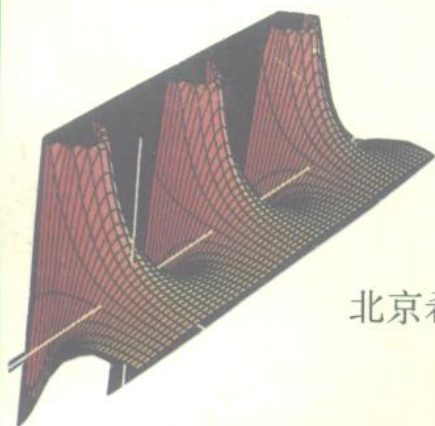
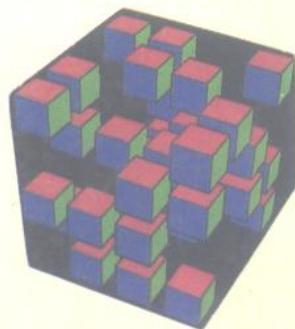




北京希望电脑公司计算机图形技术丛书

三维图形 动画编程实例

侯 阳 编
迪 克



海 洋 出 版 社

北京希望电脑公司计算机图形技术丛书

三维图形、动画编程实例

侯 阳 迪 克 编

海洋出版社

1993 年 · 北京

内 容 提 要

本书提供了三维图形和建模的详细介绍,讨论了流行的光线跟踪和计算机动画。内容包括通用例程,三维建模,光线跟踪和动画,书中给出了详尽的实例,所有C代码都用 Borland C++编译程序编译通过,可在各种PC上运行。

侯阳,张国伟,卢雄,严凯民,沈智红,黄勇,贺渊平,顾明文,徐水泉,张连生,刘海东和迪克参加了本书的编写和审校工作。

需要本书的用户,请直接与北京 8721 信箱联系,邮编 100080,电话 2562329。

(京)新登字 087 号

责任编辑:阎世尊

北京希望电脑公司计算机图形技术丛书

三维图形、动画编程实例

侯阳 迪克 编

*

海洋出版社出版(北京市复兴门外大街1号)

海洋出版社发行 (商印刷厂印刷)

开本:787×1092 1/16 印张:33.5 字数:736千字

1993年9月第一版 1993年9月第一次印刷

印数:1—5000

*

ISBN 7-5027-3798-0/TP·227 定价:35.00元

目 录

简 介	(1)
第一部分 通用例程	(6)
第一章 模块简介	(6)
1.1 头文件	(6)
1.1.1 DEFS.H	(6)
1.1.2 GLOBALS.H	(7)
1.1.3 MATHB.H	(7)
1.1.4 GRAPHB.H	(9)
1.2 编译模块	(10)
第二章 数学模块	(11)
2.1 数值函数	(24)
2.1.1 Radians 和 Degrees	(24)
2.1.2 CosD 和 SinD	(24)
2.1.3 Power	(24)
2.1.4 Log	(25)
2.1.5 Exp10	(25)
2.1.6 Sign 和 IntSign	(25)
2.1.7 IntSqrt	(25)
2.1.8 IntPower	(26)
2.1.9 MIN 和 MAX	(26)
2.1.10 MIN3 和 MAX3	(26)
2.1.11 MIN4 和 MAX4	(26)
2.2 矢量和矩阵例程	(27)
2.2.1 Vec 和 VecInt	(27)
2.2.2 UnVec 和 UnVecInt	(27)
2.2.3 VecDot	(27)
2.2.4 VecCross	(28)
2.2.5 VecLen	(29)
2.2.6 VecNormalize	(29)
2.2.7 VecMatxMult	(29)
2.2.8 VecSub 和 VecSubInt	(29)
2.2.9 VecAdd	(29)
2.2.10 VecAdd3	(30)

