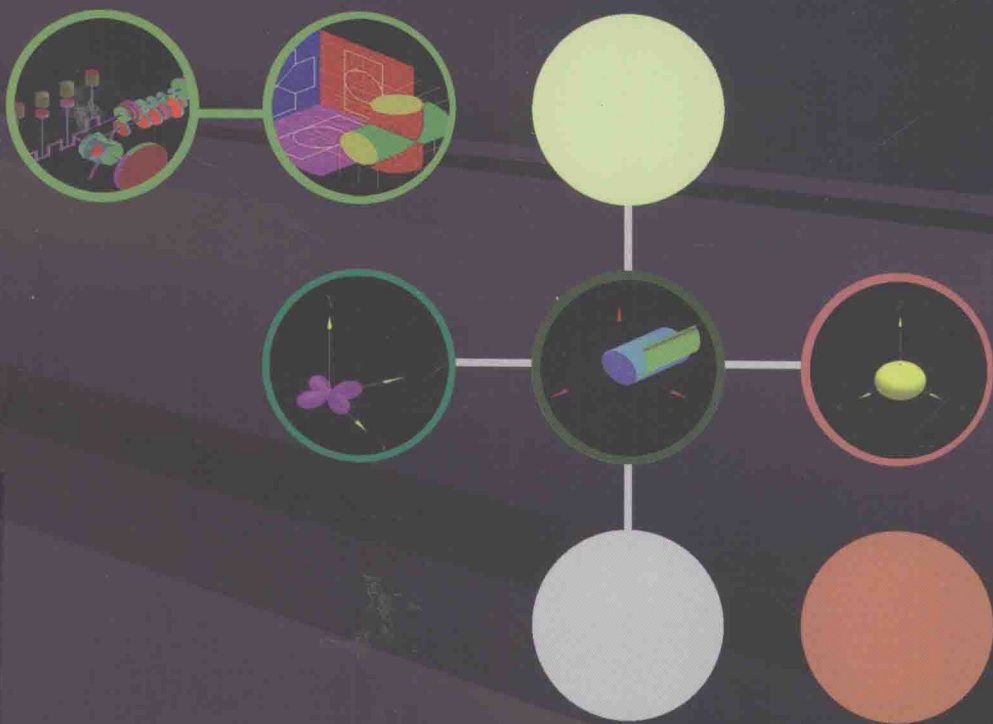


Weikeyian De Sheji Yu Kaifa Shili

微课件的设计与开发 实例

程飞 刘媛 著

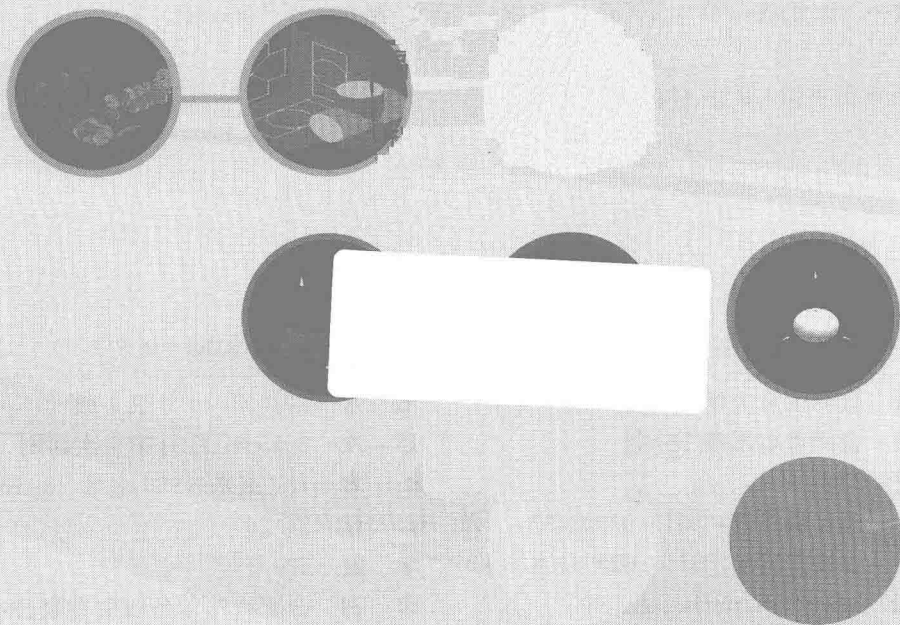


合肥工业大学出版社
HEFEI UNIVERSITY OF TECHNOLOGY PRESS

Weikejian De Sheji Yu Kai

微课件的设计与开发 实例

程 飞 刘 媛 著



合肥工业大学出版社

内容简介

本书作者结合自身多年来开发微课件的经验,介绍了以最简单 OpenGL 函数和最少的 VC++ 语言来开发一档多视交互式微课件的技术。全书共分 6 章,本书既包括简单交互的单窗口和切分窗口的物理、化学、数学微课件设计技术,又包括复杂形体的机械和古建筑微课件设计,并以 OpenGL 图形库为核心,精心设计了 43 个实例,本书同时还提供了完整的工作区文件,因此读者只要掌握基础的数学知识、较少的 OpenGL 函数知识以及基本的 VC++ 知识即可阅读本书。本书图文并茂,实例丰富,既可以作为微课件开发的指导书,又可以作为 VC++ 和 OpenGL 编程的入门教材。

图书在版编目(CIP)数据

微课件设计与开发实例/程飞,刘媛著. —合肥:合肥工业大学出版社,2015. 10

ISBN 978-7-5650-2468-9

I. ①微… II. ①程…②刘… III. ①多媒体课件—制作 IV. ①G434

中国版本图书馆 CIP 数据核字(2015)第 233233 号

微课件的设计与开发实例

程 飞 刘 媛 著

责任编辑 汤礼广

出 版 合肥工业大学出版社

版 次 2015 年 10 月第 1 版

地 址 合肥市屯溪路 193 号

印 次 2015 年 10 月第 1 次印刷

邮 编 230009

开 本 787 毫米×960 毫米 1/16

电 话 理工编辑部:0551-62903087

印 张 24.25

