



# Unity 3D

## ShaderLab开发实战详解

郭浩瑜 编著



- 全面讲解了Unity Shader渲染的实战技术
- Unity中的照明，如VertexLit渲染路径、Forward渲染路径、基于光照贴图的烘焙照明、基于LightProbes的照明
- Shadows（阴影），如平面阴影、球体阴影、体积阴影、阴影映射、内置的阴影
- Unity中的各种Shader，如固定管线、Surface Shader、凹凸材质、卡通材质、镜面材质、半透明材质、体积雾、Wrap Model新解、面积光、体积光、材质替代渲染、后期效果、地形、投影
- Shader的组织和优化，如Shader的组织 and 复用、基于渲染路径的优化、移动平台上的优化



源程序

人民邮电出版社  
POSTS & TELECOM PRESS

014036671

TP317.6  
06

# Unity 3D

## ShaderLab开发实战详解

郭浩瑜 编著



人民邮电出版社  
北京

## 图书在版编目 (C I P) 数据

Unity 3D ShaderLab开发实战详解 / 郭浩瑜编著

— 北京 : 人民邮电出版社, 2014.4

ISBN 978-7-115-33890-7

I. ①U… II. ①郭… III. ①游戏程序—程序设计

IV. ①TP311.5

中国版本图书馆CIP数据核字(2013)第292953号

## 内 容 提 要

本书全面讲解了 Unity Shader 渲染的实战技术, 全书分为 5 篇共 33 章, 主要内容为: Shader 在 3D 游戏中的作用、Shader 的实例化、Shader 的实现语言等; Unity 中 Shader 的形态、SubShader 的重要标签、可编程 Shader、Surface Shader; Shader 中用到的各种空间的概念和 Shader 中的投影矩阵; 基本的光照模型; 第一个被执行的 Pass, 包括渲染路径和 Pass 的 LightMode 标签; VertexLit 渲染路径, 包括顶点照明和 Unity 存放光源的方式; Forward 渲染路径; 基于光照贴图的烘焙照明; 基于 LightProbes 的照明; 平面阴影; 球体阴影; 体积阴影; 映射阴影; 内置的阴影; Pass 的通用指令开关; 固定管线; Surface Shader; 凹凸材质; 卡通材质; 镜面材质; 半透明材质; 体积雾; Wrap Model 新解; 面积光; 体积光; 材质替代渲染; 后期效果; 地形; 投影; Shader 的组织 and 复用。和你必须知道的渲染概念及基于渲染路径的优化、移动平台上的优化等实战内容, 是不可多得的实战教程。

本书适合移动开发者、游戏开发者、程序员阅读, 可作为大专院校相关专业的师生用书, 也可当作培训学校的教材。

---

◆ 编 著 郭浩瑜

责任编辑 张 涛

责任印制 程彦红 杨林杰

◆ 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>

北京艺辉印刷有限公司印刷

◆ 开本: 787×1092 1/16

印张: 21.25

字数: 552 千字

2014 年 4 月第 1 版

印数: 1—3 500 册

2014 年 4 月北京第 1 次印刷

---

定价: 69.00 元 (附光盘)

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

反盗版热线: (010) 81055315

# 前 言

在开始研究 Unity 中的 Shader 编程过程中，发现虽然 Unity 的官方文档对 ShaderLab 的介绍在某些方面相当详细，文档庞大，但是并非巨细无遗。如关于如何用 Unity 渲染不同类型的 Shader、光源如何存取等一些非常重要的知识，官方文档并没有给出详细的技术说明。其中一些技术疑问，有的可以在 Unity 的官方论坛中找到，但这些都是世界各地 Unity 开发者的问答集锦，文档分散且不正式，在数万条各种各样的条目中寻找答案不容易。有些解答不是很清楚，不能很好地解答读者的问题。所以，对于那些想对 Unity 的渲染框架有一个比较全面理解的人来说，一个全面而系统的知识讲解是非常迫切的。因此，就产生了写一本专注于 Unity 渲染方面的书的想法。

