格路径长度**。

# 距离变换常用距离表达

## ✓ 棋盘距离

棋盘距离指平面上A、B两点间取x, y两方向中的较大者作为距离, 其数学表达式为:  $d_{E(A, B)} = \max(|x_A - x_B|, |y_A - y_B|)$

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

棋盘距离

				2
		2	1	2
2	1	0	1	2
	2	1	2	
				2

曼哈顿距离

## ✓ 曼哈顿距离

曼哈顿距离也称出租车距离, 他特别适合于求取矩形城市街区中两点间出租车经过的路程, 它实际是指平面A、B两点间x, y两方向上的距离之和, 其数学表达式为:  $d_E(A, B) = |x_A - x_B| + |y_A - y_B|$

# 常用距离表达（续1）

## 欧氏距离

$$d_e(A, B) = [ (x_A - x_B)^2 + (y_A - y_B)^2 ]^{1/2}$$

## Minkowsky距离

$$d(A, B) = [ |x_A - x_B|^p + |y_A - y_B|^p ]^{1/p}$$

显然，前边提到的三种距离都是 Minkowsky 距离的特例。

$p=1$  时，表现为曼哈顿距离；

$p=2$  时，则为欧氏距离；

$p \rightarrow \infty$  时，表现为棋盘距离。

棋盘距离和曼哈顿距离都是对欧氏距离的近似表达。

# 常用距离表达（续2）

## 八边形距离

棋盘距离、曼哈顿距离的综合，比前两种能够更准确地描绘欧氏平面并且依然保持整数变换的特性。

$$\max(|x_A - x_B|, |y_A - y_B|) \quad (\text{棋盘距离})$$

$$\leq [(x_A - x_B)^2 + (y_A - y_B)^2]^{1/2} \quad (\text{欧氏距离})$$

$$\leq |x_A - x_B| + |y_A - y_B| \quad (\text{曼哈顿距离})$$



## Hausdorff 距离

Hausdorff 距离是广泛用于衡量两个集合之间差别的度量。

## 大地距离表达与其他

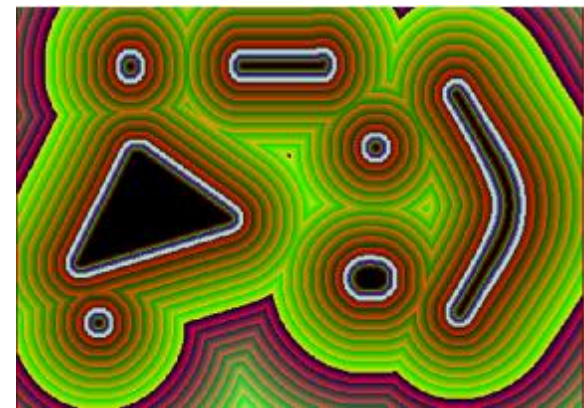
椭球面上最短程线，用二维欧氏坐标 地图投影变换 -> 椭球面经纬坐标模拟。同理，对其他距离表达处理方式类似，建立欧氏坐标与目标度量空间坐标的一一映射，则一对欧氏坐标可以确定一个目标度量空间下的距离。



# 栅格模型上的距离变换方法

距离变换的概念首先是由 **Rosenfeld** 和 **Pfalt** 在1966年引入的。

**距离变换**是计算并标识空间点集各点（对参照体）距离的变换（或过程）。





# 影响距离变换算法复杂度的因素

## ✓ 生成元个数

相应于矢量算法，矢量算法时间复杂度的唯一评估指标，一般用  $n$  表示，在矢量 Voronoi 算法中，算法时间复杂度与  $n$  严密相关；

## ✓ 栅格空间的分辨率

一般我们用  $m$  表示空间被划分成的栅格单元数量  $m = \text{图像宽度} * \text{图像高度}$ ，部分文献中也有使用  $n^2$  来衡量空间分辨率，有时候容易引起混淆；

## ✓ 生成元复杂程度

包括构成单个生成元的栅格点（黑点）个数，生成元形态的复杂程度（凹凸特性、边界平滑特性）等；

## ✓ 距离度量的计算复杂程度

使用整数变换、计算累积距离的棋盘距离、八边形距离等等复杂程度小于欧氏距离；更远小于使用迭代方法求取的大地距离度量。

## ✓ 特别的，对于二维欧氏度量下的穷举法求距离变换，其时间复杂度量

基数为  $O(nm)$ ，对于非点状生成元  $O(\text{黑点数} * m)$



# 距离变换算法分类

## ✓ 伪欧氏距离算法

Rosenfeld 1966年提出, 最具代表性 Borgefors 1984年提出的 Chamfer切削算法.

## ✓ 并行全局算法

Yamada 1984, Saito 1994

## ✓ 有序传播算法

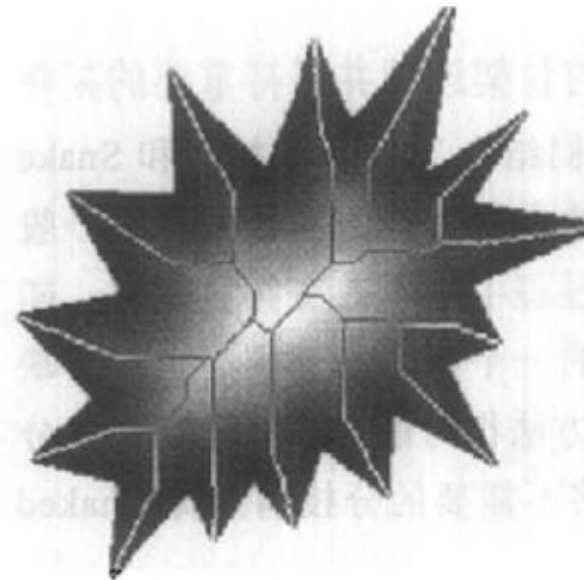
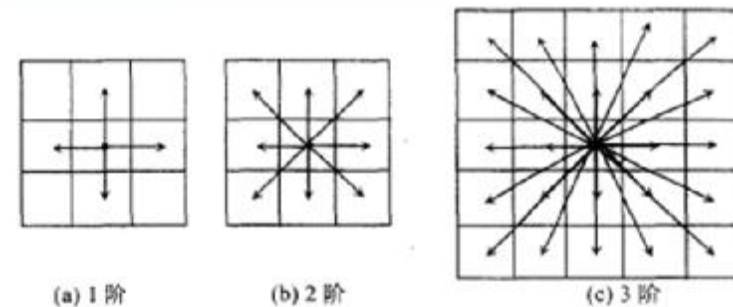
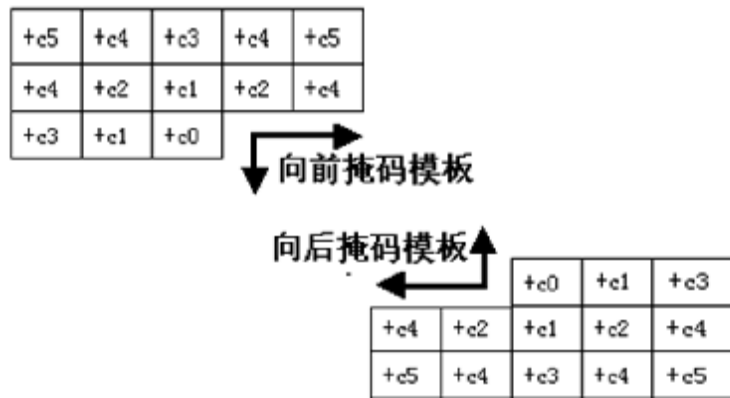
Piper 1987, Ragnemalm 1992, Cuisenaire 1999 2003

