

国外著名高等院校
信息科学与技术优秀教材

MERCURY
LEARNING AND INFORMATION LLC

异步图书
www.epubit.com

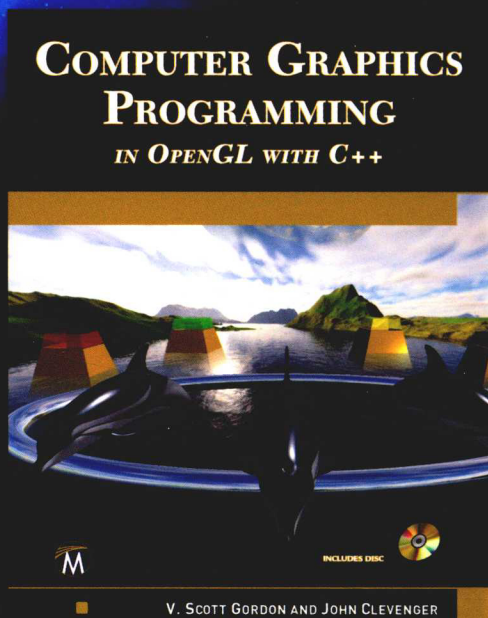
计算机图形学编程 (使用OpenGL和C++)

[美] V.斯科特·戈登 (V. Scott Gordon)

著

魏广程 沈瞳 译

[美] 约翰·克莱维吉 (John Clevenger)



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

国外著名高等院校
信息科学与技术优秀教材

计算机图形学编程

(使用OpenGL和C++)

[美] V.斯科特·戈登 (V. Scott Gordon)

著

魏广程 沈瞳 译

[美] 约翰·克莱维吉 (John Clevenger)

人民邮电出版社

北京

图书在版编目(CIP)数据

计算机图形学编程：使用OpenGL和C++ / (美) V. 斯科特·戈登 (V. Scott Gordon), (美) 约翰·克莱维吉 (John Clevenger) 著；魏广程, 沈瞳译. — 北京：人民邮电出版社, 2020. 2

国外著名高等院校信息科学与技术优秀教材
ISBN 978-7-115-52128-6

I. ①计… II. ①V… ②约… ③魏… ④沈… III. ①计算机图形学—高等学校—教材②C++语言—程序设计—高等学校—教材 IV. ①TP391.411②TP312.8

中国版本图书馆CIP数据核字(2019)第218712号

版权声明

Simplified Chinese translation copyright © 2020 by Posts and Telecommunications Press
ALL RIGHTS RESERVED

Computer Graphics Programming in OpenGL with C++, by V. Scott Gordon & John Clevenger.
ISBN: 978-1-683922-21-6

Copyright © 2019 by Mercury Learning and Information LLC.

本书中文简体版由 Mercury Learning and Information LLC 公司授权人民邮电出版社出版。未经出版者书面许可，对本书的任何部分不得以任何方式或任何手段复制和传播。

版权所有，侵权必究。

-
- ◆ 著 [美] V.斯科特·戈登 (V. Scott Gordon)
[美]约翰·克莱维吉 (John Clevenger)
译 魏广程 沈瞳
责任编辑 陈冀康
责任印制 王 郁 焦志炜
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京鑫正大印刷有限公司印刷
- ◆ 开本：787×1092 1/16
印张：16.75 彩插：4
字数：397 千字 2020 年 2 月第 1 版
印数：1-2 400 册 2020 年 2 月北京第 1 次印刷

著作权合同登记号 图字：01-2018-7739 号

定价：69.00 元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

内容提要

本书以 C++ 和 OpenGL 作为工具，教授计算机图形学编程。全书共 14 章和 3 个附录。首先从图形编程的基础和准备工作开始，依次介绍了 OpenGL 图像管线、图形编程数学基础、管理 3D 图形数据、纹理贴图、3D 模型、光照、阴影、天空和背景、增强表面细节、参数曲面、曲面细分、几何着色器，以及其他相关的图形编程技术。附录分别介绍了 Windows、macOS 平台上的安装设置，以及 Nsight 图形调试器的应用。本书每章最后配备了不同形式的习题，供读者巩固所学知识。