为了把这个想法变成现实，作者辞职后专门研究 Unity 框架，并花了几个月的时间来写这本书。在写作过程中，为了让学习 Shader 编程的读者尽快入门，书中对每一个主题的讲解都是从基础的、简单的技术讲起，然后逐步深入到更高一级的渲染技术，而且每一章都有相应的案例场景，可以方便读者自己动手实践，学习书中提到的技术点，达到学以致用目的。本书全面讲解了 Unity Shader 渲染的实战技术，主要内容为：第 1 章 Shader 的概念和在 3D 游戏中的作用，讲解了 Shader 的概念、Shader 的实例化、Shader 的实现语言等；第 2 章 Unity 中 Shader 的形态，介绍了使用 SubShader 组织 Shader 的不同实现、SubShader 的重要标签、可编程 Shader、Surface Shader；第 3 章 Shader 中用到的各种空间的概念，讲解了世界坐标空间、脚本和 Shader 中的投影矩阵；第 4 章 基本的光照模型，讲解了照明的计算方式、光照模型；第 5 章 第一个被执行的 Pass，包括渲染路径和 Pass 的 LightMode 标签，设计可以检测渲染路径的材质，Always 类型的 Pass 在 3 种渲染路径下的执行等；第 6 章 VertexLit 渲染路径，包括顶点照明和 Unity 存放光源的方式，顶点照明的一个实现实例；第 7 章 Forward 渲染路径；第 8 章 基于光照贴图的烘焙照明；第 9 章 基于 LightProbes 的照明；第 10 章 平面阴影；第 11 章 球体阴影；第 12 章 体积阴影；第 13 章 映射阴影；第 14 章 内置的阴影；第 15 章 Pass 的通用指令开关；第 16 章 固定管线；第 17 章 Surface Shader；第 18 章 凹凸材质；第 19 章 卡通材质；第 20 章 镜面材质；第 21 章 半透明材质；第 22 章 体积雾；第 23 章 Wrap Model 新解；第 24 章 面积光；第 25 章 体积光；第 26 章 材质替代渲染；第 27 章 后期效果；第 28 章 地形；第 29 章 投影；第 30 章 Shader 的组织和复用；第 31 章 你必须知道的渲染概念；第 32 章 基于渲染路径的优化；第 33 章 移动平台上的优化等实战内容。本书全面阐述了 Unity Shader 应用，是不可多得的实战教程。

## 光盘中工程文件使用说明

本书在写作过程中，针对每一章、每一节，甚至每一节的不同话题，都分别创建了不同的场景、Shader 文件和帮助脚本，因此，读者打开光盘中的工程文件后，会发现许多目录和场景文件，这些文件都用目录加以组织，为了便于查找，下面简单说明一下使用方法。

书中总共有 5 大篇，因此光盘中有相应的 5 个顶级目录。

第 1 篇 初识庐山真面目——Unity 3D Shader	Assets/NewBie
第 2 篇 Unity 中的照明	Assets/Lighting
第 3 篇 Shadows（阴影）	Assets/Shadow
第 4 篇 Unity 中的各种 Shader	Assets/Shader
第 5 篇 Shader 的组织和优化	Assets/Organize_Optimize

除了上面 5 个目录外，还有几个目录，下面简单说明一下。

- Assets/\_Models: 主要存放本书用到的模型资源。
- Assets/\_Textures: 主要存放一些贴图资源。
- Assets/MVP: 主要存放一些关于 MVP 矩阵的生成、操作的工程文件。虽然创作了这些工程文件，但是没有什么大的意义，所以没有将这部分写到书中。如果读者对 MVP 矩阵还有疑问，可以打开文件运行一下看看效果。

本书每一章的开始，都会声明本章所使用的工程文件的相对目录，如在第 19 章的开始，就告诉读者：“本章所有用到的文件和场景，其位置都相对于光盘下的 Assets/Shader/Toon 目录。”

