

计算机基础教育系列教材

邓顺华 毛先成 著
刘石年 叶柏龙 柳建新



微型机

实用图形程序设计

WEIXINGJI SHIYONG TUXING CHENGXU SHEJI



中南工业大学出版社

微型机实用图形程序设计

邓顺华 毛先成 著
刘石年 叶柏龙 柳建新

中南工业大学出版社

内 容 简 介

本书以图形和 CAD 的系统化程序设计为主线,对图表及 CAD 系统程序设计的各个方面进行了全面透彻的论述,主要包括计算机图形算法、显示系统程序设计、嵌入式汉字系统程序设计、扩展内存驱动程序设计、菜单程序设计、鼠标与键盘驱动程序设计、域编辑器程序设计、CAD 软件系统设计等内容。该书是作者科研成果(MCAD 软件系统)的总结和提炼,并给出了完整的源代码,极具实用价值。本书可供专业化程序设计人员借鉴与参考,也可作为 C/C++ 初学者进行系统化程序设计的训练教材及本科生和研究生图形程序设计方面的参考教材。

微型机实用图形程序设计

邓顺华 毛先成 刘石年

叶柏尧、柳雪新 著

责任编辑:文 刃

中南工业大学出版社出版发行

核工业中南230研究所印刷厂印装

新华书店总店北京发行所经销

开本:787×1092 1/16 印张:17 字数:418千字

1997年1月第1版 1997年3月第1次印刷

印数:0001-3000

*

ISBN 7-81020-976-0/TP·076

定价:20.00元

本书如有印装质量问题,请直接与生产厂家调换

厂址:湖南长沙市韶山路43号 邮编:410011

前 言

CAD 图形程序设计,尤其是采用 C/C++ 语言进行 CAD 图形系统程序设计,一向是技术难度较高、且具有一定挑战性的程序设计领域。这是因为,CAD 图形程序设计涉及到计算机图形学、图形数据结构、图形数据的存取与组织、图形运算算法、图形输入输出设备的驱动以及集成化的用户交互环境构造等诸多方面的内容。进行大型集成交互 CAD 系统的程序设计最能锻炼和提高程序员的程序设计水平。作者多年来一直从事 CAD 方面的科研和软件开发工作,经过几年的科研实践与探索,逐渐写成和完善了称为“MCAD”的图形处理系统,并已成功地在多个生产单位投入运行。来自用户的评价是:MCAD 系统最大的特点是具有极高的系统稳定性和很强的容错能力、很强的图形输入输出设备适应性(由于使用基于自定义的逻辑输入输出设备驱动,MCAD 系统几乎可以驱动已有各种数字化仪和绘图仪,因而有些用户甚至使用 MCAD 替代 AutoCAD 软件来作数字化)、良好的开放特性和灵活的操作方式,灵巧易用。确实,MCAD 在某些方面早已超过了 AutoCAD 软件的性能,这已是使用过 MCAD 软件用户的共识。作者现将 MCAD 软件系统中的主框架和精华部分提炼出来,并给出完整的源代码,以飨读者,希望起到抛砖引玉的作用。

本书内容翔实,逻辑性强,层次结构清晰,程序完整。主要以如何进行 CAD 的系统化程序设计为主线,对 CAD 系统程序设计的各个方面进行了全面的论述。书中所涉及的内容都是程序设计人员非常感兴趣的话题。如:显示系统程序设计、嵌入式汉字系统程序设计、扩展内存驱动程序设计、菜单程序设计、鼠标与键盘驱动程序设计、域编辑器程序设计等,对于程序员进行各种类型的程序设计都有很好的参考价值,并能在实际编程中直接使用其部分或全部。对于从事图形程序设计的程序员来说,其中的面向对象图素构造、图形存取管理结构、图形运算算法、交互式界面组织、以及数字化仪、绘图仪、打印机等图形输入输出设备的自定义驱动与基于设备逻辑化的图形输出设备的集中统一管理,又是图形程序设计方面的技术性很强的专题。因此,本书既可作为 C/C++ 程序设计的初学者来进行系统化的大型程序设计训练,也可供专业程序设计人员借鉴,是一本难得的大型图形程序设计技术参考书,可作为高等院校本科生和研究生图形程序设计方面的教材。

全书共分 11 章,分别介绍了计算机图形基本算法、显示系统、动画、嵌入式汉字系统、扩展内存驱动、数字化仪驱动、菜单、域编辑器、图形数据结构、交互 CAD 系统构造等方面的程序设计方法和技巧。每章都有与之对应的 C/C++ 源程序。最后给出了由书中全部源程序组成的 MCAD 系统的简要介绍,以利读者阅读和分析全部源程序。书中全部源程序均在 Borland C/C++ 3.1 环境下编译和调试通过。