## ✓ 硬件算法

90年代中后期, Huff 1999

# Chamfer方法 Borgefors 1986

**伪欧氏距离:** [Montanari 1980]:任意两点间的距离定义为它们之间最短的折线链路径(chain-coded path)长度, 该路径每一步可由模板内的可能步进里选择。



(b) 初始骨架 (车武军 2003)

Chamfer距离的名称首先由Borgefors在1983年正式提出[Mullikin 1992], 用以表示计算距离时该类方法先后按相反两个方向的两遍扫描过程这一特点.

# 全局算法

Saito 算法的 3 维并行形式如下：

输入：图像  $F = \{f_{ijk}\}$ ,  $1 \leq i \leq L$ ,  $1 \leq j \leq M$ ,  $1 \leq k \leq N$ 。

输出：距离变换  $S = \{s_{ijk}\}^{\text{②}}$ ,  $1 \leq i \leq L$ ,  $1 \leq j \leq M$ ,  $1 \leq k \leq N$ 。

算法：

变换 1： 根据  $g_{ijk} = \min_x \{(i-x)^2 : f_{ijk} = 0, 1 \leq x \leq L\}$  由  $F$  得到图像  $G = \{g_{ijk}\}$  ( $i$  轴方向的变换)

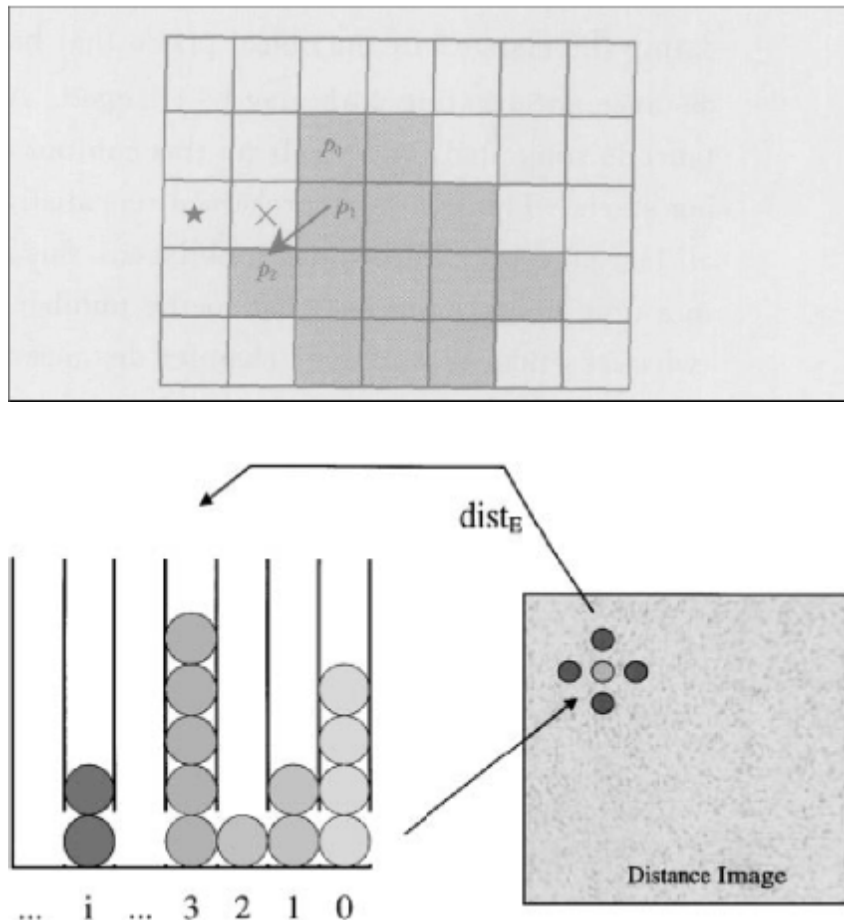
变换 2： 由上面得到的  $G$  根据下式计算图像  $H = \{h_{ijk}\}$  ( $j$  轴方向的变换)

$$h_{ijk} = \min_y \{g_{ijk} + (j-y)^2 : 1 \leq y \leq M\};$$

变换 3： 由上面得到的  $H$  根据下式计算图像  $S = \{s_{ijk}\}$  ( $k$  轴方向的变换)

$$s_{ijk} = \min_z \{h_{ijk} + (k-z)^2 : 1 \leq z \leq N\}。$$

# 有序传播方法 (Ordered Propagation)

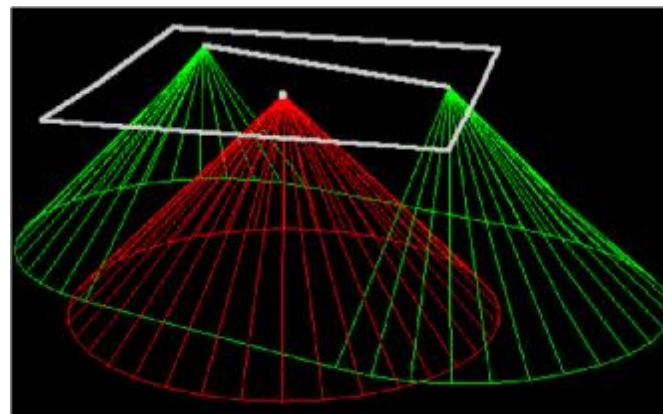
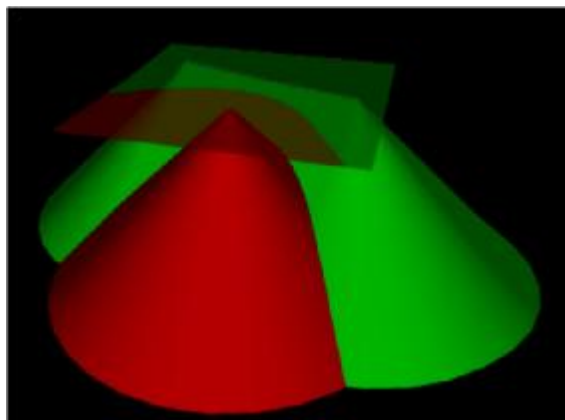
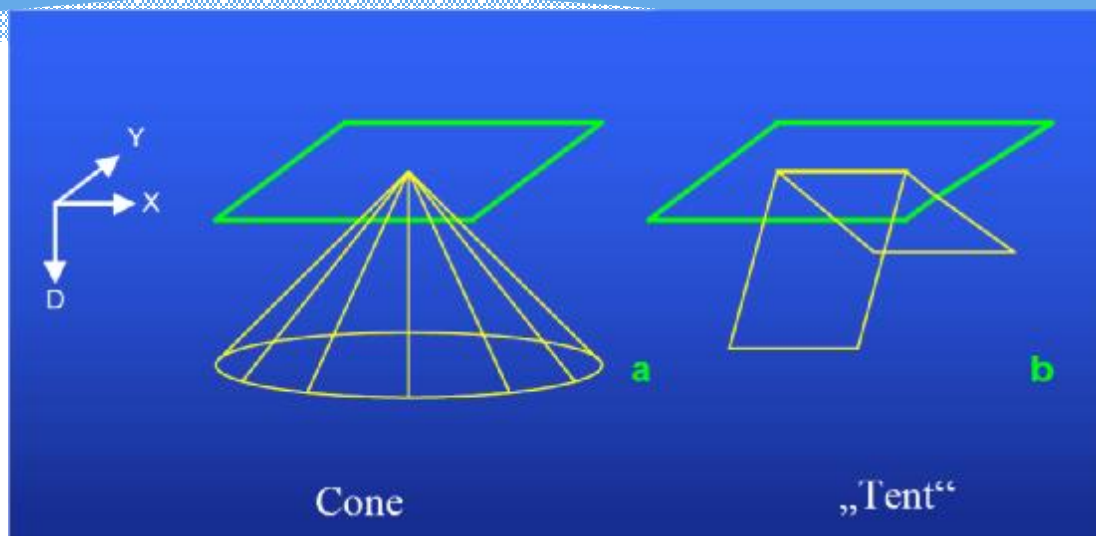