如 19.1 小节：“打开 Outline/Lab\_1 文件夹下 Lab\_1”，那么这个目录就是相对于本章，即第 19 章的目录，因此，完整路径应该是：“Assets/Shader/Toon/Outline/Lab\_1”。

如果读者对本书的某一部分或者其他问题还有疑问，可以到 QQ 群（246817781）中来提问，会把大家问得比较多的问题的解决方法贴出来。大家也可以到 Unity 圣典社区中来提问，到时会在社区开一个相关板块回答大家的问题。

如果读者对本书的某一部分或者其他问题还有疑问，可以到 QQ 群（246817781）中来提问，会把大家问得比较多的问题的解决方法贴出来。大家也可以到 Unity 圣典社区中来提问，到时会在社区开一个相关板块回答大家的问题。

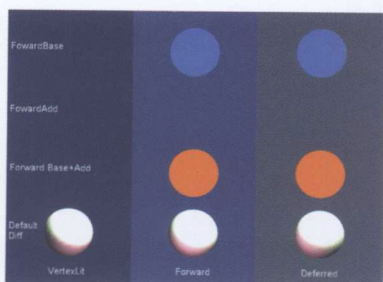
在本书出版之际，首先感谢 ShaderLab 官方论坛上的各位朋友，谢谢他们对我在论坛上一些提问的热心回答！感谢 Unity 圣典论坛和纳金网，谢谢他们在这本书筹备过程中给予的支持。其次是人民邮电出版社的编辑，感谢他们对出版这本书给予的专业指导意见和帮助，虽然这些指导意见经常导致我不得不把这本书翻来覆去地改一遍，不过他们专注和专业的意见对我写好本书很有帮助。最后，感谢网友和 QQ 群的好朋友，谢谢大家在我写这本书的过程中给予的支持和完善建议。

由于时间有限，加之本人水平有限，书中难免存有不足之处，望读者批评指正，编辑联系邮箱为 zhangtao@ptpress.com.cn。

编者

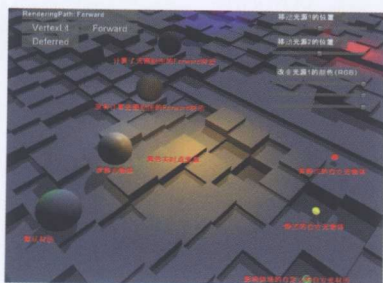
# 目 录

## 第 1 篇 初识庐山真面目—— Unity 3D Shader



### 第 1 章 Shader (着色器) 的概念和在 3D 游戏中的作用.....2

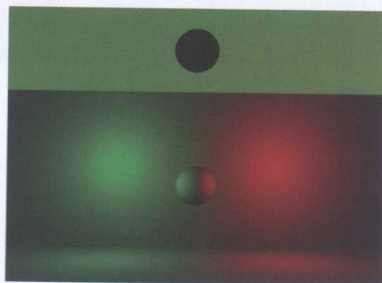
- 1.1 Shader 的概念.....2
  - 1.1.1 虚拟世界中的光明和色彩.....2
  - 1.1.2 游戏开发人员的终点.....2
  - 1.1.3 Shader (着色器) 简史.....2
- 1.2 Shader 的实例化.....3
- 1.3 Shader 的实现语言.....3
  - 1.3.1 GPU 上的编程.....3
  - 1.3.2 Unity 中的着色器编程.....3