本书由中南工业大学邓顺华副教授、毛先成副教授、刘石年教授、叶柏龙教授和柳建新教授共同完成。

由于作者水平有限,书中错误与不妥之处,恳请读者与专家指正。

作者

1996 年 1 月

目 录

1 计算机图形基本算法	(1)
1.1 图形的生成	(1)
1.2 图形的矩阵变换	(2)
1.3 图形的计算	(6)
1.4 图形的裁剪	(13)
2 显示系统程序设计	(20)
2.1 VGA 显示系统的基本原理	(20)
2.2 显示存储器	(22)
2.3 VGA 卡上的控制器	(24)
2.4 EGA/VGA 寄存器组编址及其访问	(26)
2.5 几个常用寄存器的编程	(27)
3 动画程序设计	(33)
3.1 内存映象方式(源程序 3.4.1)	(33)
3.2 图形页面技术(源程序 3.4.2)	(33)
3.3 异或(XOR)方式(源程序 3.4.3)	(34)
3.4 动画设计源程序	(34)
4 汉字系统程序设计	(38)
4.1 引 言	(38)
4.2 内嵌汉字系统程序设计的几个关键技术	(38)
4.3 内嵌汉字系统源程序	(41)
4.4 汉字输入程序设计	(51)
4.5 汉字输入源程序	(53)
5 扩展内存驱动程序设计	(58)
5.1 DOS 内存及其管理	(58)
5.2 XMS 技术编程	(59)
5.3 XMS 内存驱动程序	(60)
5.4 XMS 内存管理程序设计	(63)
5.5 XMS 内存分配子系统设计	(64)
6 数字化仪驱动程序设计	(67)
6.1 8250 通讯程序设计基础	(67)
6.2 数字化仪驱动程序的基本工作原理	(68)
6.3 数字化仪驱动源程序	(69)
6.4 数字化仪高级接口程序设计	(74)
6.5 数字化仪高级接口源程序	(75)

7	键盘及鼠标程序设计	(79)
7.1	交互式 CAD 系统中的逻辑输入设备	(79)
7.2	鼠标程序设计	(80)
7.3	鼠标驱动源程序	(81)
7.4	键盘程序设计	(87)
7.5	键盘处理源程序	(90)
8	CAD 菜单程序设计	(106)
8.1	CAD 菜单系统设计的特点	(106)
8.2	菜单数据结构设计	(106)
8.3	递归菜单程序设计	(107)
8.4	菜单设计源程序	(108)
9	域编辑器程序	(118)
9.1	域编辑器设计的指导思想与基本概念	(118)
9.2	域编辑器数据结构	(119)
9.3	域编辑器程序工作原理与调用方法	(121)
9.4	域编辑器中主要函数介绍	(122)
9.5	域编辑器编程实例	(123)
9.6	域编辑器设计源程序	(128)
10	MCAD 软件系统设计	(146)
10.1	MCAD 系统组织	(146)
10.2	MCAD 的图形数据结构	(148)
10.3	图层管理器设计	(150)
10.4	MCAD 系统的图形内核	(153)
10.5	MCAD 系统执行机构	(155)
10.6	MCAD 系统编译	(158)
10.7	MCAD 公用头文件	(158)
10.8	图形核心驱动源程序	(160)
10.9	直线图素源程序	(164)
10.10	多边形图形源程序	(167)
10.11	圆/弧图素源程序	(172)
10.12	矢量文本图素源程序	(178)
10.13	图层管理器源程序	(187)
10.14	坐标变换源程序	(199)
10.15	绘图仪驱动源程序	(205)
10.16	环境设置源程序	(210)
10.17	选择集构造源程序	(214)
10.18	图形绘制源程序	(215)
10.19	系统工具源程序	(221)
10.20	线型支持源程序	(224)
10.21	命令执行机构源程序	(227)

10.22 屏幕组织及图素生成命令集源程序	(228)
10.23 窗口组织源程序	(232)
10.24 辅助命令集源程序	(234)
10.25 图形操作源程序	(235)
10.26 快速显示支持源程序	(249)
10.27 MCAD 主程序	(250)
11 MCAD 绘图系统简介	(252)
11.1 MCAD 基本命令与特点	(252)
11.2 MCAD 的基本概念、方法和约定	(255)
11.3 MCAD 系统的定制化	(259)

1 计算机图形基本算法

1.1 图形的生成

在所有的图形中,直线段是最基本的,用直线段不仅可以生成各种直线图形,而且可以生成曲线图形和三维图形。以下给出直线和圆弧的生成算法。

1.1.1 直线的生成

如图 1.1 所示,用 t 来表示步长,误差 $e_1 = \frac{\Delta y}{\Delta x}t$,对图中实线 $e_1 < \frac{1}{2}t$,生成直线时只沿 X 方向走到 1 点, Y 方向坐标不变;对图中虚线, $e_1 = \frac{\Delta y}{\Delta x} > \frac{1}{2}t$,沿 X 、 Y 方向各走一步到 2 点。如果让 e 减去 $\frac{1}{2}t$,则 $e = \frac{\Delta y}{\Delta x} - \frac{1}{2}t$ 。此时有

$$\begin{aligned} e \leq 0 & \quad y \text{ 方向不变} \\ e > 0 & \quad y \text{ 方向增加一步} \end{aligned}$$

下一步怎样走,参见图 1.2 中的实线部分,即前一步 y 方向没有走步的情况。很明显,此时的误差为 $e_1 + e_1 - \frac{1}{2}t$ 。即原先所保留的 e 再加上 e_1 ,为 $e + \frac{\Delta y}{\Delta x}t$ 。再来分析图中的虚线部分,即 Y 增加一步的情况。同实线的情况类似,在原来的误差上又增加一个 e_2 ,但由于 Y 方向增加了一个步长,所以误差值又缩小了一个 t ,即为: $e + \frac{\Delta y}{\Delta x}t - t$ 。

