



第九艺术学院
COLLEGE OF THE NINTH ART

—— 游戏开发系列教材

DirectX游戏编程

周国庆 陈 洪 冯人果 编著



- 中国文化产业技术创新联盟
- 中国电影电视技术学会数字特效与三维动画专业委员会
- 中国系统仿真学会数字娱乐专业委员会

推 荐 教 材

清华大学出版社



第九艺术学院——游戏开发系列教材

DirectX 游戏编程

周国庆 陈 洪 冯人果 编 著

清华大学出版社

北 京

内 容 简 介

本书主要讲述了利用 Direct3D 进行图形程序开发的核心技术,从三维向量开始,由浅入深,逐步讲解 3D 图形编程中的矩阵变换相关知识、投影知识,及视口处理、裁剪处理、深度缓存的应用技术。在高级绘图技术中进一步讲解纹理处理、透明及混合技术的应用、模板的应用、光照及阴影技术、网格的应用、利用网格技术处理模型显示、在 Direct3D 程序中显示文字、粒子系统的创建及应用。书的后半部分讲解游戏设计的高级特性,如摄像机封装技术、地形显示技术、场景中的模型捡取。最后讲解 GPU 编程相关的知识,让用户了解高级着色的相关内容。

本书每章都以简单的实例来辅助对知识点的讲解,力求深入浅出,知识由浅入深,难度是逐渐递增的,另外本书的章节之间没有过分紧密的程序连接,所以有一定基础的人员,可以选择需要的章节单独阅读。

本书面向的群体为高校数字媒体艺术程序设计专业的学生,及自考学生,或有一定编程基础的专业图形编程人员。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

DirectX 游戏编程/周国庆,陈洪,冯人果编著. —北京:清华大学出版社,2010.1

(第九艺术学院——游戏开发系列教材)

ISBN 978-7-302-21593-6

I. D… II. ①周… ②陈… ③冯… III. ①多媒体—软件工具, DirectX—高等学校—教材 ②游戏—应用程序—程序设计—高等学校—教材 IV. ①TP311.56 ②G899

中国版本图书馆 CIP 数据核字(2009)第 209982 号

责任编辑:张彦青 宋延清

封面设计:杨玉兰

版式设计:北京东方人华科技有限公司

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185×230 印 张:24.25 字 数:524 千字

版 次:2010 年 1 月第 1 版 印 次:2010 年 1 月第 1 次印刷

印 数:1~4000

定 价:45.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:035309-01

丛书编委会

主编单位: 北京递归开元教育科技有限公司

主 编: 陈 洪

副主编: 许 影 任 科

编 委: (排名不分先后)

北京理工大学软件学院	丁刚毅
北京工业大学软件学院	蔡建平
西南交通大学软件学院	景 红
上海第九城市信息技术有限公司	黄雪斌
巨人网络集团有限公司	邓 昆
目标软件(北京)有限公司	毛海滨
腾讯公司	姚晓光
文睿信息研究中心	王 涛
创游传媒	孙 辉
中国文化创意产业技术创新联盟	高东旭
北京递归开元教育科技有限公司	黄 昆

前 言

这是一本讲述如何使用 DirectX 9.0 编写交互式三维计算机图形程序的书，重点在游戏开发方面。将讲述关于 Direct3D 的基础知识，使读者能够进一步地学习和使用更高级的技术。当读者看完这本书的时候，对 DirectX 就会有一个大体的了解。

从程序员的角度看，DirectX 是一个用于开发 Windows 平台下多媒体应用程序的 API (Application Programming Interfaces)函数的集合。

本书重点关注 DirectX 的一个重要子集——Direct3D。就像名字所描述的，Direct3D 是一个用于开发三维应用程序的 API 的集合。当然本书同时也涉及 DirectX 的其他内容。

本书主要分为 4 个部分：第 1 部分主要介绍贯穿于本书的基本数学概念；第 2 部分将会介绍基本的 3D 技术，例如光照、纹理、Alpha 混合以及模板缓冲区；第 3 部分主要讲述如何通过 Direct3D 去实现多种有趣的技术和应用程序，例如图元选取(Picking)、地形渲染(Terrain Render)、粒子系统(Particle System)、通用虚拟摄像机(Camera)以及如何加载和渲染 3D 模型(.x 文件)；第 4 部分的主题是 Vertex Shader、Pixel Shader 以及特效框架和新的(在 DirectX 9 中首次出现)高级 Shader 语言(HLSL)。