本书适合作为高等院校计算机科学专业的计算机图形编程课程的教材或辅导书，也适合对计算机图形编程感兴趣的读者自学。

前言

本书的主要目标是用作计算机专业本科 OpenGL 3D 图形编程相关课程的教材。同时，我们也付出了很大的努力，让本书成为一本无须配合课程使用的自学教材。在以这两者为目标的前提下，我们尽力将内容解释得简单而清晰。本书中的所有代码示例都已经尽可能地简化，同时没有破坏其完整性，以便读者可以直接运行。

我们期望本书与众不同的一点是，新手（刚接触 3D 图形编程的人）更容易学习。关于这个主题的学习资料从来都不匮乏，恰恰相反，很多新手刚入门的时候，相关的资料就扑面而来。我们刚接触 3D 图形编程的时候，就期望能遇见这样的教材——一步步解释基础概念，循序渐进并有序地梳理进阶概念，因此我们也尝试将本书编写成这样的教材。我们曾想将本书命名为 *shader programming made easy*（《轻松学着色器编程》），虽然我们并不认为有什么方法能真的让着色器编程变得“轻松”，但我们希望本书能够帮助你尽可能地达成这个目标。

本书使用 C++ 进行 OpenGL 编程教学。使用 C++ 学习图形编程有以下几个好处。

- 由于 OpenGL 的原生语言是 C，因此 C++ 程序可以直接进行 OpenGL 函数调用。
- C++ 编写的 OpenGL 应用程序通常有着很好的性能。
- C++ 提供 C 语言所没有的现代编程结构（类、多态等）。
- 在 OpenGL 社区中，C++ 是一个热门选项。许多 OpenGL 的教学资源有 C++ 版本。

值得一提的是，OpenGL 也存在着与其他语言绑定。常见的有 Java、C#、Python 等，但本书仅关注 C++。

本书与众不同的另一点是它有一个 Java 版，英文书名是 *Computer Graphics Programming in OpenGL with Java*。这两本书是按同样的节奏组织的，它们使用相同类型的章节编号、主题、图表、习题和讲解方式，其代码组织方式也尽可能地相似。诚然，使用 C++ 或 Java 编程肯定有着相当大的差异。尽管如此，我们相信这两本书提供了几乎相同的学习路径，甚至可以让选修同一门课的学生使用不同的语言版本作为教材。

需要说明的一点是，OpenGL 有着不同的版本（稍后简述）和不同的变体。例如，在标准 OpenGL（也称桌面 OpenGL）之外，还有一个变体叫作 OpenGL ES。它是为嵌入式系统（Embedded System）的开发而定制的（因此称为 ES）。“嵌入式系统”包括手机、游戏主机、汽车和工业控制系统之类的设备。OpenGL ES 的大部分内容是标准 OpenGL 的子集，删除了嵌入式系统通常用不到的很多操作。OpenGL ES 还增加了一些功能，通常是特定目标环境下的特定功能。本书侧重于标准 OpenGL。

OpenGL 的另一种变体称为 WebGL。WebGL 基于 OpenGL ES，它的设计目标是支持在浏览器中运行 OpenGL。WebGL 允许应用程序通过 JavaScript 进行 OpenGL ES 操作调用，从而简单地将 OpenGL 图形嵌入标准 HTML（Web）文档中。大多数现代 Web 浏览器，包括 Apple Safari、Google Chrome、Microsoft Internet Explorer、Mozilla Firefox 和 Opera，支

持 WebGL。由于 Web 编程超出了本书的讨论范围，因此本书不会涵盖 WebGL。不过，由于 WebGL 基于 OpenGL ES，而 OpenGL ES 又基于标准 OpenGL，因此本书所涵盖的大部分内容可以直接迁移到这些 OpenGL 变体的学习中。

3D 图形编程这个主题通常让人想起精美而宏大的画面。事实上，许多相关热门教材中充满了令人惊叹的场景，很大程度上是为了吸引读者翻阅他们的图库。虽然我们认同这些图例的激励作用，但我们的目标是教学而非令人惊叹。本书中的图像仅仅是示例程序的输出。由于本书只是入门教程，其渲染的场景应该无法让专家侧目。然而，本书呈现的技术确实是构成当今这些炫目 3D 效果的基础。

我们没有尝试写一本“OpenGL 参考大全”，因此，本书所涵盖的 OpenGL 部分只是其所有功能中的一小部分。我们的目标是以 OpenGL 作为基础工具，教授基于现代着色器的 3D 图形编程，并为读者提供足够深入的理解，以供进一步研究。