2.2.11 VecCopy 和 VecCopyInt	(30)
2.2.12 VecLinComb	(30)
2.2.13 VecScalMult, VecScalMultI 和 VecScalMultInt	(30)
2.2.14 VecAddScalMult	(30)
2.2.15 VecNull 和 VecNullInt	(30)
2.2.16 VecElemMult	(31)
2.2.17 VecMin 和 VecMax	(31)
2.2.18 VecNegate	(31)
2.3 关系变换例程	(31)
2.3.1 ZeroMatrix	(31)
2.3.2 Translate3D	(32)
2.3.3 Scale3D	(32)
2.3.4 Rotate3D	(32)
2.3.5 Multiply3DMatrices	(33)
2.3.6 MatCopy	(33)
2.3.7 PrepareMatrix	(33)
2.3.8 PrepareInvMatrix	(34)
2.3.9 Transform	(34)
2.4 伪随机数生成	(34)
2.4.1 InitRand	(34)
2.4.2 Rand	(34)
2.4.3 RandInt	(34)
第三章 图形界面模块	(35)
3.1 基本图形函数	(35)
3.1.1 Scal_Mode	(35)
3.1.2 Calc_Offsets	(35)
3.1.3 Pre_Calc	(35)
3.1.4 Plot	(36)
3.1.5 Clear_Palette	(36)
3.1.6 Set_Palette	(36)
3.1.7 Init_Palette	(36)
3.1.8 Init_Palette_2	(37)
3.1.9 Cycle_Palette	(37)
3.1.10 Swap	(37)
3.1.11 Circle	(37)
3.1.12 Draw	(38)
3.1.13 Init_Graphics	(38)
3.1.14 Set_Graphics_Mode	(38)

3.1.15 Wait_For_Key	(39)
3.1.16 Exit_Graphics.....	(39)
3.1.17 Title	(39)
3.2 三维画图例程.....	(39)
3.2.1 Init_Plotting	(39)
3.2.2 Init_Perspective	(39)
3.2.3 Map_Coordinates	(39)
3.2.4 Cartesian_Plot_3D.....	(40)
3.2.5 Cylindrical_Plot_3D	(40)
3.2.6 Spherical_Plot_3D	(41)
3.2.7 Draw_Line_3D	(41)
3.3 像素例程.....	(41)
3.3.1 Put_Pixel	(41)
3.3.2 Get_Pixel	(42)
3.4 坐标轴和调色板例程的建立.....	(42)
3.4.1 Put_Axis_And_Palette.....	(42)
3.4.2 Display_Axis.....	(42)
3.4.3 Display_Palette.....	(43)
3.4.4 Axis_And_Palette	(43)
第四章 使用模块	(58)
4.1 三维迭代功能系统程序.....	(58)
4.1.1 3D-AVR.C IFS 程序	(58)
4.1.2 3D-IFS.C IFS 程序	(65)
4.2 云彩和等离子体生成程序.....	(67)
4.3 三维轨道模拟程序.....	(71)
4.4 一个递归的植物和树生成程序.....	(79)
4.5 线性单元自动程序.....	(81)
4.6 平面单元自动程序.....	(89)
4.7 生命单元自动程序.....	(89)
第二部分 三维模型	(90)
第五章 模型理论和数据库结构	(90)
5.1 如何构造对象.....	(90)
5.2 有关坐标系统的注释.....	(94)
5.2.1 浮点和整数坐标	(94)
5.3 创建对象.....	(94)
5.4 声明 Vertex 和 Facet 数组	(95)

5.5 加载和保存数组数据	(95)
5.6 手工生成数据文件	(96)
第六章 增加对象到景象	(98)
6.1 对象数据库结构	(98)
6.2 初始化和加载对象缓冲区	(99)
6.3 增加边缘反射器到景象	(100)
6.4 从磁盘文件增加对象	(100)
第七章 排序和显示对象	(102)
7.1 排序对象	(102)
7.2 在屏幕上画对象	(103)
7.3 显示对象及其反射	(104)
7.4 前面是什么	(105)
第八章 模型建立程序和创建动画	(106)
8.1 关系变换	(106)
8.2 观察点和光源矢量	(106)
8.3 平面法线矢量	(107)
8.4 光照模型	(107)
8.5 表面可见性测试	(108)
8.6 镜像屏幕缓冲区	(108)
8.7 多边形表面填充例程	(109)
8.8 写到临时文件	(110)
8.9 3DMODEL.C 程序	(110)
第九章 创建对象数据库	(139)
9.1 增加顶点	(141)
9.2 初始化顶点数据库生成程序	(142)
9.3 用 MAKEOBJ.C 创建对象	(142)
9.4 生成锥面和金字塔数据文件	(145)
9.5 生成柱面数据文件	(148)
9.6 生成球面数据文件	(150)
9.7 生成环面数据文件	(153)
9.8 生成旋转面数据文件	(155)
9.9 生成栅格数据文件	(160)
第十章 编辑景象文件	(164)
10.1 生成动画脚本	(164)