栅格扫描距离变换算法在计算过程中不加区别地处理所有像素点，**导致大量的运算浪费**在处理已经得到正确距离的像素点或由特征点发出的距离场扩散波前根本未曾到达的像素点，这样每个点都多次被处理。  
[Ragnemalm 1992]

有序传播算法，又称为轮廓处理(Contour processing)技术，在变换的任何阶段，只有发挥作用的传播波前的像素点才在当前阶段的处理范围内。

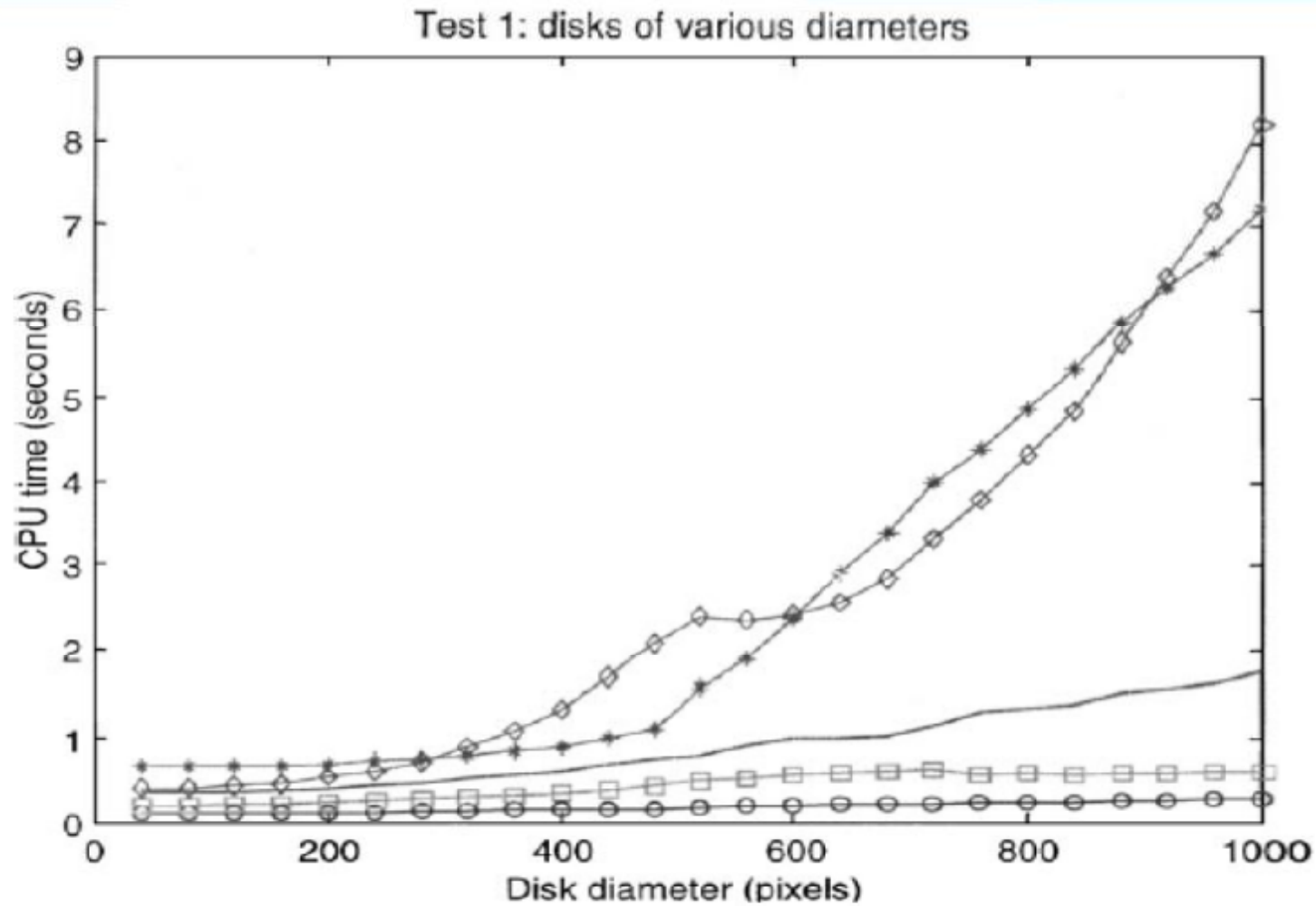


# 硬件加速算法 (Huff 1999)



[返回](#)

# 几类算法效率比较



Test 1: diameters vary from 40 to 1000 pixels: "○," Chamfer 3, 4; "□," PSN; "—," PMN; "◇," Eggers; "\*, " Saito.