### 第 2 章 Unity 中 Shader (着色器) 的形态.....4

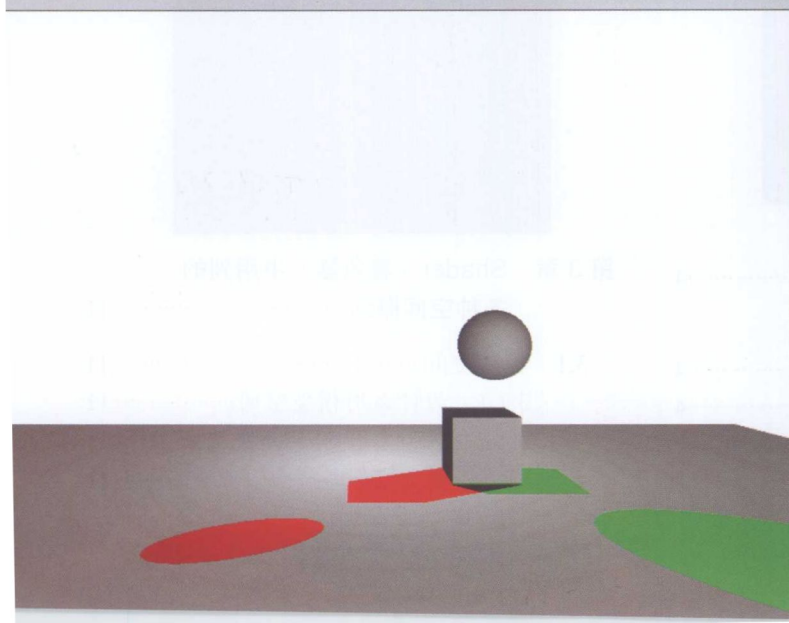
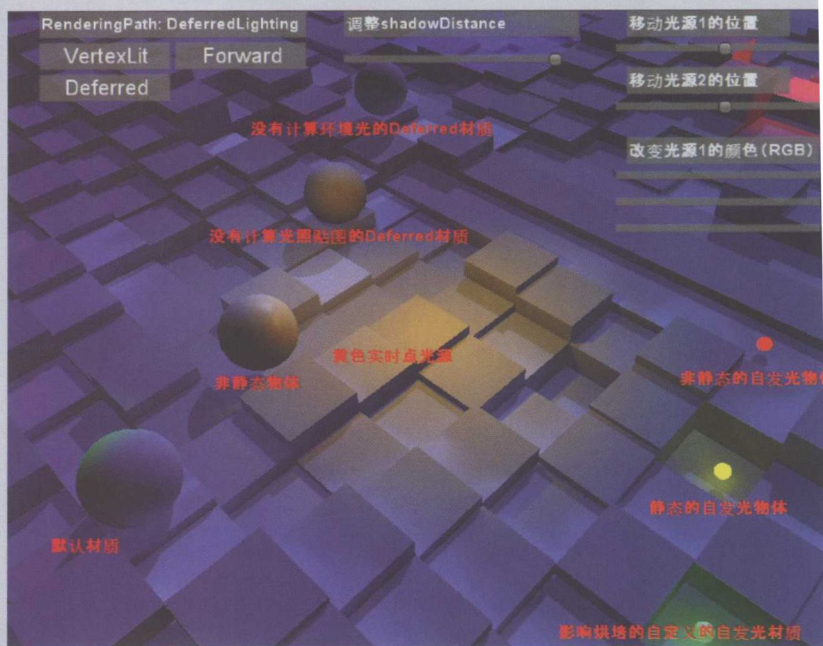
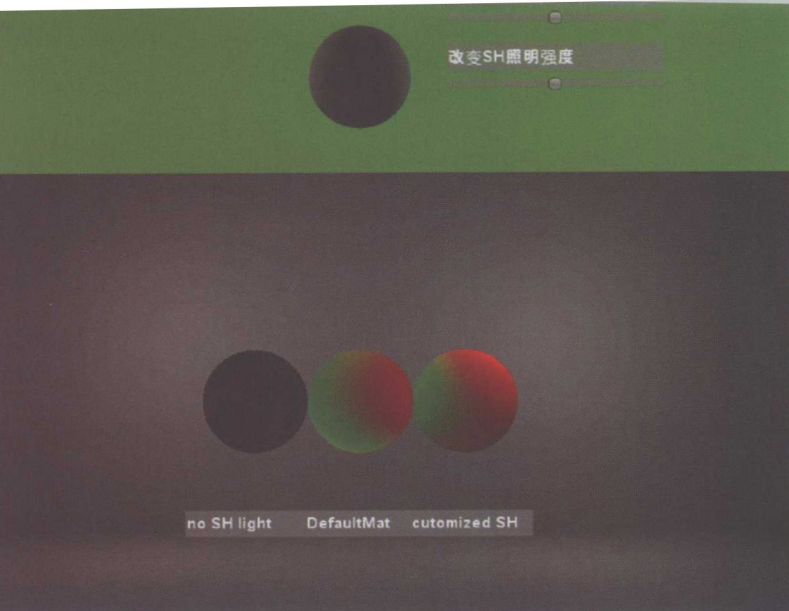
- 2.1 Unity 通过 ShaderLab 来组织  
Shader.....4
  - 2.1.1 关键字 Shader.....4
  - 2.1.2 使用 SubShader 组织 Shader  
的不同实现.....4