10.1.1 创建景象文件	(165)
10.1.2 加载已有景象文件	(165)
10.1.3 编辑景象文件	(165)
10.1.4 保存景象文件到磁盘	(165)
10.2 动画脚本	(166)
10.2.1 SPHRPLAN.SCN	(166)
10.2.2 SPHRWALL.SCN	(176)
10.2.3 SPINTOR.SCN	(198)
10.2.4 SOLOFREV.SCN	(215)
 第三部分 光线跟踪	 (263)
 第十一章 光线跟踪理论	 (263)
11.1 光线跟踪基础	(263)
11.2 计算光线/对象相交	(265)
11.3 球面相交	(265)
11.4 二次表面相交	(266)
11.5 平面对象相交	(266)
11.6 平行四边形相交	(267)
11.7 光线跟踪算法	(268)
 第十二章 光线跟踪程序	 (271)
12.1 定义景象: .RT 文件	(271)
12.2 编排约束条件	(271)
12.3 加载 .RT 文件	(273)
12.4 光线跟踪程序	(277)
12.5 常量和变量	(277)
12.6 main 程序	(278)
12.7 初始化噪音函数	(279)
12.8 扫描景象	(279)
12.9 跟踪光线	(280)
12.10 确定颜色	(281)
12.11 创建纹理表面	(282)
12.12 完整光线跟踪	(282)
 第十三章 用光线跟踪程序创建图像和动画	 (345)
13.1 光线跟踪动画文件创建程序	(346)
13.2 BOUNCE.C	(346)
13.2.1 BOUNCE.C	(347)

13.2.2	BOUNCE2.C	(351)
13.2.3	DNA.C	(355)
13.2.4	ENGINE.C	(362)
13.2.5	NEWTORAD.C	(369)
13.2.6	ORBIT.C	(376)
13.2.7	ROBOT.C	(382)
13.2.8	SPIN2.C	(387)
13.2.9	WAVES.C	(392)
13.3	光线跟踪的单帧重放	(395)
13.3.1	DIE&CARD.C	(395)
13.3.2	CANNON.C	(405)
13.3.3	DESERT.C	(413)
13.3.4	MOUNT.C	(417)
13.3.5	DESK.C	(427)
13.3.6	PIANO.C	(434)
13.3.7	POOLTABL.C	(442)
13.3.8	TETRA.C	(451)
13.3.9	TETSPHR2.C	(459)
13.3.10	WAVYORB.C	(465)
第十四章	动画技术	(469)
14.1	计算机动画和手工动画	(469)
14.2	动画方法	(470)
14.3	平滑技术	(470)
第十五章	动画程序	(471)
15.1	ANIMATE.C 变量	(471)
15.2	ANIMATE.C 的函数	(472)
15.2.1	Set_Mode	(472)
15.2.2	Init_Screen	(472)
15.2.3	Plot	(472)
15.2.4	Set_Palette	(472)
15.2.5	Init_Graphics	(472)
15.2.6	Exit_Graphics	(472)
15.2.7	Set_Text_Screen	(473)
15.2.8	Allocate_Memory	(473)
15.2.9	Free_Memory	(473)
15.2.10	Max	(473)
15.2.11	Get_Key	(473)

15.2.12	Get__ANI__File__Name	(473)
15.2.13	Init__Cells	(473)
15.2.14	Init__Window	(474)
15.2.15	Do__Title	(474)
15.2.16	Find__Brightest__Color	(474)
15.2.17	H__Line	(474)
15.2.18	V__Line	(474)
15.2.19	Draw__D__Meter	(474)
15.2.20	Update__D__Meter	(474)
15.2.21	Init__D__Meter	(474)
15.2.22	Animate	(474)
15.2.23	Load__Cells	(475)
15.3	ANIMATE.C 控制	(476)
15.4	运行 ANIMATE.C	(476)
第五部分	彩色处理	(488)
第十六章	用彩色频率分布处理程序减少颜色	(488)
16.1	彩色频率分布处理程序	(488)
16.2	PROCESS.C 程序和 PROCESS.H 头文件	(489)
16.2.1	.ANI 文件格式	(490)
16.2.2	DISPRGB.C 程序	(490)
16.2.3	关于图形卡	(490)
附录 A	数学和图形模块函数	(503)
附图		(511)

