

Direct

Direct x 6
立即模式

技术参考手册

3D



周颖恒 杨峰 等编
西南交通大学出版社

Direct 3D 立即模式技术参考手册

—————Direct X 6 技术参考手册系列丛书

杨 峰 周颖恒 等编

西南交通大学出版社

• 成 都 •

内 容 提 要

本书主要介绍 Direct 3D 立即模式的编程知识。Direct 3D 立即模式是 DirectX D3D 提供的一个低层的立即模式界面,有了它,应用程序就能完全控制整个渲染管线。Direct 3D 立即模式的一些特性包括可交换的 Z 缓冲、Flat 和 Gouraud 明暗处理、多光源多类型的 Phong 光线模型、含 mipmap 在内的完整材质和纹理支持、Ram 和 RGB 软件仿真、变换和裁剪等。

全书共分为两部分,第一部分主要介绍 Direct 3D 立即模式的一些基础知识;第二部分主要介绍 Direct 3D 立即模式中常用的接口、函数、结构、类型和返回值等。

本书编写新颖、内容全面、通俗易懂,特别是提供了 Direct 3D 立即模式的许多新技术和编程新方法。本书非常适合作为广用 Direct 3D 立即模式来进行游戏开发的人员的技术参考书。

本书无四川省版权防盗标识,不得
销售;版权所有,违者必究,举报有奖,
举报电话:(028)6636481、6241146、7600560。

Direct 3D 立即模式技术参考手册

——Direct X 6 技术参考手册系列丛书

杨 峰 周颖恒 等编

*

出 版 人 宋绍南

责任编辑 唐 晴

封面设计 小 唐

西南交通大学出版社出版发行

(成都二环路北一段 111 号 邮编 610031)

成都市报华印装厂印刷

*

开本:787mm×1092mm 1/16 印张:17.125

字数:408 千字 印数:1~5000 册

1999 年 6 月第 1 版 1999 年 6 月第 1 次印刷

ISBN7-81057-295-4/T·324

定价:30.00 元

目 录

第一部分 Direct 3D 立即模式概述

第一章 关于 Direct 3D 立即模式	(3)
第二章 为什么要用 Direct 3D 立即模式	(4)
第三章 Direct 3D 立即模式入门	(5)
3.1 三维坐标系统	(5)
3.1.1 Direct 3D 三维坐标系统	(5)
3.1.2 U-和 V-坐标系统	(6)
3.2 三维图形变换	(6)
3.2.1 平 移	(7)
3.2.2 旋 转	(7)
3.2.3 缩 放	(8)
3.3 多 边 形	(8)
3.3.1 面和顶点法向量	(8)
3.3.2 阴影模式	(9)
3.3.3 三角形插值	(11)
3.4 三角条和扇形	(11)
3.5 三角形光栅化规则	(12)
第四章 Direct 3D 立即模式的结构	(16)
4.1 结构概述	(16)
4.2 立即模式对象类型	(16)
4.2.1 DirectDraw 对象	(17)
4.2.2 DirectDrawSurface 对象	(17)
4.2.3 Direct3DDevice 对象	(17)
4.2.4 Direct3DMaterial 对象	(17)
4.2.5 Direct3DViewport 对象	(17)
4.2.6 Direct3DLight 对象	(18)
4.2.7 Direct 3D 执行缓冲对象	(18)