当前和未来的三维计算机游戏会更多地运用 Shader 技术，所以这是一本与现行图形编程密切相关的书。

1. 对读者的要求

本书属于基础性的图书，但是这并不表示本书适合于没有任何编程经验的读者。本书要求读者了解代数学、几何学的知识，熟悉游戏程序开发环境(例如 Visual C++)，掌握 C++ 语言以及链表和数组等基本的数据结构知识，如果读者对 Windows 程序设计具有一定的了解，将更有助于阅读本书，不过这不是必需的。读者可以通过附录 A 对 Windows 程序设计有一个简单的认识。

2. 需要的开发工具

本书使用 C++ 语言作为示例的主要编程语言。读者通过阅读 DirectX 9.0 的帮助文档，可以发现“DirectX 9.0 只支持 Visual C++ 6.0 或者更高版本”。因此，为了使用 DirectX 9.0 和 C++ 语言开发应用程序，需要拥有 Visual C++ 6.0(简称 VC 6.0)或者 Visual C++ 7.0(.NET)(简称 VC 7.0)编程环境。

 **注意：**本书的示例代码都是使用 VC 7.0 开发和编译的。在大多数情况下，这些代码也可以在 VC 6.0 中编译和执行。

VC 7.0 和 VC 6.0 会有如下所示的稍许不同。

在 VC 7.0 中，以下符合语法规则的代码将能编译通过，因为变量 `cnt` 将被认为是 `for` 循环的局部变量：

```
int main()
{
    for(int cnt=0; cnt<10; cnt++)
    {
        std::cout << "hello" << std::endl;
    }
    for(int cnt=0; cnt<10; cnt++)
    {
        std::cout << "hello" << std::endl;
    }
    return 0;
}
```

但是在 VC 6.0 中，这段程序不能通过编译，它会给出一个错误号为 C2374 的错误，该错误产生的原因是变量重定义；因为在 VC 6.0 中，变量 `cnt` 不会在 `for` 循环中被作为局部变量对待，从而会被多次初始化。因此，当使用 VC 6.0 时，你可能需要对程序做出一些修改，以避免程序在编译时产生错误。

3. 建议使用的硬件

如果想在一個可以接受的帧数(一般可接受的帧数为 25 帧/秒)下运行本书的示例，就需要提供一定的硬件支持。

本书所有的例子都可以在 REF 设备下运行，它用软件模拟了 Direct3D 提供的功能。但由于使用软件模拟来实现效果，所以运行速度会很慢，我们将在本书第 1 章讨论 REF。

本书第 2 部分中的示例都是最简单、最基本的，所以可以在一些低端显卡上运行，例如 NVIDIA 的 TNT 系列显卡或者同级的显卡。第 3 部分的示例中将会计算更多的基本图元，并且使用了一些新特性，例如点图元，所以建议使用 GeForce2 或其同级别的显卡。第 4 部分将会使用 Vertex Shader 和 Pixel Shader 技术，因此为了能实时地运行那些程序，需要一块支持 Shader 技术的显卡，例如 GeForce3。

4. 本书面向的读者

本书主要面对以下的三类读者：

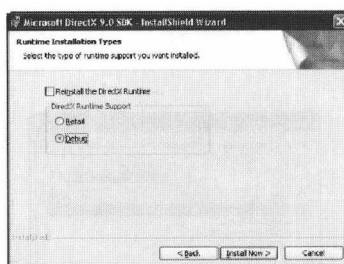
- 中级 C++ 程序员，并想通过最新的 Direct3D 版本——Direct3D 9.0 来了解 3D 编程技术。
- 三维程序开发人员，使用过除了 DirectX 之外的其他图形库 API(例如 OpenGL)并想了解 Direct3D。
- 资深的 Direct3D 程序员，并想了解最新的 Direct3D 的技术，包括 Vertex Shader 和 Pixel Shader，以及高级 Shader 语言和特效框架。