简介

演示复杂的三维景象是对制作电影的模拟。作为导演，你必须协调好各种各样的成员：演员（环境中的对象）；脚本（定义对象在动画菜单中将做些什么）；艺术指导（指定景象如何加亮以及各种元素看上去应是什么样子）；摄影师（控制摄影机的移动，聚焦，景宽，景深，等等）；以及各种各样的支持人员。所有这些元素一起工作，最终把图像显示在屏幕上。使用本书中的软件，你将能够创建描述令人吃惊的现实和只能在计算机产生的世界中存在的景象。

1. 本书的结构

全书分成五个部分，每一部分都分别讲述计算机图像生成的一个特定方面。每一章都是综合性的，都根据前面几章的信息产生更加复杂的图像。

第一部分提供处理三维计算机图形所需的矢量算术和矩阵运算的工具，组成一幅景象的所有元素都在三维空间中定义。其中的例程允许这些对象的操作——旋转、比例缩放和变换。大多数图形例程都尽可能地写成与设备无关以及与编译程序无关。所有与设备有关的调用（例如，在屏幕上画一条线或画一个点）都放在同一个组中。任何与编译程序有关的地方也都一一指出。

第二部分讨论创建计算机环境中流行的对象的三维模型建立工具。对象是由三维多边形小平面的集合组成的。例如，一个立方体可以用分别定义其六个表面的多边形（正方形）定义。这些对象可以组合在一起，构成更加复杂的对象，就跟用砌块构成建筑一样。随后可以把几个对象放入到一个景象中定义整个环境，因为不同大小和形状的建筑物可以组合在一起而构成一个城市。模型建立程序用一个简化了的轻型模型来演示这些景象，其速度优越于第三部分中介绍的光线跟踪方法。模型建立程序还提供一个动画脚本生成实用程序，定义产生一个基于文件的动画序列的立体交叉（Fly-over）摄影机路径（与那些商用软件中的相似）。除了脚本化的摄影机动作，该实用程序还让你能够在立体交叉期间平移、缩放或旋转任何对象，以在景象中增加附加的动作。动画制作程序（在第五部分中给出）可以加载这个文件以演示最终的动画序列。

第三部分讨论图片写实景象浓淡处理（rendering）程序。该程序使用称作光线跟踪（ray tracing）的技术来建立光线与景象中的对象交互的模型。不同于其他浓淡处理技术，光线跟踪方法提供处理反射表面（包括折射）、表面构造和对象之间反射的方法，这个部分中还提供了用于动画的光线跟踪和演示模型的概念化理论，以及用于文本中的C代码增强。动画序列景象是通过建立物理实体的模型、为光线跟踪创建脚本的程序生成的。光线跟踪景象的动画和基于文件的模型化景象类似。

第四部分涉及动画程序，它读取由三维模型建立程序或光线跟踪程序创建的动画文件，并重播其中存储的景象。该程序控制动画的速度和方向，并允许通过连续地循环一个序列来实现平滑的重复景象。

第五部分描述把由模型建立程序和光线跟踪程序计算出来的 32768 种颜色减少到 256 种可用的 VGA 颜色的方法。在这个部分你将会发现图形卡的专用信息。

下面是每一章中的详细描述。

2. 数学模块

第一章介绍第二和第三章中讨论的数学和图形模块的头文件。这些文件和模块是本书中所有软件的基础。数学和图形模块所使用的变量类型和全局数据都可以在这里找到。

第二章介绍计算机图形通常所需的数学函数。这些函数，位于本书中许多程序的核心之处，提供对三维对象及其环境的直接操作。三维矢量和矩阵的所有数学运算都在这一章中处理。变换——包括平移（在空间中移动一个点）、缩放（改变矢量长度）和旋转（围绕一个坐标轴旋转一个点）——都表示为矩阵运算。这些类型的变换中，最好的功能之一是它们的任意组合都可以表达为单独一个变换。这是表达一个对象或摄影机和光源的动作的紧致而有效的方法。