4.3 立即模式 COM 界面	(18)
4.4 画图元方法和执行缓冲	(18)
第五章 Direct 3D 立即模式的主要内容	(19)
5.1 DirectX 6 中立即模式的改变	(19)
5.1.1 与 DirectX 3 的兼容性	(19)
5.1.2 将 DirectX 3 的程序移植到 DirectX 6 上	(20)
5.2 画图元 (DrawPrimitive) 方法	(20)
5.2.1 画图元的 API 扩展	(20)
5.2.2 画图元功能结构	(20)
5.2.3 同时运用画图元和执行缓冲	(21)
5.2.4 一个简单的画图元例子	(22)
5.3 全局统一识别码 (GUID)	(23)
5.4 IDirect3D2 界面	(23)
5.5 设 备	(23)
5.6 视口与转换	(24)
5.6.1 变换管道	(25)
5.6.2 设置变换	(25)
5.6.3 创建和删除视口	(26)
5.6.4 矩 阵	(27)
5.6.5 世界变换	(28)
5.6.6 视口变换	(29)
5.6.7 投影变换	(31)
5.7 纹 理	(32)
5.7.1 表面、设备和句柄	(32)
5.7.2 纹理弯曲	(33)
5.7.3 纹理滤波和融合	(34)
5.7.4 递减纹理 (Mipmaps)	(34)
5.7.5 透明和半透明	(36)
5.8 灯 光	(36)
5.9 材 质	(39)
5.10 颜色与雾化	(40)
5.10.1 彩色光照	(40)
5.10.2 调色板项	(40)
5.10.3 雾	(40)
5.11 反 走 样	(41)
5.11.1 边缘反走样	(41)
5.11.2 普遍反走样	(42)
5.12 Direct 3D 与 DirectDraw 的结合	(42)

5.12.1 对象与接口	(42)
5.12.2 纹理贴图	(43)
5.12.3 Z-缓冲	(44)
5.12.4 RGBZ 支持	(44)
5.13 执行缓冲	(45)
5.14 运用执行缓冲	(45)
5.14.1 执行缓冲的结构	(46)
5.14.2 执行缓冲的内容	(47)
5.14.3 执行缓冲的创建	(49)
5.14.4 执行缓冲锁	(49)
5.14.5 填充执行缓冲	(49)
5.14.6 执行缓冲的解锁	(51)
5.14.7 执行缓冲的执行	(51)
5.15 状态和状态废弃	(51)
5.16 浮点精度	(52)
5.17 性能优化	(52)
5.17.1 PC 硬件加速	(53)
5.17.2 数据库和挑选	(53)
5.17.3 批处理方式画图元	(53)
5.17.4 光照提示	(54)
5.17.5 纹理大小	(54)
5.17.6 软件对硬件	(55)
5.17.7 三角形标志	(56)
5.17.8 执行时的剪取测试	(56)
5.17.9 整体性能提示	(58)
5.18 疑难解答	(58)
5.18.1 设备创建	(58)
5.18.2 没有可见物体	(58)
5.18.3 排错 (Debugging)	(59)
5.18.4 Borland 浮点初始化	(59)
5.18.5 杂 项	(60)
第六章 Direct 3D 执行缓冲教学	(61)
6.1 定义、原型和全局变量	(61)
6.1.1 头文件与包含	(62)
6.1.2 Imsample.c 中的常量	(62)
6.1.3 Imsample.c 中的宏	(63)
6.1.4 全局变量	(64)
6.1.5 函数原型	(66)

6.2 枚举 Direct 3D 设备	(68)
6.2.1 枚举回调函数	(68)
6.2.2 枚举函数	(70)
6.3 创建设备和接口	(71)
6.3.1 创建主 DirectDraw 表面和剪贴板对象	(71)
6.3.2 创建 Direct 3D 对象	(74)
6.3.3 创建 Direct 3D 设备	(75)
6.4 创建场景	(77)
6.5 填充执行缓冲	(82)
6.6 场景动画	(86)
6.7 渲 染	(86)
6.7.1 渲染场景	(87)
6.7.2 渲染单帧	(88)
6.8 矩阵的工作	(89)
6.8.1 设置透视变换	(89)
6.8.2 设置旋转变换	(90)
6.9 恢复和重画	(91)
6.9.1 恢复 Direct 3D 设备	(91)
6.9.2 恢复主表面	(92)
6.9.3 恢复所有的表面	(92)
6.9.4 在窗口移动后重画	(93)
6.9.5 在窗口改变大小后重画	(93)
6.9.6 重画客户区	(95)
6.9.7 更新视角	(96)
6.10 释放对象	(97)
6.10.1 释放 Direct 3D 对象	(97)
6.10.2 释放 Direct 3D 设备	(98)
6.10.3 释放主表面	(98)
6.10.4 释放场景对象	(99)
6.11 错误检查	(100)
6.11.1 检查击活的状态	(100)
6.11.2 报告标准错误	(101)
6.11.3 报告致命错误	(102)
6.11.4 显示报告字符串	(102)
6.12 转换位深度	(103)
6.12.1 将位深度转换为标志位	(103)
6.12.2 将标志位转换为位深度	(104)
6.13 主窗口程序	(105)
6.14 WinMain 函数	(107)