- 2.1.3 SubShader 的重要标签.....4
- 2.1.4 SubShader 中的 Pass 块.....5
- 2.1.5 Pass 块的标签及其名字  
的意义.....5
- 2.1.6 使用 FallBack 保证 Shader  
的广泛适应性.....6
- 2.2 Unity 的 ShaderLab 所支持的  
Shader 编程语言.....6
- 2.3 Unity 中 Shader 的 3 种形态.....6
  - 2.3.1 固定管线.....6
  - 2.3.2 可编程 Shader.....7
  - 2.3.3 ShaderLab 的骄傲: Surface  
Shader.....8
- 2.4 Shader 的数据接口: 属性和  
uniform 变量.....8
  - 2.4.1 在 Properties 块中定义  
属性.....8
  - 2.4.2 通过图形界面操作属性.....9
  - 2.4.3 通过脚本操控属性.....9
  - 2.4.4 矩阵: 不能在属性块定义  
的变量.....10
  - 2.4.5 在 Cg 代码中使用属性.....10



### 第 3 章 Shader (着色器) 中用到的 各种空间概念.....11

- 3.1 模型空间.....11
  - 3.1.1 为什么用模型空间.....11
  - 3.1.2 在脚本和 Shader 中进出  
模型空间.....11



- 3.2 世界坐标空间.....11
  - 3.2.1 统一表达: 世界坐标空间 .....11
  - 3.2.2 在脚本和 Shader 中进出世界坐标空间 .....12
- 3.3 视空间.....12
  - 3.3.1 渲染的需要: 视空间 .....12
  - 3.3.2 在脚本和 Shader 中进出视空间 .....12
- 3.4 空间的一块: 视锥体.....12
- 3.5 剪切空间.....13
  - 3.5.1 投影.....13
  - 3.5.2 脚本和 Shader 中的投影矩阵 .....13



第 4 章 基本的光照模型.....14

- 4.1 光源对物体照明的分类.....14
  - 4.1.1 间接照明 .....14
  - 4.1.2 直接照明 .....14
- 4.2 照明的计算方式: 光照模型.....14
  - 4.2.1 漫反射和 Lambert.....14
  - 4.2.2 镜面高光和 Phong.....15
  - 4.2.3 半角向量和 BlinnPhong.....16

