

详细介绍了 Vulkan 及三维图形学必知必会的基础知识  
给出了大量基于 Vulkan 实现的基本特效及高级特效  
提供了基于 Vulkan 的完整游戏案例  
多平台完整源代码下载

# Vulkan

## 开发实战详解

吴亚峰◎编著



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

# Vulkan

## 开发实战详解

多 平 台 完 整 源 代 码

吴亚峰◎编著



人民邮电出版社

北 京

## 图书在版编目 (CIP) 数据

Vulkan开发实战详解 / 吴亚峰编著. — 北京 : 人民邮电出版社, 2019.7  
ISBN 978-7-115-50939-0

I. ①V… II. ①吴… III. ①图形软件—程序设计  
IV. ①TP391.41

中国版本图书馆CIP数据核字 (2019) 第042418号

## 内 容 提 要

本书共分为 19 章, 介绍了 Vulkan 的诞生、特点、开发环境的搭建以及运行机制、渲染管线和调试技术, 着色器编程语言——GLSL、投影及各种变换、光照、纹理映射、3D 模型的加载、混合与雾、两种测试及片元丢弃、顶点着色器的妙用、片元着色器的妙用、真实光学环境的模拟、阴影及高级光照、几种高级着色器特效、骨骼动画、Vulkan 的性能优化等, 最后以一个休闲游戏——方块历险记的案例来展示 Vulkan 的功能与技术。本书按照必知必会的基础知识、基于 Vulkan 实现基本特效以及高级特效、完整游戏案例的顺序, 循序渐进地进行详细讲解, 适合不同需求、不同水平层次的读者。为了便于读者学习, 随书提供了书中所有案例的完整源代码 (书中所有案例都给出了安卓版和 Windows 版, 最后的大案例还进一步给出了 macOS、iOS 和 Linux 版), 最大限度地帮助读者快速掌握各方面的开发技术。

本书适合游戏开发者、程序员学习, 也可以作为大专院校相关专业的师生学习用书和培训学校的教材。

- 
- ◆ 编 著 吴亚峰  
责任编辑 张 涛  
责任印制 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: 46.5 彩插: 2  
字数: 1 232 千字 2019 年 7 月第 1 版  
印数: 1—2 000 册 2019 年 7 月北京第 1 次印刷
- 

定价: 139.00 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

# 前言

## 为什么要写这样的一本书

作为一种跨平台的 2D 和 3D 图形应用程序接口，Vulkan 因为高性能和低开销而大受欢迎，虽然面市不久，但市面上目前已有不少支持 Vulkan 的游戏和应用，如《Doom》《Dota2》《极品飞车——无极限》等。同时由于 3D 应用程序开发比较复杂，而 Vulkan 比传统的 OpenGL 更加复杂，造成入门门槛较高，初学者无从下手。根据这种情况，作者结合多年从事 3D 游戏及应用开发的经验编写此书。

Vulkan 起初被称为 GLNext，了解一些 3D 领域的技术人员都知道，当下多种平台上的 3D 应用开发很多是基于 OpenGL 的，但面向多线程任务设计的 OpenGL 在当下的处境比较尴尬，多线程的问题严重制约了新一代 GPU 渲染能力的发挥。

Vulkan 则良好地解决了这一问题，其原生支持多线程并发渲染，留给了开发人员充分的发挥空间。Vulkan 的许多新特性也使渲染的 3D 场景光影效果更加真实，实现过程更加迅速。本书在给出实际的案例时涉及了安卓、Windows、macOS、iOS、Linux 等主流平台，充分考虑到了各个不同主流目标平台读者的需求。

经过两年多见缝插针式的奋战，本书终于交稿了。回顾写书的这两年时间，不禁为自己能最终完成这个耗时费力的“大制作”而感到欣慰。同时也为自己能将从事游戏与应用开发近 15 年来积累的宝贵经验以及编程感悟，分享给正在开发阵线上埋头苦干的广大开发人员而感到高兴。

贾岛的《剑客》一诗有言：“十年磨一剑，霜刃未曾试，今日把示君，谁有不平事？”从 1998 年首次接触 Java 与 OpenGL 算起，到现在已经 20 多年了。作者希望用 20 多年的知识和经验打磨成的“利剑”，能够帮助广大读者在实际工作中披荆斩棘、奋勇向前。

## 本书特点

### 1. 内容丰富，由浅入深

本着“起点低，终点高”的原则，本书涵盖从 Vulkan 必知必会的基础知识到基于 Vulkan 实现各种高级特效，最后还给出了一个完整的 3D 游戏案例。这样的内容组织使得 3D 应用开发初学者可以一步一步成长为 3D 开发的资深人员，符合绝大部分想学习 3D 应用开发的学生与程序开发人员以及相关技术人员的需求。

### 2. 结构清晰，讲解到位

本书配合每个需要讲解的知识点都给出了丰富的插图与完整的案例，使得初学者易于上手，有一定基础的读者便于深入。书中所有的案例均是根据作者多年的开发心得进行设计的，结构清晰明朗，便于读者学习。同时书中还给出了很多作者多年来积累的编程技巧与心得，具有很高的参考价值。

### 3. 非常实用的案例源代码

为了便于读者学习，随书提供了书中所有案例的完整源代码（书中所有案例都给出了安卓版和 Windows 版，最后的大案例还进一步给出了 macOS、iOS 和 Linux 版），最大限度地帮助读者

