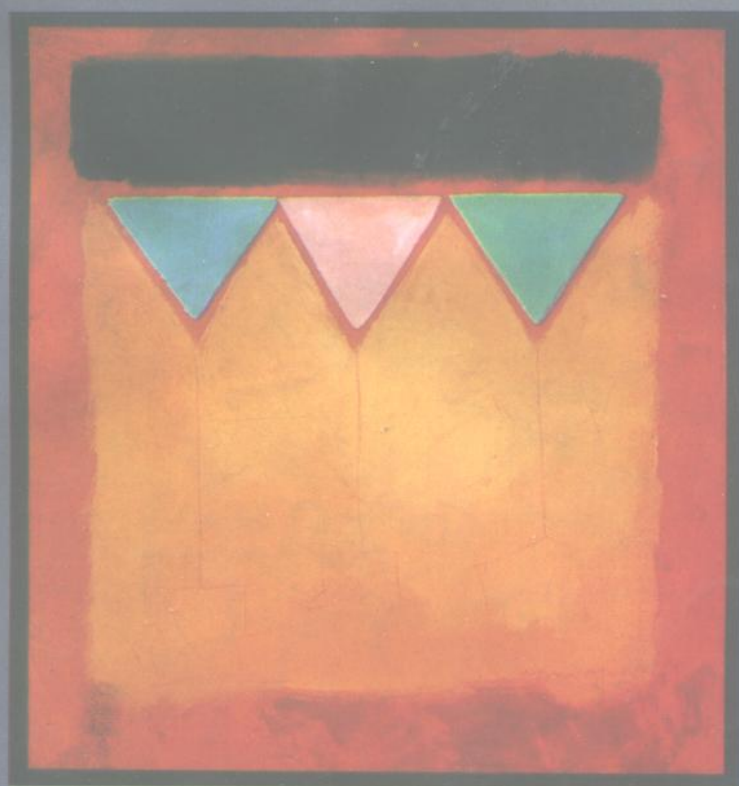


PROGRAMMING FOR
GRAPHICS FILES
IN C AND C++

C和C++的
图形文件编程

〔美〕 John Levine 著
赵英 译

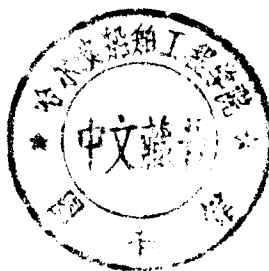


电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

385162

C 和 C++ 的图形文件编程

[美] John Levine 著
赵 英 译



TP391.4
L36

电子工业出版社

内 容 提 要

本书介绍了图形文件的一般概念,并详细论述了十余种文件格式及大图形文件的压缩技术,讲述了在分段(DOS)和虚拟(Macintosh, UNIX, NT 和 OS/2)环境下的内存管理策略。还有专门的章节介绍格式转换的算法及编码,以及如何使文件格式适应于打印机格式。书中给出了大量具有独立平台的C和C++源程序。

Copyright ©1994 by John Wiley & Sons, Inc. All rights reserved.

Authorized translation from English language edition published by John Wiley & Sons, Inc.

本书中文版由美国约翰·威利父子公司授权电子工业出版社独家出版。未经出版者书面允许,不得以任何手段复制或抄袭本书内容。

C 和 C++ 的图形文件编程

[美] John Levine 著

赵 英 译

责任编辑:魏冬

*

电子工业出版社出版

北京市万寿路 173 信箱(100036)

电子工业出版社发行 各地新华书店经销

顺义县天竺颖华印刷厂印刷

*

开本: 787 × 1092 毫米 1/16 印张: 29 字数: 706 千字

1996 年 1 月第 1 版 1996 年 1 月北京第 1 次印刷

印数: 1—4000 册 定价: 55.00 元

ISBN 7-5053-3439-5/TP·1346

著作权合同登记号 图字: 01-1995-436

前 言

从最小的 PC 绘图软件包到最大的超级计算机视觉系统,图形系统在计算机领域中普遍存在。对于有显示价值的图形图象,同样有存储价值。但要存储一幅图象,应用程序就需要有一个图形文件。随着图形系统的发展,大量不同的图形文件格式也应运而生。对于读入和生成图形文件的应用程序来讲,它应能够处理各种不同的格式,这样用户才能够处理这些文件。