## 4.4.2 传统栅格距离变换

### 一、步骤

§ ①把实体所在栅格赋距离值**0**，并把其余所有栅格赋一个足够大的距离值：

§ ②顺序访问各像元，以图左下角为原点，行号递增，列号递增。以所访问像元的左、下左、下中及下右的距离值推出所访问像元的距离值，并改写它。设像元坐标为列号*i*，行号*j*，则距离*d* (*i*, *j*) 由下式推出：

$$d_c(i, j) = \min(d_c(i-1, j-1)+1, d_c(i-1, j)+1, d_c(i-1, j+1)+1, d_c(i, j-1)+1, d_c(i, j))$$



## 4.4.2 传统栅格距离变换

- ③逆序访问各像元，以图左下为原点，右上角为起点，行号递减，列号递增（以下同），并改写距离值：

$$d_c(i, j) = \min(d_c(i+1, j+1)+1, d_c(i+1, j)+1, d_c(i+1, j-1)+1, d_c(i, j+1)+1, d_c(i, j))$$

- ④采用与距离相应的色值，把上述各像元距离值改写为位图。

## 4.4.2 传统栅格距离变换

### 二、算法分析

- § 步骤①，把待求距实体像元的距离值置为0，显然符合实物自身到自身距离为0的规律；
- § 步骤②，顺序递推距离使每个栅格空间到左、左下、下、右下四个栅格的距离正确，也使到左、左下、下、右下零栅格的距离正确，但其到右、上右、上、上左零栅格距离不正确；
- § 步骤③，逆序递推距离，使每个栅格空间到右、上右、上、上左栅格距离正确，也使到右、上右、上、上左的零栅格距离正确。与此同时，若步骤②得到的该栅格到左、下左、下、下右的距离大于这个距离时才更动，小于、等于这个距离均采用步骤②得到的距离，不更动。
- § 这样，步骤②、③使每个像元到四面八方的零栅格的距离均为最短的距离值，也即每个栅格均具有它到零栅格实体的距离。
- § 步骤④，采用适当的彩色位图，使这个距离具有更好的可视化。

## 4.4.2 传统栅格距离变换

### 三、误差分析

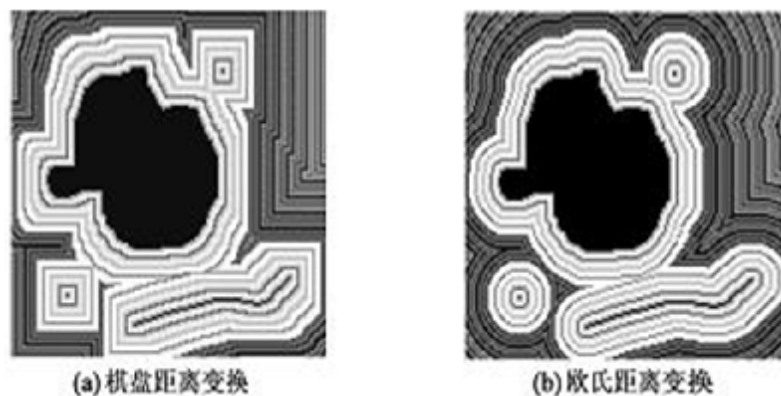


图4-4 棋盘距离、欧氏距离变换特点

考察图4-4，显然，对像元中心而言，从  $(0, 0)$  到  $(i, j)$  欧氏距离应为  $(i^2 + j^2)^{1/2}$ ，而棋盘距离为  $\max(|i|, |j|)$ ，这样，棋盘距离与欧氏距离误差  $ds$  为：

$$|i| > |j| \quad ds = (i^2 + j^2)^{1/2} - |i|$$

$$|i| \leq |j| \quad ds = (i^2 + j^2)^{1/2} - |j|$$

## 4.4.3 栅格路径距离变换

### 一、步骤

Ø 与棋盘距离变换步骤类似：

Ø ① 赋所有栅格像元为一个足够大的距离值；

Ø ② 顺序访问各栅格，并按下式改写它的距离值：

Ø  $dR(i, j) = \min(dR(i-1, j-1) + \sqrt{2}, dR(i-1, j) + 1, dR(i-1, j+1) + \sqrt{2}, dR(i, j-1) + 1, dR(i, j+1) + 1)$

Ø ③ 逆序访问各栅格，并按下式改写它的距离值：

Ø  $dR(i, j) = \min(dR(i+1, j+1) + \sqrt{2}, dR(i+1, j) + 1, dR(i+1, j-1) + \sqrt{2}, dR(i, j+1) + 1, dR(i, j-1) + 1)$

Ø ④ 把上述距离值四舍五入化为整数，并用彩色位图表达即可（一般用256色）。

### 二、算法分析

Ø 类同棋盘距离中的分析，只是对距离的理解不同，在此应理解为栅格路径距离。

## 4.4.3 栅格路径距离变换

### 三、误差分析

$$dS_R = i - j + \sqrt{2}j - (i^2 + j^2)^{1/2}$$

$$\text{令 } C = \frac{j}{i}$$

$$dS_R = [1 - C + \sqrt{2}C - (1 + C^2)^{1/2}]i = 0$$

显然， $dS_R$  决定于  $i$  及  $C$ 。其中  $dS_R$  是  $i$  的正比函数， $i$  越大， $dS_R$  越大。

### 四、方法改进

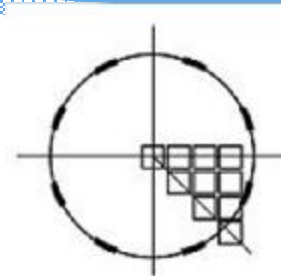


图4-5 栅格路径距离变换误差圆

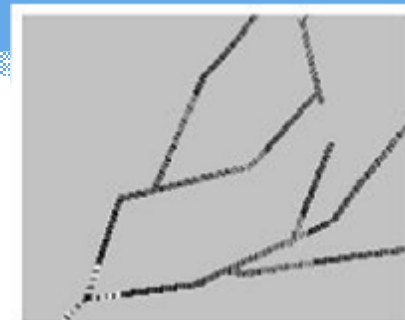


图4-6 栅格路径距离变换

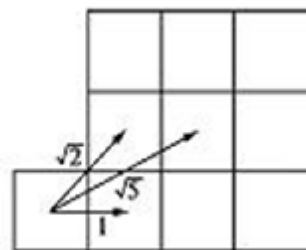


图4-7 栅格路径方向尺度1、2、3



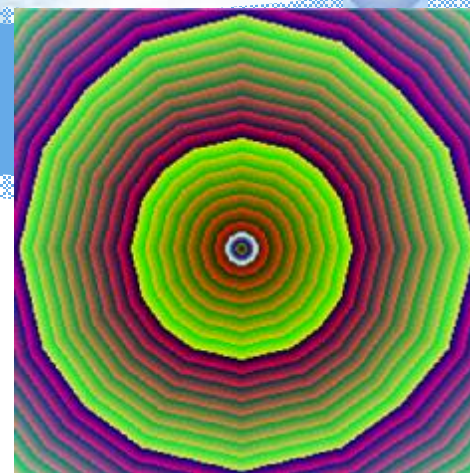
## 5×5模板距离变换效果



欧氏距离距离变换

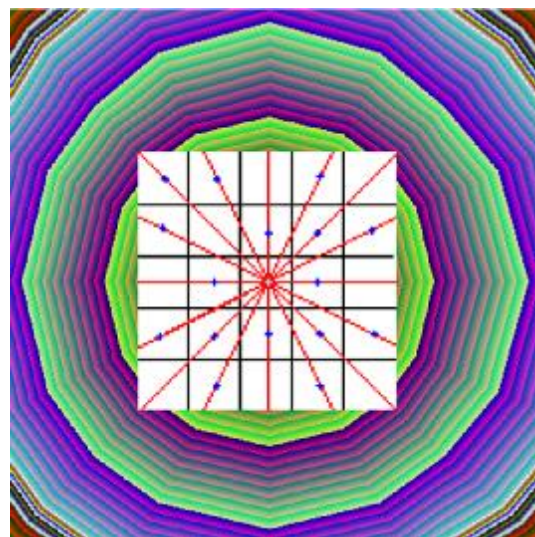


3×3模板的距离变换



5×5模板的距离变换

$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
2	1	P	1	2
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$