快速地掌握各方面的开发技术。

内容导读

本书共分为 19 章，内容按照必知必会的基础知识、基于 Vulkan 实现基本特效以及高级特效、完整游戏案例的顺序循序渐进地进行详细讲解。

| 章 名                               | 主要内容                                                                                                                                 |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| 第 1 章 初识 Vulkan                   | 本章简要介绍了 Vulkan 的诞生、特点、开发环境的搭建以及运行机制                                                                                                  |
| 第 2 章 渲染管线和调试技术                   | 本章主要介绍了渲染管线、着色器预编译和 Vulkan 调试技术等，为以后的 Vulkan 项目开发打下良好的基础                                                                             |
| 第 3 章 着色器编程语言——GLSL               | 本章对可以用于实现 Vulkan 可编程渲染管线着色器的 GLSL 进行了系统地介绍，为后面各方面的深入学习打下了基础                                                                          |
| 第 4 章 投影与各种变换                     | 本章介绍了 3D 开发中投影、各种变换的原理与实现，同时还介绍了几种不同的绘制方式                                                                                            |
| 第 5 章 光照                          | 本章介绍了 Vulkan 中光照的基本原理与实现、点法向量与面法向量的区别以及光照的顶点计算与片元计算的差别等                                                                              |
| 第 6 章 纹理映射                        | 本章介绍了纹理映射的基本原理与使用，同时还介绍了不同的纹理拉伸与采样方式、多重过程纹理技术以及压缩纹理等                                                                                 |
| 第 7 章 更逼真的场景——3D 模型的加载            | 本章介绍了如何使用自定义的加载工具类直接加载通过 3D Max 创建的 3D 立体物体模型                                                                                        |
| 第 8 章 独特的场景渲染技术——混合与雾             | 本章主要介绍了混合以及雾的基本原理与使用                                                                                                                 |
| 第 9 章 常用 3D 开发小技巧                 | 本章主要介绍了一些常用的 3D 开发技巧，包括标志板、灰度图地形、高真实感地形、天空盒与天空穹、简单镜像技术以及非真实感绘制等                                                                      |
| 第 10 章 两种测试及片元丢弃                  | 本章主要介绍了 Vulkan 中经常使用的两种测试及片元丢弃，包括剪裁测试、模板测试、片元丢弃操作以及任意剪裁平面等                                                                           |
| 第 11 章 顶点着色器的妙用                   | 本章主要介绍如何通过顶点着色器实现几种酷炫效果，包括飘扬的旗帜、扭动的软糖、展翅飞翔的雄鹰、吹气特效等                                                                                  |
| 第 12 章 片元着色器的妙用                   | 本章介绍了如何通过片元着色器实现几种酷炫效果，包括程序纹理、数字图像处理技术、分形着色器、3D 纹理的妙用、体积雾以及粒子系统火焰特效等                                                                 |
| 第 13 章 真实光学环境的模拟                  | 本章介绍如何通过 Vulkan 模拟现实环境中的一些光学效果，如反射、折射、凹凸映射、镜头光晕等。同时本章还介绍了常用的投影贴图、绘制到纹理、高级镜像以及高真实感水面倒影等                                               |
| 第 14 章 阴影及高级光照                    | 本章介绍如何通过 Vulkan 模拟现实世界的阴影及高级光照，主要包括平面映射、阴影映射、阴影贴图等几个方面。同时本章还介绍了几种常见的技术，即多重渲染目标、聚光灯高级光源、延迟渲染以及环境光遮蔽等                                  |
| 第 15 章 几种高级着色器特效                  | 本章主要介绍了一些常用的高级着色器特效，如运动模糊、遮挡透视效果、积雪效果、背景虚化、泛光以及体绘制等                                                                                  |
| 第 16 章 骨骼动画                       | 本章介绍了 3D 游戏开发中常用的骨骼动画技术，包括自主开发的骨骼动画、ms3d 骨骼动画文件的加载以及自定义格式骨骼动画的加载等                                                                    |
| 第 17 章 让应用运行得更流畅——性能优化            | 本章讨论了一些在使用 Vulkan 开发 3D 游戏、应用过程中的性能优化问题，包括着色器代码的优化、纹理使用过程中的优化以及 3D 图形绘制过程中的优化等。同时本章还介绍了几种常见的技术，即图元重启、几何体实例渲染、遮挡查询、计算着色器的使用以及多线程并发渲染等 |
| 第 18 章 杂项                         | 本章主要介绍了一些与 Vulkan 应用开发相关的不太容易分类的知识与技术，主要包括四元数旋转、3D 拾取、多重采样抗锯齿、保存屏幕图像、Windows 系统窗口缩放、曲面细分着色器与几何着色器，以及苹果与 Linux 平台下 Vulkan 应用的开发等      |
| 第 19 章 基于 Vulkan 的 3D 休闲游戏——方块历险记 | 本章将通过介绍“方块历险记”游戏在 Android 平台以及 Windows、macOS、iOS、Linux 等平台上的设计与实现，对使用 Vulkan 技术开发 3D 休闲类游戏的步骤做详细讲解                                   |

本书内容丰富，从基本知识到高级特效，从简单的应用程序到完整的 3D 游戏案例，适合不同需求、不同水平层次各类读者。

- 初学 Vulkan 应用开发的读者

本书包括在各个主流平台下进行 3D 应用开发的知识，内容由浅入深，配合详细的案例，非常适合 3D 游戏、应用开发的初学者循序渐进地进行学习，最终成为 3D 游戏、应用开发的资深人员。

- 有一定 3D 开发基础并且希望进一步深入学习 Vulkan 高级开发技术的读者

本书不仅包括了 Vulkan 开发的基础知识，同时也包括了基于 Vulkan 实现高级特效的内容以及完整的游戏案例，有利于有一定基础的开发人员进一步提高开发水平与能力。

- 想学习图形学的读者

本书内容组织不单聚焦于 Vulkan 技术本身，还介绍了很多图形学方面的知识与技术，做到了理论联系实际，也非常适合想基于 Vulkan 学习图形学的读者。

## 作者简介

吴亚峰，毕业于北京邮电大学，后留学澳大利亚卧龙岗大学取得硕士学位。1998 年开始从事 Java 应用开发，有多年的 Java 开发与培训经验。主要的研究方向为 Vulkan、OpenGL ES、手机游戏、以及 VR/AR。曾任 3D 游戏、VR/AR 独立软件工程师，并兼任百纳科技软件培训中心首席培训师。近十年来为数十家著名企业培养了上千名高级软件开发人员，曾编写过《OpenGL ES 3x 游戏开发(上下卷)》《Unity 案例开发大全》(第 1 版、第 2 版)、《VR 与 AR 开发高级教程——基于 Unity》《H5 和 WebGL 3D 开发实战详解》《Android 应用案例开发大全》(第 1 版~第 4 版)、《Android 游戏开发大全》(第 1 版~第 4 版)等多本畅销技术书。2008 年初开始关注 Android 平台下的 3D 应用开发，并开发出一系列优秀的 Android 应用程序与 3D 游戏。