●本书的读者对象

使用本书的主要对象是程序员,他们经常需要在应用中输入或输出图形文件。本书还示例了如何打印位映象格式,这个任务与写文件方式非常相似。

对于比较简单的图形格式(除 JPEG 格式外的所有位映象格式),本书包含了按所希望的格式进行文件读写的完整程序。程序员能够在他们的应用中使用这些程序。由于读入和生成 JPEG 格式比较复杂,所以本书没有足够的篇幅给出所有的程序代码。JPEG 库的代码已被存放在书后的一张附带盘上,书中只给出了应用程序与 JPEG 库的接口信息。

程序管理员也会发现这本书是非常有用的。对一个正在做的或将要做的应用程序来讲,当增加其对各种不同的图形格式的支持能力时,程序管理员将会知道要做哪些事情来完成这些工作。

●如何使用本书

在研究程序以前,请首先阅读本书的前五章内容。它们所提供的基本背景知识对理解以后的程序是非常有用的。

除 TIFF 和 JPEG 格式外,所有的位映象编码都基于在第六章到第十章中所开发的 PBM 库。在使用特定的位映象格式前,请阅读一下上述内容。

●本书源程序的来源

本书的大部分程序代码来自于一些可能的公共代码库,这些代码库中的程序已经经过几年的精炼和测试。

有关 MacPaint、PCX、GEM IMG、IFF/ILBM、Targa、GIF 和 TIFF 的应用程序和位映象库取自于由杰夫·波斯坎茨等人所编的 PBMPLUS 库。本书中的程序版本已经被修改,以使其能够象运行在 Unix 和 VMS 操作系统上一样,也能够在 MS-DOS 操作系统上运行,同时还改正了一些错误和不足。(书后附带的磁盘中不仅包括书中所提供的所有程序,而且还包括与 MS-DOS 兼容的全部 PBMPLUS 图形库。)TIFF 库中的程序由硅谷图形公司的萨姆·利弗勒编写。但为了与 MS-DOS 系统兼容,代码的内容有一些小的修改。有关 JPEG 的程序是由汤姆·赖恩所领导的“独立 JPEG 集团”编写。

本书中,关于 BMP 读写程序以及所有的 C++ 程序都是最新的内容。

●读者意见的重要性

所有的作者都希望得到读者的反馈信息。如果您喜欢此书,或者特别是不喜欢此书时,请写信告诉我。我的地址是:

John R. Levine

I. E. C. C

Post Office Box 380349

Cambridge, MA 02238-0349

能够使用电子邮件的读者可将邮件直接送入 Internet 网,地址号为:graphics @iecc.com。

约翰·利威恩
马萨诸塞州坎布里奇市
一九九三年五月

目 录

第一部分 图形文件概述

第 1 章 图形文件是什么?	(3)
1.1 文件编码	(3)
1.2 图象几何	(4)
1.2.1 屏幕坐标	(4)
1.2.2 图形坐标	(5)
1.2.3 其它坐标系统	(5)
第 2 章 图形文件类型	(7)
2.1 位映象文件	(7)
2.1.1 象素的顺序	(7)
2.1.2 扫描行	(7)
2.1.3 位平面	(7)
2.1.4 交叉行	(9)
2.1.5 有损失压缩	(9)
2.2 矢量文件	(9)
2.3 元文件	(10)
2.4 打印机数据文件	(10)
2.4.1 扩充正文格式	(10)
2.4.2 页描述语言	(10)
2.5 文件类型间的转换	(11)
2.5.1 位映象到位映象	(11)
2.5.2 矢量到矢量	(11)
2.5.3 矢量到位映象	(11)
2.5.4 位映象到矢量	(11)
第 3 章 图象的颜色	(13)
3.1 单色与彩色图象	(13)
3.2 彩色模式	(13)
3.2.1 亮度/色度模式	(13)
3.3 颜色映象表	(14)
第 4 章 压缩技术	(15)
4.1 行程编码	(15)
4.2 霍夫曼编码	(16)