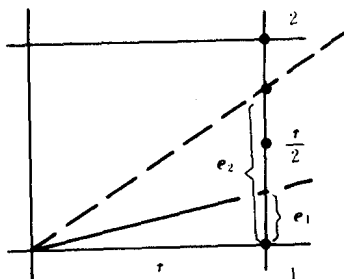


图 1.1 生成直线第一步

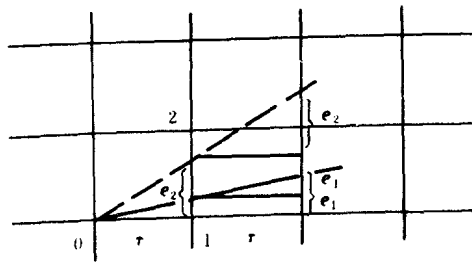


图 1.2 生成直线第二步

事实上,我们并不需要误差值的大小,而只需要知道它的符号就可以决定 Y 方向的走步情况。先对初始值进行简化

$$e = \frac{\Delta y}{\Delta x}t - \frac{1}{2}t = \frac{2\Delta y \cdot t - \Delta x t}{2\Delta x}$$

等式两边同乘 $2\Delta x/t$,并用 $2\Delta x e$ 代替 e ,则初值

$$e = 2\Delta y - \Delta x$$

对 Y 不走步时,下一步误差项可化简为

$$e = e + 2 \cdot \Delta y$$

对 Y 方向走时, 下一步误差项可写做:

$$e = e + (2 \cdot \Delta y - 2 \cdot \Delta x)$$

上面讨论的是 $0 \leq \Delta y \leq \Delta x$ 的情况, 只是平面上直线八种“走向”中的一种, 其它情况可利用对称性推导出误差项的计算公式。

1.1.2 圆弧的生成

我们可以把生成直线的基本想法用于圆弧的生成。把圆分成八个部分(如图 1.2)分别生成。

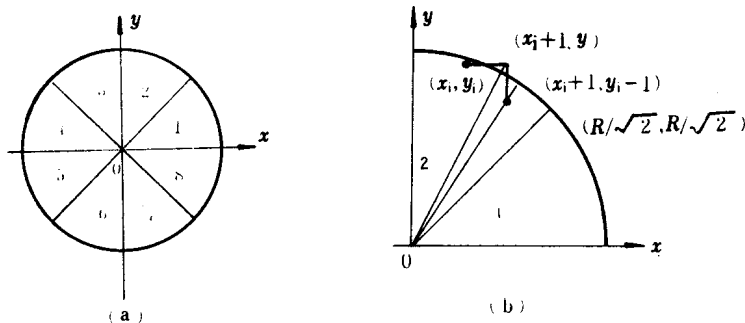


图 1.3 生成圆弧

下面以从点 $(0, R)$ 到 $(\frac{R}{\sqrt{2}}, \frac{R}{\sqrt{2}})$ 一段圆弧的生成为例, 开始 $x_0 = 0, y_0 = R$, 以后每次 X 坐标加 1, Y 坐标则根据误差 e_i 的符号决定减 1 与否, 取 $f_i = (x_i + 1)^2 + y_i^2 - R^2$ 来表示点 $(x_i + 1, y_i)$ 到圆周的距離, $g_i = (x_i + 1)^2 + (y_i - 1)^2 - R^2$ 表示点 $(x_i + 1, y_i - 1)$ 到圆周的距離, 令 $e_i = |f_i| - |g_i|$

若 $e_i \geq 0$, 下一点取 $(x_i + 1, y_i - 1)$;

若 $e_i < 0$, 下一点取 $(x_i + 1, y_i)$;

通过简化处理, 可得递推公式

$$f_{i+1} = \begin{cases} g_i + 2x_i + 3 & \text{当 } e_i \geq 0 \\ f_i + 2x_i + 3 & \text{当 } e_i < 0 \end{cases}$$

$$g_{i+1} = \begin{cases} f_{i+1} - 2y_i - 1 & \text{当 } e_i \geq 0 \\ f_{i+1} - 2y_i + 1 & \text{当 } e_i < 0 \end{cases}$$

由此得下面的算法:

$$1. f_0 = 1, g_0 = 2(1 - R)$$

2. 对 $i = 0$ 到 $i = \left[\frac{R}{\sqrt{2}} + \frac{1}{2} \right] - 1$, 有 $x_{i+1} = x_i + 1$, 且

若 $|f_i| \geq |g_i|$, 则 $y_{i+1} = y_i - 1, f_{i+1} = g_i + 2x_i + 3, g_{i+1} = f_{i+1} - 2y_i - 1$;

若 $|f_i| < |g_i|$, 则 $y_{i+1} = y_i, f_{i+1} = f_i + 2x_i + 3, g_{i+1} = f_{i+1} - 2y_i + 1$ 。

1.2 图形的矩阵变换

平面图形都是由点、线构成的。而线可以用具有一定关系的点来表示。这样我们就可以

把线、面看成是一些点的集合,称之为点集。二维平面中的图形可以用点集的矩阵表示。对图形的变换可由点集的变换来完成。二维空间中的点可以用行向量 $[x, y]$ 表示,也可以用列向量 $\begin{bmatrix} x \\ y \end{bmatrix}$ 表示。