3. 图形界面模块

第三章包含本书中软件所需的许多有用的图形例程。基本绘图、彩色调色板操作、圆和直线生成以及图形方式选择都包括在这一章中。这一章还介绍三维理论、拼字法、从三维点到二维屏幕的投影和坐标系统转换。

4. 如何使用这些模块

乐趣将从第四章开始。在这里，你将学习如何使用前三章中开发的工具，而且将会对可以创建何种类型的图像有一个印象。

5. 模型理论和数据库结构

第五章带我们进入第二部分，这里讨论三维模型建立工具。一个快速计算方法（实数转换成整数）在这里给出。在这一章的最后，可以手工地键入一个对象的数据，以获得实际数据库是如何构造的印象。

6. 增加对象到景象

第六章介绍对象是如何增加到环境中以创建一个景象。对象是用它们的大小以及在三维空间中的方向和位置来定义的。它们还给出指示它们的表示以及它们如何反射的材料特性和参数。

7. 排序和显示对象

第七章介绍为特定的视口安排对象。显示程序必须决定哪些对象离摄影机较近，这样模糊不清的对象就离得较远。还讨论了隐藏面的消除（消隐），也就是对象的不在屏幕上或背向摄影机的部分就不再画出来，就跟反射对象的问题一样。包括了安排对象然后用实体格式显示它们的显示例程；这些例程处理实际对象和它们的镜像。

8. 用模型建立程序创建动画

第八章通过介绍主要的三维模型建立程序把到这一章为止讨论过的东西都合在一起。该程序加载一个定义景象看上去应是什么样子的文件，包括对象位置、方向和大小；光源；和摄影机位置。一个简单的照明模型确定每一个对象表面应如何根据关于单独一个光源的方式而着色。第一个表面所面向的方向在照明计算机和表面可见性测试中计算和使用。一旦标记为可见，则该表面投影到屏幕，而且调用一个多边形表面填充例程以画出该多边形及其各个镜像。最终计算出来的图像用完全的 32768 色调色板写到屏幕的一个临时文件中。各个图像和动画系列的显示由第五部分中介绍的动画程序控制。

9. 创建对象数据库

第九章介绍如何为模型建立程序创建对象以加载和显示。这些对象用多边形表面构造，而这些多边形表面是根据顶点（定义表面各个角的点）定义的。使用三角和几何生成圆锥、锥面、柱面、球面、环面、栅格和旋转面的程序也在这里给出。这一章还介绍如何把这些对象组合在一起以生成更加复杂的对象。

10. 用三维模型建立程序创建景象和动画序列

第十章介绍一个定义景象中所有对象的动作以及摄影机动作的动画脚本文件生成程序。另外还介绍如何生成单个图像。

11. 光线跟踪理论和数据库结构

第三部分从第十一章开始，讨论光线跟踪和光线跟踪图像的动画。光线描述理论和基本光线跟踪算法也在这里讨论，另外还有阴影、光源、射线的反射和折射的递归、透明和构造。你将学习单个或多幅动画是如何产生的，并讨论有关 VGA 的问题。最后还讨论从屏幕文件到适当的动画序列的转换。

12. 用光线跟踪程序创建动画

第十一章中介绍的理论在第十二章中实现，第十二章讨论实际的光线跟踪程序。我们将通过环境来跟踪来自摄影机的射线以观察对象和光线是如何相互作用的。

13. 用光线跟踪程序创建景象和动画序列

在第十三章中我们将创建一个生成 RT 文件的程序，光线跟踪程序使用 RT 文件来创建动画和单幅高分辨率显示。

14. 动画理论

第十四章开始了第四部分，这里主要讨论动画。动画程序读取各幅图像的完成序列，就像电影投影仪一样，用一种重复分时的方法显示它们。这一章还给出通用的动画技术，比如彩色循环和翻页，以用 PC 完成这项工作。像动作模糊和图像过滤这样的技术也在这里给出，以解决这些问题。

