

“十五”国家重点电子出版物规划项目 · 计算机知识普及和软件开发系列

Visual C++ OpenGL DirectX 三维动画 编程宝典

北京希望电子出版社 总策划
万斌 等 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

“十五”国家重点电子出版物规划项目·计算机知识普及和软件开发系列

Visual C++ OpenGL DirectX 三维动画 编程宝典

北京希望电子出版社 总策划
万斌 等 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

这是一本关于用 Visual C++ 平台下的 OpenGL 和 DirectX 实现三维动画的方法和技巧的专著。

本书的特点是以实例的形式由浅入深地详细地讲解。本书共分为两部分共 18 章和 2 个附录。第 1 部分：关于 OpenGL 的 10 个例子的讲解。内容为制作 OpenGL 黑屏窗口框架程序；绘制无色的三角形和正方形；绘制彩色的三角形和正方形；制作彩色旋转的三角形和正方形；制作彩色旋转的金字塔和正方体；制作旋转的带纹理贴图的正方体；实现纹理过滤器、基本的光照和键盘控制等特效；制作透明的带纹理贴图的可键盘控制的正方体；制作美丽的星空；用 3D 实现的在屋子里移动。第 2 部分给出与 OpenGL 前 8 个例子相对应的 DirectX 8.0 例子（应用 DirectX 8.0 SDK），内容为制作不断变换背景颜色的 DirectX 窗口；制作黑白的三角形和正方形；制作彩色的三角形和正方形；制作彩色旋转的三角形和正方形；制作彩色旋转的金字塔和正方体；制作带纹理贴图的旋转的正方体；制作纹理过滤、光照效果以及键盘控制；制作透明效果。附录 1 为 OpenGL 指令集，附录 2 为安装 DirectX 8.0 SDK。

每章的代码都是基于第 1 章的代码，并在前一章代码的基础上进行增删而成的，易于理解。代码特点是由浅入深，内容逐步增加。

本书适合初、中级三维动画相关方面程序开发人员，高等院校学生、计算机编程爱好者阅读参考。

说明：有关书中实例的源代码，请从 www.b-xr.com 下载 3990.zip。

系 列 书 名： “十五”国家重点电子出版物规划项目·计算机知识普及和软件开发系列

书 名： Visual C++ OpenGL DirectX 三维动画编程宝典

总 策 划： 北京希望电子出版社

文 本 著 者： 万 斌 等编写

责 任 编 辑： 李东震

出版、发行者： 北京希望电子出版社

地 址： 北京市海淀区知春路甲 63 号卫星大厦三层 100080

网址： www.bhp.com.cn

E-mail: lwm@bhp.com.cn

电话： 010-62520290,62521724,62528991,62630301,62524940,62521921,82610344（发行），010-82675588-202（门市），010-82675588-501,82675588-201（编辑部）

经 销： 各地新华书店、软件连锁店

排 版： 希望图书输出中心 周宇

文 本 印 刷 者： 北京媛明印刷厂

开 本 / 规 格： 787 毫米×1092 毫米 1/16 22 印张 501 千字

版 次 / 印 次： 2003 年 1 月第 1 版 2003 年 1 月第 1 次印刷

印 数： 1~3000

本 版 号： ISBN 7-89498-012-9

定 价： 35.00 元

说明：凡我社产品如有残缺，可执相关凭证与本社调换。

前 言

阅读本书需要一定的 Visual C++ 编程经验，及 OpenGL 和 DirectX 有基本了解。

本书分为两个部分。

第一篇：

关于 OpenGL 的 10 个例子的讲解。第 1 至第 8 章为 3D 编程基础，第 9 章和第 10 章为提高。

每章的代码都是基于第 1 章的代码，并在前一章代码的基础上进行增删而成的。所以易于理解。代码特点是由浅入深，内容逐步增加，使读者可以迅速掌握要领。

OpenGL 是 1992 年由 SGI 公司发布。1995 年集成到 Windows NT，后来被 Windows 后续版本所支持如（Windows 95 OSR2、Windows 98、Windows ME、Windows 2000 等）。特点是跨平台跨操作系统易于移植，支持 UNIX、Mac OS、Linux，SGI 图形工作站等。

Windows 下 OpenGL 支持 Visual Studio、Delphi、Boland C++ Builder 等开发工具。

第二篇：

给出与 OpenGL 前 8 个例子相对应的 DirectX 8.0 例子（应用 DirectX 8.0 SDK）。

DirectX 是由微软发表，目的是编写高性能的，直接对硬件编程的图形动画开发环境，支持 2D、3D 图形动画编程，与 OpenGL 不同的是，DirectX 8.0 还支持声音、游戏杆、网络游戏、多媒体音、视频等的支持，但不支持视窗以外的操作系统。微软最近的 X-BOX 游戏机也支持 DirectX。

