

图书基本信息

书名：<<Direct 3D和XNA游戏开发基础>>

13位ISBN编号：9787302187646

10位ISBN编号：7302187649

出版时间：2009-1

出版时间：清华大学出版社

作者：耿肇英

页数：383

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

前言

运行于Windows操作系统的大部分3D游戏和图形程序都是基于微软公司DirectX开发的，以往开发DirectX程序主要使用C++语言。

由于DirectX函数库的使用十分复杂，所涉及的概念众多，而且多都不容易理解，又加上C++语言学习难度较大，以及国内适合初学者的有关DirectX程序设计书籍较少等原因，使用DirectX设计3D游戏和图形程序工作一商未能得到很好的普及。

2002年微软推出Managed DirectX（托管DirectX），支持C#语言开发DirectX程序。

2006年8月微软发布XNA Game Studio Express，是微软专门用于开发游戏的最新集成开发环境，是微软人力发展的游戏开发平台。

Managed DirectX和XNA用类重新封装了DirectX函数库，比直接使用DirectX函数库要容易得多。

两者都使用C#语言开发，减少了学习编程语言的难度。

使用C#语言编写基于Managed DirectX和XNA的3D游戏和图形程序可以降低学习和开发难度，提高开发效率，使开发人员能写更安全的代码。

基于Managed DirectX和XNA的3D游戏和图形程序的运行速度已很接近C++程序，一些商业游戏已经使用Managed DirectX和XNA来创建。

XNA编写的3D游戏程序还可运行于微软游戏机XBox 360。

Managed DirectX和XNA的推出必将促进3D游戏和图形程序设计的普及。

本书介绍使用C#语言开发基于Managed DirectX和XNA的3D游戏和图形程序技术，目的是使具有C语言基础的读者通过本书学习，掌握使用C#语言开发3D图形程序的基础，并能开发简单的3D游戏程序

。

内容概要

使用C#语言可以编写基于DirectX和微软最新游戏开发平台XNA（仅支持C#语言）的3D图形和3D游戏程序，其运行速度已接近于C++代码的运行速度，一些商业游戏已经使用C#语言创建。

用C#语言编写3D图形和3D游戏程序可以降低学习和开发难度，提高开发效率，使开发人员能写出更安全的代码。

本书目的是使具有C语言基础的读者通过本书学习，掌握用C#语言开发基于DirectX和XNA的3D图形和3D游戏程序。

本书采用实例驱动的方式进行讲解，在例子中尽量避免罗列不相关的知识点和无关代码，使例子代码短小精悍，容易理解，书中全部例程均给出了详细设计步骤，并对每一步代码给出详细解释，读者可按照书中步骤完成例子。

本书可作为学习用C#语言开发3D图形和3D游戏程序的入门书，也可作为高校计算机及游戏等相关专业教师、研究生、本专科学学生的教材或参考书，对使用C#语言开发3D图形和3D游戏程序的程序员也有很好的参考价值。

作者简介

耿肇英，教授，1948年1月出生。

1983年毕业于北方交通大学电信系，获工学硕士学位。

2000年晋升为教授。

现任河北师范大学数学与信息科学学院计算机科学系主任。

耿肇英教授目前主要从事智能仪器与虚拟仪器的研究工作。

在《计算机科学》等国内外学术刊物上发表学术论文20余篇。

1993年《动态智能心电监护仪》获河北省人民政府生产办公室河北新产品奖，1998年《智能远程心电监控装置》获国家专利，《SDA—1型微机实验系统》获全国普通高等工业学校CAI协作组优秀CAI软件三等奖。

目前承担着三项横向研究课题。

书籍目录

第1章 Managed DirectX和XNA 1.1 DirectX 1.2 Managed DirectX 1.3 XNA简介 1.4 .NET Framework 1.5 事件驱动 1.6 Windows应用程序框架 1.7 可视化程序设计 1.8 解决方案和项目 1.9 键盘事件 1.10 事件处理函数参数 1.11 鼠标事件 1.12 窗体的Paint事件 1.13 常用结构

第2章 3D图形的数学基础 2.1 3D坐标系统和坐标 2.2 向量 2.3 矩阵和Matrix结构 2.4 仿射矩阵 2.5 Matrix结构表示3D变换矩阵

第3章 Direct 3D程序框架 3.1 图形卡和GPU 3.2 Device类 3.3 3D程序基小框架 3.4 从基小框架创建项目 3.5 Transformed Colored结构 3.6 绘制静止三角形 3.7 绘制点、线和三角形 3.8 VertexBuffer类 3.9 绘制静止立方体 3.10 背面剔除 3.11 在基本框架中使用控件

第4章 Direct 3D图形原理 4.1 世界、观察和投影变换 4.2 Position Colored结构 4.3 显示三角形 4.4 使三角形旋转 4.5 三角形连续旋转 4.6 显示立方体 4.7 从不同位置观察立方体 4.8 使用顶点索引绘制立方体 4.9 模拟地板和墙壁 4.10 旋转的卒心圆柱 4.11 复杂的变换关系

第5章 灯光和材质 5.1 灯光 5.2 材质和Material结构 5.3 Position Normal结构和法线 5.4 定向光源照亮三角形 5.5 定向光源照亮立方体 5.6 定向光源照亮卒心圆柱 5.7 点光源 5.8 聚光灯光源 5.9 各种光源照射到地板上 5.10 镜面高光 5.11 材质属性Emissivre

