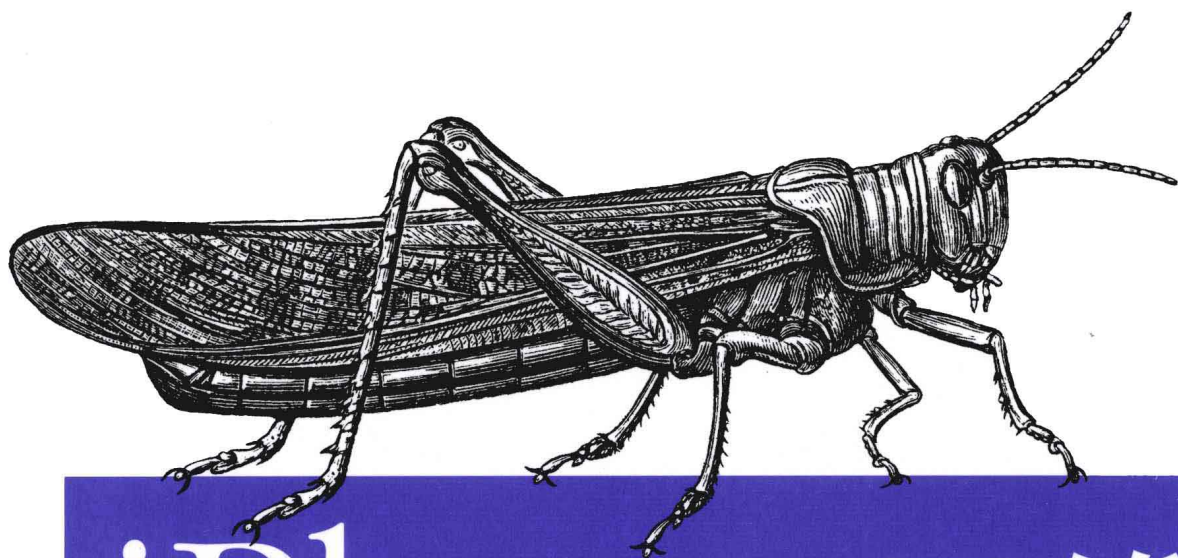
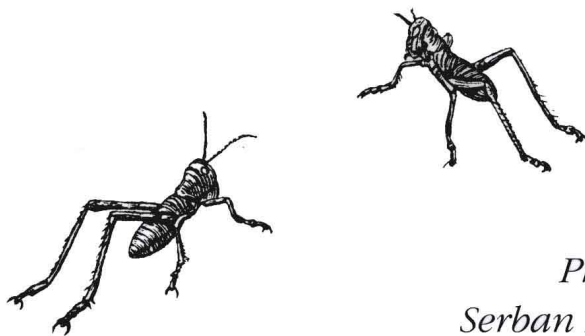


iPhone 3D Programming
基于OpenGL ES的图形应用程序设计



iPhone三维

程序设计



Philip Rideout 著
Serban Porumbescu 序
黄丽臣 译

O'REILLY®

清华大学出版社

iPhone三维程序设计

基于OpenGL ES的图形应用程序设计

Philip Rideout 著

黄丽臣 译

清华大学出版社

北 京

内 容 简 介

本书详细阐述了基于 OpenGL ES 的 3D 图形程序设计, 主要包括顶点和触摸点、景深和现实感、纹理和图像捕捉、混合操作与增强现实感、精灵和文本、高级光照和纹理以及优化操作等内容。本书还提供了相应的代码以及伪代码, 以帮助读者进一步理解渲染方案的实现过程。

本书适合作为高等院校计算机及相关专业的教学参考书, 也可作为相关开发人员的自学教材和参考手册。

Philip Rideout

iPhone 3D Programming, 1e

ISBN: 9780596804824

Copyright © 2010 by O'Reilly Media, Inc.

Simplified Chinese edition, jointly published by O'Reilly Media, Inc. and Tsinghua University Press, 2011.