市场营销部:0551-62903198

字 数 500 千字

网 址 www.hfutpress.com.cn

印 刷 合肥杏花印务股份有限公司

E-mail press@hfutpress.com.cn

发 行 全国新华书店

ISBN 978-7-5650-2468-9

定价:50.00 元

如果有影响阅读的印装质量问题,请与出版社发行部联系调换。

前言

从计算机的产生开始,计算机就被广泛应用于教育领域。从早期的 CAI (Computer-Aided Instruction 计算机辅助教学),到 CAL (Computer-Aided Learning 计算机辅助学习),再到 E-Learning (网络学习),直到目前的 M-Learning (移动学习),计算机一直发挥着重要作用。随着移动技术的发展,计算机对于教学的辅助作用呈现出碎片化、微型化的趋势,不具备交互功能的演示性课件越来越不能适应教学需要,而能够适应移动学习、微型学习的交互式微课件则在教学中发挥越来越重要的作用。微课件将静态图形动态化、文本知识图形化、科学数据可视化,在微课程、翻转课堂、非正式学习和正式学习中都有广泛应用。

微课件具有下列特性:

- (1) 微型性——微课件体积小、教学目的明确,一般只针对一个知识点。
- (2) 形象性——微课件是计算机可视化在微型学习领域的应用。
- (3) 可积性——微课件是搭建大型的系统学习情境的基础材料。

(4) 科普性——设计合理的微课件还具有一定的科普功能。

(5) 交互性——微课件具有交互功能,运行中要求学习者参与,能激发学习者的学习兴趣。

一般来说,交互性微课件的设计和开发需要计算机语言的支持,开发者要掌握一门计算机语言,本书基于 VC++6.0。VC++6.0 的图形功能较为底层,因而要有图形引擎支持,本书采用 OpenGL。研究表明,OpenGL 有强大的三维场景绘制能力,能够高效支持交互式微课件的设计和开发。基于 VC++6.0 与 OpenGL 开发的微课件,能达到引人入胜的效果,能够有效地促进学生的学习。