4.3	字典系统	(16)
4.4	更复杂的方法	(17)
4.4.1	算术编码	(17)
4.4.2	有损失压缩	(17)
第5章	处理文件时的内存管理	(19)
5.1	处理小型和中型文件	(19)
5.2	在虚拟环境下处理大型文件	(20)
5.3	在 MS-DOS 环境下处理大型文件	(21)
5.3.1	用 C++ 实现虚拟分段像素数组	(21)
5.4	处理部分文件	(26)
5.5	对矢量和元文件的处理技术	(27)

第二部分 位映象文件的结构

第6章	位映象文件结构	(31)
6.1	PBM 方法	(31)
6.2	PBM 格式	(31)
6.2.1	单个位 PBM 格式	(32)
6.2.2	灰度 PGM 格式	(32)
6.2.3	彩色 PPM 格式	(32)
6.2.4	通用格式	(33)
6.3	支持库	(33)
6.3.1	头文件	(33)
第7章	单色图象库	(39)
7.1	PBM 例程	(39)
7.2	内存分配例程	(41)
7.3	程序启动例程	(43)
7.4	错误报告	(46)
7.5	多种支持	(48)
7.6	文件处理	(49)
7.7	整数读/写例程	(51)
7.8	读入 PBM 格式文件	(54)
7.9	初级文件头读入例程	(58)
7.10	生成 PBM 格式文件	(60)
第8章	灰度图象库	(63)
8.1	PGM 头文件	(63)
8.2	PGM 初始化	(65)
8.3	读 PGM 文件	(65)
8.4	生成 PGM 文件	(69)

第 9 章 全彩色图象库	(73)
9.1 PPM 初始化	(75)
9.2 读 PPM 文件	(76)
9.3 生成 PPM 文件	(80)
9.4 颜色映象表	(85)
9.4.1 颜色映象表的数据结构	(85)
9.4.2 颜色映象表建立例程	(87)
第 10 章 通用象素图象库	(95)
10.1 PNM 头文件	(95)
10.2 PNM 初始化	(96)
10.3 读 PNM 文件	(97)
10.4 生成 PNM 文件	(100)

第三部分 行程压缩格式

第 11 章 MacPaint 格式	(105)
11.1 概述	(105)
11.2 读 MACPAINT 文件	(105)
11.3 生成 MACPAINT 文件	(107)
第 12 章 PCX 格式	(111)
12.1 读 PCX 文件	(111)
12.1.1 解压缩已存储的图象数据	(116)
12.1.2 将位平面数据转换成象素	(117)
12.1.3 还原被压缩的象素	(118)
12.2 生成 PCX 文件	(120)
12.2.1 对 PCX 文件编码	(123)
12.2.2 图象编码帮助例程	(126)
12.2.3 压缩位平面数据	(128)
12.3 进一步扩展	(130)
第 13 章 GEM IMG 格式	(131)
13.1 IMG 文件	(131)
13.2 读 IMG 文件	(131)
13.3 生成 IMG 文件	(136)
13.3.1 位的压缩和写入	(138)
13.4 进一步扩展	(140)
第 14 章 IFF/ILBM 位映象图象	(141)
14.1 ILBM 头文件	(141)
14.2 读 ILBM 文件	(142)
14.2.1 读入程序的辅助例程	(151)

14.3 生成 ILBM 文件.....	(152)
14.3.1 生成 ILBM 文件的辅助例程.....	(159)

第四部分 未压缩的格式

第 15 章 Windows 中的位映像	(163)
15.1 通用的 BMP 文件头	(163)
15.2 读 BMP 文件	(164)
15.2.1 对 BMP 文件解码	(165)
15.3 生成 BMP 文件	(169)
15.3.1 对 BMP 文件编码	(171)
第 16 章 Targa 格式	(175)
16.1 Targa 包含文件	(175)
16.2 读 Targa 文件.....	(176)
16.2.1 读 Targa 文件的辅助例程.....	(181)
16.3 生成 Targa 文件.....	(185)
16.3.1 生成 Targa 文件的辅助例程.....	(192)