Authorized translation of the English edition, 2010 O'Reilly Media Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

本书之英文原版由 O'Reilly Media Inc. 于 2010 年出版。

本中文简体翻译版由 O'Reilly Media Inc. 授权清华大学出版社于 2011 年出版发行。

版权所有, 未经书面许可, 本书的任何部分和全部不得以任何形式复制。

北京市版权局著作权合同登记号 图字: 01-2011-1817

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

iPhone 三维程序设计: 基于 OpenGL ES 的图形应用程序设计 / (美) 莱德奥特 (Rideout, P.) 著; 黄丽臣译. —北京: 清华大学出版社, 2011. 12

ISBN 978-7-302-25128-6

I. ①i… II. ①莱… ②黄… III. ①图形软件, 'OpenGL ES-程序设计 IV. ①TP391.41

中国版本图书馆 CIP 数据核字 (2011) 第 051659 号

责任编辑: 熊 健

封面设计: Karen Montgomery, 张 健

版式设计: 文森时代

责任校对: 张彩凤

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 装 者: 北京国马印刷厂

经 销: 全国新华书店

开 本: 185×230 印 张: 25.75 字 数: 530 千字

版 次: 2011 年 12 月第 1 版 印 次: 2011 年 12 月第 1 次印刷

印 数: 1~4000

定 价: 69.00 元

O'Reilly Media, Inc.介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 Make 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人员获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar 博客有口皆碑。”

——Wired

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——CRN

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去 Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

译者序

OpenGL ES (OpenGL for Embedded Systems) 是 OpenGL 三维图形 API 的子集, 针对手机、PDA 和游戏主机等嵌入式设备而设计。API 由图形软硬件行业协会 Khronos Group 定义推广, 该协会主要关注图形和多媒体方面的开放标准。

OpenGL ES 是从 OpenGL 裁剪定制而来的, 去除了 glBegin/glEnd、四边形 (GL_QUADS)、多边形 (GL_POLYGONS) 等复杂图元中许多非绝对必要的特性。经过多年发展, 现在主要有两个版本: OpenGL ES 1.x, 针对固定管线硬件; OpenGL ES 2.x, 针对可编程管线硬件。本书对上述两个版本均以支持。

iPhone 是结合照相手机、个人数码助理、媒体播放器以及无线通信设备的掌上设备, 支持电子邮件、移动电话、短信、网络浏览以及其他无线通信服务。iPhone 没有键盘, 而是创新地引入了多点触摸 (Multi-touch) 触摸屏界面, 在操作性上领先于其他品牌的手机。iPhone 无疑是一件非常伟大的产品, 甚至有人打出“无所不能的 iPhone”这样的口号。

OpenGL ES 与 iPhone 的完美结合定会释放出巨大的能量。针对于此, 本书深入 iPhone 图形程序设计这一核心问题的内部, 内容涉及顶点和触摸点、景深和现实感、纹理和图像捕捉、混合操作与增强现实感、精灵和文本、高级光照和纹理以及优化操作等。希望本书能够唤起读者的编程乐趣。

在本书的翻译过程中, 除黄丽臣外, 张超、赵雅利、吕莹莹、贾亚飞、古庆梅、张文京、左雪飞、孙方旭、庞云凤、李孝静、刘冬悦、李淑清、宋立帅、杨丽丽、李娜等也参与了部分翻译工作。

限于译者的水平, 译文中定有错误和不妥之处, 恳请读者指正。

译者

序

2010年2月23日这一天令人印象深刻,iTunes Music Store上的歌曲下载量突破了100亿首。遥想2003年4月 iTunes Music Store 刚刚上线之时,这一数字的确令人深思。是的,达到100亿首歌曲下载量仅用了7年的时间。

当读者阅读本书时,定对 iPhone 应用程序开发怀有莫大的兴趣。2010年1月5日,Apple 公司声称,在18个月内,App Store 中的应用程序下载量已达到了30亿,相信这一数字还将不断增长。这也创造了一个时代的奇迹!