对非计算机专业人员而言,学习计算机语言常常有一定难度,设计与开发微课件就更不必说了,为了解决部分人员制作微课件难的问题,我们特编写本书。本书有以下主要特点:

(1) 不需要系统学习 VC++6.0。本书作者力图采用最简单的 VC++6.0 与 OpenGL 语句开发微课件。读者仅

需要很少的 C 语言知识基础即可阅读本书。只要按照本书步骤,读者完全可以开发出适合教学需要的微课件。

(2) 本书预先搭建了微课件开发的三维环境框架(见本书第一章),读者在具体开发的时候,只要在必要的地方添加代码,就可以开发交互式微课件。在该环境下,开发微课件的难度将大大降低。

(3) 本书中所举示例,大部分为笔者多年来参加各类课件大赛的获奖作品,并在教学中经过了教学实践的检验。

(4) 本书配有所有实例的 exe 文件,可以直接作为教学课件使用。

(5) 本书提供了所有微课件工作区文件,读者可以在 VC++6.0 下对其进行二次开发,以适合自己的教学需要。

(6) 本书所有资料均可在百度云地址 <http://pan.baidu.com/s/106Kbp0Y> 下载,资料有:

① 全部工作区文件夹,包含本书所有代码。

② OpenGL 预添加文件夹,包括需要预先添加的 lib 文件、h 文件、dll 文件。

③ 可以直接使用的课件及其系统文件夹,包括按照本书的技术开发的微课件集和微课件系统,可以直接应用于教学,其中有“机械制图”微课件集和微课件系统、“机械设计”微课件集和微课件系统、“汽车传动”微课件集和微课件系统、“高等数学”部分微课件、“物理”部分微课件、“基础化学”部分微课件。

本书结构如下:在第 0 章,首先梳理了微课件设计所需要的基本的数学知识和计算机知识;在第 1 章,提供了搭建微课件的开发框架的方法;在第 2 章,提供了简单的演示性微课件的设计方法;在第 3 章,提供了交互式微课件的设计方法;在第 4 章,提供了动画式微课件的设计方法;在第 5 章,提供了复杂的机械形体的微课件设计方法;在第 6 章,介绍了微课件系统的设计方法。

本书第 0 章、第 1 章、第 3 章、第 5 章由安徽电子信息职业技术学院电子工程系程飞副教授撰写,第 2 章、第 4 章、第 6 章由安徽电子信息职业技术学院机电工程系刘媛副教授撰写。

作 者

目 录

第 0 章 基础知识	(1)
0.1 数学基础知识	(1)
0.2 VC++ 基础知识	(3)
0.3 OpenGL 基础	(6)
第 1 章 微课件开发框架的搭建	(9)
1.1 OpenGL 环境搭建	(9)
1.2 微课件开发框架	(9)
1.3 单窗口的微课件环境搭建	(28)
第 2 章 简单微课件设计	(30)
2.1 空间直角坐标系的绘制	(30)
2.2 基本形体的绘制	(34)
2.3 简单数学微课件设计	(46)
2.4 简单化学微课件设计	(64)
第 3 章 交互式微课件设计	(81)
3.1 方向键交互	(81)
3.2 参数式交互	(84)
3.3 一档多视原理	(94)

3.4	一档多视实例	(95)
第4章	动画微课件设计	(129)
4.1	“机械制图”微课件设计	(129)
4.2	“机械设计”微课件设计	(223)
第5章	复杂微课件设计	(258)
5.1	“汽车传动机构”微课件设计	(258)
5.2	结合其他软件设计微课件	(283)
5.3	主控界面的设计	(324)
5.4	基于 OpenGL 透明技术设计微课件	(326)
第6章	微课件系统设计实例	(358)
6.1	“机械制图”微课件系统设计	(358)
6.2	“机械设计”微课件系统设计	(368)
6.3	“中国古代建筑”微课件系统设计	(375)
附 录		(378)
参考文献		(382)