第五部分 字典压缩格式

第 17 章 GIF 格式	(199)
17.1 生成 GIF 文件	(199)
17.1.1 对 GIF 文件编码	(203)
17.1.2 LZW 压缩方法	(209)
17.2 读 GIF 文件	(218)
17.2.1 对压缩的图象进行解码.....	(224)
第 18 章 TIFF 格式	(233)
18.1 TIFF 的子集和类别	(233)
18.2 TIFF 库	(234)
18.2.1 使用 TIFF 库	(234)
18.2.2 库的头文件.....	(234)
18.2.3 处理图象文件目录(IFD)	(253)
18.3 处理 TIFF 的 IFD	(276)
18.3.1 读 TIFF 的 IFD	(276)
18.3.2 生成 TIFF 的 IFD	(299)
18.4 打开和关闭 TIFF 文件	(315)
18.4.1 关闭 TIFF 文件	(323)
18.5 压缩和编码技术.....	(325)
18.5.1 未压缩方式.....	(330)

18.5.2	PackBits 压缩	(334)
18.5.3	其它压缩方法	(340)
18.6	读 TIFF 数据	(341)
18.7	写 TIFF 数据	(347)
18.7.1	其它内容	(357)
18.8	TIFF 的读入和应用程序	(363)
第 19 章	JPEG 格式	(365)
19.1	JPEG 是什么	(365)
19.1.1	JPEG 格式	(366)
19.2	对 JPEG 进行编码的步骤	(366)
19.2.1	颜色编码	(366)
19.2.2	DCT 转换	(367)
19.2.3	量化过程	(367)
19.2.4	霍夫曼或算术编码	(367)
19.3	使用 JPEG 库	(368)
19.3.1	独立程序	(368)
19.3.2	库代码	(368)
19.3.3	JPEG 压缩	(368)
19.3.4	JPEG 的解压缩	(373)

第六部分 矢量格式

第 20 章	HP-GL 格式	(385)
20.1	读 HP-GL 文件	(385)
20.2	初级 HP-GL 读入程序	(385)
20.3	高级 HP-GL 读入程序	(391)
第 21 章	MS Windows 元文件	(395)
21.1	在 MS WINDOWS 下用 C 语言读写元文件	(395)
21.2	在 MS WINDOWS 下用 C++ 语言读写元文件	(396)
21.3	用 C 语言读元文件	(397)

第七部分 打印机格式

第 22 章	HP 打印机控制语言	(407)
第 23 章	PostScript 格式	(411)
23.1	生成 POSTSCRIPT 文件	(411)
23.2	读 POSTSCRIPT 文件	(425)
23.2.1	读 EPS 预观察数据	(425)

第八部分 格式转换

第 24 章 位映象到位映象:PBM 方法	(431)
24.1 共同使用 PBM 程序	(431)
24.2 位映象转换	(431)
24.2.1 彩色到灰度转换	(431)
24.2.2 颜色量化	(432)
24.2.3 并列调色	(432)
第 25 章 矢量格式到位映象格式:光栅化	(437)
25.1 光栅化库	(437)
25.2 HPGL 格式到 PPM 格式的转换	(444)
25.3 扩展的光栅化程序	(447)
第 26 章 判断文件的格式	(449)
附录 1 参考文献	(451)
附录 2 有关附带盘上的软件说明	(453)

第一部分 图形文件概述

第 1 章 图形文件是什么？

图形文件其实就是一个包含图象的文件。它有许多种类型，从图 1.1 所示的用折线组成的图形到图 1.2 所示的一幅扫描照片。当要显示被存储的图象时，程序应该能够读入图形文件。同样，为了以后使用图象，处理图象的程序也应该能够存储图象。