第6章 纹理 6.1 纹理图案和坐标 6.2 包含纹理坐标的顶点结构 6.3 为墙壁增加纹理 6.4 纹理寻址模式 6.5 光照三角形增加纹理 6.6 为立方体增加纹理 6.7 为空心圆柱增加纹理 6.8 添加背景 6.9 纹理滤波器 6.10 多层纹理 6.11 多级渐进纹理滤波

第7章 Mesh类 7.1 Mesh类预定义的几何体 7.2 显示茶壶 7.3 改变观察点和旋转茶壶 7.4 增加多个茶壶 7.5 克隆mesh实现纹理 7.6 3D字体 7.7 显示.x文件中的3D图形 7.8 地形图 7.9 Mesh的优化 7.10 Mesh的简化 7.11 Progressive Mesh类 7.12 将3ds MaX文件转换为.x文件

第8章 透明效果和雾化 8.1 深度测试 8.2 透明物体的绘制原理 8.3 顶点颜色透明 8.4 材质颜色透明 8.5 纹理透明 8.6 有纹理的透明立方体 8.7 多个透明体 8.8 雾化 8.9 顶点雾化 8.10 像素雾化 8.11 基于范围的雾化

第9章 其他特殊效果 9.1 倒影 9.2 使用TextureFactor来设置顶点颜色 9.3 阴影 9.4 后视镜 9.5 广告牌技术 9.6 模板测试

第10章 可编程流水线入门 10.1 可编程流水线的基本概念 10.2 HLSL基础 10.3 fx文件 10.4 Effect类 10.5 使用HLSL程序基本框架 10.6 简单HLSL渲染 10.7 HLSL光照模型 10.8 HLSL表示环境光 10.9 HLSL定向光源漫反射光 10.10 HLSL表示镜面高光 10.11 纹理渲染 10.12 effect编辑器的使用

第11章 移植到XNA游戏框架 11.1 XNA基本框架 11.2 Basic Effect类 11.3 键盘 11.4 用顶点索引绘制图形 11.5 灯光 11.6 纹理 11.7 .x和.fbx文件 11.8 HLSL 11.9 SpriteBatch类 11.10 输出字符串 11.11 鼠标及按钮实现

第12章 XNA透明效果和雾化 12.1 深度测试 12.2 将顶点颜色设置为透明 12.3 将材质颜色设置为透明 12.4 纹理透明 12.5 像素雾化

第13章 XNA特殊效果 13.1 倒影 13.2 阴影 13.3 广告牌技术 13.4 模板测试 13.5 后视镜 13.6 声音

第14章 用XNA实现粒子系统 14.1 Point Sprite 14.2 描述粒子的结构 14.3 粒子系统原理 14.4 粒子系统类 14.5 模拟曳光弹 14.6 模拟爆炸 14.7 模拟雪景 14.8 HLSL粒子系统 14.9 SpriteBatch粒子系统

第15章 XNA实现阶层动画 15.1 常用的动画技术 15.2 阶层关系 15.3 .x文件格式分析 15.4 显示有阶层关系的.x文件 15.5 逼真的坦克 15.6 蒙皮骨骼动画

第16章 XNA游戏实例 16.1 显示一辆汽车 16.2 汽车停在公路上 16.3 汽车在公路上行驶 16.4 汽车左右移动 16.5 在公路上增加障碍物 16.6 汽车是否碰到障碍物 16.7 完成游戏附录 C# 语言入门

章节摘录

插图：第1章 Managed DirectX和XNA本章首先介绍DirectX、Managed DirectX和XNA有关知识，然后详细解释用C#语言设计3D程序时所用到的但一般教科书又很少讲到的重要概念，最后列出C#语言3D程序设计中用到的一些结构。

本书认为读者已经掌握了C#语言基本语法，并能够使用微软Visual C# 2005速成版（简称VS 2005）编写Windows应用程序。

如果读者不具备这方面的技能，请先阅读附录，掌握最基本的C#语言语法，然后再学习本章，学习事件驱动有关概念和用VS 2005编写Windows应用程序的最基本步骤，为学习用C#语言编写Direct 3D和XNA程序提供必要的基础。

当然，仅靠这些内容就完全掌握C#语言是不可能的，如需进一步学习C#语言，还需要阅读C#语言的专著。

如果读者对这方面的内容比较熟悉，也可跳过本章1.4节以后的内容，直接阅读第2章。

编辑推荐

《Direct 3D和XNA游戏开发基础(C#语言版)》可作为学习用C#语言开发3D图形和3D游戏程序的入门书，也可作为高校计算机及游戏等相关专业教师、研究生、本专科学学生的教材或参考书，对使用C#语言开发3D图形和3D游戏程序的程序员也有很好的参考价值。

XNA+C#新一代游戏开发利器，采用实例教学法，在讲清基本知识点的基础上，借助于丰富的实例加以说明，涵盖关键知识点，提供短小精悍的范例代码，并辅之以详细设计步骤和解释。

随着游戏产业的迅猛发展，快速、高效地开发出精彩的3D游戏已经成为众多游戏开发企业的终极梦想。

从DirectX到XNA，借助于C#的优势，具体的实现细节被封装，游戏开发变得不再神秘，所以不管是游戏开发新手，还是具有C/C++基础的高级程序员，都能轻松开发出优秀的游戏软件。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>