在受邀为本书作序之前,本人也曾为自己的 iPhone 图形专著编写大纲并四处搜集素材,考虑到两书之间的相似性,这一邀请一度令我犹豫不决。但作为 O'Reilly 出版社的坚定支持者,我即刻改变了自己的初衷,因为我们都清醒地认识到,本书将会得到出版社方面的大力支持。

Philip 为本书的出版付出了卓有成效的工作,并引领读者快速步入 iPhone OpenGL ES 开发殿堂。本书主要包含 Xcode 以及 Objective-C 语言基础知识、固定管线程序设计(基于 OpenGL ES 1.1)和可编程图形管线程序设计(基于 OpenGL ES 2.0)、通过试验方法获取有效的数据、高级光照知识以及基于 Python 语言的距离场等内容。阅读完本书后,相信读者将具备坚实的 OpenGL ES 知识,并掌握某些基于 iPhone 以及其他平台的、较为高级的图形处理技术。

从阅读本书部分手稿之日起,文中的诸多内容即颇合我意,待拜读完本书的全部内容之后,这种感觉愈发强烈,相信读者亦会如此。

Serban Porumbescu 博士
Gameplay 高级工程师
Tapulous Inc.

前言

如何阅读本书

“程序设计将不再索然无味”——针对某些新晋程序设计语言或开发平台，这是一句最为常见的广告语。经过深思熟虑，我决定不再向 iPhone 图形程序设计添加任何冠名。无论您是专业开发人员抑或是狂热的图形爱好者，希望本书能够再次唤起您对编程的乐趣。

本书并非 OpenGL 编程指南，但书中内容涉及了大量的 OpenGL 概念，以服务于基于 iPhone 和 iPod touch 平台的 3D 程序设计，因此本书大部分内容均采用编程指南这一常见风格加以编写。这里，建议读者下载相应的示例代码并尝试对其进行修改、调试。需要说明的是，本书并不要求读者具备相关的图形学以及 iPhone SDK 背景知识，但坚实的 C++ 语言基础知识还是必要的。另外，Objective-C 语言并非阅读本书的必要条件，第 7 章中涵盖了与 Python 语言相关的若干内容，本书也将对此进行深入分析。

本书尽量避免涉及过多的数学知识，但也希望读者不必过分恐惧于某些较为基础的线性代数知识，相关内容都将在书中加以详细讨论。

如果读者精通 3D 图形学知识但不熟悉 iPhone 平台，则可根据相关内容实现针对性的学习并略过对应章节。例如，第 2 章主要讨论常见的 3D 图形学概念，略过本章对这一类读者而言将不会产生任何影响；相反，如果读者仅拥有 iPhone 平台经验但不具备相关的 3D 图形学知识，则可略过第 1 章中的 Objective-C 语言以及 Xcode 等内容。

最后，预祝读者阅读愉快，一如我编写本书时那样。

本书约定



图标对应于提示、建议等相关信息。



图标对应于警告、注意事项等相关信息。

本书代码

本书可以有效地帮助读者完成自己的工作。一般情况下，读者可能希望将本书的代码用于自己的开发和文档写作中，如果程序中只是用到了本书中的一小部分代码，读者不需要联系我们来获取相应的使用许可，除非准备大量地复制和使用书中大部分代码。销售或发布 O'Reilly 图书案例光盘是必须经过授权的。如果用户只是在解答问题时引用了本书的内容或者某段示例代码，那么不需要授权；如果用户打算将本书中的大部分示例代码移植到自己的产品或文档中，则需要经过我们的许可。

我们十分欢迎用户在自己的文章和产品中注明对本书的引用，请使用标题、作者、出版社和国际标准书号（ISBN）的顺序来编写引用说明。例如，*iPhone 3D Programming by Philip Rideout*. Copyright 2010 Philip Rideout, 978-0-596-80482-4.。

如果读者认为自己对例程代码的使用不属于上述情形的任何一种，欢迎与我们联系，地址为 permissions@oreilly.com。

联系方式

请把对本书的评论和问题发送给出版社：

美国：

O'Reilly Media, Inc.