本书在编写过程中得到了百纳科技软件培训中心的大力支持，同时刘易周、宋润坤、张杰义、毛煜、尹豆、官端亮、李昀阳、刘聪颖、梁超以及作者的家人为本书的编写提供了很多帮助，在此表示衷心地感谢！

由于作者的水平和学识有限，且书中涉及的知识较多，难免有错误疏漏之处，敬请广大读者批评指正，并多提宝贵意见。本书责任编辑联系邮箱为 [zhangtao@ptpress.com.cn](mailto:zhangtao@ptpress.com.cn)。

作者



# 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。

## 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](mailto: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](http://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、网络技术等。



异步社区



微信服务号

# 目 录

|                                  |    |                               |     |
|----------------------------------|----|-------------------------------|-----|
| 第 1 章 初识 Vulkan                  | 1  | 2.4 Vulkan GPU 大 PK           | 89  |
| 1.1 Vulkan 概览                    | 1  | 2.4.1 移动端 GPU 的 4 大家族         | 89  |
| 1.1.1 Vulkan 简介                  | 1  | 2.4.2 PC 端 GPU 中 3 大家族        | 91  |
| 1.1.2 支持 Vulkan 的游戏概览            | 2  | 2.5 本章小结                      | 92  |
| 1.2 搭建开发环境                       | 4  | 第 3 章 着色器编程语言——GLSL           | 93  |
| 1.2.1 Android 平台开发环境的配置          | 4  | 3.1 着色语言概述                    | 93  |
| 1.2.2 Windows 平台开发环境的配置          | 8  | 3.2 着色语言基础                    | 94  |
| 1.3 第一个 Vulkan 程序                | 15 | 3.2.1 数据类型概述                  | 94  |
| 1.3.1 案例的运行效果                    | 15 | 3.2.2 数据类型的基本使用               | 98  |
| 1.3.2 Vulkan 应用程序的基本架构           | 15 | 3.2.3 运算符                     | 100 |
| 1.3.3 3 色三角形案例相关类的介绍             | 20 | 3.2.4 各个数据类型的构造函数             | 102 |
| 1.3.4 Vulkan 中的常用方法              | 22 | 3.2.5 类型转换                    | 104 |
| 1.3.5 MyVulkanManager 类的<br>基本结构 | 28 | 3.2.6 存储限定符                   | 104 |
| 1.3.6 创建 Vulkan 实例               | 31 | 3.2.7 插值限定符                   | 107 |
| 1.3.7 获取物理设备列表                   | 32 | 3.2.8 一致块                     | 108 |
| 1.3.8 创建逻辑设备                     | 32 | 3.2.9 layout 限定符              | 109 |
| 1.3.9 创建命令缓冲                     | 33 | 3.2.10 流程控制                   | 111 |
| 1.3.10 获取设备中支持图形<br>工作的队列        | 35 | 3.2.11 函数的声明和使用               | 113 |
| 1.3.11 初始化交换链                    | 35 | 3.2.12 片元着色器中浮点及整型<br>变量精度的指定 | 114 |
| 1.3.12 创建深度缓冲                    | 41 | 3.2.13 程序的基本结构                | 114 |
| 1.3.13 创建渲染通道                    | 43 | 3.3 特殊的内建变量                   | 115 |
| 1.3.14 创建帧缓冲                     | 46 | 3.3.1 顶点着色器中的内建变量             | 115 |
| 1.3.15 创建绘制用物体                   | 47 | 3.3.2 片元着色器中的内建变量             | 116 |
| 1.3.16 初始化渲染管线                   | 51 | 3.3.3 内建常量                    | 117 |
| 1.3.17 创建栅栏和初始化呈现信息              | 62 | 3.4 着色语言的内置函数                 | 117 |
| 1.3.18 初始化基本变换矩阵、<br>摄像机矩阵、投影矩阵  | 63 | 3.4.1 角度转换与三角函数               | 118 |
| 1.3.19 执行绘制                      | 63 | 3.4.2 指数函数                    | 119 |
| 1.3.20 销毁相关对象                    | 66 | 3.4.3 常见函数                    | 120 |
| 1.3.21 整体流程的执行                   | 67 | 3.4.4 几何函数                    | 123 |
| 1.3.22 顶点着色器和片元着色器               | 68 | 3.4.5 矩阵函数                    | 124 |
| 1.4 Vulkan 中立体物体的构建              | 70 | 3.4.6 向量关系函数                  | 125 |
| 1.5 本章小结                         | 71 | 3.4.7 纹理采样函数                  | 126 |
| 第 2 章 渲染管线和调试技术                  | 72 | 3.4.8 微分函数                    | 129 |
| 2.1 渲染管线                         | 72 | 3.4.9 浮点数打包与解包函数              | 129 |
| 2.2 着色器的预编译                      | 79 | 3.5 用 invariant 修饰符避免值变<br>问题 | 130 |
| 2.3 Vulkan 调试技术                  | 81 | 3.6 预处理器                      | 131 |
|                                  |    | 3.7 本章小结                      | 132 |