模板从3×3 扩大到5×5时，最大误差从0.089i降到了0.028i

结论：5×5邻域所有象元中心的斜率方向线段上均没有长度误差，其它斜率方向线段有误差，实际上其距离采用上述斜率方向上线段代替，其实质上是以 $1$ ， $2^{1/2}$ ， $5^{1/2}$ 三种尺度基底对整个空间进行度量

$(2n+1) \times (2n+1)$ 模板中各方向路径表达的最大误差

一线段用 $(2n+1) \times (2n+1)$ 邻近域内各方向栅格路径表达误差:

$$dS_R = [1 - nC + (1 + n^2)^{1/2}C - (1 + C^2)^{1/2}]i$$

(C为斜率)

当 $C = [((1 + n^2)^{1/2} - n)^2 / (1 - ((1 + n^2)^{1/2} - n)^2)]^{1/2}$  时,  
误差有极大值

$$2n+1=13 \text{ 时, } \text{MAX } dS_R = 0.00343i$$

$$2n+1=41 \text{ 时, } \text{MAX } dS_R = 0.000312i$$

## 4.4.4 栅格欧氏距离变换

### 一、栅格平方平面

§ 计算欧氏距离的困难之一在于根号，把根号去掉即可使用整数运算。如图4-10所示，栅格平面上一个栅格  $(i, j)$  到原点的欧氏距离为  $(i^2 + j^2)^{1/2}$ ，其坐标值在栅格平面上均为整数，其距离值与其平方  $i^2 + j^2$  值为一映射关系，除少数点以外，如  $3^2 + 4^2 = 5^2$ （勾、股、弦定理），对于  $45^\circ$  线下三角平面大部分是1-1对应的，显然可分别以  $45^\circ$  直线为对称轴镜像对称扩展为第一象限，同样也可以坐标轴为对称轴，扩展到其余三个象限，称之为平方平面栅格，这样就把栅格平面映射为平方平面。

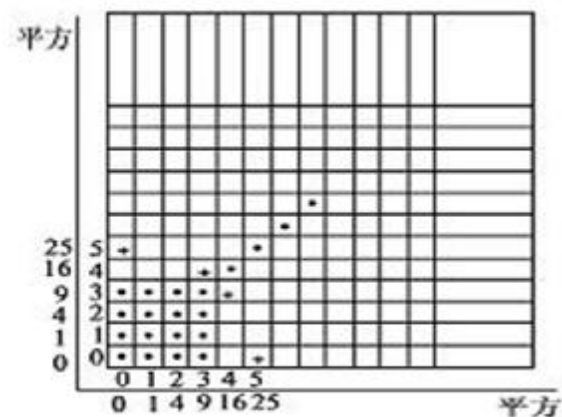


图4-10 栅格平方平面

### 二、栅格平方平面规律

§ ①左右关系:  $de2_1(i, j) = 2de2(i-1, j) - de2(i-2, j) + 2$

§ ②下上关系:  $de2_2(i, j) = 2de2(i, j-1) - de2(i, j-2) + 2$

§ ③左下与右上关系:  $de2_3(i, j) = de2(i-1, j-1) + 2(i-1)_{02} + 2(j-1)_{02} + 2$



## 4.4.4 栅格欧氏距离变换

### √三、变换方法

§ ①赋所有空间点为一足够大的正数**M1**，并赋所有实体点为**0**值。

§ ②顺序访问，即行号由**0, 1, 2, ...**递增，列号由**0, 1, 2, ...**递增，按下式改写各点平方值：

$$de2(i, j) = \min(de2_1(i, j), de2_2(i, j), de2_3(i, j), de2_4(i, j), de2(i, j))$$

§ ③逆序访问并改写各点平方值：

$$de2(i, j) = \min(de2_5(i, j), de2_6(i, j), de2_7(i, j), de2_8(i, j), de2(i, j))$$