目标读者

本书的主要目标读者是计算机科学专业的学生（可以是本科在读学生），其实任何想要学习计算机科学相关知识的人也适合阅读本书。因此，我们假设读者有扎实的面向对象编程基础，至少相当于计算机科学专业大二或大三学生的水平。

还有一些本书没有涵盖的内容，因为我们假设读者已经掌握了足够的背景知识，包括：

- C++ 及其常用库，如标准模板库（Standard Template Library）；
- 熟悉集成开发环境（Integrated Development Environment, IDE），如 Visual Studio；
- 事件驱动编程概念；
- 基础矩阵代数、三角函数；
- 了解颜色模型，如 RGB、RGBA 等。

希望本书的潜在受众能够因为对其 Java 版的喜爱而进一步支持本书。正如前面所说的，我们期望看到这样一种情景——学生在同一门课中可以自由选择使用 C++ 或 Java 版本的教材。由于这两本书按同样的节奏对教学内容进行组织编排，因此我们认为可以尝试以这种开放的方式来开展课程教学和学习。

如何使用本书

本书从内容安排上适合从前往后阅读，即后面各章中的内容经常依赖于前面各章中所讲的内容。因此，在各章中来回跳跃地选择性阅读可能并不适合本书，读者最好逐章顺序阅读。

本书同时可以作为实用的动手指南。由于已经有许多其他偏理论的学习材料，因此读者应该将本书作为一本“练习册”，通过一边参考本书一边自己动手编程来理解基础概念。虽然我们为所有的示例提供了代码，但是想要真正理解这些概念，还是得自己动手“实现”这些代码——通过编程来搭建你自己的 3D 场景。

本书在第 2 章到第 14 章的最后都留给读者一些习题。有的题比较简单，仅仅需要对提供的代码进行简单改动就可以解决。那些标记为“项目”的习题，则需要读者花费更多的时间来解答，因为可能需要编写大量代码或者使用多个示例中用到的技术。少数标记为“研

究”的习题，则在本书中并没有提供解题需要学习的细节知识，我们鼓励读者进行自主学习并解答。

OpenGL 调用通常会有很长的参数列表。在撰写本书时，两位作者在每种情况下都会讨论是否要描述所有的参数。最终我们决定在最初的部分描述所有参数。随着主题深入，我们避免在每次 OpenGL 调用中陷入细枝末节的描述过程（因为调用的次数很多），以防读者失去对全局的理解。因此，在浏览示例时，读者需要在手边准备 OpenGL 和所使用的各种库的参考资料。

为此，我们建议结合一些优秀的在线资源使用本书。OpenGL 的文档是绝对必要的。有关各种命令的详细信息，可以利用搜索引擎，或访问 OpenGL 的官方网站获取。

我们的示例中用到了称作 GLM 的数学库。在安装 GLM（见附录）后，读者应该找到其在线文档并将其加入书签。

本书中经常用到的另一个库是 SOIL2，用于读取和处理纹理图像文件，读者可能也需要定期查阅它的文档。SOIL2 没有中心化的文档资源，但读者通过 Web 搜索可以找到一些例子。

还有许多关于 3D 图形编程的图书，我们建议与本书并行阅读（例如要解决各章后的“研究”问题）。以下是我们经常提到的 5 本。

- [SW15] Sellers et al. *OpenGL SuperBible*.
- [KS16] Kessenich et al. *OpenGL Programming Guide*.
- [WO13] Wolff, *OpenGL 4 Shading Language Cookbook*.
- [AS14] Angel and Shreiner, *Interactive Computer Graphics*.
- [LU16] Luna, *Introduction to 3D Game Programming with DirectX 12*.