|                              |     |                             |     |
|------------------------------|-----|-----------------------------|-----|
| 第4章 投影与各种变换                  | 133 | 5.3 定位光与定向光                 | 212 |
| 4.1 矩阵相关类的介绍                 | 133 | 5.4 点法向量和面法向量               | 215 |
| 4.1.1 矩阵数学计算类——Matrix        | 133 | 5.5 光照的每顶点计算与每片元计算          | 218 |
| 4.1.2 矩阵状态管理类——MatrixState3D | 134 | 5.6 本章小结                    | 223 |
| 4.2 摄像机的设置                   | 137 | 第6章 纹理映射                    | 224 |
| 4.3 两种投影方式                   | 138 | 6.1 初识纹理映射                  | 224 |
| 4.3.1 正交投影                   | 138 | 6.2 一个简单的案例                 | 225 |
| 4.3.2 透视投影                   | 144 | 6.2.1 开发前的准备工作              | 225 |
| 4.3.3 推送常量                   | 146 | 6.2.2 纹理相关类                 | 226 |
| 4.4 各种变换                     | 148 | 6.2.3 案例代码的开发               | 241 |
| 4.4.1 基本变换的相关数学知识            | 148 | 6.2.4 图像的瓦片组织方式             | 248 |
| 4.4.2 平移变换                   | 149 | 6.2.5 色彩通道的灵活组合             | 250 |
| 4.4.3 旋转变换                   | 150 | 6.3 纹理拉伸                    | 251 |
| 4.4.4 缩放变换                   | 151 | 6.3.1 四种拉伸方式概览              | 251 |
| 4.4.5 基本变换的实质                | 152 | 6.3.2 不同拉伸方式的案例             | 254 |
| 4.5 所有变换的完整流程                | 153 | 6.4 纹理采样                    | 259 |
| 4.6 绘制方式                     | 156 | 6.4.1 纹理采样概述                | 259 |
| 4.6.1 各种绘制方式概览               | 156 | 6.4.2 最近点采样                 | 259 |
| 4.6.2 点与线段绘制方式               | 158 | 6.4.3 线性纹理采样                | 260 |
| 4.6.3 三角形条带与扇面绘制方式           | 161 | 6.4.4 MIN 与 MAG 采样          | 261 |
| 4.6.4 索引法绘制                  | 164 | 6.4.5 不同纹理采样方式的案例           | 262 |
| 4.7 设置合理的视角                  | 169 | 6.5 Mipmap 纹理               | 266 |
| 4.8 设置合理的投影参数                | 174 | 6.5.1 基本原理                  | 266 |
| 4.9 深度偏移                     | 178 | 6.5.2 一个简单的案例               | 268 |
| 4.10 卷绕和背面剪裁                 | 181 | 6.6 多重纹理与过程纹理               | 272 |
| 4.10.1 基本知识                  | 181 | 6.6.1 案例概览                  | 273 |
| 4.10.2 一个简单的案例               | 182 | 6.6.2 将 2D 纹理映射到球面上的策略      | 273 |
| 4.11 间接绘制                    | 183 | 6.6.3 案例的场景结构               | 274 |
| 4.11.1 背景知识                  | 184 | 6.6.4 开发过程                  | 275 |
| 4.11.2 顶点法间接绘制               | 184 | 6.7 压缩纹理的使用                 | 283 |
| 4.11.3 索引法间接绘制               | 186 | 6.7.1 Android 平台下的压缩纹理      | 284 |
| 4.12 本章小结                    | 188 | 6.7.2 PC 中 Windows 平台下的压缩纹理 | 286 |
| 第5章 光照                       | 189 | 6.8 点精灵                     | 289 |
| 5.1 曲面物体的构建                  | 189 | 6.8.1 基本知识                  | 289 |
| 5.1.1 球体构建的基本原理              | 189 | 6.8.2 一个简单的案例               | 289 |
| 5.1.2 案例效果概览                 | 190 | 6.9 3D 纹理                   | 292 |
| 5.1.3 开发步骤                   | 190 | 6.9.1 3D 棋盘纹理的准备            | 293 |
| 5.2 基本光照效果                   | 194 | 6.9.2 3D 棋盘纹理案例的开发          | 294 |
| 5.2.1 光照的基本模型                | 194 | 6.10 2D 纹理数组                | 297 |
| 5.2.2 环境光                    | 195 | 6.10.1 2D 纹理数组的准备           | 297 |
| 5.2.3 散射光                    | 199 | 6.10.2 2D 纹理数组案例的开发         | 298 |
| 5.2.4 镜面光                    | 205 | 6.11 各向异性过滤                 | 302 |
| 5.2.5 3 种光照通道的合成             | 210 |                             |     |

|        |                        |     |        |              |     |
|--------|------------------------|-----|--------|--------------|-----|
| 6.11.1 | 背景知识                   | 303 | 9.4    | 天空盒与天空穹      | 358 |
| 6.11.2 | 一个简单的案例                | 303 | 9.4.1  | 天空盒          | 358 |
| 6.12   | 本章小结                   | 305 | 9.4.2  | 天空穹          | 359 |
| 第7章    | 更逼真的场景——3D模型的加载        | 306 | 9.4.3  | 天空盒与天空穹的使用技巧 | 361 |
| 7.1    | obj模型文件概述              | 306 | 9.5    | 简单镜像效果       | 361 |
| 7.1.1  | obj文件的格式               | 306 | 9.5.1  | 基本原理         | 361 |
| 7.1.2  | 用3ds Max设计3D模型         | 307 | 9.5.2  | 基本效果案例       | 362 |
| 7.2    | 加载obj文件                | 308 | 9.5.3  | 升级效果案例       | 364 |
| 7.2.1  | 加载仅有顶点坐标数据与面数据的obj文件   | 308 | 9.6    | 非真实感绘制       | 366 |
| 7.2.2  | 加载后自动计算面法向量            | 312 | 9.6.1  | 基本原理与案例效果    | 366 |
| 7.2.3  | 加载后自动计算平均法向量           | 315 | 9.6.2  | 具体开发步骤       | 368 |
| 7.2.4  | 加载纹理坐标                 | 317 | 9.7    | 描边效果的实现      | 370 |
| 7.2.5  | 加载顶点法向量                | 320 | 9.7.1  | 沿法线挤出轮廓      | 370 |
| 7.3    | 双面光照                   | 321 | 9.7.2  | 视空间中挤出       | 373 |
| 7.4    | 本章小结                   | 324 | 9.8    | 本章小结         | 374 |
| 第8章    | 独特的场景渲染技术——混合与雾        | 325 | 第10章   | 两种测试及片元丢弃    | 375 |
| 8.1    | 混合技术                   | 325 | 10.1   | 剪裁测试         | 375 |
| 8.1.1  | 基本知识                   | 325 | 10.1.1 | 基本原理与核心代码    | 375 |
| 8.1.2  | 源因子和目标因子               | 326 | 10.1.2 | 一个简单的案例      | 375 |
| 8.1.3  | 简单混合效果的案例              | 328 | 10.2   | 模板测试         | 377 |
| 8.2    | 地月系云层效果的实现             | 330 | 10.2.1 | 基本原理         | 377 |
| 8.3    | 雾                      | 333 | 10.2.2 | 一个简单的案例      | 380 |
| 8.3.1  | 雾的原理与优势                | 333 | 10.3   | 片元丢弃操作       | 383 |
| 8.3.2  | 雾的简单实现                 | 334 | 10.4   | 任意剪裁平面       | 385 |
| 8.4    | 本章小结                   | 336 | 10.4.1 | 基本原理         | 385 |
| 第9章    | 常用3D开发小技巧              | 337 | 10.4.2 | 茶壶被任意平面剪裁的案例 | 386 |
| 9.1    | 标志板                    | 337 | 10.5   | 本章小结         | 387 |
| 9.1.1  | 案例效果与基本原理              | 337 | 第11章   | 顶点着色器的妙用     | 388 |
| 9.1.2  | 开发步骤                   | 338 | 11.1   | 飘扬的旗帜        | 388 |
| 9.2    | 灰度图地形                  | 341 | 11.1.1 | 基本原理         | 388 |
| 9.2.1  | 基本原理                   | 341 | 11.1.2 | 开发步骤         | 389 |
| 9.2.2  | 开发前的准备工作               | 342 | 11.2   | 扭动的软糖        | 392 |
| 9.2.3  | 普通灰度图地形                | 342 | 11.2.1 | 基本原理         | 392 |
| 9.2.4  | 过程纹理地形                 | 344 | 11.2.2 | 开发步骤         | 393 |
| 9.2.5  | Mipmap地形               | 347 | 11.3   | 风吹椰林场景的开发    | 394 |
| 9.2.6  | 顶点着色器采样纹理地形            | 351 | 11.3.1 | 椰子树随风摇摆的基本原理 | 395 |
| 9.3    | 高真实感地形                 | 353 | 11.3.2 | 开发步骤         | 396 |
| 9.3.1  | 基本思路                   | 353 | 11.4   | 展翅飞翔的雄鹰      | 400 |
| 9.3.2  | 地形设计工具EarthSculptor的使用 | 354 | 11.4.1 | 基本原理         | 400 |
| 9.3.3  | 一个简单的案例                | 356 | 11.4.2 | 开发步骤         | 400 |
|        |                        |     | 11.5   | 二维扭曲         | 402 |
|        |                        |     | 11.5.1 | 基本原理         | 403 |