5. 安装 DirectX 9.0

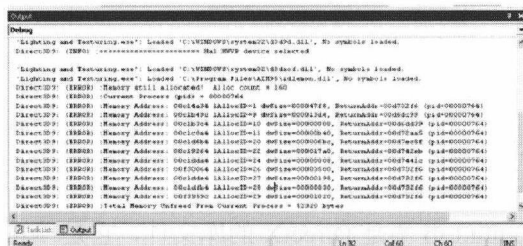
如果要编写和执行 DirectX 9.0 的程序，除了需要安装 DirectX 9.0 的运行时(RunTime)文件外，还需安装 DirectX 9.0 的 SDK(Software Development Kit, 软件开发包)。需要说明的是，在安装 DirectX 9.0 的 SDK 的同时，DirectX 9.0 的运行时文件会被自动安装。

可以从微软的网站上免费获得 DirectX 9.0 的 SDK。

安装过程非常简单，但是需要提示一点，那就是一旦遇到如下图所示的对话框时，就需要选择 Debug 选项。



Debug 选项会将 Debug(调试模式)和 Retail(发布模式)版本的所有 DirectX 动态链接库都安装到用户的计算机中，如果选择 Retail 选项，那么就只会安装 Retail 版本的动态链接库文件。因为在开发程序时，Debug 版的动态链接库可以在 Visual Studio 集成开发环境的输出窗口中输出 Direct3D 相关的调试信息，当然，这时程序需要在 debug 模式下运行，这些信息对于调试 DirectX 程序是非常有用的。下图中给出了 Direct3D 设备没被释放时所显示的错误调试信息。



 **注意：**如果使用 Debug 版的动态链接库，程序的运行速度会比使用 Retail 版的动态链接库时慢，所以对于用于商业销售的应用程序，应该附带 Retail 版本的动态链接库。

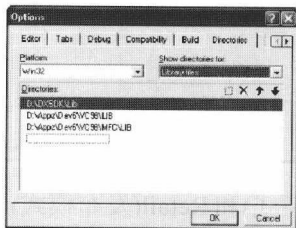
6. 设置开发环境

在开发 DirectX 应用程序的时候，需要创建的工程类型为 Win32 Application。

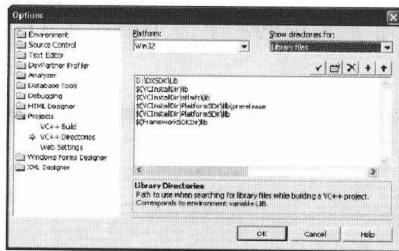
在 VC 6.0 和 VC 7.0 中，都需要指定 DirectX 头文件和库文件的路径，从而使 VC 开发环境可以找到这些文件。通常情况下 DirectX 的头文件和库文件的路径为 D:\DXSDK\Include 和 D:\DXSDK\Lib。

 **注意：**在不同的计算机中，DirectX 的安装路径可能不一样，这决定于当初你在安装 DirectX SDK 时所选择的路径。

一般来说，DirectX SDK 安装程序将会自动地将这些路径添加到 VC++ 集成开发环境中。但如果它没有这样做，也可以手动设置：在 VC 6.0 中选择“工具”→“选择”菜单命令，在弹出的 Options 对话框中选中 Directories(目录)选项卡，然后输入 DirectX 的头文件和库文件路径，如下图所示。

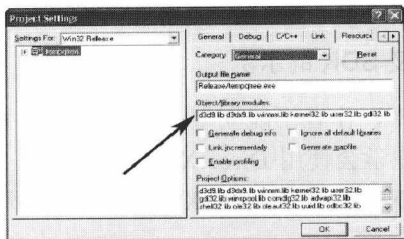


在 VC 7.0 中，可以选择“工具”→“选项”菜单命令，在弹出的 Options 对话框中选择项目中的 VC++ 目录，然后输入 DirectX 9.0 的头文件和库文件的路径，如下图所示。

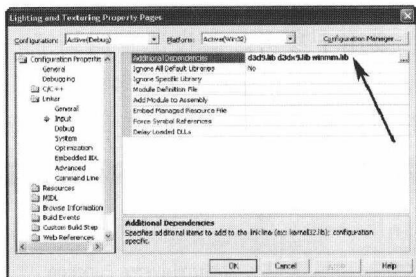


接下来，为了能顺利地编译示例程序，需要将 D3D9.Lib、D3DX9.Lib 和 Winmm.Lib 链接到工程中。注意 Winmm.Lib 不是 DirectX 的库文件，它是 Windows 多媒体库文件，之所以链接它，是因为要使用到它的时钟函数。