配套资源

本书提供随书的配套资源供读者下载，其内容有：

- 书中所有 C++ / OpenGL 程序和相关的实用类文件以及 GLSL 着色器代码；
- 各种程序和示例中使用的模型和纹理文件；
- 用于制作天空和地平线的天空顶和立方体贴图图像文件；
- 用于照明和表面细节效果的法线贴图和高度贴图；
- 本书中所有的图表以图像文件形式提供。

上述文件也可以通过访问异步社区（www.epubit.com）上的本书页面获取。

教师辅助

我们鼓励大学或学院的教师获取本书的教师辅助包，其中包含以下附加项：

- 一套完整的 PowerPoint 幻灯片，涵盖本书中的所有主题；
- 本书中大多数章末习题的答案和所需代码；
- 基于本书的课程大纲示例；
- 每章用于讲解材料的额外内容。

教师辅助包可以通过联系出版商获取：contact@epubit.com.cn。

致谢

本书中的许多内容是基于我们之前出版的 *Computer Graphics Programming in OpenGL with Java*。我们需要感谢许多帮助我们完成上一本书的人，他们继续为本书的编写提供了帮助。Java 版早期的草稿被用于加州州立大学萨克拉门托分校的 CSc-155（高级计算机图形编程）课程中，得到了学生的指正，他们还给出了修改建议（包括代码）。两位作者要特别感谢 Mitchell Brannan、Tiffany Chiapuzio-Wong、Samson Chua、Anthony Doan、Kian Faroughi、Cody Jackson、John Johnston、Zeeshan Khaliq、Raymond Rivera、Oscar Solorzano、Darren Takemoto、Jon Tinney、James Womack 及 Victor Zepeda 的建议。

我们也从许多教师那里得到了很好的反馈，他们采用 *Computer Graphics Programming in OpenGL with Java* 作为课程教材，同时向我们分享了他们的教学经验。塔尔萨大学的 Mauricio Papa 博士和我们进行了几次对我们非常有帮助的邮件沟通。Sean McCrory 对光照（第 7 章）和柏林噪声（第 14 章）出现的问题进行了非常详细的修正。他们的建议帮助我们对本书进行了改进。我们还收到了许多来自不同学校的学生提出的问题，这些问题帮助我们评估了我们编写方法的优缺点。

我们对本系列书的第一次试用是在 2017 年的秋天，当时我们的同事 Pinar Muyan-Ozcelik 博士在她教授的 CSc-155 课程上第一次使用了 *Computer Graphics Programming in OpenGL with Java*，这让我们有机会评估我们是否实现了让本书成为“自学”资源的目标。课程进展顺利的同时，Pinar Muyan-Ozcelik 博士也为每章留存了问题和更正日志。这份日志帮我们对本书进行了许多改进。

Martín Lucas Golini 是 SOIL2 纹理图像处理库的开发者和维护者，也对本书表现出了极大的支持和热情。我们对他的帮助表示非常感谢。

Jay Turberville 来自于亚利桑那州 Scottsdale 的 Studio 522 Productions。他创建了本书英文版的封面和书中用到的海豚模型，学生们非常喜欢。Studio 522 Productions 制作出极高质量的 3D 动画和视频，以及自定义 3D 建模。我们很感谢 Turberville 慷慨地为本书创建这个精美的模型。

我们还要感谢其他一些艺术家和研究人员。他们非常慷慨地让我们使用他们的模型和纹理。来自 Planet Pixel Emporium 的 James Hastings-Trew 提供了许多行星表面纹理。Paul Bourke 允许我们使用他拥有的精彩的星域。斯坦福大学的 Marc Levoy 博士授权我们使用著名的“斯坦福龙”模型。Paul Baker 的凹凸贴图教程是我们在许多例子中使用的“圆环”模型的基础。我们还要感谢 Mercury Learning 允许我们使用《DirectX 12 3D 游戏开发实战》^①中的一些纹理。

Danny Kopec 博士向我们介绍了 Mercury Learning 公司，并向它的出版商 David Pallai 引荐了我们。Kopec 博士的《人工智能（第 2 版）》^②的成功出版让我们考虑与 Mercury 合作，我们与 Kopec 的电话交谈也对我们非常有帮助。Kopec 博士的早逝让我们深感悲痛，也对他没有机会看到本书的成书感到遗憾。

① 《DirectX 12 3D 游戏开发实战》已由人民邮电出版社出版（ISBN 978-7-115-47921-1）。——编者注

② 《人工智能（第 2 版）》已由人民邮电出版社出版（ISBN 978-7-115-48843-5）。——编者注

最后，我们要感谢 Mercury Learning 的 David Pallai 和 Jennifer Blaney，他们一直保持着对这个项目的热情并引导我们完成了本书的整个出版流程。