|                             |     |                   |     |
|-----------------------------|-----|-------------------|-----|
| 11.5.2 开发步骤                 | 403 | 13.3 色散效果模拟       | 452 |
| 11.6 吹气膨胀特效                 | 406 | 13.3.1 案例效果与基本原理  | 452 |
| 11.6.1 基本原理                 | 406 | 13.3.2 开发步骤       | 452 |
| 11.6.2 开发步骤                 | 406 | 13.4 菲涅尔效果的模拟     | 453 |
| 11.7 本章小结                   | 407 | 13.4.1 案例效果与基本原理  | 453 |
| 第12章 片元着色器的妙用               | 408 | 13.4.2 开发步骤       | 454 |
| 12.1 程序纹理技术                 | 408 | 13.5 凹凸映射         | 455 |
| 12.1.1 砖块着色器                | 408 | 13.5.1 案例效果与基本原理  | 455 |
| 12.1.2 沙滩球着色器               | 410 | 13.5.2 法向量纹理图的生成  | 456 |
| 12.2 数字图像处理                 | 411 | 13.5.3 切向量的计算     | 458 |
| 12.2.1 卷积的基本知识              | 411 | 13.5.4 案例的开发      | 459 |
| 12.2.2 平滑过滤                 | 412 | 13.6 镜头光晕         | 464 |
| 12.2.3 边缘检测                 | 413 | 13.6.1 案例效果与基本原理  | 464 |
| 12.2.4 锐化处理                 | 414 | 13.6.2 镜头光晕案例     | 465 |
| 12.2.5 浮雕效果                 | 414 | 13.7 投影贴图         | 468 |
| 12.2.6 图像渐变                 | 415 | 13.7.1 案例效果与基本原理  | 468 |
| 12.2.7 卡通渲染                 | 416 | 13.7.2 开发步骤       | 470 |
| 12.3 分形着色器                  | 418 | 13.8 绘制到纹理        | 472 |
| 12.3.1 曼德布罗集简介              | 418 | 13.8.1 案例效果与基本原理  | 472 |
| 12.3.2 曼德布罗集着色器的实现          | 419 | 13.8.2 开发步骤       | 473 |
| 12.3.3 将曼德布罗集纹理应用到<br>实际物体上 | 421 | 13.9 高级镜像         | 476 |
| 12.3.4 茉莉亚集着色器的实现           | 421 | 13.9.1 案例效果与基本原理  | 476 |
| 12.4 3D 纹理的妙用               | 423 | 13.9.2 开发步骤       | 477 |
| 12.4.1 噪声函数的基础知识            | 423 | 13.10 高真实感水面倒影    | 480 |
| 12.4.2 柏林噪声生成工具             | 426 | 13.10.1 案例效果与基本原理 | 480 |
| 12.4.3 3D 噪声木纹理茶壶的<br>开发    | 427 | 13.10.2 开发步骤      | 481 |
| 12.5 体积雾                    | 429 | 13.11 本章小结        | 484 |
| 12.5.1 基本原理                 | 429 | 第14章 阴影及高级光照      | 485 |
| 12.5.2 开发步骤                 | 430 | 14.1 多重渲染目标       | 485 |
| 12.6 粒子系统火焰的开发              | 431 | 14.1.1 基本知识       | 485 |
| 12.6.1 火焰的基本原理              | 431 | 14.1.2 一个简单的案例    | 486 |
| 12.6.2 普通版火焰                | 433 | 14.2 阴影的重要性       | 491 |
| 12.6.3 点精灵版火焰               | 441 | 14.3 平面阴影         | 493 |
| 12.7 本章小结                   | 444 | 14.3.1 案例效果与基本原理  | 493 |
| 第13章 真实光学环境的模拟              | 445 | 14.3.2 开发步骤       | 494 |
| 13.1 反射环境模拟                 | 445 | 14.4 阴影映射         | 497 |
| 13.1.1 案例效果与基本原理            | 445 | 14.4.1 案例效果与基本原理  | 497 |
| 13.1.2 开发前的准备工作             | 446 | 14.4.2 距离纹理的生成    | 498 |
| 13.1.3 开发步骤                 | 447 | 14.4.3 阴影场景的绘制    | 501 |
| 13.2 折射环境模拟                 | 449 | 14.5 阴影贴图         | 506 |
| 13.2.1 案例效果与基本原理            | 449 | 14.5.1 案例效果与基本原理  | 506 |
| 13.2.2 开发步骤                 | 450 | 14.5.2 开发步骤       | 507 |
|                             |     | 14.6 静态光照贴图       | 508 |
|                             |     | 14.6.1 案例效果与基本原理  | 508 |