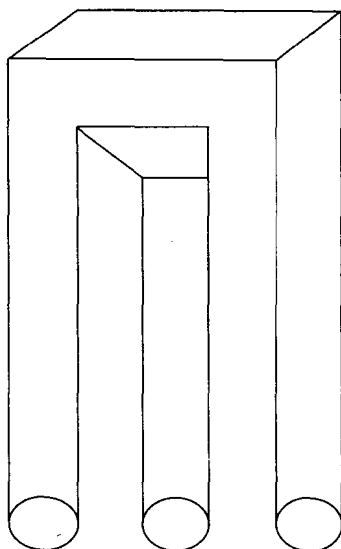


图 1.1 一幅折线图

我们将在第 2 章讨论有关图象的一些基本的编码方法。除了这些文件格式外，还有许多不兼容的文件格式，它们都拥有自己的特性。为了能够处理每一类文件格式，我们首先要从一些基本结构出发，然后再详细论述针对每一个特定文件格式的读写方法。

1.1 文件编码

对一个图形文件进行编码有许多不同的方法。一种极端的方法是编码的信息为可读的 ASCII 正文。这对我们来说，是很容易读懂的，对它进行编辑也是比较容易的。从编程角度来说，ASCII 正文也是比较容易读写的。但它的数量庞大，程序对它的读写速度非常慢。另一种方法是使用压缩的二进制格式，虽然这种格式很紧凑，但与人类所读的信息不相容，并且需要复杂的读写例程。

就图象的大小、形状、颜色及其它属性而言，它的灵活程度不同，图形存储格式也有很大差异。对于 MacPaint 文件格式而言，它只有唯一的存储格式，且仅具有两种颜色。而对于 TIFF 格式（带标记的图象格式），它有很多的选项和特征，目前还没有一个 TIFF 应用程序

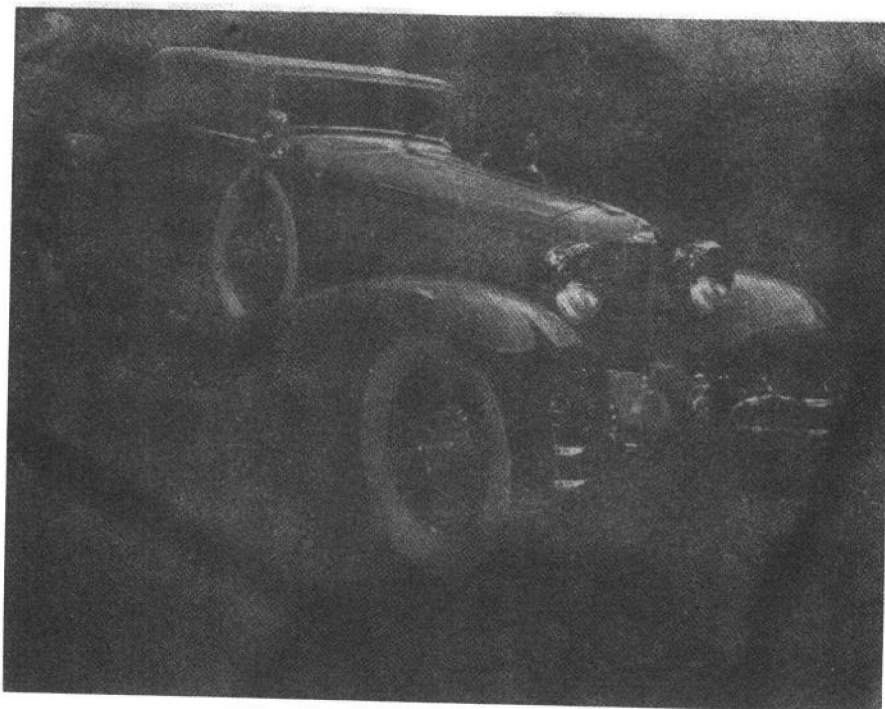


图 1.2 一幅扫描照片

能够读取所有的 TIFF 文件。

象我们将在第 2 章中所看到的一样,文件格式被划分成几种基本类型。最通用的格式为位映象格式,它用一个矩形点阵数组来描述一幅照片,使人想起电视图象的画面。一种不很常用的格式为矢量格式,它用一系列的线段和弧来描述图象。复杂的图形系统提供图元文件,此文件给出了图形系统中各种命令的列表。对于当今复杂的激光打印机而言,图象输出所需要的数据流也都拥有自己的格式。