勘误

如果你在阅读本书时发现任何错误，请告诉我们！尽管我们尽了最大努力，但本书肯定还有错误。当收到错误报告时，我们将会尽最大努力尽快发布。我们建立了一个用于收集并发布勘误的网页：

<http://athena.ecs.csus.edu/~gordonvs/errata.html>^①

出版商 Mercury Learning 也保留了本书勘误表页面的链接。因此，如果我们的勘误页面的 URL 有变动，请查看 Mercury Learning 网站以获取最新链接。

参考资料

[AS14] E. Angel and D. Shreiner, *Interactive Computer Graphics: A Top-Down Approach with WebGL*, 7th ed. (Pearson, 2014).

[KS16] J. Kessenich, G. Sellers, and D. Shreiner, *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V*, 9th ed. (Addison-Wesley, 2016).

[LU16] F. Luna, *Introduction to 3D Game Programming with DirectX 12*, 2nd ed. (Mercury Learning, 2016).

[SW15] G. Sellers, R. Wright Jr., and N. Haemel, *OpenGL SuperBible: Comprehensive Tutorial and Reference*, 7th ed. (Addison-Wesley, 2015).

[WO13] D. Wolff, *OpenGL Shading Language Cookbook*, 2nd ed. (Packt Publishing, 2013).

① 本书中文版已经对这些勘误进行了修改。——编者注

作者简介

V.斯科特·戈登 (V. Scott Gordon) 博士已经在加州州立大学系统担任教授有 20 多年，目前在加州州立大学萨克拉门托分校教授高级图形和游戏工程课程。他撰写及合著了 30 多部出版物，涉及人工智能、神经网络、进化计算、软件工程、视频和策略游戏编程，以及计算机科学教育等多个领域。戈登博士在科罗拉多州立大学获得博士学位。他同时也是爵士鼓手和优秀的乒乓球运动员。

约翰·克莱维吉 (John Clevenger) 博士拥有超过 40 年的教学经验，教学内容包括高级图形、游戏架构、操作系统、VLSI 芯片设计、系统仿真和其他主题。他是多个用于图形和游戏架构教学的软件框架和工具的开发人员，其中包括我们 Java 版第一版书中所用到的 `graphicslib3D` 库。他是国际大学生程序设计竞赛 (ICPC) 的技术总监，负责监督 PC^2 的持续开发。 PC^2 是目前世界上使用较为广泛的编程竞赛支持系统。克莱维吉博士在加州大学戴维斯分校获得博士学位。

资源与支持

本书由异步社区出品，社区（<https://www.epubit.com/>）为您提供相关资源和后续服务。

配套资源

本书提供如下资源：

- 本书配套源代码；
- 书中用到的模型、图形、纹理和贴图等；
- 书中彩图文件。

要获得以上配套资源，请在异步社区本书页面中单击 **配套资源**，跳转到下载界面，按提示进行操作即可。注意：为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

如果您是教师，希望获得教学配套资源，请在社区本书页面中直接联系本书的责任编辑。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，单击“提交勘误”，输入勘误信息，单击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的 100 积分。积分可用于在异步社区兑换优惠券、样书或奖品。

详细信 息	写书 评	提交勘 误
----------	---------	----------

页码: 页内位置(行数): 勘误印次:

B I U -

字数统计
提交

扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并请在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术 etc。



