

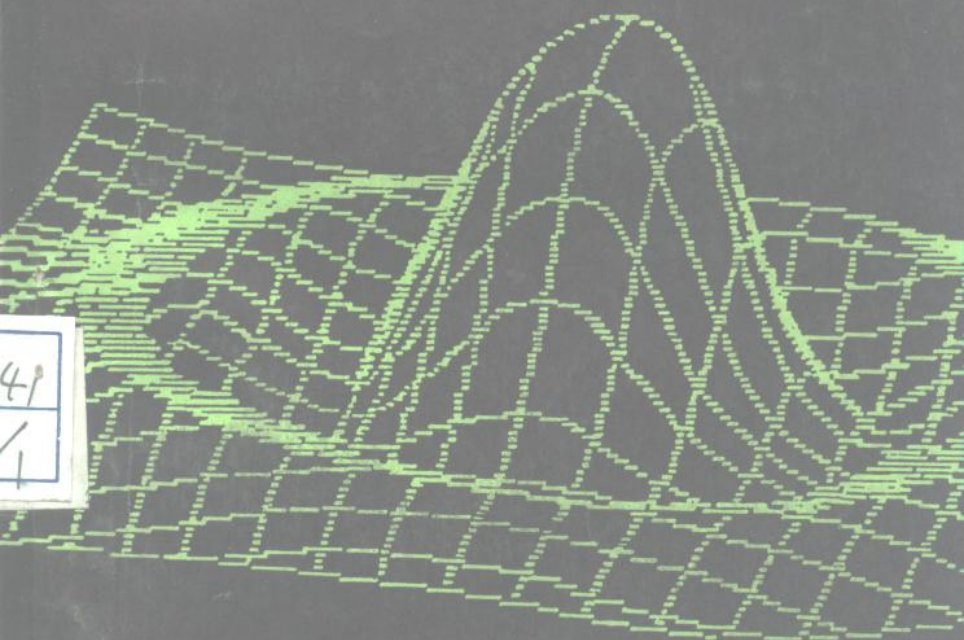
IBM PC

GAOJI ZUOTU YUANLI

[英] I.O.安格尔著 王世迪等译 / 重庆大学出版社

IBM PC

高级作图原理



41
1

TT391.61
AGE/1

IBM PC高级作图原理

(英) I.O. 安格尔 著

王世迪 等 译



15738

重庆大学出版社

IBM PC高级作图原理

(英) I.O.安格尔 著

王世迪 等译

责任编辑 谭 敏

•

重庆大学出版社出版发行

新华书店经销

重庆大学出版社印刷厂印刷

•

开本: 787×1032 1/32 印张: 13.375 字数: 301 千

1988年12月第一版 1988年12月第一次印刷

印数: 1—4600

ISBN 7-5624-0107-1

TP·7

定价: 2.69元

内 容 提 要

本书全面地介绍了计算机图形学的基本概念、基本原理和计算方法。全书共分十三章，分别讨论了IBM PC的作图指令；空间坐标与显示屏象素点的关系，二维、三维几何原理和空间变换，消隐面算法，正投影、透视投影和体视投影，以及字符作图和数值图表等图形学的各个主要领域的内容。

全书给出了大量的程序和例题。全部程序都在IBM PC上用版本A2.00的BASIC实现，不需另外配置其它硬件设备。所有程序都采用模块化结构。只需要一定的解析几何知识就可以很好地掌握本书内容。