1.2.1 图形的矩阵变换

如果用行向量 $[x, y]$ 来表示二维图形中的点,用二阶方阵来表示二维变换矩阵,则有:

$$[x, y] \begin{bmatrix} a & b \\ c & d \end{bmatrix} = [(ax + cy), (bx + dy)] = [x^*, y^*]$$

式中的 x^*, y^* 表示变换后新点的坐标。由上式看出一点经二维变换后所得的新点的坐标由变换矩阵的各个元素来决定。

1) 恒等变换

当 $a = d = 1, b = c = 0$ 时,

$$[x, y] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = [x, y] = [x^*, y^*]$$

变换矩阵是单位矩阵,变换前后的点相等。

2) 比例变换

当 $b = c = 0$ 时,

$$[x, y] \begin{bmatrix} a & 0 \\ 0 & d \end{bmatrix} = [ax, dy] = [x^*, y^*]$$

元素 a 控制 x 坐标的变化, d 控制 y 坐标的变化。 $a = d > 1$ 为等比例放大, $a = d < 1$ 为等比例缩小, $a \neq d$ 则产生畸变。

3) 反射变换

(1)当 $b = c = 0, a = 1, d = -1$ 时,

$$[x, y] \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = [x, -y] = [x^*, y^*]$$

产生对 x 轴的反射变换(图 1.4(a))。

(2)当 $b = c = 0, a = -1, d = 1$ 时,

$$[x, y] \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} = [-x, y] = [x^*, y^*]$$

产生对 y 轴的反射变换(图 1.4(b))。

(3)当 $a = d = 0, b = c = 1$ 时,

$$[x, y] \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = [y, x] = [x^*, y^*]$$

产生对直线 $y = x$ 的反射变换(图 1.4(c))。

(4)当 $a = d = 0, b = c = -1$ 时,

$$[x, y] \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} = [-x, -y] = [x^*, y^*]$$

产生对直线 $y = -x$ 的反射变换(图 1.4(d))。

(5)当 $b = c = 0, a = d = -1$ 时,

$$[x, y] \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = [-x, -y] = [x^*, y^*]$$

产生对原点的反射变换(图 1.4(e))。

4) 错切变换

(1) 当 $a = d = 1, c = 0$ 时,

$$[x, y] \begin{bmatrix} 1 & b \\ 0 & 1 \end{bmatrix} = [x, bx + y] = [x^*, y^*]$$

产生沿 y 方向的错切变换(图 1.5(a))。

(2) 当 $a = d = 1, b = 0$ 时,

$$[x, y] \begin{bmatrix} 1 & 0 \\ c & 1 \end{bmatrix} = [x + cy, y] = [x^*, y^*]$$

产生沿 x 方向的错切变换(图 1.5(b))。

5) 旋转变换

旋转变换指点绕着原点旋转 θ 角, 设 $a = \cos\theta, b = \sin\theta, c = -\sin\theta, d = \cos\theta$, 则:

$$[x, y] \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} = [x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta] [x^*, y^*]$$

旋转角 θ 逆时针为正, 顺时针为负。

1.2.2 点的齐次坐标及其变换

上一节研究的二阶方阵变换中没有包含平移变换, 这是因为变换矩阵 $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ 中不可能引入平移的元素。为了解决这一问题并使变换矩阵具有更多的功能, 我们引入齐次坐标的概念, 所谓点的齐次坐标就是将平面上一个点 $[x, y]$ 表示为 $[u, v, \xi]$ 而令 $x = u/\xi, y = v/\xi$ 。一般地, 我们把二维点的齐次坐标形式写为 $[x, y, 1]$ 。经某种变换后为 $[\bar{X}, \bar{Y}, H]$, 它的真正二维坐标为 $[\bar{X}/H, \bar{Y}/H, 1]$ 或表示为 $[x^*, y^*, 1]$ 对应点的齐次坐标的变换矩阵是一个三阶方阵 T :

$$T = \begin{bmatrix} a & b & p \\ c & d & q \\ l & m & s \end{bmatrix}$$

一个二维点的线性变换可以写成

$$[x, y, 1] \cdot T = [\bar{X}, \bar{Y}, H] = \left[\frac{\bar{X}}{H}, \frac{\bar{Y}}{H}, 1 \right] = [x^*, y^*, 1]$$

在齐次坐标变换矩阵中元素 a, b, c, d 可以控制恒等、比例、错切、反射和旋转变换, l, m 控制平移变换, p, q 控制透视变换, s 控制全比例变换。

对纯平移变换:

$$[x, y, 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l & m & 1 \end{bmatrix} = [x + l, y + m, 1]$$

对单纯的全比例变换

$$[x, y, 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s \end{bmatrix} = [x, y, s] = \left[\frac{x}{s}, \frac{y}{s}, 1 \right]$$