在 VC 6.0 中可以选择“工程”→“设置”菜单命令，在弹出的 Project Settings 对话框中选择 Link(链接)选项卡，然后输入需要链接的库名，如下图所示。



在 VC 7.0 中可以选择“项目”→“属性”菜单命令，在弹出的对话框中选择“链接器”→“输入”，然后输入库名，如下图所示。



7. D3DX 库的使用

从 DirectX 7.0 开始，DirectX 都会附带一个 D3DX 库(Direct3D 扩展库)，该库提供一系列函数、类和接口，可以大大地简化 3D 图形相关的通用操作，例如数学运算、纹理和图像操作、网格处理以及 Shader 操作(例如对 Shader 文件的编译和汇编)。也就是说，D3DX 库包含了很多原本需要由开发者自己实现的而且非常繁琐的操作。

在本书中，将始终使用 D3DX 扩展库所提供的功能，从而可以使读者将精力更多地放在感兴趣的部分。例如，作者不希望浪费太多的篇幅来讲述如何读取各种各样的图片格式(如 BMP、JPEG)文件，所以，当需要使用纹理数据的时候，可以通过一个 D3DX 库函数 D3DXCreateTextureFromFile 进行纹理数据的加载。一句话，D3DX 库使我们的开发更高效，并能更多地关注实际内容，而不是花费大量的时间来改造各种难以继续改进的算法。

使用 D3DX 库的其他理由如下：

- D3DX 库是通用的，可以适用于不同范围的 3D 应用程序。
- D3DX 库的执行效率很高(至少不会慢于同类的图形库)。
- 其他程序员也使用 D3DX 库。因此，你将会在很多情况下遇到使用 D3DX 库书写的代码。所以，无论你是否选择使用 D3DX 库进行开发，都应该掌握它，这

样才能读懂别人所写的代码。

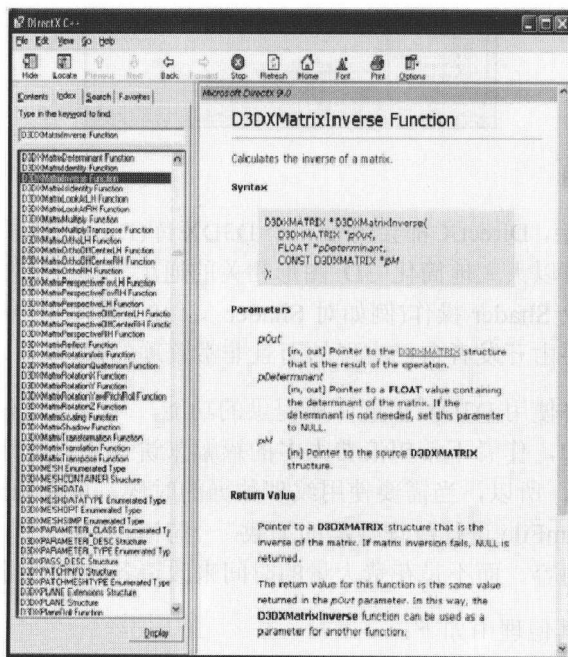
- D3DX 库所提供的函数使用方便，并经过了长时间的检验。此外，在经过数个版本的 DirectX 升级之后，它已经变得非常强大，并且具有更多的特性。

8. 使用 DirectX 文档以及 SDK 的示例

Direct3D 是庞大的 API 库，作者不能期望在本书中涵盖关于它的所有细节。因此，为了更好地掌握其他信息，必须学会使用 DirectX 的在线文档。可以通过在 DirectX SDK 的安装目录\DXSDK\Doc\DirectX9 中运行 DirectX9_c 文件来打开 DirectX C++ 在线文档。

DirectX 在线的文档涵盖了关于 DirectX API 的所有内容,所以它是非常好的参考文档,但是由于它的覆盖范围非常大,所以没有比较深入地探讨一些问题,因此它并不是很好的学习工具。但是,随着 DirectX 不同版本的发布,它将会变得更加完善。

就像前面所说的，文档主要用于参考。假设读者遇到了一个 DirectX 相关的类型或函数(例如 D3DXMatrixInverse)，并想对这个函数有更多的了解，这时就可以在帮助文档的索引里进行简单的搜索，从而获得更多的类型信息或者函数信息，如下图所示。