异步社区



微信服务号

目 录

第 1 章 入门	1	第 3 章 数学基础	26
1.1 语言和库	1	3.1 3D 坐标系统	26
1.1.1 C++	2	3.2 点	26
1.1.2 OpenGL / GLSL	2	3.3 矩阵	27
1.1.3 窗口管理	2	3.4 变换矩阵	29
1.1.4 扩展库	3	3.4.1 平移矩阵	29
1.1.5 数学库	3	3.4.2 缩放矩阵	29
1.1.6 纹理管理	3	3.4.3 旋转矩阵	30
1.1.7 可选库	4	3.5 向量	31
1.2 安装和配置	4	3.5.1 点积的应用	32
参考资料	4	3.5.2 叉积的应用	33
第 2 章 OpenGL 图像管线	5	3.6 局部和世界空间	33
2.1 OpenGL 管线	5	3.7 视觉空间和合成相机	34
2.1.1 C++/OpenGL 应用 程序	6	3.8 投影矩阵	36
2.1.2 顶点着色器和片段 着色器	9	3.8.1 透视投影矩阵	36
2.1.3 曲面细分着色器	12	3.8.2 正射投影矩阵	37
2.1.4 几何着色器	13	3.9 LookAt 矩阵	38
2.1.5 光栅化	14	3.10 用来构建矩阵变换的 GLSL 函数	39
2.1.6 片段着色器	15	补充说明	40
2.1.7 像素操作	16	习题	40
2.2 检测 OpenGL 和 GLSL 错误	17	参考资料	41
2.3 从文件读取 GLSL 源代码	19	第 4 章 管理 3D 图形数据	42
2.4 从顶点构建对象	20	4.1 缓冲区和顶点属性	42
2.5 场景动画	21	4.2 统一变量	44
2.6 C++ 代码文件结构	23	4.3 顶点属性插值	45
补充说明	24	4.4 模型-视图和透视矩阵	46
习题	24	4.5 我们的第一个 3D 程序——一个 3D 立方体	47
参考资料	25	4.6 渲染一个对象的多个副本	53
		4.7 在同一个场景中渲染多个不同	

模型	57	补充说明	106
4.8 矩阵堆栈	59	习题	107
4.9 应对“Z冲突”伪影	64	参考资料	107
4.10 图元的其他选项	65	第7章 光照	108
4.11 性能优先的编程方法	66	7.1 光照模型	108
4.11.1 尽量减少动态内存空间 分配	66	7.2 光源	109
4.11.2 预先计算透视矩阵	67	7.3 材质	111
4.11.3 背面剔除	68	7.4 ADS 光照计算	112
补充说明	69	7.5 实现 ADS 光照	114
习题	70	7.5.1 Gouraud 着色(双线性 光强插值法)	115
参考资料	70	7.5.2 Phong 着色	120
第5章 纹理贴图	71	7.6 结合光照与纹理	124
5.1 加载纹理图像文件	71	补充说明	126
5.2 纹理坐标	72	历史记录	126
5.3 创建纹理对象	74	习题	126
5.4 构建纹理坐标	74	参考资料	127
5.5 将纹理坐标载入缓冲区	75	第8章 阴影	128
5.6 在着色器中使用纹理: 采样器 变量和纹理单元	76	8.1 阴影的重要性	128
5.7 纹理贴图: 示例程序	77	8.2 投影阴影	129
5.8 多级渐远纹理贴图	79	8.3 阴影体	129
5.9 各向异性过滤	83	8.4 阴影贴图	130
5.10 环绕和平铺	84	8.4.1 阴影贴图(第1轮)—— 从光源位置“绘制” 物体	131
5.11 透视变形	85	8.4.2 阴影贴图(中间步骤)—— 将Z缓冲区复制到 纹理	132
5.12 材质——更多 OpenGL 细节	86	8.4.3 阴影贴图(第2轮)—— 渲染带阴影的场景	132
补充说明	86	8.5 阴影贴图示例	135
习题	87	8.6 阴影贴图的伪影	139
参考资料	87	8.7 柔和阴影	142
第6章 3D 模型	88	8.7.1 现实世界中的柔和 阴影	142
6.1 程序构建模型——构建一个 球体	88	8.7.2 生成柔和阴影——百分比 邻近滤波(PCF)	143
6.2 OpenGL 索引——构建一个 环面	94		
6.2.1 环面	94		
6.2.2 OpenGL 中的索引	95		
6.3 加载外部构建的模型	99		