对于每一种格式类型,都存在着一系列方法将其转换成其它格式类型。

1.2 图象几何

为定位图象中的元素,每一幅计算机图象都有它的内部几何结构。幸运的是图象几何结构的种类并不很多。它们都使用一个“图形页”模型,图形中的位置都用相对于原点的 x 和 y 坐标来描述。 x 坐标是相对于原点的水平距离, y 坐标是相对于原点的垂直距离。使用这种模型的两个最通用的坐标版本为屏幕坐标和图形坐标。

1.2.1 屏幕坐标

当在屏幕上显示图象时,最常用的坐标是图 1.3 所示的屏幕坐标。一般来讲,坐标用括号中的内容来表示即 (x,y) 。(0,0) 点为图象的左上角, x 坐标的方向是从原点向右递增, y 坐标的方向是从原点向下递增。例如,点(1,0)在原点的右边,点(0,1)在原点的下边。

通常来讲,x 方向的距离单位与 y 方向的距离单位是一样的。但在有些情况下,它们的单位尺度是不一样的。造成这种情况的原因大部分是由于存储屏幕图象,这些屏幕会造成图象的变形,例如 IBM CGA 或 VGA 屏幕和以前的 Amiga 屏幕。

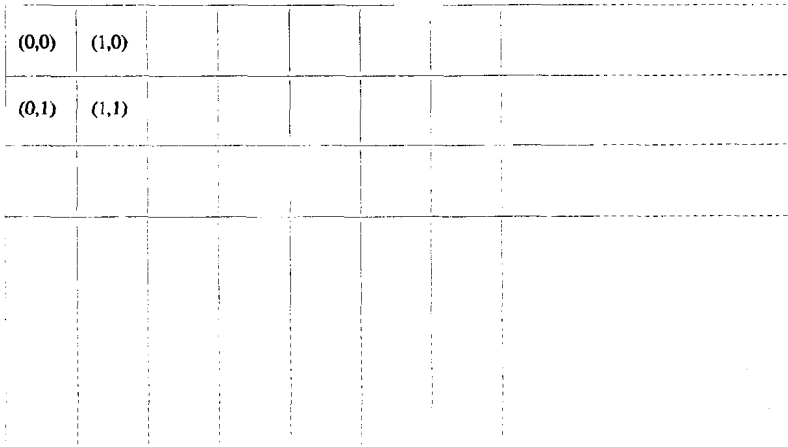


图 1.3 屏幕坐标

1.2.2 图形坐标

当将一幅图象打印到一张纸上时,通常使用图 1.4 所示的图形坐标。原点在图象的左下角,x 坐标的方向是从原点向右递增,y 坐标的方向是从原点向上递增。

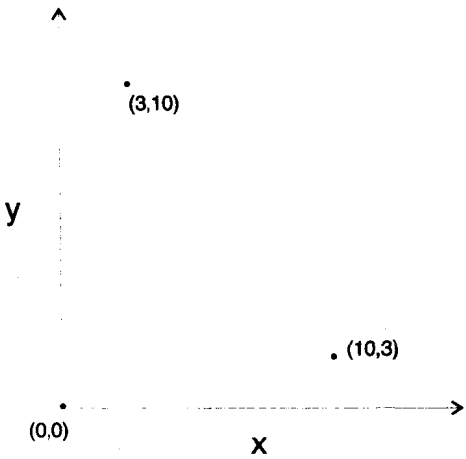


图 1.4 图形坐标

1.2.3 其它坐标系统

有时,你会发现一幅图象的原点会在其它位置,例如在图象右上方或在图象的左下方。象 PostScript 文件和 HP-GL 文件一样,程序员可随意设置文件格式所使用的坐标系统。x 的

递增方向既可向左也可向右, y 坐标的方向既可向上也可向下; 原点也可放置在图象的任意位置, 这包括图象以外的区域。PostScript 格式甚至允许坐标旋转以及比例不同的 x 坐标和 y 坐标, 这样它的 x 轴和 y 轴就有可能指向任何方向, 而不一定要垂直。

很幸运, 除了产生象旋转正文这样美观的 PostScript 效果外, 这些特殊的坐标系统并不是非常有用的, 所以我们在这里不再进一步讨论它们。