本书可以作为计算机图形学的教材，或从事有关工作的科技人员的参考书，也可以作为一本计算机作图程序手册使用。



译者序

计算机图形学 (Computer Graphics) 在计算机辅助设计CAD、计算机辅助教学、生物医学中的形态学科以及计算机图象发生CIG等各个领域都有十分广泛的应用。已经成为计算机科学中一个重要的发展方向。由于人们所获取的信息中有70%以上来自眼睛, 所以利用计算机产生图形或对图形进行各种变换处理, 对帮助人们获取信息、提高获取信息的效率都有十分重要的意义。但计算机图形处理对硬件设备的运行速度和内存容量都有很高的要求, 并且需要一定的理论基础, 因而限制了推广应用。本书较好地解决了这些问题, 选用广泛使用的IBM PC作为硬件设备, 不需要另外配置其它设施。本书所讨论的内容深入浅出, 从最基本的概念开始, 逐步深入到图形学中的各个领域, 包括通用消隐面算法等这样一些很深的课题, 并给出了大量的程序和例题, 具有较高的理论价值和实用意义, 可以作为一本计算机图形学的教材或参考书; 也可以作为一本计算机作图程序集或手册使用。

本书内容共分十三章。第一章介绍IBM PC版本A2.00 BASICA的作图指令。第二章讨论实际空间坐标系和显示屏象素点的关系。这两章是本书的基础。第三章和第四章讨论二维空间的几何原理和空间变换。与此对应, 第七章和第八章讨论了三维空间的几何原理及空间变换。这几章着重讨论建立物体图形, 移动图形到画面中的真实位置, 以及在不同的位置观察图形等这样一些图形学的重要概念。第九章正投

影和第十一章透视投影和体视投影是构成实际图形所必需的，也是图形处理中的难点。第十章和第十二章讨论了消隐面（线）算法，这是图形学中很深的内容。第五章和第十三章介绍字符作图和产生动画的方法。第六章相对独立，给出了多种绘制图表的程序。可以根据需要阅读有关章节。图形学的内容也可分为二维和三维两部分，如只需掌握二维图形的处理，即可以阅读第一到第四章。而三维图形部分则需要掌握第一、二章之后，阅读第七章到第十二章的内容（如能参考第三、四章的内容也会有一定的帮助）。只对数值图表感兴趣的可以只看第五、六两章。阅读第五章、第十三章可以了解字符产生、字符作图、动画产生等方面的内容。

本书给出了大量的程序，全部采用 BASICA 版本 A2.00 编写，容易掌握。如果感到运行速度较慢，最好能采用编译 BASIC，这样程序基本无需改动。本书全部采用模块化结构程序，这样对改进程序、增加功能以及调试程序都有很大好处。一般格式为：主程序调用一个建立图形子程序，有些主程序还需要再调用一个作图子程序。作图过程中每个程序（或子程序）都需要调用公共作图子程序库程序。这种公共作图子程序库共有 3 个，参见附录 B。

本书在讨论空间变换时采用了绝对坐标系的概念，观察者所处的位置被变换到绝对坐标系中规定的特定位置上，被观察物体的位置只在绝对坐标系中变换。明确提出被观察物体的建立图形位置、在画面中的真实位置和观察位置这样 3 个有特定意义的位置，对理解空间变换的意义和目的有独到之处。特别是以不同观察位置观察同一幅画面，或在一幅画面中观察多个相同的物体，对简化程序和加快程序的运行速度有很大好处。

本书着重讨论了图形处理的计算方法，避免了过多的数学推导。并给出了大量的程序和例题帮助理解，有利于掌握图形处理的基本思想和方法，对于以后改进程序和自行编制更复杂的程序提供了很好的机会。学习本书内容需要一定的平面解析几何和空间解析几何知识，以及一些拓扑学的基本常识。当然，希望已经比较熟悉 IBM PC 的 DOS 指令和 BASIC 编程技术。

本书中的程序可分为 4 类：完成一个作图问题的主程序。主程序都是从第 100 行开始，执行主程序需要并入相应的作图子程序库（见附录 B），一般还需并入建立图形子程序和（或）作图子程序；统一从 6500 行开始的“建立图形”子程序，建立一个物体的图形，有些还可以画出物体的图形；专门用于画出图形的作图子程序‘drawit’统一从 7000 行开始；最后一种是一些相对独立的程序，其中有些本身已包含很多子程序，一般在执行时只需并入由一个作图子程序组成的子程序库 lib0。