第0章

基础知识

微课件的设计和开发,由于需要掌握必要的数学知识和有关的计算机语言知识及相关软件的使用方法,因此下面对相关基础知识做一些简单介绍。

0.1 数学基础知识

(1) 空间直角坐标系

在空间内作三条相互垂直且相交的数轴 Ox 、 Oy 、 Oz , 这三条数轴的长度单位相同。它们的交点 O 称为坐标原点。 Ox 、 Oy 、 Oz 称为 x 轴、 y 轴和 z 轴。一般地, 取从后向前, 从左向右, 从下向上的方向作为 x 轴、 y 轴、 z 轴的正方向, 如图 0-1 所示, Ox 、 Oy 、 Oz 统称为坐标轴。由两个坐标轴所确定的平面, 称为坐标平面, 简称坐标面。 x 轴、 y 轴、 z 轴可以确定 xOy 、 yOz 、 zOx 三个坐标面。图 0-1 表示的空间直角坐标系也可以用右手来确定。用右手握住 z 轴, 当右手的四个手指从 x 轴正向以 90° 的角度转向 y 轴的正向时, 大拇指的指向就是 z 轴的正向。这样建立的坐标系又称为右手系。

(2) 空间一点的坐标

已知 M 为空间一点。过点 M 作三个平面分别垂直于 x 轴、 y 轴和 z 轴, 它们与 x 轴、 y 轴、 z 轴的交点分别为 P 、 Q 、 R , 如图 0-2 所示。这三点在 x 轴、 y 轴、 z 轴上的坐标分别为 x 、 y 、 z 。于是空间的一点 M 就唯一确定了一个有序数组 x 、 y 、 z 。这组数 x 、 y 、 z 就叫作点 M 的坐标。坐标为 x 、 y 、 z 的点 M 通常记为 $M(x, y, z)$ 。

(3) 矢量

我们把既有大小, 又有方向的量叫作矢量。以 M_1 为始点、 M_2 为终点的有向线段所表示的矢量, 记为 $\overrightarrow{M_1M_2}$, 如图 0-3 所示。如果 $M_1(x_1, y_1, z_1)$, $M_2(x_2, y_2, z_2)$, 则 $\overrightarrow{M_1M_2} = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$ 。矢量的大小称为矢量的模, 记为 $|\overrightarrow{M_1M_2}|$ 。模等于 1 的矢量称为单位矢量。计算 $\frac{\overrightarrow{M_1M_2}}{|\overrightarrow{M_1M_2}|}$ 就得到模等于 1 的矢量, 这称为矢量的规范化。

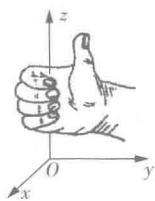


图 0-1 右手系

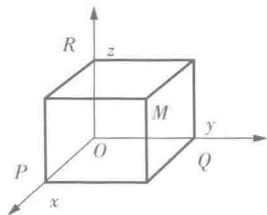


图 0-2 空间一点的坐标



图 0-3 矢量

(4) 矢量 a, b 的矢量积

两个矢量 a, b 的矢量积规定为一个矢量 c 。 c 由下列条件确定：

- ① c 的模等于 $|a| |b| \sin\theta$, 其中 θ 为 a, b 之间的夹角, 即

$$|c| = |a| |b| \sin\theta \quad (0-1)$$

- ② $c \perp a, c \perp b$ 。

③ 面向 c 看, a 以逆时针方向转 θ 角到 b 。如果右手的食指指向 a 的方向, 中指指向 b 的方向, 那么垂直食指和中指的大拇指的指向为 c 的方向, 即遵循右手系。

设 $a = (x_1, y_1, z_1), b = (x_2, y_2, z_2)$, 则

$$a \times b = (y_1 z_2 - y_2 z_1, z_1 x_2 - z_2 x_1, x_1 y_2 - x_2 y_1) \quad (0-2)$$