Windows 下 DirectX 8.0 支持 Visual C++、Visual BASIC、Boland C++ Builder。

读者可根据 OpenGL，DirectX 各自特点，并通过对本书的学习，根据应用程序的目的，得出自己编程的选择。

本书的编写参考了 Jeff Molofee's NeHe OpenGL 教程及 Drunken Hyena Direct 3D 教程。

序

关于 OpenGL

一、什么是 OpenGL

OpenGL 应该被严格的定义为“图形硬件的软件接口”。基本来说，它是一个方便的和快速的三维图形和模型库。使用 OpenGL，可以创建接近虚拟光线跟踪质量的精彩和漂亮的三维图形。应用 OpenGL 的最先进之处是与光线跟踪相比较它有极高的速度。它的算法是由 Silicon Graphics, Inc.(SGI)开发及优化的。

OpenGL 愿意使用可以显示和处理经过设计及优化的三维图形的计算机硬件。现在，越来越多的 PC 图形硬件开发商推出了许多增加三维硬件加速的产品，使得应用三维图形加速显示卡的 PC 越来越普遍了。

OpenGL 的用途也十分广泛，从工程设计和建筑设计到一鸣惊人的电影里由计算机生成的恐龙。随着三维图形加速卡及高速 PC 微处理器的不断普及，三维图形将不仅仅应用于科学和游戏方面，它会很快成为消费者及商业应用的典型部分。

二、OpenGL 的历史

OpenGL 是一个刚出现几年的并赢得了大量跟随者的新的工业标准。OpenGL 的先驱是 Silicon Graphics 的 GL。'IRIS GL' 是 SGI 公司高端图形工作站的三维图形程序接口。这些不是通常用途的计算机，它们是经过特殊的硬件优化并被用于显示精致图形。其硬件支持极高速度的矩阵变换（三维图形的先决条件），支持深度缓冲区及其它功能。当 SGI 试着将 IRIS GL 移植到其它的硬件平台时，问题发生了。

OpenGL 是 SGI 增加 IRIS GL 的移植功能的结果。新的名字代表既有 'GL' 的能力，又有开放性 'Open'，即允许很容易地适用于其它的硬件平台及操作系统(SGI 还在维护 IRIS GL，但其功能的增强没有错误的产生多)。

OpenGL 1.0 是 1992 年发表的。五年后，在 Win32 开发者会议上，SGI 展示了 OpenGL 在他们的 IRIS Indigo 硬件上的运行效果。视频片段从电影终结者二：审判日，和医学图象的应用吸引了展览大厅里的参观者。这时，SGI 和微软早已一起将 OpenGL 放在了下一代的 Windows NT 中（即 Windows2000）。

OpenGL 在 1995 年被集成到 Windows NT，后来被 Windows 后续版本所支持如（Windows 95 OSR2、Windows 98、Windows me、Windows 2000 等）。

Windows 下 OpenGL 支持 Visual Studio、Delphi、Boland C

关于 DirectX 8 SDK

Microsoft®开发DirectX的首要目的就是要提高Windows环境下游戏开发的容易性。利用DirectX，游戏开发人员就可以获得DirectX的硬件独立性的优点，同时又可以直接访问硬件。DirectX的最主要的目标就是提供MS-DOS中直接的硬件访问特性，同时去掉个人电脑中再添加新的硬件时所带来的硬件识别问题。

因为硬件的升级是不可避免的，所以，DirectX实际上也提供了另一种即插即用的方法。因此，利用DirectX可以开发出高性能实时的应用程序，可以直接访问计算机中的硬件和将来系统中会具有的硬件设备。DirectX在硬件和应用之间提供了一致的接口以减少安装和配置的复杂性，并且使硬件的利用达到最优。利用DirectX提供的接口，程序员能充分利用硬件的特性而不需要考虑其具体细节。一个优秀的基于Windows的高性能游戏应该能够利用系统的如下技术：

- 为提高性能而专门设计的加速卡

- 即插即用PnP（Plug and Play）

- 内嵌于Windows的通信服务，包括DirectPlay

1. DirectX 8.0 的组成

（1）DirectX Graphics

DirectX Graphics 将以前版本的 DirectX 里的 DirectDraw，Direct3D 混合在一起成为一个单独的应用程序接口（API），使你可以将它应用在所有图形编程中。这个部分包括了Direct3D 应用程序库，这样做简化了许多图形编程任务。

（2）DirectX Audio

DirectX Audio 将以前版本的 DirectX 里的 DirectSound 和 DirectMusic 混合在一起成为一个单独的应用程序接口（API），使你可以将它应用在所有声音编程中。