§ ④改写各点距离平方值为距离值：

$$c(i, j) = INT\{[de2(i, j)]^{1/2} + 0.5\}$$

## 4.4.4 栅格欧氏距离变换

### ✓ 四、几点讨论

Ø1、误差分析

Ø2、实用环境与速度

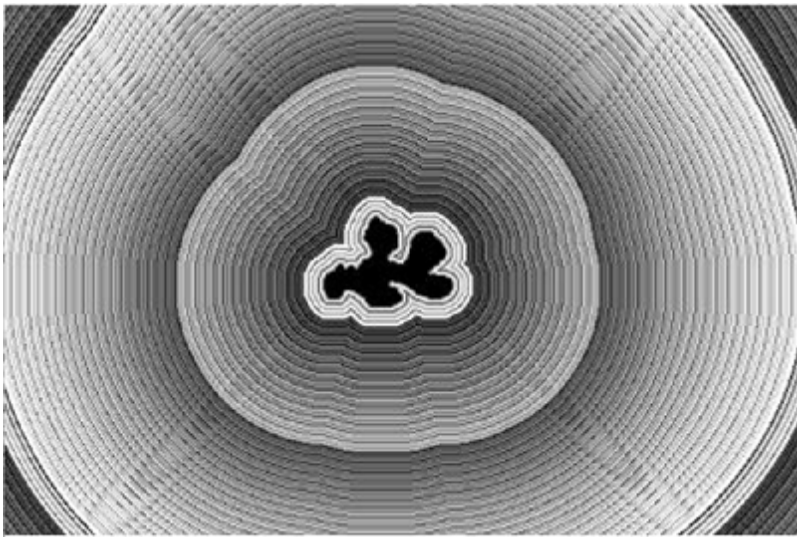


图4-11 一个面（实体）的欧氏距离变换图

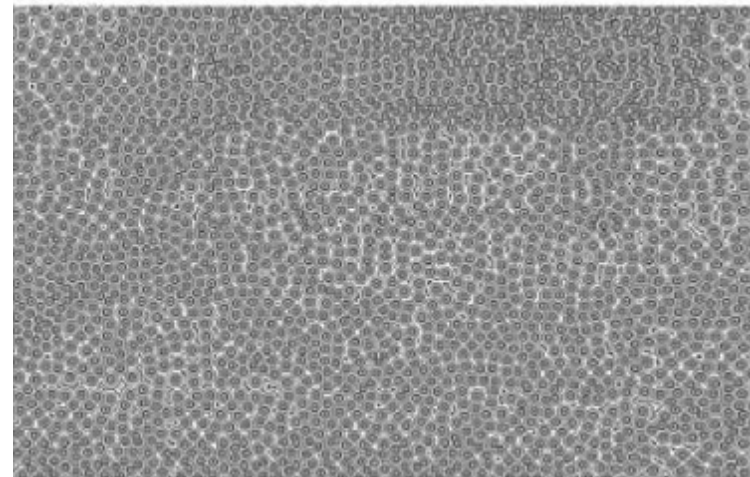


图4-12 众多点实体的欧氏变换图

## 4.4.4 栅格欧氏距离变

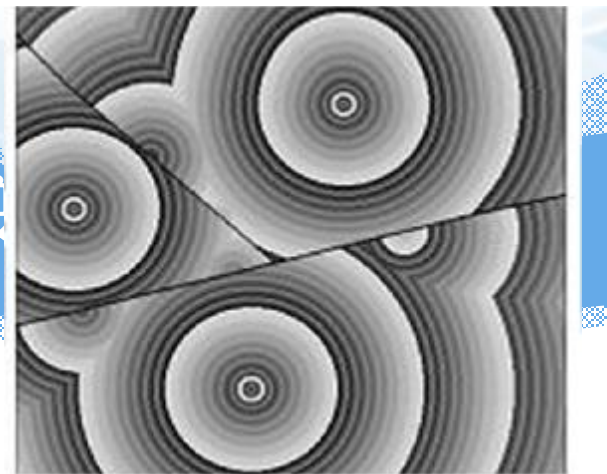


图4-13 障碍欧氏距离变换

### 五、障碍欧氏变换

- § 所谓障碍是本身不计算距离，而且不“传递”距离的点。
- § 对此，利用地图代数方法，只要在前述欧氏变换算法基础上把障碍物拓展进原始点集，即可生成障碍欧氏变换，概念简单、准确、严密，广泛适用于一般情况。具体操作如下：
- § （1）在原算法“①赋所有空间为一足够大的正数M1，并赋所有实体点为0值。”的基础上增加：并赋所有障碍体点为另一足够大整数M2，可设 $M2 \gg M1$ 。
- § （2）在原算法②、③、④步骤中对所有点值 $de2(i, j)$ 为M2的点均不改写。
- § （3）原算法③、④步骤中所有 $de2(i, j)$ 邻域中邻元点值为M2的点均不参与比较（实际上参与比较一般也无妨）。

## 4.4.5 出租车距离变换及时间距离变换

### 一、出租车距离变换方法

出租车距离尺度是与坐标系有关的，是矩形街区城市特定结构下产生的。这里设定坐标系坐标轴平行于街道，则本距离变换方法类同前述棋盘距离：

$$d_M(i, j) = \min(d(i-1, j) + 1, d(i, j-1) + 1, d(i, j))$$

$$d_M(i, j) = \min(d(i+1, j) + 1, d(i, j+1) + 1, d(i, j))$$

### 二、时间距离变换方法

假设坐标系纵轴与地球子午线投影平行，这时方法与前述棋盘距离变换一样：

$$d_T(i, j) = \min(d_T(i-1, j), d_T(i, j-1) + 1, d_T(i, j), d_T(i-1, j-1) + 1, d_T(i-1, j+1) + 1)$$

$$d_T(i, j) = \min(d_T(i+1, j), d_T(i, j+1) + 1, d_T(i, j), d_T(i+1, j+1) + 1, d_T(i+1, j-1) + 1)$$

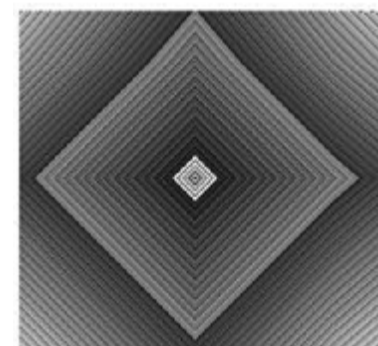


图4-14 单点出租车距离变换



图4-15 若干实体的出租车（外）距离变换

## 4.4.6 距离变换特性

### 一、距离变换应用同尺度定义空间量度

Ø 距离变换是在确定特性的尺度下对全空间的度量。外距变换度量实体图形外的空间，内距变换度量实体图形本身空间（这里空间与实体互为依存）。它是对空间量度性质的总体和全部的把握。

### 二、各空间点距离数值上的惟一性

- Ø ①各空间点到惟一实体点的距离平方是正确的。
- Ø ②各空间点到惟一实体点的距离正确且惟一。
- Ø ③当空间有 $n$ 个实体点时，任一空间点将在变换中得到它到最近实体点的距离（平方）；而当有2个、3个或若干个实体点到该空间点均具有这最小距离时，则有④。
- Ø ④该点即成为距此2点、3点或若干点的等距点。
- Ø ⑤任一空间点将标有离它最近的（点、线、面）实体的距离，它是惟一的，符合本章前述点到点、线、面实体的距离定义。
- Ø ⑥Voronoi多边形与欧氏距离路径



## 4.4.6 距离变换特性

### 三、欧氏距离变换的幂等算子

Ø 设欧氏距离变换算子用**De2**表示，空间**X**由实体点集和空的空间点集组成，则有

即**De2**为幂等算子。

$$X \bullet De2 \bullet De2 \cdots De2 = X \bullet (De2)^n = X \bullet De2$$

### 四、平移不变性

Ø 这是指原点集的点，由于它们是由实体及相互之间关系决定的，理论上与坐标系关系不大。平移是不变的。

### 五、局部更动性

Ø 当加入或减少、合并新的实体点集时，只与更动“周围”的实体与其空间有关，这个“周围”是：对更动点集的点位产生更动距离数值的实体，以及“面向”更动点集的“一方”的空间。

Ø 这是个十分重要的动态特性。

## 4.4.7 加权距离变换

- ✓ 实际地理世界中，各实体对空间实际影响能力、强度不同，相应的度量空间也发生变化，因而各实体或各类实体对周围空间的距离影响可简单地赋予相应权因子而成为加权距离变换，以解决该尺度空间更广义的距离问题。

## 4.5 形态变换

- √ 4.5.1 缓冲区变换
- √ 4.5.2 加壳变换和蜕皮变换
- √ 4.5.3 粘连变换和蚀断变换



## 4.5.1 缓冲区变换

### 一、变换方法

矢量方法:

栅格方法:

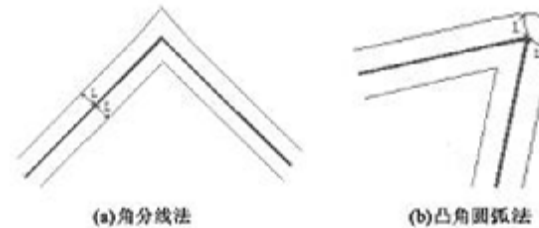


图4-16 矢量方法制作缓冲区

### 二、地图代数的缓冲区生成

图形X的L缓冲区可定义为距该图形距离不大于L的点集。对应地可定义内、外L缓冲区，定义L1~L2间的缓冲区为距离大于L1而不大于L2的点集，则取该内缓冲区的过程或变换可表示为 $B_i(L)$ 或 $B_i(L1, L2)$ ，取外缓冲区的过程或变换可表示为 $B_o(L)$ 或 $B_o(L1, L2)$ 。