(5) 球面坐标与空间直角坐标系的转换

如图 0-4 所示, 空间一点 P 对应的直角坐标为 $P(x, y, z)$, 对应的球面极坐标为 (r, θ, ϕ) , 二者有如下关系:

$$\begin{cases} x = r \sin\theta \cos\phi \\ y = r \sin\theta \sin\phi \\ z = r \cos\theta \end{cases} \quad (0-3)$$

其中, 极径 r 以及 θ 角与 ϕ 角的意义如图 0-4 所示。

(6) 曲面法矢量计算

如图 0-5 所示, 矢量 $a = \overrightarrow{P(i, j)P(i+1, j)}, b = \overrightarrow{P(i, j)P(i, j+1)}$, 则表面点 $P(i, j)$ 对应的法矢量为

$$c = a \times b = (b_y a_z - b_z a_y, b_z a_x - b_x a_z, b_x a_y - b_y a_x)$$

其中 a_x, a_y, a_z 和 b_x, b_y, b_z 分别为矢量 a, b 的 x, y, z 分量。

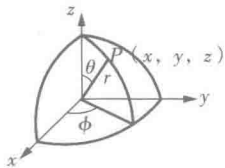


图 0-4 球面坐标

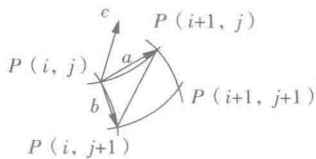


图 0-5 表面点的法矢量计算

0.2 VC++基础知识

(1)数据类型

VC++语言的数据类型很多,微课件开发环境下仅用到:

- ① 基本数据类型,包括 int(整数型)、float(实数)、bool(值为 0 或者 1)。
- ② 复杂数据类型,本书仅用到数组,如 p[100][50][3],表示 5000 个点的三维坐标。

(2)语句

- ① 赋值语句,如

```
int r=5;
```

- ② for 语句,如下列语句,表示计算球体表面的点的坐标,并将坐标值保存在数组 p[100][50][3]中。

```
for(i=0;i<100;i++)//计算各个顶点坐标
```

```
{
```

```
    for(int j=0;j<50;j++)
```

```
    {
```

```
        p[i][j][0]=a*sin(j*3.14/50)*cos(i*6.28/100);
```

```
        p[i][j][1]=b*sin(j*3.14/50)*sin(i*6.28/100);
```

```
        p[i][j][2]=c*cos(j*3.14/50);
```

```
    }
```

```
}
```

(3)函数

函数由函数名、函数类型、参数表、函数体构成。例如定义椭球体函数如下:

```
void tuoqiuti(float r){函数体}
```

(4) VC++ 的结构

VC++ 6.0 开发环境左边有三个选项卡, 本书用到两个: Resource 和 FileView。在 Resource 选项卡中, 展开 Dialog, 本书用到两种类型: 表单 (Form, 如图 0-6 所示) 和对话框 (如图 0-7 所示)。在 FileView 选项卡 (如图 0-8 所示) 中, 上方是源文件 (.cpp Source), 本书主要的操作集中在 DrawView.cpp、kuangjiaView.cpp、kuangjiaDoc.cpp 文件; 下方是头文件 (.h), 本书主要的操作集中在 kuangjiaView.h、kuangjiaDoc.h 文件; 双击打开相应的文件, 定位到相应的位置就可以编写代码, 详细内容请参阅文献[3]。本书中, DrawView.cpp 中在标记“//此处添加变量”处添加变量, 在标记“//此处添加函数”处添加函数, 在标记“//此处添加代码”处添加对于函数的调用。