（3）DirectInput

DirectInput 广泛支持各种输入设备，包括对力反馈技术的完全支持。

（4）Microsoft DirectPlay

支持多人网络游戏。

（5）Microsoft DirectShow

支持高质量采集和回放多媒体流。

（6）Microsoft DirectSetup

支持对 DirectX 组件的一次性安装。

2. DirectX 8.0 新增内容

(1) 将 DirectDraw 和 Direct3D 完全的集成在一起。

Microsoft DirectDraw 和 Microsoft Direct3D 被融合成一个单独的 DirectX Graphics 组件。这个应用程序接口被扩展更新使其更容易使用并且支持最新的硬件。

(2) DirectMusic 和 DirectSound 更加的集成。

Microsoft DirectMusic 和 Microsoft DirectSound 早已被 DirectX 7.0 很紧密的集成在一起，波形文件（WAV）或资源可以被 DirectMusic 读取，并通过 DirectMusic 的功能进行播放，并与 MIDI 同步。

(3) DirectPlay 的升级

The Microsoft DirectPlay 组件被升级以增加它的性能并增进其易用性。通常情况下，DirectPlay 现在支持两个玩家之间的语音通信。

(4) DirectInput 的升级

Microsoft DirectInput 介绍了一个主要的新功能——动作映射。动作映射允许你在输入动作和输入设备之间建立一个连接并且不依赖在通常的设备对象上的扩充。这简化了输入循环并减少了对客户游戏驱动、设备外形和客户在游戏里的用户配置界面的需求。

(5) DirectShow 被 DirectX 所包含

Microsoft DirectShow 现在成为了 DirectX 的一部分并进行了升级。

(6) 调试构建 Debug build

你可以使用 DirectX 控制面板应用程序打开或关闭调试和零星构建 DirectInput, Direct3D, 和 DirectMusic。为了实现这个功能，在安装 SDK 时选择调试 debug 选项。这个选项将调试 debug 和基本的 DLL 程序（运行 DirectX 8.0 应用程序的最低要求）装到你的系统里。Retail 选项仅安装基本的 DLL 程序。

设计与制作优秀教科书



CX-83263
定价: 100.00 元
ISBN: 7900056424



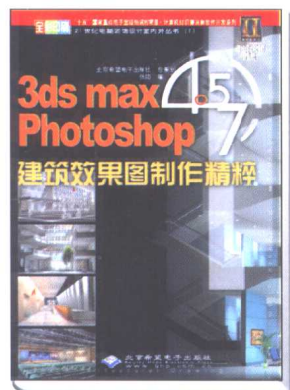
CX-3938
定价: 68.00 元
ISBN: 7900118746



CX-83346
定价: 68.00 元
ISBN: 7900056319



CX-3993
定价: 68.00 元
ISBN: 7894980145



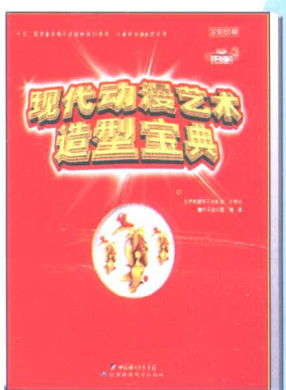
CX-3918
定价: 59.00 元
ISBN: 7900118543



CX-82891
定价: 55.00 元
ISBN: 7900031146



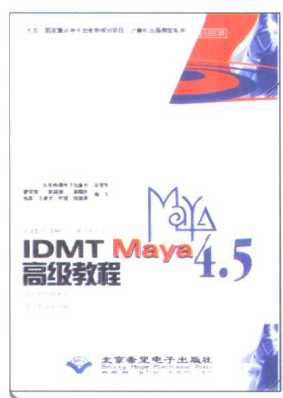
CX-3892
定价: 58.00 元
ISBN: 7900118241



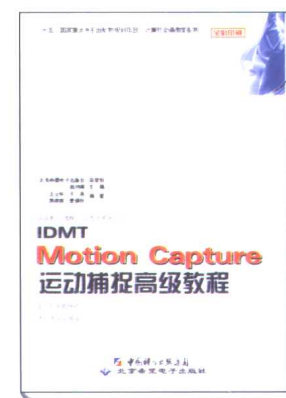
CX-3672
定价: 50.00 元
ISBN: 7900088504



CX-3776
定价: 68.00 元
ISBN: 7900101144



CX-3846
定价: 68.00 元
ISBN: 7900101489



CX-3730
定价: 50.00 元
ISBN: 7900088954



CX-3841
定价: 88.00 元
ISBN: 7900101349



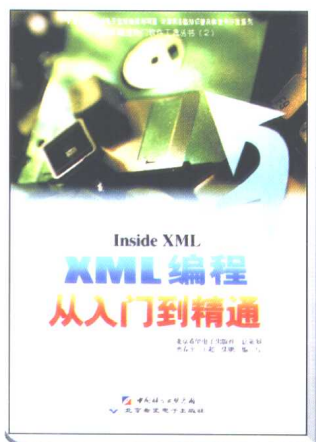
北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀知春路甲 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱(100080)
电话: (010) 82675588(总机) 传真: (010) 62520573