第二部分 Direct 3D 立即模式参考

第一章 接口函数	(113)
1.1 IDirect3D2 接口	(113)
1.1.1 IDirect3D2::CreateDevice	(114)
1.1.2 IDirect3D2::CreateLight	(114)
1.1.3 IDirect3D2::CreateMaterial	(115)
1.1.4 IDirect3D2::CreateViewport	(115)
1.1.5 IDirect3D2::EnumDevices	(116)
1.1.6 IDirect3D2::FindDevice	(117)
1.1.7 IDirect3D2::Initialize	(117)
1.2 IDirect3DDevice 接口	(117)
1.2.1 IDirect3DDevice::CreateExecuteBuffer	(118)
1.2.2 IDirect3DDevice::CreateMatrix	(119)
1.2.3 IDirect3DDevice::DeleteMatrix	(120)
1.2.4 IDirect3DDevice::Execute	(120)
1.2.5 IDirect3DDevice::GetMatrix	(121)
1.2.6 IDirect3DDevice::GetPickRecords	(121)
1.2.7 IDirect3DDevice::Initialize	(122)
1.2.8 IDirect3DDevice::Pick	(122)
1.2.9 IDirect3DDevice::SetMatrix	(123)
1.3 IDirect3DDevice2 接口	(124)
1.3.1 IDirect3DDevice2::AddViewport	(125)
1.3.2 IDirect3DDevice2::Begin	(126)
1.3.3 IDirect3DDevice2::BeginIndexed	(127)
1.3.4 IDirect3DDevice2::BeginScene	(128)
1.3.5 IDirect3DDevice2::DeleteViewport	(129)
1.3.6 IDirect3DDevice2::DrawIndexedPrimitive	(129)
1.3.7 IDirect3DDevice2::DrawPrimitive	(131)
1.3.8 IDirect3DDevice2::End	(133)
1.3.9 IDirect3DDevice2::EndScene	(133)
1.3.10 IDirect3DDevice2::EnumTextureFormats	(134)
1.3.11 IDirect3DDevice2::GetCaps	(134)
1.3.12 IDirect3DDevice2::GetClipStatus	(135)
1.3.13 IDirect3DDevice2::GetCurrentViewport	(135)
1.3.14 IDirect3DDevice2::GetDirect3D	(136)
1.3.15 IDirect3DDevice2::GetLightState	(136)