第 2 篇 让你的应用更炫彩——Unity 中的照明



第 5 章 第一个被执行的 Pass.....18

- 5.1 不同的 LightMode 被选择的顺序.....18

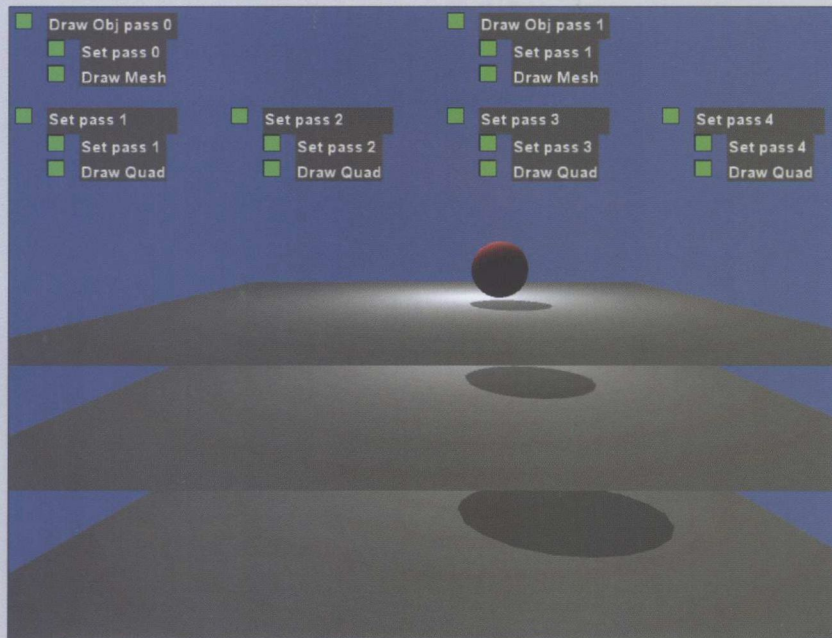
- 5.1.1 渲染路径和 Pass 的 LightMode 标签.....18
- 5.1.2 设计可以检测渲染路径的材质.....18
- 5.1.3 设计便于检测渲染路径的场景.....21
- 5.1.4 VertexLit 渲染路径下 Pass 的执行.....21
- 5.1.5 Forward 渲染路径下 Pass 的执行.....21
- 5.1.6 Deferred 渲染路径下 Pass 的执行.....21
- 5.1.7 不同渲染路径下的 Pass 执行规则总结.....22
- 5.2 3 个渲染路径之外.....22
  - 5.2.1 LightMode 的其他值.....22
  - 5.2.2 设计检测用的材质.....23
  - 5.2.3 Always 类型的 Pass 在 3 种渲染路径下的执行.....24
  - 5.2.4 LightMode 的默认值及其在 3 种渲染路径下的执行.....25



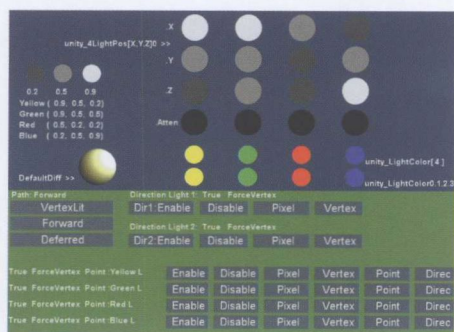
第 6 章 VertexLit 渲染路径.....26

- 6.1 顶点照明.....26
  - 6.1.1 什么是顶点照明 .....26
  - 6.1.2 存取光源的变量 .....26
- 6.2 顶点照明和 Unity 存放光源的第一种方式.....27
  - 6.2.1 用于调试输出的材质.....27
  - 6.2.2 设计用于检测的场景.....27
  - 6.2.3 在 Vertex Pass 中的检测结果.....28
  - 6.2.4 无效数据 .....28
- 6.3 顶点照明和 Unity 存放光源的第二种方式.....29