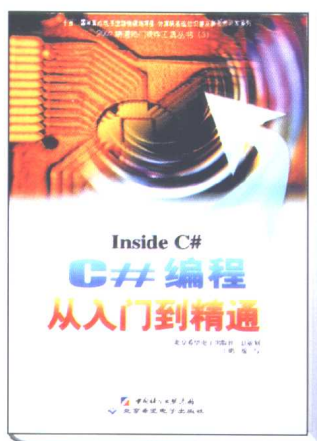
专业编程人员成长之路



CX-3747
定价: 39.00 元
ISBN: 7900101012



CX-3660
定价: 35.00 元
ISBN: 7900088490



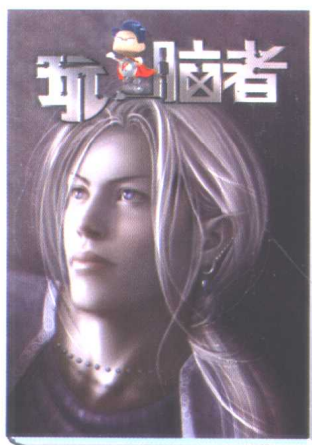
CX-3686
定价: 42.00 元
ISBN: 7900088717



CX-83657
定价: 46.00 元
ISBN: 7900088458

精彩、时尚、实用

邮发代号 2-699 国内统一刊号: CN11-2992/TP 定价: 12.00 元



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀知春路甲 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱 (100080)
电话: (010) 82675588 (总机) 传真: (010) 62520573