为了便于调用作图所需的各种共用的作图子程序，组成了 3 个作图子程序库：第七到第十二章中三维作图用的 lib3，第二到第四章中二维作图用的 lib2 和第五、六、十三等章中一些独立程序所需的，仅由一个设置彩色显示屏子程序‘Colormonitor’组成的 lib0。作图子程序库由若干个子程序组成。请查阅附录 B。

执行本书中所有程序的硬件要求是带有彩色图形适配器和彩色显示屏的 IBM PC 或 IBM PC/XT。要求内存存在 96k 字节以上。采用 IBM PC 版本为 A2.00 的解释型 BASICA。

为了在图形显示模式状态下硬拷贝（Shift PrtSc）显示屏内容，在进入 BASICA 以前必须先执行 DOS 指令

GRAPHICS。这是一个DOS外部命令，在DOS 2.00系统磁盘上。适用的打印机为FX-100，FX-80，RX-80等采用EPSON标准的打印机。

本书统一规定在图形显示状态下建立的可变字符集（ASCII码128-255）在段址&H1700中偏移量0处（10进制绝对地址92k处，占1k字节）开始存放。如果在DOS盘上有GRAFTABL文件，也可以执行这个DOS指令，装入可变字符集，并存放在BASIC段以前的DOS区内。这时需删去有关程序行中DEF SEG = 0；后的POKE127，&H17指令。如还需自己定义可变字符集，可设置中断&H1F的中断向量（段址0中的124-127），指向开始存放可变字符集的地址。请参看第五章。

本书附录A：执行各个主程序时所需调用的子程序清单及建议选用的参数表。

附录B：公共作图子程序库的组成表。

附录C：本书所有程序的清单。

本书由江正同志译第一、二章，邱贤菊同志译三、四章，孟民果同志译第五、六章，王世迪同志译七、八、九、十、十一、十二、十三章，最后全书由王世迪同志统一校定。由于译者水平有限，错误在所难免，恳切希望专家及广大读者给予批评指正。

译者

1987年9月20日

绪 论

阅读本书可以有各种不同的深度。首先,如果只是简单地希望在IBM PC上画出各种图形,那么本书可以作为一本图形程序手册,在绘制图形的过程中很自然地会希望更深入地掌握本书的具体内容,以了解这些程序是如何编制的,为什么这样编制;其次,书中绘制数值图表和标记图标等一些程序(扇形图,直方图,曲线图等)可以作为在办公事务和实验室工作中实际应用的程序包使用。最后,也是本书的主要目的是作为一本计算机图形学的教材,可以使读者从对图形学的一些初浅的认识,深入到研究如象字符图形、三维物体的图形、消隐面(线)算法等这样一些高深的课题。

本书中很多较为复杂的程序都是由若干独立的子程序组成的,在运算过程中,反复应用了一些基本的算法。因此,在程序的编制及解释过程中采用了自顶向下逐级处理的程序设计方案,也就是模块化结构方法。绘制一个图形的问题可以理解为是解决一系列子问题的过程,而这些问题还可以进一步由一些分解得更细的问题(模块)来解释。这些模块设计为BASIC语言中的子程序。每一个子程序都有一个程序名(在单引号中的小写字母程序名,如'scene 3'),解决一个特定的问题。由这些程序所组成的一个完整的程序解决具体的作图问题。每一个子程序都是解决一种子问题的算法,子程序的程序名是这种算法的简单说明。小写字母只用于程序中的注释和程序名(和由若干子程序组成的完整程序的名称),而程序中的其它变量名和数组名则统一用大写字母表示,以防混淆。

为了方便读者，可以提供包括本书全部程序在内的磁盘，以及包括象素图形块集和可变字符集的数据文件（这些程序和数据文件也可以由读者自行输入和定义，但这是相当费时间的）。