 **注意：**在本书的内容讲解中，将会引导读者在文档中寻找更多的细节。

在 DirectX 9 对应的开发文档(\DirectX9_c.chm::directx/graphics/programmingguide/tutorialsandsamplesandtoolsandtips/tutorials/tutorials.htm)中还包括了一些介绍性的教程。这些教

程是与本书的第 2 部分的某些主题相对应的。因此建议在学习本书的这部分内容的同时，阅读帮助文档中的这些教程，以得到对同一个问题的更多解释。

值得注意的是，DirectX SDK 还附带了一些关于 Direct3D 的示例程序。这些程序位于 \DXSDK\Samples\C++\Direct3D 目录中。每个示例都示范了怎样在 Direct3D 中实现某个特效。这些示例对一些图形编程新手来说有相当的难度，但是在读完这本书后你就可以很容易地学习它们了。在学完本书的内容后，下一步最好是学习这些示例。

9. 代码约定

本书的示例所用的代码约定都很简单，但是需要指出两点。

(1) 用下划线标识成员变量。例如：

```
class C
{
public:
    // ...定义公共接口
private:
    float _x; // 成员变量以下划线开头
    float _y;
    float _z;
};
```

(2) 全局变量和全局函数用大写字母开头，而局部变量和局部函数以小写字母开头。可以通过这些来了解变量或者函数的作用域。

10. 错误处理

一般在本书的示例中不会进行错误处理，因为在讲述一个重要概念和技术的主要代码时，与错误处理相关的代码会分散读者的注意力。也就是说，没有了错误处理代码，示例可以更清楚地示范或讲述一个概念或技术。当你自己的工程中利用本书所提供的代码时，应该加入错误处理的相关代码。

11. 代码清晰度

需要强调一点，在本书示例中代码的书写原则是清晰度第一，性能第二。因此，某些示例程序的执行效率可能不高。如果要在你自己的工程中使用这些代码，可能需要重新编写代码，并使程序的运行效率更高。

12. 示例程序及其他在线功能

本书的中文网站 <http://www.cngda.com> 提供了对本书的技术支持。你可以在这个网站上

找到本书每一个示例程序的完整代码。建议读者在学习本书各章节的过程中，或者在阅读过这些章节之后学习完整的源代码；一般来说，读者需要花一部分时间来实现每章中的示例。事实上，学习本书的一个好方法是以本书的源代码为参考，然后自己编写实例。

除了示例程序之外，网站还提供了讨论区。我们希望读者可以在网站上交流经验，以及讨论一些仍未明白的问题。在大多数情况下，了解对同一概念的不同看法能加快读者领悟概念的速度。

还计划在网站上添加由于各种原因不适合置于本书中的一些示例程序及教程。如果读者提出对某个特定的概念存在疑问，或对之难以理解，那么针对该问题的额外示例和解释也会被添加到网站上。



读者回执卡

欢迎您立即填写回函

您好！感谢您购买本书，请您抽出宝贵的时间填写这份回执卡，并将此页剪下寄回我公司读者服务部。我们会在以后的工作中充分考虑您的意见和建议，并将您的信息加入公司的客户档案中，以便向您提供全程的一体化服务。您享有的权益：

- ★ 免费获得我公司的新书资料；
- ★ 寻求解答阅读中遇到的问题；

- ★ 免费参加我公司组织的技术交流及讲座；
- ★ 可参加不定期的促销活动，免费获取赠品；

读者基本资料

姓 名 _____ 性 别 ☐男 ☐女 年 龄 _____
电 话 _____ 职 业 _____ 文化程度 _____
E-mail _____ 邮 编 _____
通讯地址 _____

请在您认可处打√ (6至10题可多选)