|        |                           |     |
|--------|---------------------------|-----|
| 14.6.2 | 使用 3ds Max 制作静态光照贴图 ..... | 509 |
| 14.6.3 | 案例的开发 .....               | 510 |
| 14.7   | 聚光灯高级光源 .....             | 512 |
| 14.7.1 | 第一种实现的案例效果与基本原理 .....     | 513 |
| 14.7.2 | 第一种实现案例的开发步骤 .....        | 514 |
| 14.7.3 | 第二种实现的案例效果与基本原理 .....     | 515 |
| 14.7.4 | 第二种实现案例的开发步骤 .....        | 516 |
| 14.8   | 延迟渲染 .....                | 517 |
| 14.8.1 | 案例效果与基本原理 .....           | 517 |
| 14.8.2 | 开发步骤 .....                | 518 |
| 14.9   | 环境光遮挡 .....               | 522 |
| 14.9.1 | 基本原理 .....                | 522 |
| 14.9.2 | 一个简单的案例 .....             | 524 |
| 14.9.3 | 效率的提升 .....               | 526 |
| 14.9.4 | 平滑处理 .....                | 528 |
| 14.10  | 本章小结 .....                | 529 |
| 第 15 章 | 几种高级着色器特效 .....           | 530 |
| 15.1   | 运动模糊 .....                | 530 |
| 15.1.1 | 普通运动模糊 .....              | 530 |
| 15.1.2 | 高级运动模糊 .....              | 533 |
| 15.2   | 遮挡透视效果 .....              | 536 |
| 15.2.1 | 案例效果与基本原理 .....           | 536 |
| 15.2.2 | 开发步骤 .....                | 538 |
| 15.3   | 积雪效果 .....                | 541 |
| 15.3.1 | 案例效果与基本原理 .....           | 541 |
| 15.3.2 | 开发步骤 .....                | 541 |
| 15.4   | 背景虚化 .....                | 542 |
| 15.4.1 | 案例效果与基本原理 .....           | 542 |
| 15.4.2 | 开发步骤 .....                | 543 |
| 15.5   | 泛光效果 .....                | 545 |
| 15.5.1 | 案例效果与基本原理 .....           | 545 |
| 15.5.2 | 开发步骤 .....                | 546 |
| 15.6   | 色调映射 .....                | 548 |
| 15.6.1 | 案例效果与背景知识 .....           | 548 |
| 15.6.2 | 开发步骤 .....                | 549 |
| 15.7   | 体绘制 .....                 | 550 |
| 15.7.1 | 案例效果与基本原理 .....           | 550 |
| 15.7.2 | 开发步骤 .....                | 551 |
| 15.8   | 本章小结 .....                | 553 |

|        |                                 |     |
|--------|---------------------------------|-----|
| 第 16 章 | 骨骼动画 .....                      | 554 |
| 16.1   | 开发骨骼动画 .....                    | 554 |
| 16.1.1 | 骨骼动画的基本原理 .....                 | 554 |
| 16.1.2 | 开发步骤 .....                      | 557 |
| 16.1.3 | 机器人模型在地面上运动时的问题 .....           | 563 |
| 16.2   | ms3d 骨骼动画文件的加载 .....            | 566 |
| 16.2.1 | ms3d 文件的格式 .....                | 566 |
| 16.2.2 | 将 3ds Max 动画文件转换为 ms3d 文件 ..... | 570 |
| 16.2.3 | 加载 ms3d 文件案例的开发 .....           | 574 |
| 16.3   | 自定义格式骨骼动画的加载 .....              | 591 |
| 16.3.1 | bnggdh 文件的格式 .....              | 591 |
| 16.3.2 | Bnggdh 类 .....                  | 594 |
| 16.3.3 | 加载 bnggdh 文件的案例 .....           | 594 |
| 16.4   | 本章小结 .....                      | 606 |
| 第 17 章 | 让应用运行得更流畅——性能优化 .....           | 607 |
| 17.1   | 着色器的优化 .....                    | 607 |
| 17.1.1 | 计算量及计算频率的相关问题 .....             | 607 |
| 17.1.2 | 其他需要注意的问题 .....                 | 608 |
| 17.2   | 纹理图的优化 .....                    | 609 |
| 17.3   | 3D 图形绘制的优化 .....                | 610 |
| 17.3.1 | CPU 阶段的优化 .....                 | 610 |
| 17.3.2 | 几何阶段的优化 .....                   | 611 |
| 17.3.3 | 光栅化阶段的优化 .....                  | 612 |
| 17.4   | 图元重启 .....                      | 612 |
| 17.4.1 | 基本原理 .....                      | 612 |
| 17.4.2 | 一个简单的案例 .....                   | 613 |
| 17.5   | 几何体实例渲染 .....                   | 614 |
| 17.5.1 | 基本原理 .....                      | 614 |
| 17.5.2 | 基于实例渲染的土星光环案例 .....             | 615 |
| 17.6   | 视锥体剪裁 .....                     | 620 |
| 17.6.1 | AABB 包围盒的基本原理 .....             | 620 |
| 17.6.2 | AABB 包围盒的计算 .....               | 621 |
| 17.6.3 | 视锥体剪裁的基本思路及实现策略 .....           | 622 |
| 17.6.4 | 一个简单的案例 .....                   | 623 |
| 17.7   | 遮挡查询 .....                      | 626 |
| 17.7.1 | 相关方法 .....                      | 626 |
| 17.7.2 | 基本原理和实现策略 .....                 | 626 |