1.3.16	IDirect3DDevice2::GetRenderState.....	(137)
1.3.17	IDirect3DDevice2::GetRenderTarget.....	(138)
1.3.18	IDirect3DDevice2::GetStats	(138)
1.3.19	IDirect3DDevice2::GetTransform.....	(139)
1.3.20	IDirect3DDevice2::Index	(139)
1.3.21	IDirect3DDevice2::MultiplyTransform	(140)
1.3.22	IDirect3DDevice2::NextViewport.....	(141)
1.3.23	IDirect3DDevice2::SetClipStatus.....	(142)
1.3.24	IDirect3DDevice2::SetCurrentViewport.....	(142)
1.3.25	IDirect3DDevice2::SetLightState.....	(143)
1.3.26	IDirect3DDevice2::SetRenderState	(143)
1.3.27	IDirect3DDevice2::SetRenderTarget	(144)
1.3.28	IDirect3DDevice2::SetTransform	(145)
1.3.29	IDirect3DDevice2::SwapTextureHandles	(145)
1.3.30	IDirect3DDevice2::Vertex	(146)
1.4	IDirect3DExecuteBuffer 接口	(146)
1.4.1	IDirect3DExecuteBuffer::GetExecuteData	(147)
1.4.2	IDirect3DExecuteBuffer::Initialize	(148)
1.4.3	IDirect3DExecuteBuffer::Lock	(148)
1.4.4	IDirect3DExecuteBuffer::Optimize.....	(149)
1.4.5	IDirect3DExecuteBuffer::SetExecuteData.....	(149)
1.4.6	IDirect3DExecuteBuffer::UnLock	(149)
1.4.7	IDirect3DExecuteBuffer::Validate	(150)
1.5	IDirect3DLight 接口	(150)
1.5.1	IDirect3DLight::GetLight	(151)
1.5.2	IDirect3DLight::Initialize.....	(151)
1.5.3	IDirect3DLight::SetLight.....	(152)
1.6	IDirect3DMaterials2 接口	(152)
1.6.1	IDirect3DMaterial2::GetHandle	(153)
1.6.2	IDirect3DMaterial2::GetMaterial	(153)
1.6.3	IDirect3DMaterial::Initialize	(154)
1.6.4	IDirect3DMaterial::Reserve.....	(154)
1.6.5	IDirect3DMaterial2::SetMaterial	(154)
1.6.6	IDirect3DMaterial::Unreserve.....	(155)
1.7	IDirect3DTexture2 接口	(155)
1.7.1	IDirect3DTexture2::GetHandle	(155)
1.7.2	IDirect3DTexture::Initialize.....	(156)
1.7.3	IDirect3DTexture2::Load	(156)
1.7.4	IDirect3DTexture2::PaletteChanged.....	(157)

1.7.5 IDirect3DTexture::UnLoad	(157)
1.8 IDirect3DViewport2 接口	(157)
1.8.1 IDirect3DViewport2::AddLight	(158)
1.8.2 IDirect3DViewport2::Clear	(159)
1.8.3 IDirect3DViewport2::DeleteLight	(159)
1.8.4 IDirect3DViewport2::GetBackground	(160)
1.8.5 IDirect3DViewport2::GetBackgroundDepth	(161)
1.8.6 IDirect3DViewport2::GetViewport	(161)
1.8.7 IDirect3DViewport2::GetViewport2	(162)
1.8.8 IDirect3DViewport2::Initialize.....	(162)
1.8.9 IDirect3DViewport2::LightElements.....	(162)
1.8.10 IDirect3DViewport2::NextLight	(163)
1.8.11 IDirect3DViewport2::SetBackground	(163)
1.8.12 IDirect3DViewport2::SetBackgroundDepth	(164)
1.8.13 IDirect3DViewport2::SetViewport	(165)
1.8.14 IDirect3DViewport2::SetViewport2	(165)
1.8.15 IDirect3DViewport2::TransformVertices	(166)

第二章 D3D 重载符

(168)

2.1 D3D 重载结构	(168)
2.1.1 D3DLVERTEX 结构	(169)
2.1.2 D3DTLVERTEX 结构	(169)
2.1.3 D3DVECTOR 结构	(169)
2.1.4 D3DVERTEX 结构	(170)
2.2 D3D 重载操作符	(170)
2.2.1 传送授权操作符 (D3D 重载)	(170)
2.2.2 加运算重载符	(171)
2.2.3 赋值运算符	(171)
2.2.4 位等运算符	(172)
2.2.5 D3DMATRIX (D3D 模型重载)	(172)
2.2.6 除运算重载符	(173)
2.2.7 乘运算重载符	(174)
2.2.8 减运算符重载符	(174)
2.2.9 增减量运算符	(174)
2.2.10 向量优先操作符	(175)
2.3 D3D 重载辅助函数	(175)
2.3.1 矢量叉乘函数	(176)
2.3.2 点乘函数 (标量乘)	(176)
2.3.3 取模函数 (Magnitude)	(177)