目 录

第一篇 OpenGL

第1章 制作 OpenGL 黑屏窗口

框架程序1

1.1 在 Visual C++下建立 Win32 应用

程序2

1.2 程序的开头部分——include 文件

设置及全局变量4

1.3 改变 OpenGL 场景尺寸5

1.4 OpenGL 的初始化7

1.5 OpenGL 的绘制9

1.6 关闭 OpenGL10

1.7 激活 OpenGL 及改变屏幕分辨率10

1.8 处理窗口的消息及键盘事件处理11

1.9 WinMain13

1.10 本章小结17

1.11 本章源码17

第2章 绘制无色的三角形和正方形25

2.1 绘制三角形25

2.2 绘制矩形27

2.3 本章小结29

2.4 本章源码29

第3章 绘制彩色的三角形和正方形37

3.1 绘制彩色带混色的三角形37

3.2 绘制单一色彩的矩形39

3.3 本章小结40

3.4 本章源码40

第4章 制作彩色旋转的三角形和

正方形48

4.1 彩色带混色的三角形的旋转48

4.2 单一色彩的矩形的旋转51

4.3 本章小结52

4.4 本章源码53

第5章 制作彩色旋转的金字塔和

正方体61

5.1 绘制彩色旋转的金字塔61

5.2 绘制彩色旋转的正方体66

5.3 本章小结71

5.4 本章源码71

第6章 制作旋转的带纹理贴图的

正方体80

6.1 程序 include 及全局变量定义81

6.3 读取文件82

6.3 将位图转换成纹理83

6.4 OpenGL 初始化（包括纹理的初 始化）85

6.5 绘制带纹理的正方体86

6.6 本章小结90

6.7 本章源码90

第7章 实现纹理过滤器，基本的光照 和键盘控制等特效101

7.1 新增的设置103

7.2 读取位图文件106

7.3 读取位图文件并将其转换为纹理107

7.4 OpenGL 的初始化110

7.5 绘制带纹理贴图的正方体112

7.6 增加键盘交互功能115

7.7 本章小结118

7.8 本章源码119

第8章 制作透明的带纹理贴图的可键 盘控制的正方体131

8.1 使用混合功能实现透明的原理131

8.2 程序新的全局变量的设置132

8.3 OpenGL 初始化程序 SceneInit()的 修改134

8.4 增加'B'键的处理134

8.5 本章小结135

8.6 本章源码135

第9章 制作美丽的星空148

9.1 新的全局变量的设置149

9.2 读取纹理位图图像150

9.3 读取位图并将其转换成纹理151

9.4 OpenGL 的初始化152

9.5 绘制星星153

9.6 修改键盘事件的处理154

9.7 本章小结154

9.8 本章源码155

第10章 用 3D 实现的在屋子里

移动165

10.1 建立三维结构166

10.2 读取模型数据166

10.3 控制相机实现在 3D 世界中真实的移动.....	168	14.3 本章源码.....	246
10.4 绘制三维场景.....	170	第 15 章 制作彩色旋转的金字塔和正方体.....	256
10.5 本章小结.....	174	15.1 定义金字塔及正方体的顶点坐标.....	256
10.6 本章源码.....	174	15.2 修改 D3DInit 函数.....	258
第二篇 DirectX 3D		15.3 修改函数 D3DSceneShow().....	259
第 11 章 制作不断变换背景颜色的 DirectX 窗口.....	189	15.4 本章小结.....	260
11.1 include 及全局变量的设置以及重要的声明.....	190	15.5 本章源码.....	260
11.2 初始化场景.....	191	第 16 章 制作带纹理贴图的旋转的正方体.....	271
11.3 检测是否支持 16 位色彩模式.....	191	16.1 定义新的全局变量及修改顶点的定义.....	271
11.4 对 Direct3D 进行初始化.....	191	16.2 修改 D3DDisable()函数.....	273
11.5 释放 Direct3D 资源.....	193	16.3 修改 D3DInit()函数.....	274
11.6 绘图.....	193	16.4 读取纹理位图文件并实现对象的纹理贴图.....	274
11.7 处理视窗消息.....	194	16.5 绘制带纹理贴图的正方体.....	275
11.8 编写 WinMain().....	195	16.6 本章小结.....	275
11.9 本章小结.....	197	16.7 本章源码.....	275
11.10 本章源码.....	198	第 17 章 制作纹理过滤、光照效果以及键盘控制.....	287
第 12 章 制作黑白的三角形和正方形.....	205	17.1 为使用灯光效果及键盘交互而增加的设置.....	288
12.1 定义新的数据结构.....	205	17.2 在 D3DInit 中增加对灯光及纹理过滤的支持.....	290
12.2 修改 D3DDisable 函数.....	206	17.3 修改函数 D3DSceneShow().....	292
12.3 对函数 D3DInit()进行修改.....	207	17.4 增加键盘控制功能.....	293
12.4 绘制.....	209	17.5 本章小结.....	295
12.5 本章源码(1).....	210	17.6 本章源码.....	295
12.6 制作黑白的三角形和正方形的另一种方法.....	219	第 18 章 制作透明效果.....	309
12.7 对函数 D3DDisable 的修改.....	220	18.1 增加一些设定.....	309
12.8 修改函数 D3DInit.....	220	18.2 修改子程序 D3DInit.....	311
12.9 为 D3DSceneShow()增加新的功能.....	222	18.3 更新 D3DDisable 函数.....	313
12.10 本章小结.....	223	18.4 修改用于绘制的 D3DSceneShow()函数.....	313
12.11 本章源码(2).....	223	18.5 本章小结.....	314
第 13 章 制作彩色的三角形和正方形.....	233	18.6 本章源码.....	314
13.1 实现多边形的着色.....	233	附录 1 OpenGL 指令集.....	330
13.2 本章小结.....	234	OpenGL 核心函数库.....	330
13.3 本章源码.....	234	OpenGL 应用函数库.....	333
第 14 章 制作彩色旋转的三角形和正方形.....	244	附录 2 安装 DirectX 8.0 SDK.....	336
14.1 实现旋转功能.....	244		
14.2 本章小结.....	246		

第一篇 OpenGL

第 1 章 制作 OpenGL 黑屏窗口框架程序

本章重点：

本章讲述如何用 Visual C++实现 OpenGL 窗口的编程，制作 OpenGL 黑屏窗口，结果如图 1-1 所示。

所用功能：

本章使用了下面一些基本的 OpenGL 函数：

`glViewport()`,`glMatrixMode()`,`glLoadIdentity()`,`glMatrixMode()`,`gluPerspective()`,`glShadeModel()`,`glClearColor()`,`glClearDepth()`,`glEnable()`,`glDepthFunc()`,`glHint()`,`glClear()`