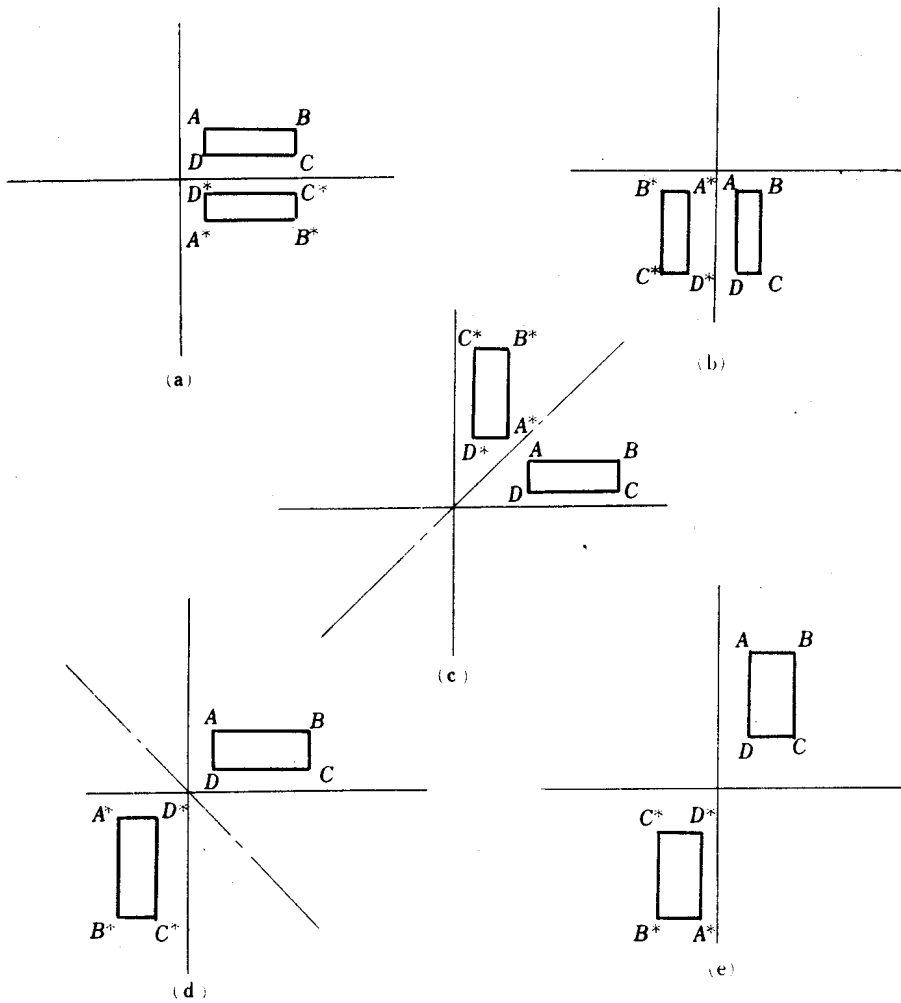


图 1.4 反射变换

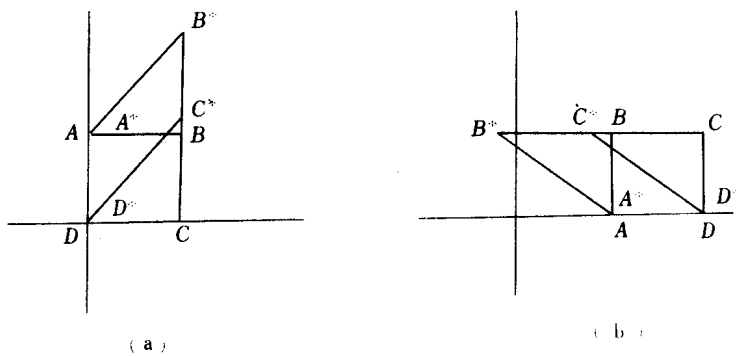


图 1.5 错切变换

很明显,当 $s > 1$ 时为全比例缩小, $s < 1$ 时为全比例放大。在这里应该注意一下控制比例变换的元素 a, d 及控制比例变换的元素 s 间的不同点,以便明确全比例变换的含义。

在解决实际问题时,有时需要对一个点集按一定顺序进行二个以上的变换,我们称为连续变换。由于矩阵乘法不适用于交换律,所以在连续变换时,一定要注意不能改变变换的次序,否则是不会得到预期结果的。

1.3 图形的计算

图形处理中经常求直线段与直线段、直线段与曲线段、曲线段与曲线段的交点。也会遇到求点与区域及区域之间的相互关系。这些问题构成了图形的算法基础。本章对此作必要的介绍。

1.3.1 交点的计算方法

1) 二直线的交点

求直线段与直线段的交点这是图形处理中经常遇到的问题。设有直线段 AB 和 CD , 其端点分别为 $A(X_A, Y_A), B(X_B, Y_B)$ 和 $C(X_C, Y_C), D(X_D, Y_D)$ 。

直线段 AB 的参数方程为:

$$\begin{cases} X = X_A + (X_B - X_A)t \\ Y = Y_A + (Y_B - Y_A)t \end{cases} \quad (0 \leq t \leq 1)$$

直线段 CD 的参数方程为:

$$\begin{cases} X = X_C + (X_D - X_C)u \\ Y = Y_C + (Y_D - Y_C)u \end{cases} \quad (0 \leq u \leq 1)$$

很明显,如二直线段存在交点,其交点的参数 (t, u) 满足:

$$\begin{cases} X_A + (X_B - X_A)t = X_C + (X_D - X_C)u \\ Y_A + (Y_B - Y_A)t = Y_C + (Y_D - Y_C)u \end{cases}$$