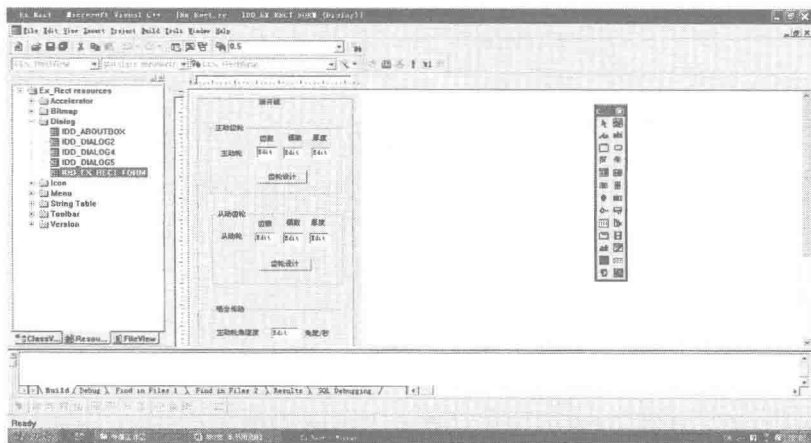


图 0-6 VC++ 6.0 开发环境

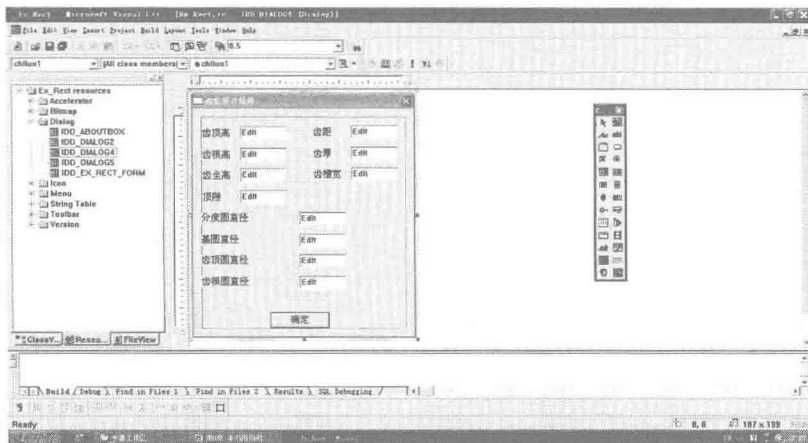


图 0-7 对话框

(5)控件

本书中需要向表单或者对话框添加控件,本书用到的控件有:编辑框(Edit)、旋转按钮(Spin)、按钮(Button)、静态文本(Static)、组框(Group Box),如图 0-9 所示。编辑框与变量对应,语句

```
UpdateData(FALSE);
```

表示变量内容进控件,而

```
UpdateData(TRUE);
```

表示控件内容进变量。

本书采用模态对话框,按钮启动对话框的语句如下例:

```
qujianfenge dlg;

*****

dlg.DoModal();
```



图 0-8 FileView 选项卡

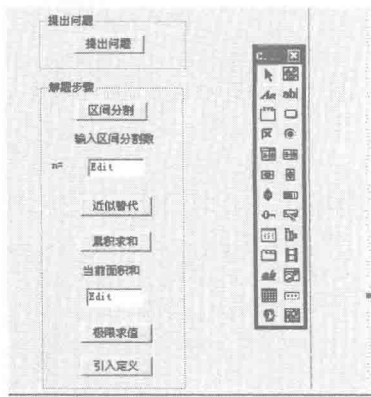


图 0-9 控件

(6)消息函数。

本书使用

```
AfxMessageBox("");
```

显示消息。

(7)打开工作区

VC++通过打开文件夹中的 dsw 文件打开已经存在的工作区。

0.3 OpenGL 基础

OpenGL 是由 SGI 公司发布的低层 3D API。其性能优越,对硬件的要求相对较低,从个人计算机到分布式计算机系统,OpenGL 都可以实现高性能的三维图形功能。本书用到的重要的 OpenGL 函数有:

(1) `gluLookAt()`

```
gluLookAt(10.0f,10.0f,10.0f,0.0f,0.0f,0.0f,0.0f,0.0f,1.0f);
```