如果只希望运行这些程序，附录A有一个完整的程序清单及适用的数据。在学习每一章的内容前，先从磁盘装入（LOAD）和（或）并入（MERGE）这一章的程序，并运行这些程序，观察结果，是一种很好的学习方法。

程序中有大量的注释（REMARK）。然而有些程序已经接近IBM PC存储量的极限，在这种情况下可以删除这些注释程序行，并用另一个文件名存入，或存入另一张磁盘。

在本书中统一规定在每个（子）程序最前面行号以0结尾的注释程序行说明程序名，并指示程序从此开始。紧接着下面行号以1结尾的注释行**in**列出执行该程序所需的输入参数。再下面行号以2结尾的注释程序行**out**列出执行程序后的输出参数。除了以上3行特殊的注释外，程序中所有其他的注释语句行的行号都以9结尾。

程序中的注释程序占用了大量的存储空间，并降低了程序运行的速度。在这种情况下，可以用程序I.1来删去各个程序中的注释程序行。本书中统一规定所有程序开始的行号不小于100。因此，当把程序I.1并入其他程序时，一定在原程序的前面。执行并入了I.1的程序，将删除原程序中除了行号以0结尾的注释行外的所有注释程序行，然后程序要求输入一个新的磁盘文件名，以存入删除了所有注释和程序I.1的程序，以便在以后使用。

在掌握程序I.1的原理以前，首先要了解在IBM PC内存中开始存放BASIC程序的地址，BASIC段址中的&H30

和 &H31 分别是这个地址的低位和高位字节 (偏移量)。

Program I.1

- 1 'MERGE this program to the front of a target program which contains REMarks to be delete. All target lines with numbers ending in zeros are executable. The rest are REMarks.
- 2 DEF SEG
- 3 'INPUT the name of the file which will hold the deREMed listing.
- 4 PRINT " Type name of disk file for deREMed listing" ; "e.g. NOREMS. BAS" : INPUT FILE\$
- 5 'locations PLR% and PTR% + 1 of the BASIC segment hold the lo-and hi-byte values of the location of a program line. Initially the first program line. Store these two bytes.
- 6 PTR%=&H30 : STO1%=PEEK(PTR%+1) : STO0% =PEEK(PTR%)
- 7 '129 signifies the END of the derem program in the memory. Program lines that follow are from the target program.
- 8 IF PEEK(PTR%+4)<>129
 THEN PTR%=PEEK(PTR%+1) * &H100 + PEEK(PTR%) : GOTO 8
- 9 PTR%=PEEK(PTR%+1) * &H100 + PEEK(PTR%)
- 10 'TARGETPTR% and TARGETPTR%+1 hold the start of target program. PROG PTR% (and+1) will be usee to over-write the target program locations, with program lines that are not RE-Marks.

```

11 PROGPTR%=PTR% : TARGETPTR%=PTR%
12 'PTR% (and +1) point to the present line under
    consideration, NEXTPTR% ( and +1) point to the
    next program line.
13 NEXTPTR%=PEEK (PTR% + 1) * &H100+PEEK
    (PTR%)
14 'NEXTPTR% is zero at the end of target program.
15 IF NEXTPTR%=0 THEN GOTO 36
16 'find the line number of the program line being
    considered.
17 LINENUM%=PEEK (PTR% + 3) * &H100+PEEK
    (PTR%+2)
18 PRINT "considering line number" ; LINENUM%
19 'if LINENUM% doesn't end in a zero it is a
    REMark.
20 IF LINENUM% MOD 10 <> 0 THEN GOTO 33
21 'program lines are moved up the memory, over-
    writing other lines of the target program that
    have already been checked.
22 LINESIZE%=NEXTPTR%-PTR%
23 'store the address of the next program line. That
    is add the size of the program line to the present
    address.
24 POKE PROGPTR%, (PROGPTR%+LINESIZE%) MOD
    &H100
25 POKE PROGPTR%+1, INT((PROGPTR%+LINE-
    SIZE%)/&H100)
26 'move up the locations hold program line.
27 FOR I%=2 TO LINESIZE%-1
28 POKE PROGPTR%+I%, PEEK(PTR%+I%)