即

$$(X_B - X_A)t - (X_D - X_C)u = X_C - X_A$$

$$(Y_B - Y_A)t - (Y_D - Y_C)u = Y_C - Y_A$$

如果

$$\Delta = \begin{vmatrix} X_B - X_A & -(X_D - X_C) \\ Y_B - Y_A & -(Y_D - Y_C) \end{vmatrix} = 0$$

说明 AB 与 CD 平行(包括重合或部分重合)。对这种情况视为 AB, CD 不相交。如果

$$\Delta = \begin{vmatrix} X_B - X_A & -(X_D - X_C) \\ Y_B - Y_A & -(Y_D - Y_C) \end{vmatrix} \neq 0$$

则可求得:

$$t = \frac{1}{\Delta} \begin{vmatrix} X_B - X_A & -(X_D - X_C) \\ Y_C - Y_A & -(Y_D - Y_C) \end{vmatrix}$$

$$u = \frac{1}{\Delta} \begin{vmatrix} X_C - X_A & X_C - X_A \\ Y_B - Y_A & Y_C - Y_A \end{vmatrix}$$

当 $0 \leq t \leq 1, 0 \leq u \leq 1$ 时, 说明 AB 、 CD 确实有一个交点。否则, 交点在 AB 、 CD 的某一侧延长线上。

2) 直线段与曲线段的交点

(1) 直线段与圆弧的交点

设直线段 AB 的参数方程为:

$$\begin{cases} X = X_A + (X_B - X_A)t \\ Y = Y_A + (Y_B - Y_A)t \end{cases} \quad (0 \leq t \leq 1)$$

半径为 R , 中心为 $C(X_C, Y_C)$ 的圆的参数方程为:

$$\begin{cases} X = X_C + R \cdot \cos \theta \\ Y = Y_C + R \cdot \sin \theta \end{cases} \quad \theta_s \leq \theta \leq \theta_e$$

θ_s, θ_e 分别为圆弧的起始角和终止角。直线段与圆弧交点处的参数 (t, θ) 应满足:

$$\begin{cases} X_A + (X_B - X_A)t = X_C + R \cdot \cos \theta \\ Y_A + (Y_B - Y_A)t = Y_C + R \cdot \sin \theta \end{cases}$$

消去 θ 得:

$$\left[\frac{(X_A - X_C) + (X_B - X_A)t}{R} \right]^2 + \left[\frac{(Y_A - Y_C) + (Y_B - Y_A)t}{R} \right]^2 = 1$$

整理后, 可得:

$$At^2 + Bt + C = 0 \quad (1.1)$$

其中:

$$A = (X_B - X_A)^2 + (Y_B - Y_A)^2$$

$$B = 2(X_A - X_C)(X_B - X_A) + 2(Y_A - Y_C)(Y_B - Y_A)$$

$$C = (X_A - X_C)^2 + (Y_A - Y_C)^2 - R^2$$

当 $B^2 - 4AC < 0$ 时, 式 1.1 无实根, 说明直线段与圆弧不相交。

当 $B^2 - 4AC > 0$ 时, 式 1.1 有二个实根 t_1, t_2 。由此求得:

$$\cos \theta = \frac{(X_A - X_C) + (X_B - X_A)t}{R}$$

$$\sin \theta = \frac{(Y_A - Y_C) + (Y_B - Y_A)t}{R}$$

的二个根 θ_1, θ_2 。当 $0 \leq t_i \leq 1 (i=1, 2)$ 求出的 θ_i 满足 $\theta_s \leq \theta_i \leq \theta_e (i=1, 2)$ 时为实有交点, 否则为直线段所在直线与圆弧所在圆的交点。

当 $B^2 - 4AC = 0$ 时, 说明直线段与圆弧相切。只有当切点处的参数 t_0, θ_0 满足 $0 \leq t_0 \leq 1, \theta_s \leq \theta_0 \leq \theta_e$ 时, 才是实有的切点。

(2) 直线段与曲线交点

将上面的讨论进行推广, 对直线段与二次曲线交点的计算, 可以有类似的分析。

设二次曲线方程为:

$$AX^2 + BXY + CY^2 + DX + EY + F = 0$$

直线段 AB 与其相交交点, 可由下式求出:

$$\bar{A}t^2 + \bar{B}t + \bar{C} = 0$$

此式与式 1.1 相同的形式,故对根及交点的讨论也与式 1.1 相同。

对于三次、四次曲线与直线段相交问题只需解 t 的三次、四次方程,仍有公式解。由于五次以上的方程不存在公式解,故直线段与五次以上曲线的交点求解只能采用特殊的求解方法或采用近似解。

3) 直线段与拟合曲线的交点

直线段与拟合曲线的交点计算一般采用近似解法。下面以 Bezier 曲线为例,求拟合曲线与直线段的交点。由于 Bezier 曲线的可分割性,可以用控制多边形的分割多边形去逼近。因此,可以用直线段与分割多边形的交点去逼近它与 Bezier 曲线的交点。

设 $P_i (i=0, 1 \cdots n)$ 为平面 N 次 Bezier 曲线特征多边形顶点, L 为已知平面直线,见图 1.6。特征多边形第 i 边可记为: $P_{i-1}P_i = P_i + (P_i - P_{i-1})t$ 。直线 L 与 Bezier 曲线交点可按下面步骤计算。