学习目的：

学会最简单的 OpenGL 应用程序的设计，理解每一个函数的作用，理解基本的色彩，透视及投影概念。

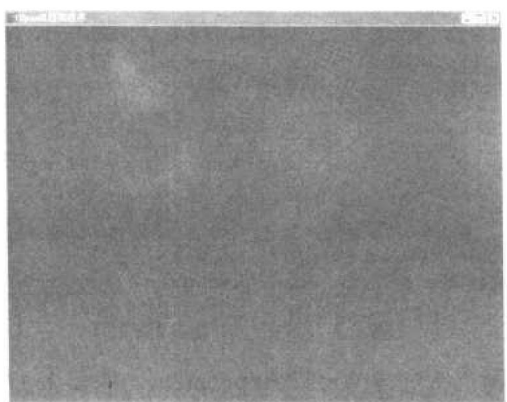


图 1-1 程序运行后的屏幕截图

本书所有 OpenGL 程序的窗口可以是窗口或全屏，可以做到任意的尺寸，任意的分辨率，任意的颜色深度。

本书所附代码可以灵活地用在所有 OPENGL 编程项目中。这些代码既灵活、同时功能很强。并且这些代码很容易被理解及修改。在学习这本书之前，应对 Visual C++ 和 OpenGL 有一定的了解。在本书后面列出一些参考书目可供参考。

这一章的内容是如何建立 OpenGL 窗口。知识点是 OpenGL 色彩的设定，透视及投影的概念。创建最简单的 OpenGL 窗口所要用到的 OpenGL 函数。在 Windows 下使用 VC++ & OpenGL 进行三维动画设计所用的程序结构。