15. 动画程序的操作

第十五章演示 ANIMATE.C 动画程序，介绍如何加载屏幕动画文件。像幅面速度控制、动画方向控制和垂直图像交换这样的特殊功能也在这一章中给出。

16. 条形图最小面积彩色归约

第十六章开始第五部分和彩色处理。在这里由显示程序计算出来的 32768 种颜色减少到 256 种 VGA / SVGA 可显示的颜色。创建颜色的彩色条形图；保持最常使用的颜色，而最少使用的颜色则丢弃。这一章还提供有关 STB ERGO PowerGraph VGA 和 Ilon VGA 图形卡的信息。

17. 所需硬件和软件

本书假设读者具有 C 语言的基本工作知识，尤其是对类型定义、结构和指针的使用有较好的了解。

本书中的软件是用 Borland C++ 2.0 中的 C 编译程序并使用标准的 C 约定开发的，因此可以很容易地移植到其他编译程序。只要作很少的修改，就可以用 Microsoft C 或 Turbo C++ 编译程序来编译。

如果还在使用 C 的早期版本，则建议转到 Borland C++ 来，因为这个编译程序对于初学的 C 程序员特别好。

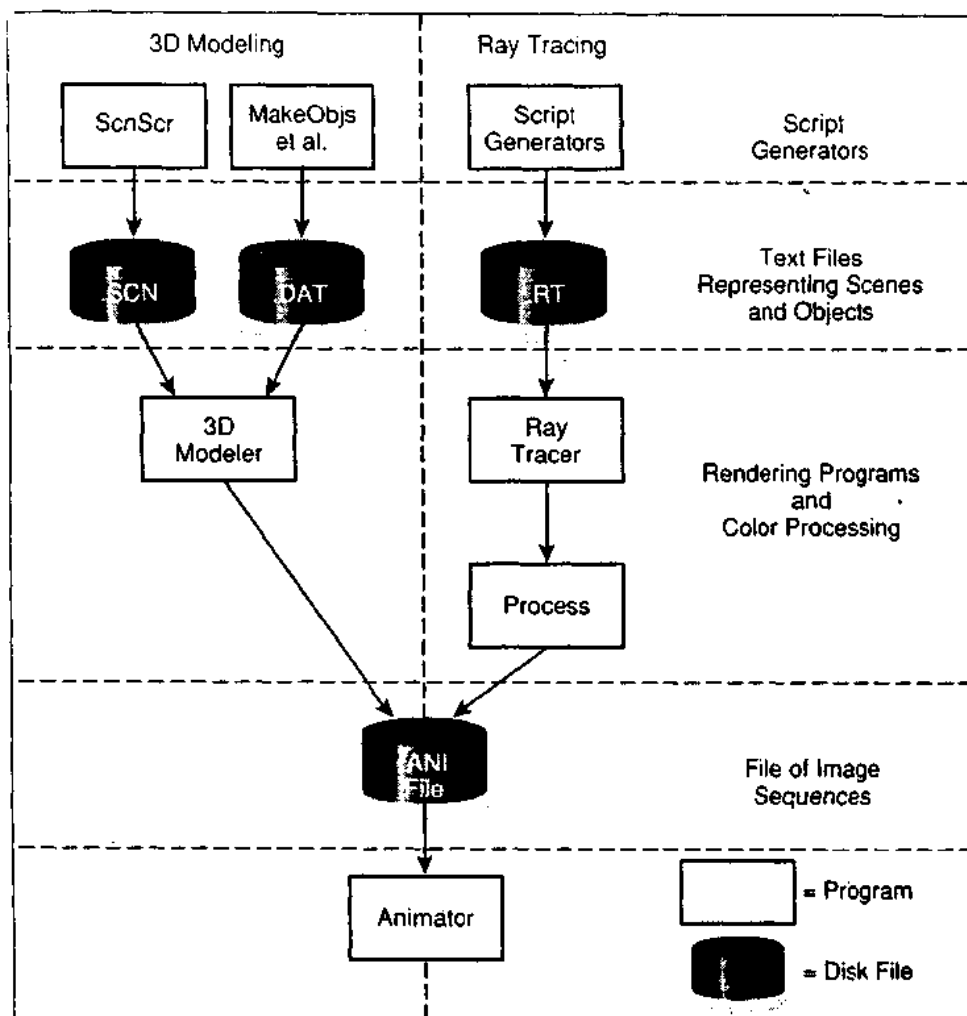


图 1.1 程序和文件如何一起工作的综述

本书中的所有程序都可以在任何带有 VGA 卡和显示器（或更好的）的 IBM PC 或兼容机上运行。这里使用的是图形方式 19 ($320 \times 200 \times 256$) 和 56 ($1024 \times 768 \times 256$)。本书使用 STB System PowerGraph ERGO VGA 和 Ilon VGA，两者都可以处理超级 VGA 的 1024×768 、256 色方式和使用 Tseng 4000 芯片。因为驱动 SVGA 的技术还没有标准化，所以本书中的驱动程序可以保证可以驱动上述两种图形卡（和其他使用 Tseng 4000 芯片的图形卡）。如果你可以从制造商得到必要的寻址和端口信息，则可以很容易地使书中的驱动程序适用于你的图形卡。

软件将在任何 IBM PC 或兼容机上运行，不过越快的机器越好。这些程序是与计算机密切相关的；建议使用 486 或者带有协处理器的 286 或 386，以使用合理的时间生成动画和单幅显示。

第一部分 通用例程

第一章 模块简介

我们用对所需的基本软件工具的综述来开始计算机图形和动画的讨论。这些工具包括执行常用数学函数和提供图像软件（如光线跟踪程序和三维模型建立程序）与计算机的图形硬件之间接口的例程。本章描述这些数学和图形例程正确运行所需的头文件。头文件包含第二章和第三章中数学和图形模块所使用的变量类型和全局变量。在那二章中，你将了解到这些模块中的各个函数是如何工作的以及它们为什么对于计算机图形很重要。第四章通过演示一个使用这些模块中许多函数的样本程序将所有这些都综合在一起。

下面让我们分别看一下各个头文件。

1.1 头文件

头文件——DEFS.H、GLOBALS.H、MATHB.H 和 GRAPHB.H——包含数学和图形界面模块使用的所有各种变量类型和全局数据。没有这些头文件，本书中的其他模块都不能正确编译；因此虽然短小，但是它们包含使模块正确编译和连接的关键数据。