2.3.4 求最大函数 (Max)	(177)
2.3.5 最大化函数 (Maximize)	(178)
2.3.6 求最小值函数 (Min)	(178)
2.3.7 求最小化函数 (Minimize)	(179)
2.3.8 单位化函数 (Normalize)	(180)
2.3.9 模平方函数 (SquareMagnitude)	(180)
第三章 宏	(181)
3.1 D3DDivide 宏 (D3D 除)	(181)
3.2 D3DMultiply 宏	(182)
3.3 D3DRGB 宏	(182)
3.4 D3DRGBA 宏	(182)
3.5 D3DSTATE_OVERRIDE 宏	(183)
3.6 D3DVAL 宏	(183)
3.7 D3DVALD 宏	(184)
3.8 RGB_GETBLUE 宏	(184)
3.9 RGB_GETGREEN 宏	(184)
3.10 RGB_GETRED 宏	(185)
3.11 RGB_MAKE 宏	(185)
3.12 RGB_TORGBA 宏	(185)
3.13 RGBA_GETALPHA 宏	(186)
3.14 RGBA_GETBLUE 宏	(186)
3.15 RGBA_GETGREEN 宏	(186)
3.16 RGBA_GETRED 宏	(187)
3.17 RGBA_MAKE 宏	(187)
3.18 RGBA_SETALPHA 宏	(187)
3.19 RGBA_TORGB 宏	(188)
第四章 回调函数	(189)
4.1 D3DENUMDEVICESCALLBACK	(189)
4.2 D3DENUMTEXTUREFORMATSCALLBACK	(190)
4.3 D3DVALIDATECALLBACK	(190)
第五章 结 构	(192)
5.1 D3DBRANCH 结构	(193)
5.2 D3DCLIPSTATUS 结构	(193)
5.3 D3DCOLORVALUE 结构	(194)
5.4 D3DDEVICEDESC 结构	(195)
5.5 D3DEXECUTEBUFFERDESC 结构	(197)

5.6	D3DEXECUTEDATA 结构.....	(198)
5.7	D3DFINDDEVICERESULT 结构.....	(199)
5.8	D3DFINDDEVICESEARCH 结构.....	(200)
5.9	D3DHVERTEX 结构.....	(202)
5.10	D3DINSTRUCTION 结构.....	(202)
5.11	D3DLIGHT2 结构.....	(203)
5.12	D3DLIGHTDATA 结构.....	(205)
5.13	D3DLIGHTINGCAPS 结构.....	(205)
5.14	D3DLIGHTINGELEMENT 结构.....	(206)
5.15	D3DLINE 结构.....	(207)
5.16	D3DLINEPATTERN 结构.....	(207)
5.17	D3DLVERTEX 结构.....	(208)
5.18	D3DMATERIAL 结构.....	(209)
5.19	D3DMATRIX 结构.....	(210)
5.20	D3DMATRIXLOAD 结构.....	(211)
5.21	D3DMATRIXMULTIPLY 结构.....	(211)
5.22	D3DPICKRECORD 结构.....	(211)
5.23	D3DPOINT 结构.....	(212)
5.24	D3DPRIMCAPS 结构.....	(213)
5.25	D3DPROCESSVERTICES 结构.....	(221)
5.26	D3DRECT 结构.....	(222)
5.27	D3DSPAN 结构.....	(222)
5.28	D3DSTATE 结构.....	(223)
5.29	D3DSTATS 结构.....	(224)
5.30	D3DSTATUS 结构.....	(224)
5.31	D3DTEXTURELOAD 结构.....	(226)
5.32	D3DTLVERTEX 结构.....	(226)
5.33	D3DTRANSFORMCAPS 结构.....	(227)
5.34	D3DTRANSFORMDATA 结构.....	(228)
5.35	D3DTRIANGLE 结构.....	(230)
5.36	D3DVECTOR 结构.....	(232)
5.37	D3DVERTEX 结构.....	(232)
5.38	D3DVIEWPORT 结构.....	(234)
5.39	D3DVIEWPORT2 结构.....	(234)
第六章	枚举类型	(237)
6.1	D3DANTIALIASMODE.....	(237)
6.2	D3DBLEND.....	(238)
6.3	D3DCMPFUNC.....	(239)