```

```

29 NEXT I%
30 'go on to next line in deREMed program.
31 PROGPtr%=PROGPtr%+LINESize%
32 'go on to next line in the target program.
33 PTR%=NEXTPtr%
34 GOTO 13
35 'mark the end of the program with two zeros.
36 POKE PROGPtr%, 0
37 POKE PROGPtr+1, 0
38 'the start of program pointers show the deREMed
   program.
39 POKE &H30, TARGETPtr% MOD &H100
40 POKE &H31, INT(TARGETPtr%/&H100)
41 'save the deREMed program.
42 SAVE FILE$
43 'reset the start of program pointers.
44 POKE &H30, ST00% : POKE &H31, ST01%
45 END

```

程序中的有些变量名很长，必然会降低程序的效率，书中往往也不考虑程序中子程序的多少。这是因为一个程序主要是表达一种算法，而执行的速度乃是第2位的。甚至程序行中为了清晰起见还留了不少空格。如果在实际应用中希望使用一个较为实用的程序，那么可以缩短一些意义不大的变量名，减少程序行中留的空格，合理组织程序的结构，以提高程序的效率。如，调用一些不太长的子程序（如‘lineTo’），可以把程序组织得更简洁、紧凑。

程序I.2是绘制一个酒杯这种旋转体线条框图的主程序，见图I.1。这是本书中很多比较复杂的程序中的一个例子。

Program I.2

```
100 'scene3 / goblet
110 GOSUB 9700 'start
120 DIM X(24),Y(24),
    XD(12), YD(12)
130 DIM A(4, 4), B(4,
    4), R(4, 4)
140 DATA 0, -8, 4,
    -8, 4, -7.5, 0.5,
    -7.5, 0.5, -1, 3,
    -0.5, 4, 1, 7, 7,
    6.5, 7, 3.5, 1, 2.5,
    0, 0, 0
```

```
150 RESTORE 140
```

```
159 'generate definition
    set.
```

```
160 NUMV%=11 : N1%=NUMV%+1
170 INPUT "Type number of horizontal lines" ,NUMH%
180 INPUT "Type initial angle" , PHI
190 FOR I%=1 TO N1%
200 READ XD(I%), YD(I%)
210 NEXT I%
220 GOSUB 9300 'idR3
230 GOSUB 8200 'look3
240 GOSUB 60000 'color monitor
250 XMOVE=HORIZ/2 : YMOVE=VERT/2
260 GOSUB 9600 'setorigin
270 GOSUB 6500 'revbod
290 END
```