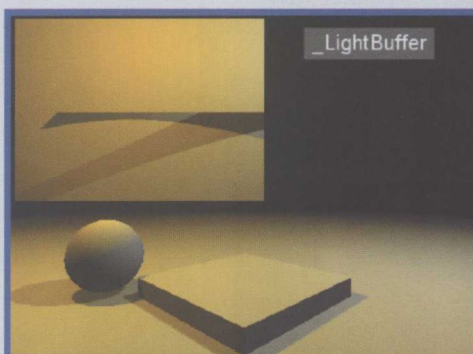
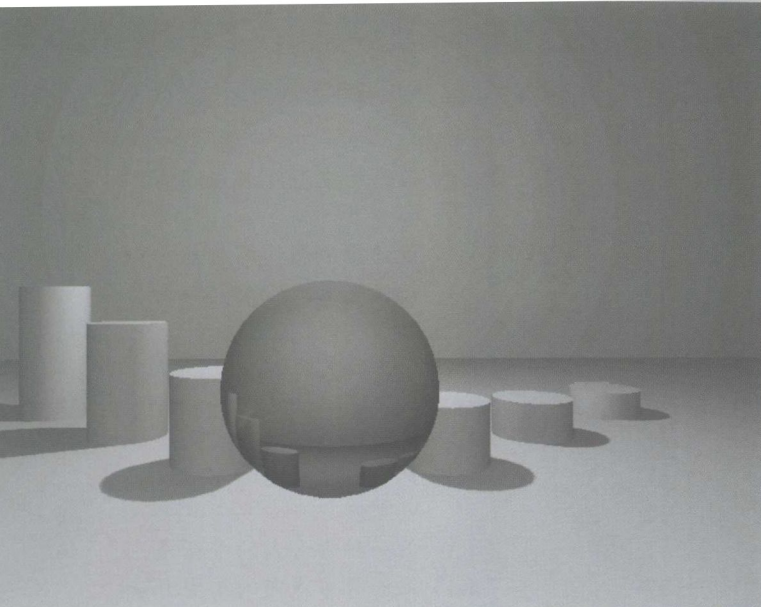
6.3.1	用于调试输出的材质	29
6.3.2	设计用于检测的场景	30
6.3.3	在 Vertex Pass 中的检测结果	30
6.4	顶点照明和 Unity 存放光源的第三种方式	31
6.4.1	Unity 为 Vertex Pass 准备的光源	31
6.4.2	设计用于检测的场景	31
6.4.3	顶点照明中的点光源	32
6.4.4	计算顶点照明的 ShadeVertexLights 函数	32
6.4.5	顶点照明中的 Pixel 光源	33
6.4.6	顶点照明中的平行光	33
6.4.7	顶点照明中的灯光信息小结	35
6.4.8	一个顶点照明的实现例子	35



## 第 7 章 Forward 渲染路径

7.1	ForwardBase 和 ForwardAdd	37
7.1.1	设计检测用的场景和材质	37
7.1.2	ForwardBase 和 ForwardAdd 的表现	39
7.2	Forward 渲染路径下的重要光源	39
7.2.1	设计检测用的材质	39
7.2.2	不存在 Pixel 光源时的情况	40
7.2.3	存在 Pixel 平行光时的情况	40
7.2.4	存在 Pixel 点光源时的情况	40
7.2.5	有多种类型的 Pixel 光源时的情况	41
7.2.6	Forward 渲染路径下的 Pixel 光源小结	41
7.3	重要光源在 ForwardAdd 内的执行	41

7.3.1	设计用来检测 Pixel 光源的材质	41
7.3.2	设计检测用的场景	44
7.3.3	检测结果: ForwardAdd 如何被执行	44
7.4	ForwardBase 和 Unity 存放光源的第一种方式	45
7.4.1	设计检测用的材质	45
7.4.2	第一种方式内的 Vertex 点光源	46
7.4.3	第一种方式内的平行光	46
7.4.4	第一种方式内的重要 Pixel 点光源	46
7.4.5	只有 ForwardBase 时的情况总结	47
7.4.6	ForwardAdd 对 ForwardBase 内光源的影响	47
7.4.7	有 ForwardAdd 时存放光源数据第一种方式的总结	49
7.5	ForwardAdd 和 Unity 存放光源的第一种方式	49
7.5.1	设计检测用的材质	49
7.5.2	设计检测用的场景	50
7.5.3	ForwardAdd 内的 Pixel 光源	50
7.5.4	ForwardAdd 内的平行光	51
7.5.5	数组变量 unity_4LightPos 的使用情况分析	51
7.6	Forward 渲染路径和 Unity 存放光源的第三种方式	51
7.6.1	检测 ForwardBase 内情况的材质	51
7.6.2	检测结果: 第三种方式不包含对 ForwardBase 有效的数据	52
7.6.3	检测结果: 第三种方式不包含对 ForwardAdd 有效的数据	52
7.7	Forward 渲染路径总结	53
7.7.1	Forward 渲染路径下材质的适应性	53
7.7.2	Unity 如何为 Forward 渲染路径设置光源	53