注意：接下来的几章都是基于这 7 个函数进行 OpenGL 应用程序的编写，并仅对有改动的函数或新增的函数进行讲解，所以对第一章的学习要有足够的重视。

接下来是讲述如何一步一步实现 OpenGL 黑屏窗口的编程。

1.1 在 Visual C++ 下建立 Win32 应用程序

这里要做的第一件事是在 VC++ 中建立一个新的项目，如图 1-2 所示。

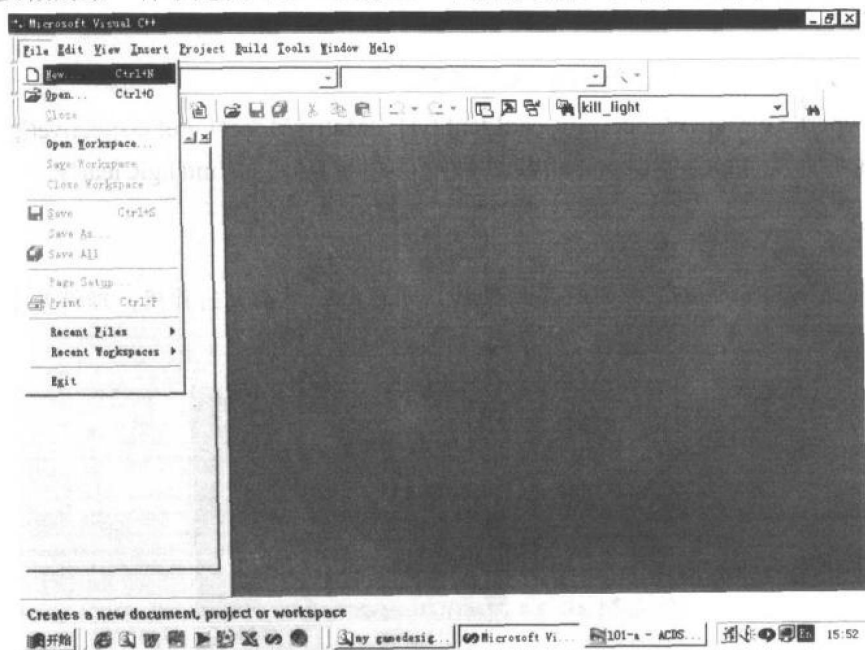


图 1-2 建立一个新的程序

(1) 首先要建立一个新的 VC++ Win32 Application，如图 1-3 所示。

注意：不要选择 Win32 Console Application，而要选择 Win32 Application。

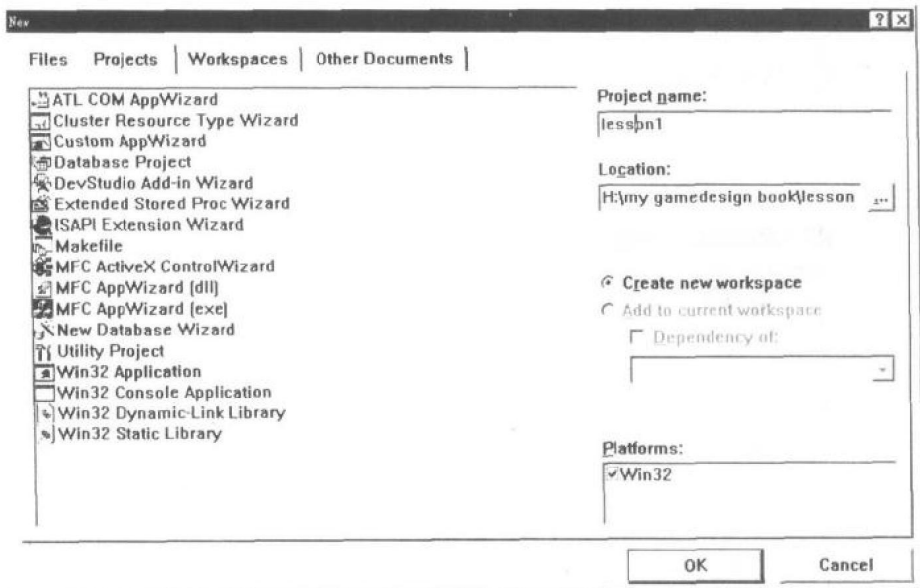


图 1-3 建立一个新的 VC++ Win32 Application

(2) 接着需要在 Visual Studio 中设置链接 OpenGL 库文件。在 VC++ 菜单中选择 Project->Settings, 如图 1-4 所示。



图 1-4 在 VC++ 菜单中选择 Project->Settings

(3) 最后选择 LINK 选项, 然后在 "Object/Library Modules" 下增加 OpenGL 所需的库程序。具体库程序是 OpenGL32.lib、Glu32.lib、Glaux.lib, 如图 1-5 所示。

注意：加入库程序的方法是，在开始行(在 kernel32.lib 前)加上 OpenGL32.lib (OpenGL 核心函数库)、Glu32.lib (OpenGL 应用函数库) 和 GLaux.lib (OpenGL 辅助函数库)。

做完这些后点击 OK。然后就可以写 OpenGL 窗口程序了。

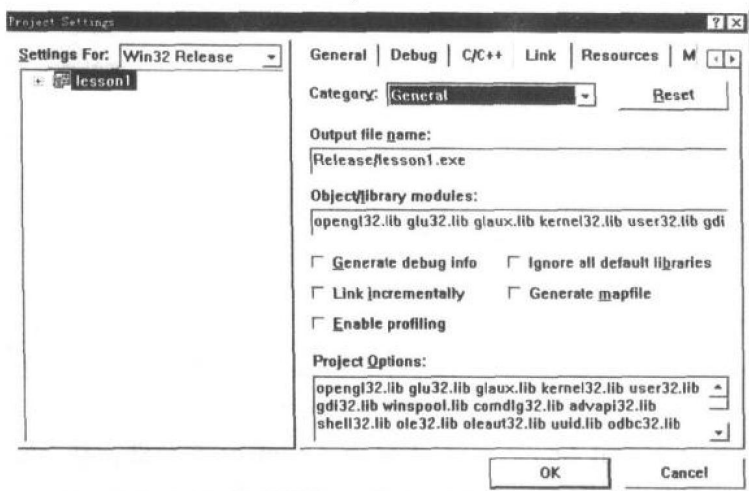


图 1-5 加上 OpenGL32.lib、Glu32.lib 和 Glaux.lib

1.2 程序的开头部分——include 文件设置及全局变量

这里会详细介绍程序的开头是如何设置的。

(1) 开始的 4 行包括了窗口及 OpenGL 所使用的库的头文件。

程序代码如下：

```
#include <windows.h> //视窗口文件
#include <gl\gl.h> //OpenGL32库文件的头文件
#include <gl\glu.h> //GLu32.lib 的头文件
#include <gl\glaux.h> //GLaux.lib的头文件
```

(2) 接下来是定义在程序中计划使用的所有变量

由于这个程序只是建立一个空的 OpenGL 窗口，所以不需定义太多的变量。这些变量虽然不多但很重要的，以后各章的大部分程序都会用到。

第 1 行是建立一个渲染环境(Rendering Context)。

每一个 OpenGL 程序都会被链接到渲染环境。渲染环境的作用是将 OpenGL 调用连接到设备环境(Device context)。OpenGL 渲染环境被定义为 hRC。

第 2 行任务是为了使程序在窗口中绘制需要而创建一个设备环境。

窗口设备环境被定义为 hDC.DC 将窗口连接到 GDI(Graphics Device Interface 图形设备接口)。RC 将 OpenGL 连接到 DC。

在第3行中参数 `hWnd` 将得到 windows 分配给窗口的句柄(handle)。

最后, 第4行为程序建立一个程序的例子 Instance(事件)。

程序代码如下:

```
HGLRC hRC=NULL;//定义渲染环境
HDC hDC=NULL;//私有的GDI设备环境
HWND hWnd=NULL//得到窗口句柄
HINSTANCE hInstance;//得到程序的例子
```