表示视点方向。前三个参数表示视起点(10,10,10),中间三个参数表示视终点(0,0,0)(即坐标原点)。最后三个参数(0,0,1)表示建立了 z 轴向上的右手系,如图 0-10 所示。该语句表示从(10,10,10)点向坐标原点(0,0,0)观察。如果物体过远或者过近,需要改变视点近距离观察或者远距离观察,如图 0-11 所示。一般情况下,本书三维绘图均采用视点(10,10,10)。

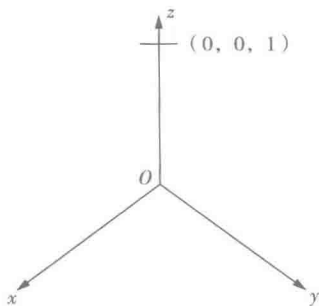


图 0-10 z 轴向上的右手系

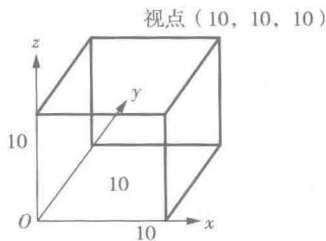


图 0-11 视点方向

(2) `glOrtho()`

```
glOrtho(-50.0f,50.0f,-50.0f,50.0f,-20.0f,20.0f);
```

表示正射投影。此时坐标原点在屏幕中心,物体必须出现在如图 0-12 所示长方体范围内。同样,如果物体相对于该长方体过大或者过小,需要改变正射投影的视口参数,以使得物体出现在长方体内部。

(3) `glTranslatef()`

```
glTranslatef(a,b,c)
```

表示将坐标系平移到(a, b, c)处绘图,是实现动画的重要函数。

(4) `glRotatef()`

`glRotatef` 函数是实现动画的重要函数。

```
glRotatef(x,0.0f,1.0f,0.0f);
```

表示 y 轴不动,从 y 轴正方向向原点观察, xOz 平面逆时针旋转 x (角度)。

```
glRotatef(-x,1.0f,0.0f,0.0f);
```

表示 x 轴不动,从 x 轴正方向向原点观察, yOz 平面顺时针旋转 x (角度)。

```
glRotatef(x,0.0f,0.0f,1.0f);
```

表示 z 轴不动,从 z 轴正方向向原点观察, xOy 平面逆时针旋转 x (角度)。

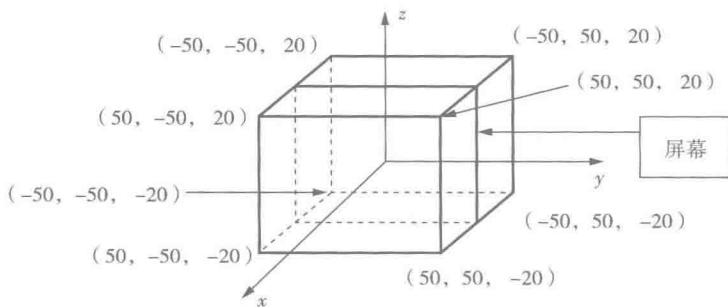


图 0-12 正射投影的视口

本书均采用正射投影。微课件环境下,采用透视投影会发生严重的形变,如图 0-13 和图 0-14 所示。

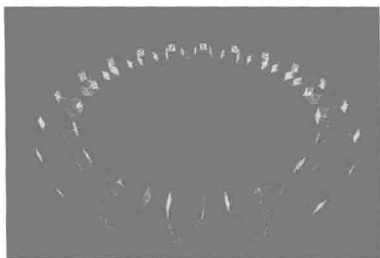


图 0-13 透视投影下绘制的齿轮框架

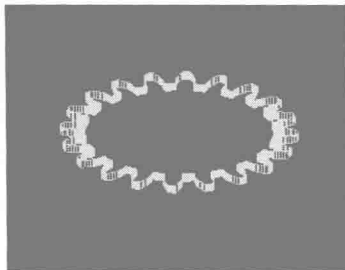


图 0-14 正射投影下绘制的齿轮框架

(5) `glBegin(GL_QUADS);.....glEnd();`

```
glBegin(GL_QUADS);.....glEnd();
```

表示绘制图形。GL_QUADS 表示四边形, GL_LINES 表示绘制直线, GL_POLYGON 表示绘制凸多边形。

```
(6)glNormal3f();
```

```
glNormal3f();
```