1005 Gravenstein Highway North
Sebastopol, CA 95472

中国:

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室 (100035)
奥莱利技术咨询 (北京) 有限公司

O'Reilly 的每一本书都有专属网站, 你可以在那找到关于本书的相关信息, 包括勘误列表、示例代码以及其他的信息。本书的网站地址是:

<http://www.oreilly.com/catalog/9780596804824>

对于本书的评论和技术性的问题, 请发送电子邮件到:

bookquestions@oreilly.com

关于本书的更多信息、会议、资料中心和网站, 请访问以下网站:

<http://www.oreilly.com>

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

致谢

本书的出版源于 Harsha Kuntur, 他使得每次午餐均充斥着与 iPhone 相关的丰富的话题, 当然还包括 Stephen Holmes (他的无心插柳使我加入了 Apple 阵营) 和 David Banks (感谢他引领我步入图形领域)。

这里, 我还要感谢 O'Reilly 出版社的编辑 Brian Jepson, 感谢 John T. Kennedy 和 Jon C. Kennedy, 他们为本书的出版提供了极具价值的参考意见。感谢 Serban Porumbescu 为本书作序, 他的经验为本书平添了许多色彩。同时, 衷心感谢 Medical Simulation Corporation 的 Alex MacPhee 和 David Schmitt, 本书的编写得益于他们的帮助与耐心。感谢 Alex 对本书错误之处所做出的指正。

最后, 衷心感谢 Mona, 本书得以出版, 源于她的大力支持和不断鼓励。

目 录

O'Reilly Media, Inc.介绍.....	I
译者序	III
序	V
前 言	VII
第 1 章 快速入门.....	1
向 Apple 技术过渡.....	2
Objective-C 语言.....	3
OpenGL ES 的发展简史.....	3
选择正确的 OpenGL ES 版本.....	4
开始项目	5
安装 iPhone SDK	5
利用 Xcode 创建 OpenGL 模板程序.....	6
将程序发布至 iPhone 上	7
具有固定功能的 HelloArrow 程序	8
3D 应用程序的层次结构.....	8
启动项目	10
链接至 OpenGL 库和 Quartz 库	11
UIView 子类	12
挂接程序委托	17

生成图标并发布镜像文件	19
状态栏	19
渲染引擎接口的定义和使用	20
渲染引擎的实现	24
处理设备的方向	27
旋转动画	29
利用着色器编写 HelloArrow 程序	31
着色器	32
框架结构	34
GLView	34
实现 RenderingEngine	35
小结	40
第 2 章 数学知识	41
装配线	41
基于顶点的图元装配	42
顶点属性	44
顶点的生命周期	47
装配线类比	48
设置模型矩阵	52
缩放	52
平移操作	53
旋转操作	54
设置视图转换	55
设置投影转换	56
利用矩阵栈存储和恢复转换操作	59
动画	61
插值技术	61
基于四元数的旋转动画	62
基于 C++ 的向量	63

基于固定功能的 HelloCone 程序	66
RenderingEngine 声明	66
OpenGL 初始化操作和锥体的细分操作	68
3D 平滑旋转	72
Render()方法	73
基于着色器的 HelloCone 程序	75
小结	77
第 3 章 顶点和触摸点	79
读取触摸屏	79
采用顶点索引以节省内存空间	85
通过顶点缓冲对象获取性能提升	90
创建线框模式的观察器	95
参数化表面	96
接口设计	100
跟踪球旋转	102
渲染引擎的实现	105
简化的标签栏	107
渐变动画	111
小结	114
第 4 章 景深和现实感	115
深度缓冲区	115
深度缓冲冲突	117
创建并使用深度缓冲区	118
利用三角形填充线框	120
表面法线	124
向 OpenGL 传递法线	124
基于法线的数学知识	125

法线的转换	127
从参数表面中生成法线	128
光照计算	131
环境光	131
粗糙表面的漫反射光照	131
镜面光	133
向 ModelViewer 添加光照	134
使用光照特征	138
着色器分析	139
向 ModelViewer 中添加着色器	142
新型渲染引擎	144
像素光照	147
卡通着色	149
采用多边形偏移渲染线框对象	151
从 OBJ 文件中加载几何体	152
资源文件的管理	154
实现 ISurface 接口	157
小结	158
第 5 章 纹理和图像捕捉	159
向 ModelViewer 程序添加纹理	159
改进 IResourceManager 接口	161
生成纹理坐标	163
在 ES1::RenderingEngine 中开启纹理功能	166
在 ES2::RenderingEngine 中开启纹理功能	169
再访纹理坐标	174
基于过滤机制的抗锯齿操作	175
通过纹理链提升渲染质量和性能	178
改进 ModelViewer 程序以支持纹理链	180
纹理格式和类型	181