6.4	D3DCOLORMODEL	(240)
6.5	D3DCULL	(241)
6.6	D3DFILLMODE	(241)
6.7	D3DFOGMODE	(242)
6.8	D3DLIGHTSTATETYPE	(243)
6.9	D3DLIGHTTYPE	(244)
6.10	D3DOPCODE	(244)
6.11	D3DPRIMITIVETYPE	(246)
6.12	D3DRENDERSTATETYPE	(247)
6.13	D3DSHADEMODE	(254)
6.14	D3DTEXTUREADDRESS	(255)
6.15	D3DTEXTUREBLEND	(255)
6.16	D3DTEXTUREFILTER	(257)
6.17	D3DTRANSFORMSTATETYPE	(258)
6.18	D3DVERTEXTYPE	(259)
第七章	其它类型	(260)
7.1	D3DCOLOR	(260)
7.2	D3DVALUE	(260)
第八章	返回值	(261)

第一部分 Direct 3D立即模式概述

本部分是对 Direct 3D 立即模式的概述。内容分为以下章节：

- 关于Direct 3D立即模式
- 为什么要用Direct 3D立即模式
- Direct 3D立即模式入门
- Direct 3D立即模式的结构
- Direct 3D立即模式的主要内容
- Direct 3D执行缓冲教学

第一章 关于Direct 3D立即模式

Direct 3D是为了在运行Windows操作系统的计算机上开发各种各样的游戏和交互的计算机图形应用程序接口。它的任务是以与硬件无关的方式操作三维图形显示器。简单地说，Direct 3D是三维硬件的图形界面。

Direct 3D可以在两种方式下运行：立即模式和保留模式。保留模式是为下列人员开发的一种高级应用程序接口（API）：想加快开发速度或希望利用保留模式中预建的类和动画的程序员。

Microsoft是将Direct 3D立即模式作为低层的3-D API进行开发的。立即模式对于需要将游戏或者其它高性能多媒体程序移植到Windows操作系统上的程序员是很理想的。立即模式是一种在低层同图形加速硬件通讯的方法。Direct 3D保留模式建立在立即模式之上。

Direct 3D的一些先进特点如下：

- 可转换的z-缓冲
- 阴影
- 虚拟光照模型，有多盏灯及灯光种类
- 全面支持材质及纹理，包括递减纹理
- Ramp和RGB软件模拟
- 模型变换及剪裁（clipping）
- 与硬件无关
- 全面支持NT
- 支持Intel MMX结构

用立即模式而不用保留模式的程序员通常都精通编程，可能也对3-D图形有所了解。了解立即模式的最佳途径就是随SDK提供的源代码，它阐明了如何在真实程序中使用立即模式。

本部分并非介绍Direct 3D立即模式的编程。关于其编程，请参阅Direct 3D执行缓冲教程。

第二章 为什么要用Direct 3D立即模式

立即模式的基本方法是基于顶点、多边形和控制它们的命令。它允许对三维图形管道的立即转换、光照和光栅化，并且模拟硬件加速功能。（程序员将会被告知哪些功能由硬件提供，哪些由软件模拟。）已经有现成三维程序的和希望同硬件接触最少而达到最佳效果的程序员应当用立即模式，而不用保留模式。