1.1.1 DEFS.H

DEFS.H 头（表 1-1）包含常量定义（#define）和重新声明的变量类型。

表 1-1 DEFS.H 头文件

```
/*
                                DEFS.H Header File
*/

typedef unsigned char      Byte;
typedef unsigned int       Word;
typedef unsigned long      DWord;
typedef enum {false, true} Boolean;

typedef struct
{
    Byte Red;
    Byte Grn;
    Byte Blu;
} RGB;

typedef RGB Palette_Register[256];
```



```

typedef float TDA[3];
typedef int TDIA[3];
typedef float FDA[4];
typedef float Matx4x4[4][4];

#define MaxCol 7
#define MaxInten 35

#define Ln10 2.30258509299405E+000
#define OneOverLn10 0.43429448190325E+000
#define Pi 3.1415927
#define PiOver180 1.74532925199433E-002
#define PiUnder180 5.72957795130823E+001

```

这里重新定义了几个变量类型，以使它们易于认识和理解。例如，unsigned char（无符号字符）现在称作 Byte，而 unsigned int（无符号整数）则称作 Word（这些都是每一个变量类型最常用的名字），#define 常量由数学和图形界面模块所使用；其中的一个例子是无理数 π 。

1.1.2 GLOBALS.H

GLOBALS.H 头文件（表 1-2）定义数学和图形界面模块的全局变量。模块和主程序代码存取的这些全局变量主要集中在图像分辨率和三维图形例程。

表 1-2 GLOBALS.H 头文件

```

/*
    GLOBALS.H Header File
*/

int      XRes, YRes;
Word     MaxXRes, MaxYRes;
Word     MaxX, MaxY;
float     Asp;
Boolean   PerspectivePlot;
float     Mx, My, Mz, ds;
Boolean   Draw__Axis__And__Palette;
int       Angl, Tilt;

```

1.1.3 MATHB.H

MATHB.H 头文件（表 1-3）包含了在数学模块中定义的所有数学函数的函数原型。函数原型是 ANSI 标准 C 的一个特性，它指定在数学模块中找到的函数的名字以及它们的参量。这些原型是找到函数参考的好地方，因为模块中所有函数都在这里列出。

表 1-3 MATHB.H 头文件