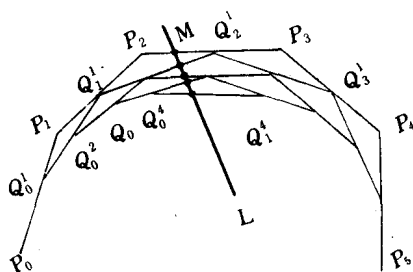


图 1.6 直线与拟合曲线交点

(1) 计算 L 与特征多边形第 i 边交点 M 的参数 t 。如果 $t \leq 0.5$, 用 $S = L_0$ 表示; $t > 0.5$, 用 $S = R_0$ 表示。无交点时用 $S = -1$ 表示。

(2) 以特征多边形各边中点构成分割多边形 $Q_i (i=0, 1, \cdots, n-j)$ 。 j 为所做分割多边形的次数。

(3) 若 $S = L_0$ 则计算 L 与 $Q_{i-2}Q_{i-1}$ 的交点, 并计算参数 t' , 由此确定 S 值。如果 $S = -1$ 计算结束。

(4) 若 $S = R_0$, 则计算 L 与 $Q_{i-1}Q_i$ 的交点, 并计算参数 t' , 由此确定 S 值, 如果 $S = -1$ 计算结束。

(5) 当交点参数 $t \leq 0.5$, 且交点在第一边上时, 取 $S = L_0, t_0 = 2t$ 。交点参数 $t > 0.5$, 且交点在最后一边时, 取 $S = R_0, t_0 = 2(t - 0.5)$ 。否则, 根据 t 的值重复第三或第四步的计算。

由于本算法的效率是有选择地计算 L 与分割多边形某一边的交点, 而不是逐边去求交点, 因而大大提高了算法的效率。同时, 本算法保证 L 与分割多边形表的交点一定收敛于 L 与 Bezier 曲线的交点, 而有足够的正确性。

4) 圆弧与圆弧的交点

设有二段圆弧。圆弧 A 的中心为 (X_A, Y_A) , 半径为 R_A , 参数方程为:

$$\begin{cases} X = X_A + R_A \cdot \cos\theta \\ Y = Y_A + R_A \cdot \sin\theta \end{cases} \quad (\theta \in u)$$

圆弧 B 的中心为 (X_B, Y_B) , 半径 R_B , 参数方程为:

$$\begin{cases} X = X_B + R_B \cdot \cos\alpha \\ Y = Y_B + R_B \cdot \sin\alpha \end{cases} \quad (\alpha \in u)$$

交点处的 θ, α 应满足:

$$\begin{cases} X_A + R_A \cdot \cos\theta = X_B + R_B \cdot \cos\theta \\ Y_A + R_A \cdot \sin\theta = Y_B + R_B \cdot \sin\alpha \end{cases} \quad (1.2)$$

消去 θ 得:

$$\left[\frac{(X_B - X_A) + R_B \cdot \cos \alpha}{R_A} \right]^2 + \left[\frac{(Y_B - Y_A) + R_B \cdot \sin \alpha}{R_A} \right]^2 = 1 \quad (1.3)$$

令:

$$\rho^2 = (X_B - X_A)^2 + (Y_B - Y_A)^2$$

$$\cos \varphi = \frac{X_B - X_A}{\rho}$$

$$\sin \varphi = \frac{Y_B - Y_A}{\rho}$$

式 1.3 经整理可写为:

$$\frac{\rho^2 + (R_B^2 - R_A^2)}{2R_B\rho} + \cos(\varphi - \alpha) = 0 \quad (1.4)$$

当 $\left| \frac{\rho^2 + (R_B^2 - R_A^2)}{2R_B\rho} \right| \leq 1$ 时, 令

$$\delta = \cos^{-1} \left(\frac{\rho^2 + (R_B^2 - R_A^2)}{2R_B\rho} \right)$$

式 1.4 有解:

$$\alpha_1 = \varphi + \pi + \delta \pmod{2\pi}$$

$$\alpha_2 = \varphi + \pi - \delta \pmod{2\pi}$$

将 1.4 的二个解 α_1 和 α_2 代入式 1.2 可以求得 θ_1, θ_2 。

当 $\delta = 0$ 或 $\delta = \pi$ 时, 表示二圆弧相切, 求出的是切点。

当 $\alpha_{1(2)} \in v, \theta_{1(2)} \in u$ 时, 可求得二个交点 $(X_1, Y_1), (X_2, Y_2)$ 。

当 α_1, α_2 中只有一个在域 v 内, θ_1, θ_2 也只有一个在域 u 内时, 二圆弧只存在一个实际交点。 $\alpha_{1(2)}, \theta_{1(2)}$ 均不在域 v, u 内, 二圆弧无实际交点。

1.3.2 包含与重叠

在许多图形处理问题中都涉及点与区域或区域与区域间的相互关系, 这些关系有包含、重叠等等。下面讨论这类关系的若干判断方法。

1) 点在区域内外的判断

(1) 转角法

若判断点 $M_0(X_0, Y_0)$ 是否在区域 D 内, 我们可以把点 M_0 与 D 边界点 P 沿边界移动时, $\overline{M_0P}$ 与有向直线的有向夹角就发生变化。如果 P 沿边界移动一周, 夹角的变化值为零, 那么 M_0 必在区域 D 之外, 如图 1.7(a); 如果夹角的变化值为 2π , 那和 M_0 必在区域 D 之内如图 1.7(b)。 M_0 在边界上的情况比较复杂, 如果边界是连续光滑的, 则变化角为 π , 如图 1.7(d)。如果边界是分段光滑的, 则变化角即为两端切线间的夹角, 如图 1.7(c)。