多种格式的数据加载	182
基于 PVRTC 的纹理压缩..	185
PowerVR SDK 和低精度纹理....	193
利用 Quartz 生成并转换 OpenGL 纹理..	195
纹理尺寸限制	198
POT 纹理的缩放操作..	201
利用相机创建纹理	202
实现 CameraTexture 程序中的渲染引擎	209
小结	214
第 6 章 混合操作与增强现实感	216
混合操作	216
预乘 alpha 值.....	218
混合操作中的注意事项	220
混合扩展操作	221
混合操作的作用	222
利用顶点颜色值调整纹理颜色	223
利用模板缓冲区实现反射计算	224
渲染圆形对象	226
利用模板缓冲区渲染反射对象	229
渲染对象	230
利用由前向后的混合方式渲染圆形对象	230
早期 iPhone 设备的模板替代方案	230
基于离屏 FBO 的抗锯齿技术.....	232
超采样示例	233
抖动采样	241
其他 FBO 效果	245
基于纹理的抗锯齿直线渲染	246

Holodeck 示例程序	249
程序框架	250
渲染穹顶、云彩以及文本	252
渲染 HUD	257
通过方位传感器替换按钮对象	262
添加加速计功能	262
使用真实的相机图像	268
小结	272
第 7 章 精灵和文本	273
文本渲染：绘制 FPS 计数器	273
利用 Python 语言生成字符纹理	275
渲染 FPS 文本	279
利用 glDrawTexOES 实施简化操作	284
基于距离场的文本	287
通过 Python 生成距离场	289
在 ES 1.1 中通过 alpha 测试使用距离场	291
利用片元着色器添加文本效果	292
平滑渲染与导数计算	292
实现轮廓线、光晕以及阴影效果	295
精灵表动画	298
图像合成以及多重纹理	299
OpenGL ES 与 UIKit 之间的混合操作	302
利用点精灵渲染纸屑、焰火等	304
SpringyStars 示例程序	305
弹簧体的物理原理	306
C++ 接口和 GLView	309
实现 ApplicationEngine	310
OpenGL ES 1.1 版本渲染引擎和加法混合计算	313
基于 OpenGL ES 2.0 的点精灵	315

小结	318
第 8 章 高级光照和纹理	319
OpenGL ES 1.1 中的纹理环境	319
纹理合成操作	322
凹凸纹理和 DOT3 光照	325
线性代数知识	327
生成基向量	327
基于 OpenGL ES 2.0 的法线贴图	329
基于 OpenGL ES 1.1 的法线贴图	332
生成对象空间内的法线贴图	333
立方体贴图反射	335
渲染立方体贴图	339
各向异性过滤	340
图像处理示例：模糊操作	342
基于混合方法的高性能解决方案	345
高斯模糊计算示例代码	346
小结	352
第 9 章 优化操作	353
度量方法	353
理解 CPU 和 GPU	354
使用 VBO	354
批处理	355
混合顶点属性	356
优化顶点格式	358
使用最佳拓扑结构和索引机制	359
光照优化	359
基于对象空间的光照	360
重述 DOT3 光照	360

烘焙光照	360
纹理优化	361
剔除和剪裁	361
多边形焊接	361
剪裁面	362
基于 CPU 的剔除操作	363
着色器性能	363
运行条件	364
kill 指令	364
纹理查找	365
利用顶点混合优化	365
蒙皮计算示例代码	366
基于 OpenGL ES 2.0 的蒙皮计算	368
基于 OpenGL ES 1.1 的蒙皮计算	370
计算权值和索引	373
聚缩效应	374
进一步阅读	374
附录 C++向量库	376