- 您购买的图书名称是什么：_____
- 您在何处购买的此书：_____
- 您对电脑的掌握程度：
☐不懂 ☐基本掌握 ☐熟练应用 ☐精通某一领域
- 您学习此书的主要目的是：
☐工作需要 ☐个人爱好 ☐获得证书
- 您希望通过学习达到何种程度：
☐基本掌握 ☐熟练应用 ☐专业水平
- 您想学习的其他电脑知识有：
☐电脑入门 ☐操作系统 ☐办公软件 ☐多媒体设计
☐编程知识 ☐图像设计 ☐网页设计 ☐互联网知识
- 影响您购买图书的因素：
☐书名 ☐作者 ☐出版机构 ☐印刷、装帧质量
☐内容简介 ☐网络宣传 ☐图书定价 ☐书店宣传
☐封面、插图及版式 ☐知名作家（学者）的推荐或书评 ☐其他
- 您比较喜欢哪些形式的学习方式：
☐看图书 ☐上网学习 ☐用教学光盘 ☐参加培训班
- 您可以接受的图书的价格是：
☐20元以内 ☐30元以内 ☐50元以内 ☐100元以内
- 您从何处获知本公司产品信息：
☐报纸、杂志 ☐广播、电视 ☐同事或朋友推荐 ☐网站
- 您对本书的满意度：
☐很满意 ☐较满意 ☐一般 ☐不满意
- 您对我们的建议：_____

1 0 0 0 8 4

北京100084—157信箱

读者服务部

收

邮政编码：□□□□□□

贴 票
邮 处

请剪下本页填写清楚，放入信封寄回，谢谢！

技术支持与课件下载: <http://www.tup.com.cn> <http://www.wenyuan.com.cn>

读者服务邮箱: service@wenyuan.com.cn

邮购电话: (010)62791865 (010)62791863 (010)62792097-220

组稿编辑: 张彦青

投稿电话: (010)62788562-312

投稿邮箱: zhangyq-tup@163.com

目 录

第 1 部分 必备的数学知识

第 1 章 基本的数学概念.....	3	1.2.8 平移变换.....	20
1.1 三维空间中的向量.....	3	1.2.9 旋转矩阵.....	21
1.1.1 向量的相等.....	6	1.2.10 缩放矩阵.....	23
1.1.2 向量模的计算.....	7	1.2.11 组合变换.....	23
1.1.3 向量归一化.....	8	1.2.12 一些用于向量变换的 函数.....	25
1.1.4 向量加法.....	8	1.3 平面.....	25
1.1.5 向量减法.....	9	1.3.1 D3DX 中的平面.....	26
1.1.6 数乘.....	10	1.3.2 点和平面之间的关系.....	27
1.1.7 点乘.....	10	1.3.3 构建平面.....	27
1.1.8 叉乘.....	11	1.3.4 标准化一个平面.....	28
1.2 矩阵.....	12	1.3.5 对平面进行变换操作.....	29
1.2.1 矩阵的相等、矩阵数乘和 矩阵加法.....	13	1.3.6 求到一个指定点距离最短的 平面上的点.....	29
1.2.2 矩阵乘法.....	13	1.4 射线.....	30
1.2.3 单位矩阵.....	15	1.4.1 射线的表达式.....	30
1.2.4 逆矩阵.....	15	1.4.2 射线与平面的相交.....	31
1.2.5 矩阵的转置.....	16	1.5 小结.....	31
1.2.6 D3DX 中的矩阵.....	16		
1.2.7 基本变换.....	19		

第 2 部分 Direct3D 基础知识

第 2 章 Direct3D 初始化.....	35	2.3.2 重采样.....	39
2.1 Direct3D 体系结构.....	35	2.3.3 像素格式.....	40
2.1.1 REF 设备.....	36	2.3.4 存储池.....	40
2.1.2 D3D 设备类型.....	36	2.3.5 交换链和页面翻转.....	41
2.2 组件对象模型.....	37	2.3.6 深度缓冲区.....	42
2.3 预备知识.....	37	2.3.7 顶点处理.....	43
2.3.1 表面缓冲区(Surface).....	37	2.3.8 设备性能.....	43