(3) 下面设置窗口大小、斜率及全屏标志变量

变量 `fullscreen` 的作用是显而易见的。

如果程序运行在全屏模式, `fullscreen` 为 `TRUE`, 如果程序运行在窗口模式, `fullscreen` 为 `FALSE`。将其设置为全局变量是很重要的, 这可使程序的每个过程和函数都知道程序是否运行在全屏模式。

程序代码如下:

```
RECT      rect;

int        sw = 640;
int        sh = 480;

bool       fullscreen = 1;

GLfloat    aspect;
```

(4) 现在开始定义程序链接时所需要调用的 OpenGL 程序库

做这个的理由是可以灵活地切换 Debug 模式及 Release 模式, 而不用担心链接时由于没有设定 OpenGL 程序库而出现错误信息。

程序代码如下:

```
#pragma comment( lib, "opengl32.lib" )
#pragma comment( lib, "glu32.lib" )
#pragma comment( lib, "glaux.lib" )
```

1.3 改变 OpenGL 场景尺寸

下一段代码的功能是当窗口尺寸改变时改变 OpenGL 场景的尺寸 (假设更喜欢窗口模式而不是全屏模式)。

注意: 在全屏模式下不能改变窗口尺寸。但当程序在第一次运行中建立透视视景时仍会至少调用一次这段代码。

OpenGL 场景尺寸的改变是基于其显示所在的窗口的高和宽。

(1) 开始编写 `SceneResizeViewport` 函数代码。

程序代码如下：

```
void SceneResizeViewport(GLsizei w, GLsizei h)
{
    if(h==0)
    {
        h=1;
    }
    aspect = (GLfloat)w/(GLfloat)h;
```

(2) 下面的代码段为屏幕设置透视视景。透视的作用是景物距离越远其尺寸将越小，如图 1-6，这样做可以创建出真实的场景。透视（OpenGL 使用 `gluPerspective`）是基于窗口的高和宽并以 45 度视角进行计算。0.1f 和 100.0f 是绘制在屏幕上相对于屏幕距离的开始点和结束点。

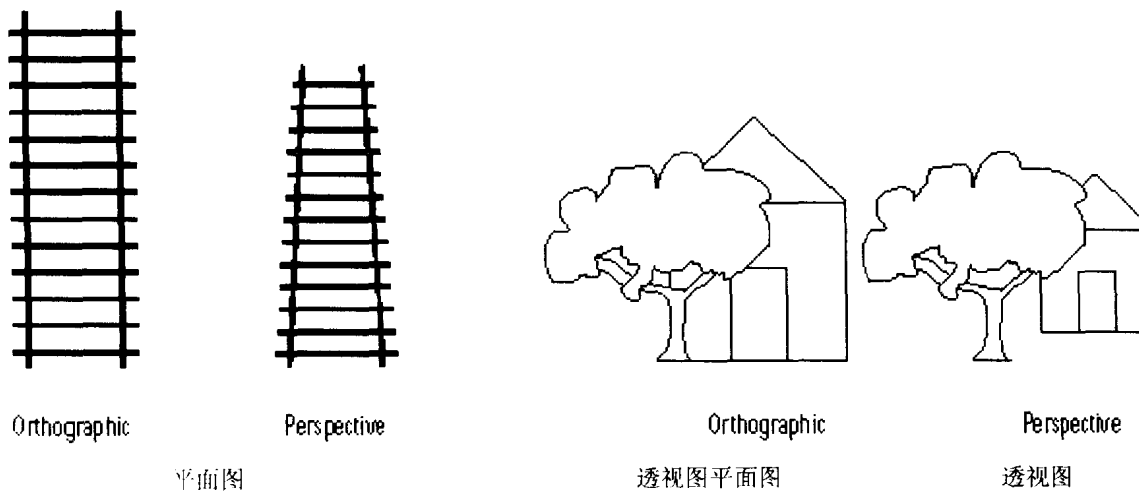


图 1-6 透视效果图示

(3) `glMatrixMode(GL_PROJECTION)` 确定下面两行代码会影响到投影矩阵。透视矩阵为场景加上投影，投影效果如图 1-7 所示。

(4) `glLoadIdentity()` 的功能相当于重置，其作用是被选择的矩阵还原为初始状态。在 `glLoadIdentity()` 被调用后，开始为场景设置透视视点。`glMatrixMode(GL_MODELVIEW)` 确定任何新的变换都会影响模型视点矩阵 `ModelView`。

注意：模型视点矩阵、投影矩阵及透视矩阵的变换过程并最终在屏幕上显示的过程如图 1-8 所示。

模型视点矩阵 `ModelView` 存储了对象的信息。最后会重置模型视点矩阵 `ModelView`。如果不理解这些也不用担心，在后面的教程中会作更详细解释。学到这里仅需知道做这些是为了得到极佳效果的透视场景就可以了。