|                                               |            |
|-----------------------------------------------|------------|
| 17.7.3 一个简单的案例 .....                          | 627        |
| 17.8 计算着色器的使用 .....                           | 630        |
| 17.8.1 基础知识 .....                             | 630        |
| 17.8.2 动态法向量光照水面案例 .....                      | 633        |
| 17.9 多线程并发渲染 .....                            | 640        |
| 17.9.1 基本原理 .....                             | 641        |
| 17.9.2 飞船案例的开发 .....                          | 641        |
| 17.10 多子通道渲染 .....                            | 646        |
| 17.10.1 基本原理 .....                            | 646        |
| 17.10.2 一个简单的案例 .....                         | 647        |
| 17.11 细节级别 LOD .....                          | 652        |
| 17.12 本章小结 .....                              | 653        |
| <b>第 18 章 杂项 .....</b>                        | <b>654</b> |
| 18.1 四元数旋转 .....                              | 654        |
| 18.1.1 基本原理与案例效果 .....                        | 654        |
| 18.1.2 开发步骤 .....                             | 656        |
| 18.2 3D 拾取技术 .....                            | 662        |
| 18.2.1 案例效果与基本原理 .....                        | 662        |
| 18.2.2 开发步骤 .....                             | 665        |
| 18.3 多重采样抗锯齿 .....                            | 670        |
| 18.3.1 基本知识 with 案例效果 .....                   | 670        |
| 18.3.2 一个简单的案例 .....                          | 671        |
| 18.4 保存屏幕截图 .....                             | 674        |
| 18.5 Windows 系统窗口缩放 .....                     | 676        |
| 18.6 曲面细分着色器 .....                            | 677        |
| 18.6.1 基本知识 .....                             | 677        |
| 18.6.2 一个简单的案例 .....                          | 680        |
| 18.7 几何着色器 .....                              | 685        |
| 18.8 macOS、iOS 与 Linux 平台下 Vulkan 应用的开发 ..... | 688        |
| 18.8.1 macOS 与 iOS 平台下 Vulkan 应用程序的开发 .....   | 688        |
| 18.8.2 Linux 平台下 Vulkan 图形应用程序的开发 .....       | 692        |

|                                                |            |
|------------------------------------------------|------------|
| 18.9 本章小结 .....                                | 700        |
| <b>第 19 章 基于 Vulkan 的 3D 休闲游戏——方块历险记 .....</b> | <b>701</b> |
| 19.1 游戏背景及功能概述 .....                           | 701        |
| 19.1.1 游戏开发背景概述 .....                          | 701        |
| 19.1.2 游戏功能简介 .....                            | 701        |
| 19.2 游戏的策划及准备工作 .....                          | 702        |
| 19.2.1 游戏的策划 .....                             | 703        |
| 19.2.2 游戏的开发准备工作 .....                         | 703        |
| 19.3 游戏的架构 .....                               | 705        |
| 19.3.1 各类的简要介绍 .....                           | 705        |
| 19.3.2 游戏框架简介 .....                            | 706        |
| 19.4 布景相关类 .....                               | 707        |
| 19.4.1 3D 布景类——My3Dlayer .....                 | 707        |
| 19.4.2 总绘制类——MyDraw .....                      | 712        |
| 19.4.3 摄像机管理类——CameraUtil .....                | 716        |
| 19.4.4 3D 物体创建类——Square3D .....                | 717        |
| 19.5 机关与方块相关类 .....                            | 718        |
| 19.5.1 方块管理类——ZFTManager .....                 | 718        |
| 19.5.2 地刺机关管理类——DiCiBox .....                  | 721        |
| 19.5.3 掉落块机关管理类——DiaoLuoManager .....          | 722        |
| 19.6 游戏入口及辅助工具类 .....                          | 724        |
| 19.6.1 游戏入口类——main .....                       | 724        |
| 19.6.2 触控监听类——Touch2D .....                    | 726        |
| 19.7 着色器的开发 .....                              | 728        |
| 19.8 游戏的优化及改进 .....                            | 730        |

# 第1章 初识 Vulkan

Vulkan 是一种跨平台的 2D 和 3D 图形应用程序接口，最早由 Khronos 组织在 2015 年 GDC 上发布。其本质上是 AMD Mantle 的后续版本，继承了前者强大的低开销架构，使开发人员能够方便全面地获取 GPU 与多核 CPU 的性能、功能和提升效率。

相比于 OpenGL，Vulkan 支持深入硬件底层进行控制，并能大幅度降低 CPU 在高负载绘制任务中的开销。同时其对多核心 CPU 的支持也更加完善，更加适应当下从高端工作站到 PC 平台到移动平台的多核战略。

## 1.1 Vulkan 概览

介绍具体的开发技术之前，本节将首先介绍 Vulkan 的历史传承以及一些技术特点，同时将 Vulkan 与其他的图形应用程序接口（OpenGL、DirectX、Metal 等）进行简要的比较，最后还会介绍一下当下支持 Vulkan 的游戏，具体内容如下。

### 1.1.1 Vulkan 简介

了解 Vulkan 的具体知识之前，我们有必要首先了解一下市面上主流的各 3D 图形应用程序接口。目前各平台下主流的 3D 图形 API 有 OpenGL、OpenGL ES、DirectX、Metal 以及 Vulkan，各自的应用领域及特点如下。

- OpenGL 的应用领域较为广泛，支持多种操作系统平台（如 Windows、UNIX、Linux、macOS 等）。基于其开发的应用可以方便、低成本地在不同操作系统平台之间移植。既可以用于开发游戏，又可以用于开发工业、行业应用。

- OpenGL ES 是专门针对移动嵌入式平台而设计的，实际是 OpenGL 的剪裁版本。去除了 OpenGL 中许多不必要的特性，优化了对性能、供电受限的移动嵌入式平台的支持。

- DirectX 为微软的专有技术，主要用于 Windows 下游戏的开发，在此领域占有极高的比例。最新的版本为 DirectX 12，此版本也是大大优化了对多核 CPU 的支持，但仅支持 Windows 10。