2.4 初始化 Direct3D	44	4.1.3 获得顶点缓冲区以及索引 缓冲区的相关信息	75
2.4.1 获得 IDirect3D9 接口	45	4.2 渲染状态	76
2.4.2 确认硬件顶点处理能力	45	4.3 渲染前的准备工作	77
2.4.3 填充 D3DPRESENT_ PARAMETERS 结构	46	4.4 使用顶点/索引缓冲区进行渲染	78
2.4.4 创建一个 IDirect3DDevice9 接口	48	4.4.1 IDirect3DDevice9:: DrawPrimitive	78
2.5 示例: 初始化 Direct3D	49	4.4.2 IDirect3DDevice9:: DrawIndexedPrimitive	78
2.5.1 d3dUtility.h/cpp 文件	50	4.4.3 Begin/End Scene	80
2.5.2 程序框架	52	4.5 D3DX 自带的几何物体	80
2.5.3 D3D 初始化示例	52	4.6 示例: 三角形、立方体、 茶壶、D3DXCreate*	81
2.6 小结	55	4.7 小结	85
第 3 章 渲染管道	57	第 5 章 颜色	87
3.1 描述的模型	57	5.1 颜色的描述方法	87
3.1.1 顶点格式	58	5.2 顶点颜色	90
3.1.2 三角形	59	5.3 着色处理	90
3.1.3 索引	60	5.4 示例: 带颜色的三角形	91
3.2 虚拟摄像机	60	5.5 小结	93
3.3 渲染管道	61	第 6 章 光照	95
3.3.1 本地空间	62	6.1 光的组成	95
3.3.2 世界空间	62	6.2 材质	96
3.3.3 视图空间	63	6.3 顶点法线	98
3.3.4 背面裁剪	65	6.4 光源	100
3.3.5 光照	66	6.5 示例: 光照	104
3.3.6 裁剪	66	6.6 其他例子	106
3.3.7 投影	67	6.7 小结	107
3.3.8 视口变换	69	第 7 章 纹理	109
3.3.9 光栅化	70	7.1 纹理坐标	109
3.4 小结	70	7.2 创建并应用纹理	111
第 4 章 在 Direct3D 中绘制图形	71	7.3 纹理过滤	112
4.1 顶点/索引缓冲区	71	7.4 多纹理映射	113
4.1.1 创建顶点缓冲区和 索引缓冲区	71	7.4.1 多纹理映射过滤	113
4.1.2 访问缓冲区内存	74		

7.4.2 在 Direct3D 中使用多纹理映射	114	9.1.1 获取模板缓冲区	131
7.5 纹理坐标寻址模式	114	8.1.2 模板测试	131
7.6 程序示例: 带纹理的四边形	116	9.1.3 模板测试的控制	131
7.7 小结	118	9.1.4 更新模板缓冲区	133
第 8 章 混合技术	119	9.1.5 模板写入掩码	134
8.1 混合方程	119	9.2 示例: 镜子	134
8.2 混合因子	121	9.2.1 镜面反射的相关数学知识	134
8.3 透明	121	9.2.2 镜面效果的实现概述	136
8.3.1 Alpha 通道	122	9.2.3 代码和解释	137
8.3.2 指定源像素的 Alpha 值	122	9.3 示例: 平面阴影	140
8.4 用 DirectX 的纹理工具创建 Alpha 通道	123	9.3.1 平行光阴影	141
8.5 示例: 透明效果	124	9.3.2 点光源阴影	142
8.6 小结	127	9.3.3 阴影矩阵	142
第 9 章 模板缓冲区	129	9.3.4 用模板缓冲区来防止二次混合	143
9.1 使用模板缓冲区	130	9.3.5 代码和解释	144
		9.4 小结	146

第 3 部分 Direct3D 的应用

第 10 章 文字处理	149	第 11 章 网格(1)	157
10.1 ID3DXFont、ID3DXFont 接口	149	11.1 几何信息	157
10.1.1 创建一个 ID3DXFont 接口	149	11.2 网格子集(Subset)和属性缓冲区	158
10.1.2 文字输出	150	11.3 网格的渲染	160
10.1.3 计算帧速(FPS)	151	11.4 网格优化	160
10.2 CD3DFont、CD3DFont 类	152	11.5 网格属性表	162
10.2.1 构造 CD3DFont 对象	152	11.6 邻接信息	164
10.2.2 DrawText 输出文字	153	11.7 网格的克隆(Cloning)	165
10.2.3 销毁 CD3DFont 对象	153	11.8 网格的创建 (D3DXCreateMeshFVF)	166
10.3 三维网格(3D Mesh)在文字显示中的应用	154	11.9 示例: 创建并渲染一个网格	168
10.3.1 D3DXCreateText 方法	154	11.10 小结	173
10.4 小结	155	第 12 章 网格(2)	175
		12.1 ID3DXBuffer 接口	175
		12.2 .X 文件	176