表示平面的法线方向。该命令十分重要,会影响光照效果。

```
(7)glVertex3f();
```

```
glVertex3f();
```

表示顶点。

```
(8)glColor3f(R,G,B)
```

```
glColor3f(R,G,B)
```

表示画笔颜色。R、G、B 都取 0、1 之间的数, glColor3f(1,0,0)表示红色, glColor3f(1,1,0)表示黄色。

符号“//”表示注释,该语句将不会被执行。

注意:所有程序区分大小写! 程序中用到的标点符号都是英文符号,否则编译时会有错!

第 1 章

微课件开发框架的搭建

本章基于 OpenGL 环境搭建微课件的单窗口开发框架。

1.1 OpenGL 环境搭建

(1) VC++6.0 程序的安装

要搭建微课件开发环境,必须先安装 VC++6.0 系统。关于 VC++6.0 系统的安装,请查阅相关资料。

(2) 部署 OpenGL

① 打开配套资料(下载地址见前言)中的《OpenGL 预添加文件》文件夹,将其中的 glaux.lib、glut32.lib、glu.lib、glut32.lib、glut.lib、opengl32.lib、opengl.lib 拷贝到 C:\Program Files\Microsoft Visual Studio\VC98\lib 文件夹中。

② 将 glaux.h、glext.h、glu.h、gl.h、glut.h、wglext.h 拷贝到 C:\Program Files\Microsoft Visual Studio\VC98\Include 文件夹中。

③ 将 glaux.h、glext.h、glu32.dll、glu.dll、glut32.dll、glut.dll、opengl32.dll、opengl32.dll 拷贝到 C:\WINDOWS\system32 文件夹中。

1.2 微课件开发框架

1.2.1 建立微课件的工作文件夹

在 D 盘建一个文件夹,名为“微课件开发”。

1.2.2 搭建微课件开发框架

(1) 建立工程

建立工程步骤如下:

① 启动 VC++6.0 系统,出现如图 1-1 界面。



图 1-1 VC++6.0 启动界面

② 选择 File→new 菜单,在弹出的 New 对话框中选择 Projects 选项卡中选择 MFC AppWizard[exe]项目类型,然后在 Project name 框中写入工程英文名称“kuangjia”(意为框架),如图 1-2 所示。

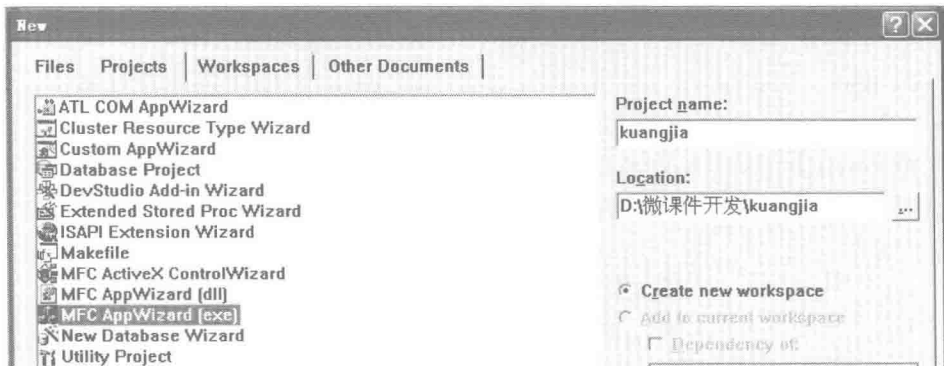


图 1-2 New 对话框

③ 单击 Location 框右侧按钮,出现 Choose Directory 对话框,如图 1-3 所示。

④ 保存地址修改为“D:\微课件开发\kuangjia”,这样所有的微课件开发工作就会被保存在该文件夹中,如图 1-4 所示。单击 OK 按钮,回到 New 对话框。