8.7.3 柔和阴影/PCF 程序	146
补充说明	148
习题	148
参考资料	149
第 9 章 天空和背景	150
9.1 天空盒	150
9.2 天空穹顶	152
9.3 实现天空盒	153
9.3.1 从头开始构建天空盒	153
9.3.2 使用 OpenGL 立方体 贴图	156
9.4 环境贴图	158
补充说明	161
习题	162
参考资料	163
第 10 章 增强表面细节	164
10.1 凹凸贴图	164
10.2 法线贴图	166
10.3 高度贴图	172
补充说明	174
习题	175
参考资料	175
第 11 章 参数曲面	176
11.1 二次贝塞尔曲线	176
11.2 三次贝塞尔曲线	177
11.3 二次贝塞尔曲面	180
11.4 三次贝塞尔曲面	181
补充说明	183
习题	183
参考资料	183
第 12 章 曲面细分	184
12.1 OpenGL 中的曲面细分	184
12.2 贝塞尔曲面细分	188
12.3 地形、高度图的细分	193
12.4 控制细节级别 (LOD)	198
补充说明	200
习题	201
参考资料	201
第 13 章 几何着色器	202
13.1 OpenGL 中的逐个图元 处理	202
13.2 修改图元	203
13.3 删除图元	206
13.4 添加图元	207
13.5 更改图元类型	209
补充说明	211
习题	211
参考资料	211
第 14 章 其他技术	212
14.1 雾	212
14.2 复合、混合、透明度	213
14.3 用户定义剪裁平面	218
14.4 3D 纹理	219
14.5 噪声	223
14.6 噪声应用——大理石	227
14.7 噪声应用——木材	230
14.8 噪声应用——云	233
14.9 噪声应用——特殊效果	236
补充说明	238
习题	238
参考资料	239
附录 A PC (Windows) 上的安装与 设置	240
A.1 安装库和开发环境	240
A.1.1 安装开发环境	240
A.1.2 安装 OpenGL / GLSL	240
A.1.3 准备 GLFW	240
A.1.4 准备 GLEW	241
A.1.5 准备 GLM	241
A.1.6 准备 SOIL2	241

A.1.7 准备共享的“lib”和 “include”文件夹	241
A.2 在 MS Visual Studio 中开发和 部署 OpenGL 项目	242
参考资料	243

附录 B Macintosh (macOS) 平台上的 安装与设置244

B.1 安装库和开发环境	244
B.1.1 准备并安装依赖库	244
B.1.2 准备开发环境	245
B.2 修改 Mac 的 C++ / OpenGL / GLSL 应用程序代码	246

B.2.1 修改 C++代码	247
B.2.2 修改 GLSL 代码	247
B.2.3 补充说明	248
参考资料	248

附录 C 使用 Nsight 图形调试器249

C.1 关于 NVIDIANSight	249
C.2 设置 Nsight	249
C.3 在 Nsight 中运行 C++/OpenGL 应用程序	250
参考资料	252

第 1 章 入门

图形编程是计算机科学中最具挑战性的主题之一，并因此而闻名。当今，图形编程是基于着色器的——也就是说，有些程序是用诸如 C++ 或 Java 等标准编程语言编写的，并运行在 CPU 上；而另一些是用专用的着色器语言编写的，并直接运行在显卡（GPU）上。着色器编程的学习曲线很陡峭，以致哪怕是绘制简单的东西，也需要一系列错综复杂的步骤，把图形数据从一个“管线”中传递下去才能完成。现代显卡能够并行处理数据，即使是绘制简单的形状，图形程序员也必须理解 GPU 的并行架构。

虽然这并不简单，但回报是超强的渲染能力。电子游戏中涌现出来的令人惊艳的虚拟现实和好莱坞电影中越来越逼真的特效，很大程度上是由着色器编程的进步带来的。如果阅读本书是你进入 3D 图形世界的第一步，那么你正在开始接受一个对自己的挑战。挑战的奖励不仅仅是漂亮的图片，还有过往不敢想象的对机器的掌控程度。欢迎来到激动人心的计算机图形编程世界！

1.1 语言和库

现代图形编程使用图形库完成，也就是说，程序员编写代码时，调用一个预先定义的库（或者一系列库）中的函数，由这个库来提供对底层图形操作的支持。现在有很多图形库，但常见的平台无关图形编程库叫作 OpenGL（Open Graphics Library，开放图形库）。本书将会介绍如何在 C++ 中使用 OpenGL 进行 3D 图形编程。

在 C++ 中使用 OpenGL 需要配置多个库。这里按照个人需求，可以有一系列令人眼花缭乱的选择。在本节中，我们会介绍哪几种库是必要的，各种库的一些常见选择，以及我们在本书中选择的库。

总的来说，你需要以下这些语言和库：

- C++ 开发环境；
- OpenGL / GLSL；
- 窗口管理；
- 扩展库；
- 数学库；
- 纹理管理。

读者可能需要进行几个准备步骤，以保证这几种库已安装在系统中，并可以正常使用。下面几个小节将简单介绍每一种语言和库。安装和配置的更多细节请参阅附录。