算法如下：

- ①根据不同性质缓冲区，使用相应距离尺度直接进行距离变换（内、外），得到全空间各点距离。
- ②取距离值为1~L的所有像元进行缓冲区提取、成图。

## 4.5.1 缓冲区变换

✓ 外缓冲区变换过程可记为：

$$\S \quad XB_o(L1, L2) = X \cdot D_o(X) \cdot B(L1, L2)$$

这里， $B(L1, L2)$  表示取距离值为  $L1 \sim L2$  的像元。

✓ 由于点和单线实体本身宽度均为1，故内缓冲区算法主要针对面实体，内缓冲区变换过程记为：

$$\S \quad XB_i(L1, L2) = X \cdot D_i(X) \cdot B(L1, L2)$$

✓ 这时称**外缓冲区为壳**，**内缓冲区为皮**，实体加外缓冲区为“**加壳**”，去内缓冲区为“**蜕皮**”。

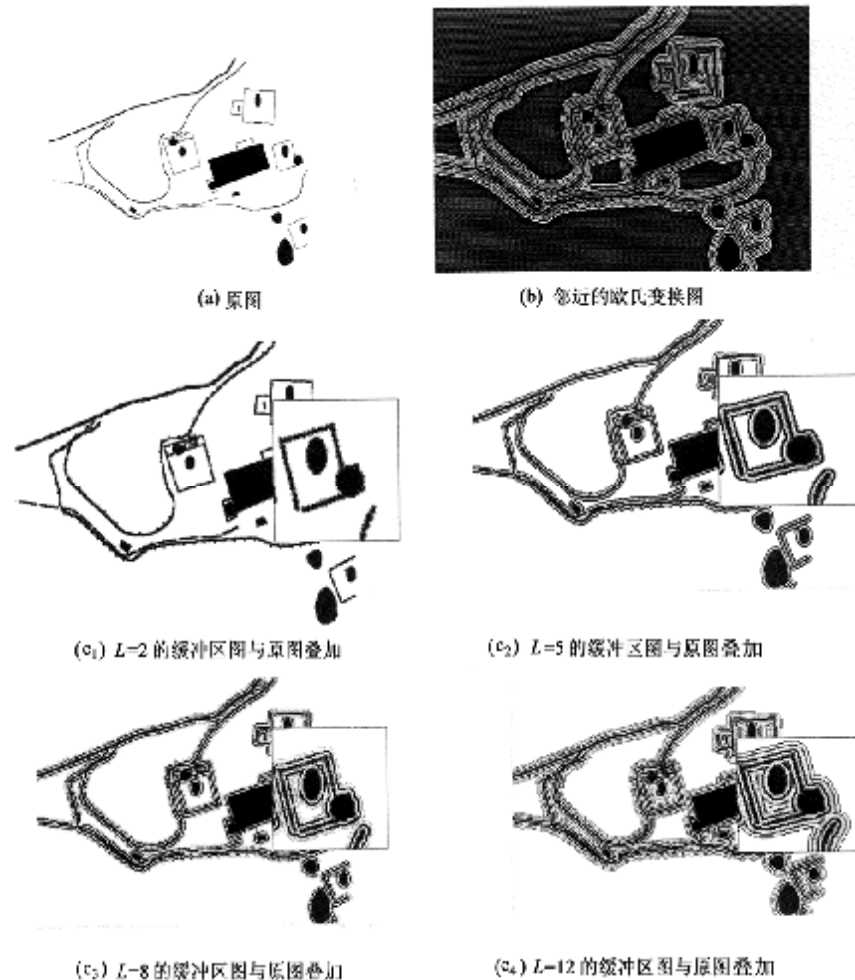


图4-17 经过欧氏变换后的缓冲区

## 4.5.2 加壳变换和蜕皮变换

### 一、定义

Ø 对图形 $X$ 的加壳变换 $K_0(L)$ 定义为:

$$Ø \quad XK_0(L) = X \cup D_0(X) \cdot B_0(L) = X + D_0(X) \cdot B_0(L)$$

对图形 $X$ 的蜕皮变换 $K_1(L)$ 定义为:

$$Ø \quad XK_1(L) = X \setminus D_1(X) \cdot B_1(L) = X - D_1(X) \cdot B_1(L)$$

### 二、形态特征

Ø 在加壳过程中, 所有外表原先为 $2L$ 宽以下的“缝隙”将不复存在, 而大的凹部仍存在, 且凸处将加宽, 高差不变, 整个实体包上 $L$ 宽的外壳, 变得厚、粗、大了。间隙小于 $2L$ 的实体将被合并。

Ø 在蜕皮过程中, 外表面宽度不大于 $2L$ 的凸部将蚀去。凸部大范围更光滑一点, 大的凸部仍存在, 而凹部发展, 小缝变大缝, 更深更宽。整个实体变得薄、细、瘦了, 一些颈部小于 $2L$ 宽的部分断开, 一分为二。

## 4.5.2 加壳变换和蜕皮变换

Ø 加壳变换有填平凹部或缝隙即减凹的作用，而凸部不变。  
缓冲区宽度 $L$ 与缝隙或凹部宽度 $r$ 关系如图4-18所示。

Ø 有  $L1-L= [r2+L2] 1/2 - L < 0.5$   
解之得  $L > r2 - 0.25$

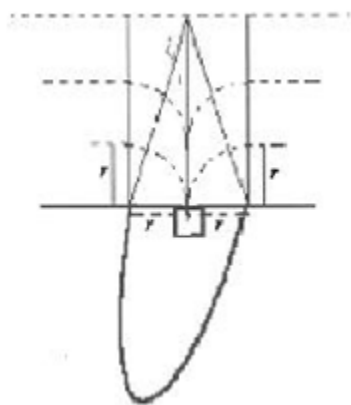


图4-18 凹部填平的厚度原理图



图4-19 点、线、面实体栅格的加壳变换和蜕皮变换

## 4.5.2 加壳变换和蜕皮变换

✓ 1、平移不变性

✓ 2、互换性、结合性及分解性

对加壳变换有：

✓ 分解性  $XK_O(L) = XK_O(L1) \cdot K_O(L2)$

✓ 互换性  $XK_O(L1) \cdot K_O(L2) = XK_O(L2) \cdot K_O(L1)$

✓ 组合性  $XK_O(L1) \cdot K_O(L2) = XK_O(L1+L2) = XK_O(L)$

对蜕皮变换有：

✓ 分解性  $XK_I(L) = XK_I(L1) \cdot K_I(L2)$

✓ 互换性  $XK_I(L1) \cdot K_I(L2) = XK_I(L2) \cdot K_I(L1)$

✓ 组合性  $XK_I(L1) \cdot K_I(L2) = XK_I(L1+L2) = XK_I(L)$

✓ 3、乘法与加法分解和结合律

✓  $(X1+X2) \cdot K_O(L) = XK_O(L) = X1 \cdot K_O(L) + X2 \cdot K_O(L)$

## 4.5.3 粘连变换和蚀断变换

### 一、定义

对图形 $X$ 的粘连变换 $L$  ( $L1, L2$ ) 定义为:

$$\textcircled{0} XL(L1, L2) = X \cdot K_0(L1) \cdot K_1(L2)$$

对图形 $X$ 的蚀断变换 $C$  ( $L1, L2$ ) 定义为:

$$\textcircled{0} XC(L1, L2) = X \cdot K_1(L1) \cdot K_0(L2)$$

### 二、形态特性

粘连变换的形态变化为:

① 总的大形态不变，各处光滑度增加，保凸，保平，减凹，这种变化趋势随 $L$ 的增大而加剧。

② 视粘连宽度 $L$ 的大小，连接 $2L$ 宽的沟隙；那些离 $X$ 距离不大于 $L$  ( $X$ 的影响大)，而又离“外”（到 $X$ 距离大于 $L$ 的点集）的距离大于 $L$ 的点（非 $X+L$ 的影响小）被保留下来。

③ 填平 $2(L) 1/2$ 以下宽的缝隙。



## 4.5.3 粘连变换和蚀断变换

蚀断变换的形态变化为：

- ①大的形态不变，保凹，保平，减凸，光滑度增加，随 $L$ 的增加而加剧这种趋势。
- ②视蚀断宽度 $L$ 的大小，蚀断 $2L$ 宽的桥，从而分离实体内离外界距离不大于 $L$ ，而又离“内”（到非 $X$ 的距离又大于 $L$ 的 $X$ 内的点集）的距离大于 $L$ 的点被蚀去（图4-20（b））。
- ③蚀掉宽 $2L$ 的实体部分，蚀平 $2(L)1/2$ 以下宽的凸部。
- ④以上形态特点对形状综合概括特别有意义。

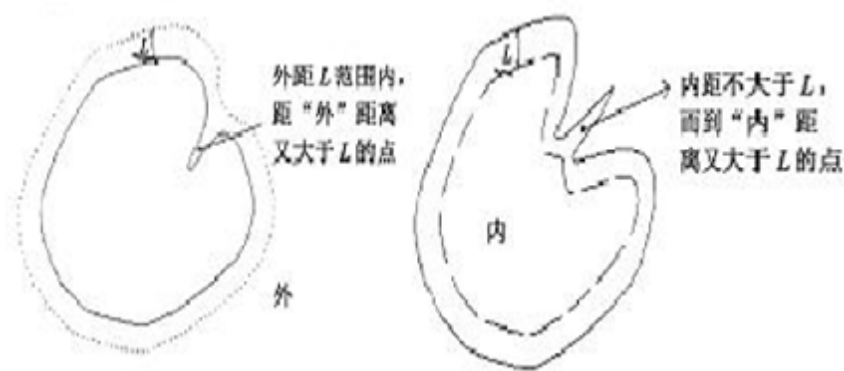


图4-20 粘连变换和蚀断变换形态变形原理



图4-21 粘连变换生成凸壳



## 4.5.3 粘连变换和蚀断变换

### 三、性质

Ø1、粘连变换具有增长性、蚀断变换具有减少性

当 $L1=L2$ 时，对粘连变换有：

Ø $X \hat{=} XL (L1, L2)$

对蚀断变换有：

Ø $X \hat{=} XC (L1, L2)$

Ø2、递增性

Ø3、幂等性

## 4.5.3 粘连变换和蚀断变换

### 四、最小凸壳算法

粘连变换 $L(L, L)$ 可以在一定的宽度 $L$ 内补凹为凸。具体算法如下：

- ①把图形 $X$ 数字化，并把 $X$ 各像元“染黑”；
- ②量测或估算图形 $X$ 欲填的最大凹部宽 $r$ ，计算加壳的缓冲区宽度 $L$ ；
- ③对图形 $X$ 实施粘连变换 $L(L, L)$ ：

首先对 $X$ 实施 $K_0(L)$ 变换，即为

- 1) 对 $X$ 进行外距变换，得到全空间各点到 $X$ 距离；
- 2) 取距离值为 $0 \sim L$ 的像元并把它变“黑”；

然后对 $X K_0(L)$ 实施 $K_1(L)$ 变换，即为

- 1) 对 $X K_0(L)$ 进行内距变换，得到 $X K_0(L)$ 中各点到非 $X K_0(L)$ 的距离；
- 2) 取距离值为 $0 \sim L$ 的像元并把它变“白”，距离值大于 $L$ 的像元变黑。

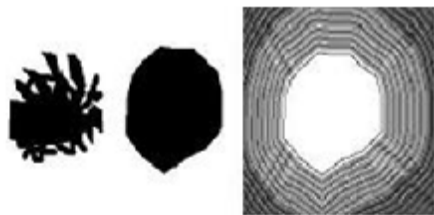


图4-21 粘连变换生成凸壳

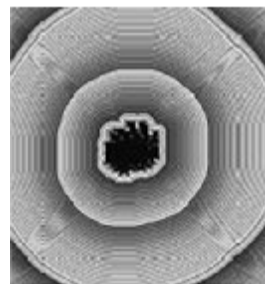


图4-22 外距变换（欧式）的等距离条纹

## 4.6 小结

- ✓ 一切空间分析首先是一种定性、定量、定位的分析，而量的刻画标准便是尺度。距离变换是尺度空间量度的基础，也是地理空间分析的基础，是起点。本章较详尽地分析了二维空间的各种尺度及量度方法，各种方法的实现没有显著的困难和障碍，它们均具有各自的特性和适用范围。

# 作业

- ✓ 4-1、距离是什么？距离变换的本义是什么？有什么意义？
- ✓ 4-2、矢量方法是如何求距的？如何对空间实施度量？
- ✓ 4-3、实体间有哪些类型的距离？试比较它们的拓扑性和距离大小性质的关系。
- ✓ 4-4、地图代数的距离变换有哪些类型，它们和实数下的运算差别在哪里？
- ✓ 4-5、什么是栅格路径距离？它的误差应如何估计？
- ✓ 4-6、距离变换有哪些特性？对图形本身和图形外空间进行距离变换的物理意义是什么？
- ✓ 4-7、形态是什么？
- ✓ 4-8、形态变换有哪些类型？其形态特征变化是如何控制的？对形状综合有何启示？
- ✓ 4-9、求图形的凸壳算法是怎样的？
- ✓ 4-10、试设计一个自然图形在八边形距离下的距离变换程序（实验，采用BMP位图作输入输出通用界面）。



# Thank You !

[www.themegallery.com](http://www.themegallery.com)