(2) 标号法

此法适用于判断点与多边形区域间的关系。以点 M_0 为原点, 划分出四个象限, 按其顺序分别赋以标号 0, 1, 2, 3。用 $\alpha(A), \alpha(B) \cdots$ 表示点 $A, B \cdots$ 所在象限的标号, 对多边形的某一边 AB , 令

$$\Delta\alpha_{AB} = \begin{cases} \alpha(B) - \alpha(A) & \text{当 } -1 \leq \alpha(B) - \alpha(A) \leq 1 \\ \alpha(B) - \alpha(A) - 4 & \text{当 } \alpha(B) - \alpha(A) > 2 \\ \alpha(B) - \alpha(A) + 4 & \text{当 } \alpha(B) - \alpha(A) < -2 \\ [\alpha(B) - \alpha(C)] + [\alpha(C) - \alpha(A)] & \text{当 } \alpha(B) - \alpha(A) = \pm 2 \end{cases}$$

其中 C 在 AB 上, 但属于 A, B 所在的象限。 $\Delta\alpha_{AB}$ 表示多边形一边 AB 的标号。如图 1.8 所示。用 W 表示多边形各边标号的总和, 记为 $W = \sum \Delta\alpha$ 。于是, 当 M_0 在多边形外时 $W = 0$, 在多边形内部时 $W = \pm 4K$ (K 为整数)。

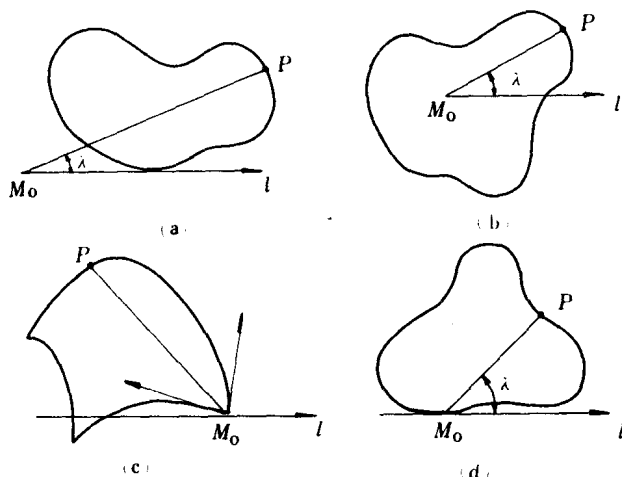


图 1.7 转角法

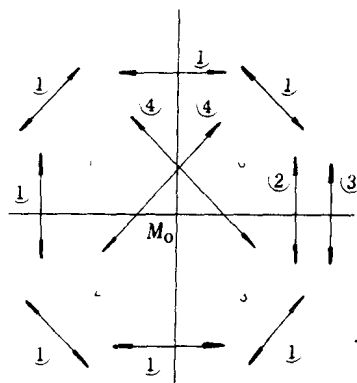


图 1.8 标号法

(3)射线法

由点 M_0 引半射线 l , 与边界 C 交点共 K 个。当 K 为奇数, 则 M_0 在 D 内; 为偶数, 则 M_0 在 D 外。不过这里规定, 射线与边界相切时, 如果边界在 l 的异侧, 则作为一个交点; 边界在 l 的同侧, 则作为无交点处理, 如图 1.9 所示。

(4)符号法

假定区域 D 的边界由方程 $f(X, Y) = 0$ 给出, 且在边界内外有不同的符号。于是, 可以由 $f(X, Y)$ 的符号来决定点 $M_0(X_0, Y_0)$ 在区域 D 的内外。如果 D 的边界由若干段曲线 $f_i(X, Y) = 0$ ($i = 1, 2, \dots, K$) 围成, 且 D 内的点 $f_i(X, Y)$ 具有相同的符号, 都为 f_i ($i = 1, 2, \dots, K$), 这样就可以根据 $f_i(X_0, Y_0)$ 的符号是否与 f_i 完全一致来判断 M_0 是否在 D 内。如果完全一致, 则 M_0 在 D 域内; 只要有一个符号不一致, M_0 就处在 D 域之外; 如果有一个为零, 其余符号全一致, 则 M_0 在边界上。

2) 线段与区域的重叠

所谓重叠是指直线与区域的公共部分或直线段在区域内的部分。利用直线段的参数方程可求出重叠存在的条件。

设区域 D 与直线段 AB 所在直线相交于 C 和 D , 如图 1.10, 若直线段的充分必要条件是

$$\begin{cases} X = X_A + (X_B - X_A)t \\ Y = Y_A + (Y_B - Y_A)t \end{cases} \quad (0 \leq t \leq 1)$$

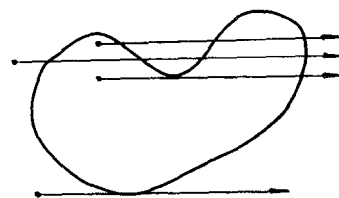


图 1.9 射线法