- Metal 是 Apple 的专有技术，仅仅能够在 macOS 以及 iOS 下使用，应用的领域相对比较狭窄，目前基于它的应用相对较少。

- Vulkan 与 OpenGL 类似，是跨平台的 3D 图形应用程序接口，同时支持 Windows 7、Windows 8.1、Windows 10、Linux 以及 Android 等平台。

Vulkan 最早被称为下一代 OpenGL，项目名称为 GLNext。其设计考虑到了统一各个平台的开发，因此不像 OpenGL 与 OpenGL ES 那样，根据硬件性能、供电区分不同版本，而是工作站、PC、移动嵌入式等平台完全一致。这对广大开发人员来说，是一个极大的利好。

2016 年 2 月 16 日，Khronos 组织发布了 Vulkan 的首个正式版本。从此，数字图形技术产业

诞生了一个真正意义上能与 DirectX 12、Metal 分庭抗礼的全新图形应用程序接口。到 2016 年 4 月，Google 在第二个 Android N 的开发预览版中也正式加入了对 Vulkan 的支持。Vulkan 的主要特点如下。

- Vulkan 提供更低的运行开销、更直接的 GPU 控制和较低的 CPU 负载。其通过批处理方式有效减少 CPU 的负载，将 CPU 从额外的运算和渲染中解放出来去执行其他的任务。
- 相比于以往面向 CPU 单核心设计的 OpenGL，Vulkan 原生支持多线程并发处理，能够更好地与当下普遍采用多核战略的 CPU 协同工作。DirectX 12、Metal 等厂商专有的新一代图形应用程序接口也都在多线程并发方面提供了支持，可见这是业界发展的大趋势。
- 着色器方面，Vulkan 也不再像 OpenGL 一样指定高层的着色器编程语言（OpenGL 指定采用 GLSL 着色器编程语言），而是采用一种被称为 SPIR-V（Standard Portable Intermediate Representation）的二进制中间层格式。这样，开发人员在开发 Vulkan 着色器时可以选用自己青睐的着色器编程语言，诸如 GLSL、HLSL 等，然后将着色器源代码采用着色器专用编译器编译为 SPIR-V 格式即可在 Vulkan 中使用，大大提高了灵活性。
- Vulkan 将计算任务和图形着色渲染任务统一管理，无需使用单独的計算和图形应用程序接口进行连接。
- 不同于 OpenGL 的状态机，在运行任务时会自动进行各种错误检查（不可关闭）。Vulkan 为了追求更高的执行效率，将各种错误的检查设计为可插拔模式。开发人员可以在开发调试时打开所需的错误检查项目，在发布时关闭错误检查项目，以达到更好的性能。
- Vulkan 在架构层面提供了对多轮渲染的支持，使得可以以更高的效率实现延迟渲染以在特定场景下大大提高渲染效率。

Vulkan 本身博大精深，其革命性的设计远远不止上述这些，读者可以跟随本书的脚步逐渐深入地学习 Vulkan 的方方面面。

1.1.2 支持 Vulkan 的游戏概览

通过前面简单的介绍，读者已基本了解到 Vulkan 相比于传统图形应用程序接口的多项优势。正因为 Vulkan 这些突出的特性，目前市面上已有几款知名游戏开始使用 Vulkan。但由于 Vulkan 诞生的时间不长，故使用 Vulkan 的游戏数量还不是很多。接下来，我们将对使用 Vulkan 的几款游戏进行简单的介绍。

● Dota

作为一款广受玩家欢迎的巨作，早在 2016 年 Dota 2 便推出官方补丁使其支持 Vulkan。如图 1-1 所示为原版 Dota 2 的游戏场景图，图 1-2 所示为在 Vulkan 支持下运行的 Dota 2 游戏场景图。



▲图 1-1 原版 Dota 2 游戏场景



▲图 1-2 Vulkan 支持下的 Dota 2 游戏场景

## 说明

通过对比图 1-1、图 1-2 可以看出，在游戏画面方面，Vulkan 支持下的 Dota 2 较原版 Dota 2 场景更加逼真、细腻。在游戏的实际对比测试中，可以感觉到 Vulkan 支持下的 Dota 2 运行更加流畅，并且可以观察到 CPU 使用率更低，这正体现了 Vulkan 降低 CPU 开销的特点。

## ● 极品飞车

通过对比 Dota 2 在使用 Vulkan 前后的场景画面，我们已经观察到了 Vulkan 在 3D 图形处理方面的进步。接下来将通过展示 Electronic Arts 开发的赛车竞技类游戏“极品飞车：无极限”，进一步感受 Vulkan 的 3D 图形处理能力，具体情况如图 1-3 和图 1-4 所示。



▲图 1-3 极品飞车：无极限场景 1



▲图 1-4 极品飞车：无极限场景 2

## 说明

可以看出上述两幅使用 Vulkan API 渲染出的“极品飞车：无极限”游戏场景画面光影效果极其逼真，烟雾、运动模糊效果都很真实。

## ● Dream League Soccer

介绍完上述两款支持 Vulkan 的游戏 Dota 2 和极品飞车之后，不得不介绍 First Touch 开发的体育类游戏——Dream League Soccer。该游戏自发布以来一直广受玩家的好评，现在更是推出了 Vulkan 版本，其效果分别如图 1-5 和图 1-6 所示。



▲图 1-5 Dream League Soccer 场景 1



▲图 1-6 Dream League Soccer 场景 2

通过对上述几款游戏画面的观察，我们可以领略到 Vulkan 在 3D 图形处理方面的能力提升。前面的内容中，多次提到 Vulkan 的一大优势是能够大幅度降低渲染时的 CPU 开销，这将直接影响游戏运行及画面的流畅度，有关权威组织对 Vulkan 这方面的测试也不少。

比如早在 2016 年 Bethesda 和 Nvidia 就进行了相关测试，测试结果表明使用 DirectX 11 在 1080P 分辨率下运行《毁灭战士 4》，平均帧率在 55~60 之间。之后，使用 Vulkan 进行同样的渲染工作，