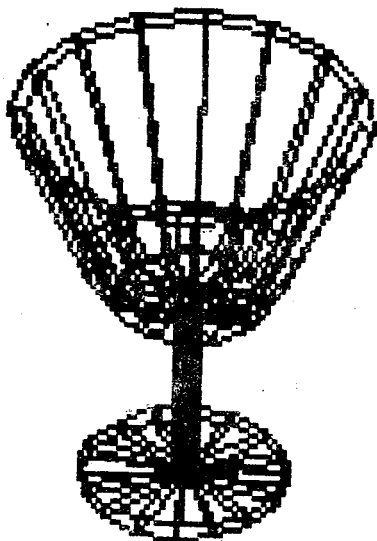


图 I. 1

程序I.2是一个主程序。要执行这一程序画出图形，需并入三维作图子程序lib 3(见附录B)和一个建立图形子程序9.9(‘revbod’)。

在彩色显示屏上执行该程序需首先输入：显示模式MODE%；调色板PALETTE%；背景色BACKGROUND%；前景色FOREGROUND%和页面色PAPER%这5个有关彩色信息的参数。可先分别试用1, 1, 0, 0, 3输入。绘制图I.1还需输入：图形显示水平宽度HORIZ=36，观察点位置 $(EX, EY, EZ) = (1, 2, 3)$ ，观察方向指向 $(DX, DY, DZ) = (0, 0, 0)$ 。以及旋转体水平线段数NUMH%=16和旋转体初位相 $\text{PHI} = 0$ 。所有这些参数都是应答程序的提问分别输入。可以改变参数观察程序运行的结果，如HORIZ=60，其他参数不变；HORIZ=16， $(EX, EY, EZ) = (1, -2, 3)$ ， $(DX, DY, DZ) = (1, 0, 0)$ ；或NUMH%=20， $\text{PHI} = 0.1$ ；还可以试在MODE%=2，PAPER%=1的状态下执行。在学习了第九章后，可以充分理解这些现象。该程序I.2也是本书中所有程序结构格式的一个范例。

希望能在学习本书内容的基础上尽可能多地上机通过本书中给出的例题程序，以掌握计算机图形学的原理和算法。进而自己编制更为复杂的作图程序。

目 录

第一章 IBM PC的作图指令

真实彩色(4) 文本输出(5) 文本状态:显式模式0
(7) 中分辨率状态:显示模式1(9) 高分辨率状态:显
示模式2(10) 图形输出(10) PSET和PRESET指令(11)
显示屏格式化(15) DRAW指令(20) PAINT命令(21)
CIRCLE画图指令(24) GET和PUT(25) WINDOW语句
(33) VIEW(35)

第二章 从实坐标到像素点

用WINDOW的初始子程序(44)

第三章 二维坐标系的几何性质

两条直线的交点(70) 剪辑(73) 曲线的方程表示法和参数
形式(81) 两凸多边形的相交(85) 多边形的纹理(94)

第四章 二维空间变换的矩阵表示法

复合变换(101) 平移变换(103) 比例变换(104) 绕原点的
旋转变换(105) 逆变换(106) 物体的空间位置(107) 观察物体
图形(107) 画出物体图形(108) 复杂图形——图形块构图法
(109) 矩阵的有效利用(118) 贮存图形的信息(122)

第五章 字符图形和像素图形块

字符生成(138) 像素图形块的生成(153)

第六章 数值图表

预处理和监视管理主程序(179) 图标(189) 条样图和直方图
(203) 扇面图(212) 连续的和连续的曲线(215)

第七章 三维坐标几何

直线的定义(227) 两个方向矢量之间的夹角(230) 平面的定
义(230) 直线和平面的交点(231) 点到平面的距离(231) 两直

线的交点(234) 通过3个非共线点的平面(237) 3个平面的交点
(240) 两个平面的交线(244) 曲面的函数表示法(247)

第八章 三维空间变换的矩阵表示法

平移变换(254) 比例变换(255) 绕坐标轴旋转(256) 逆变
换(258) 空间一点绕任意轴 $\vec{P} + \mu \vec{Q}$ 旋转 γ 角(259)

第九章 正投影

正投影(269) 由真实位置到观察位置的变换矩阵 Q 的产生(273)
建立物体的图形(278) 建立物体图形子程序和图形块构图法(279)

第十章 简单消隐线和消隐面算法

三维空间中三角形的方向(298) 单个闭合凸面体的消隐面算法
(300) 旋转体(308) 球体的显示(312) “先后再前”消隐面算
法(画家算法)(319) 绘制特殊数学函数的三维曲面图形(320)

第十一章 透视投影和体视投影

透视(328) 透视图(328) 一个点的透视投影(330) 透视变
换的性质(331) 透视变换程序的编写(334) 透视平面的选取(336)
剪裁(340) 体视投影(341)

第十二章 通用消除消隐线和消隐面算法

第十三章 文本模式作图的高级编程技术

文本显示编辑器(390)

附录A: 作图程序调用子程序清单及选用参数(406)

附录B: 公共作图子程序组成(408)

附录C: 程序清单(409)