更换材质: :Forward

Sphere Left

Sphere Middle

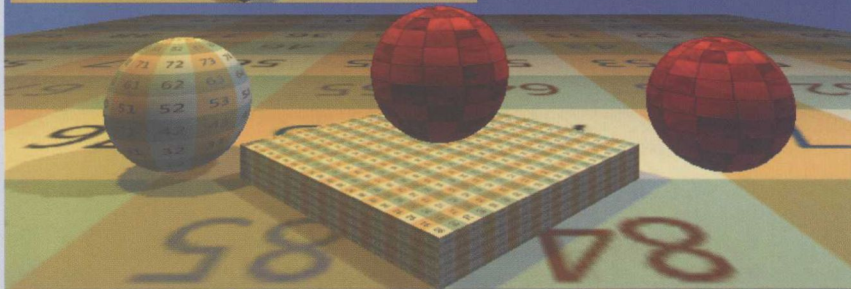
Sphere Right

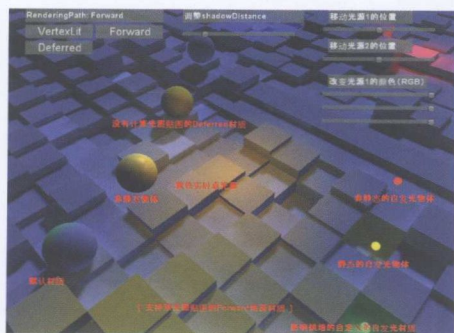
更换材质: :Deferred

Sphere Left

Sphere Middle

Sphere Right

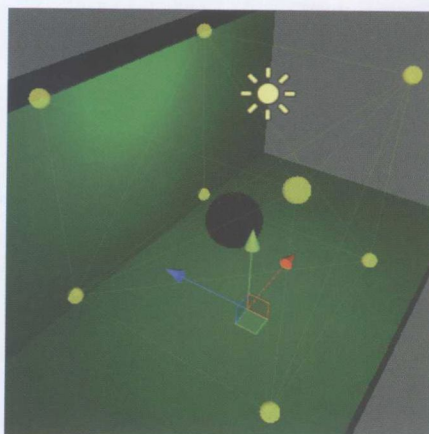




## 第 8 章 基于光照贴图的烘焙照明.....54

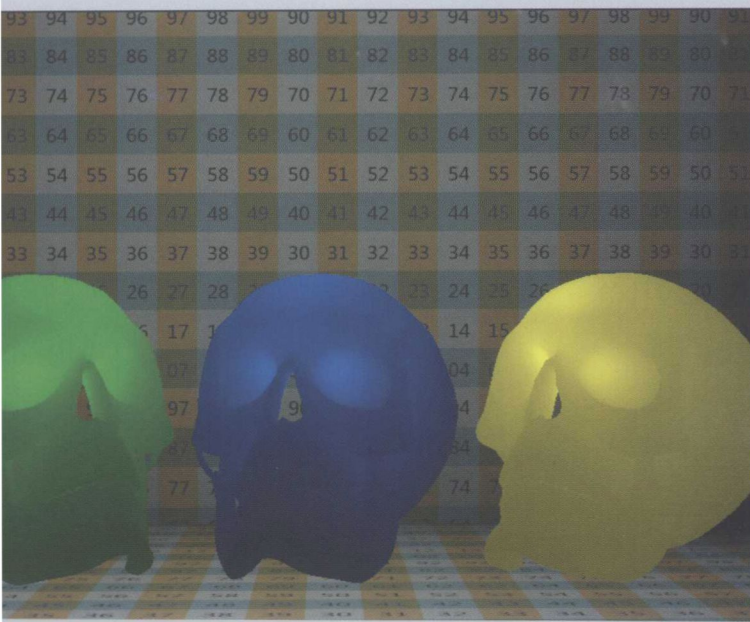
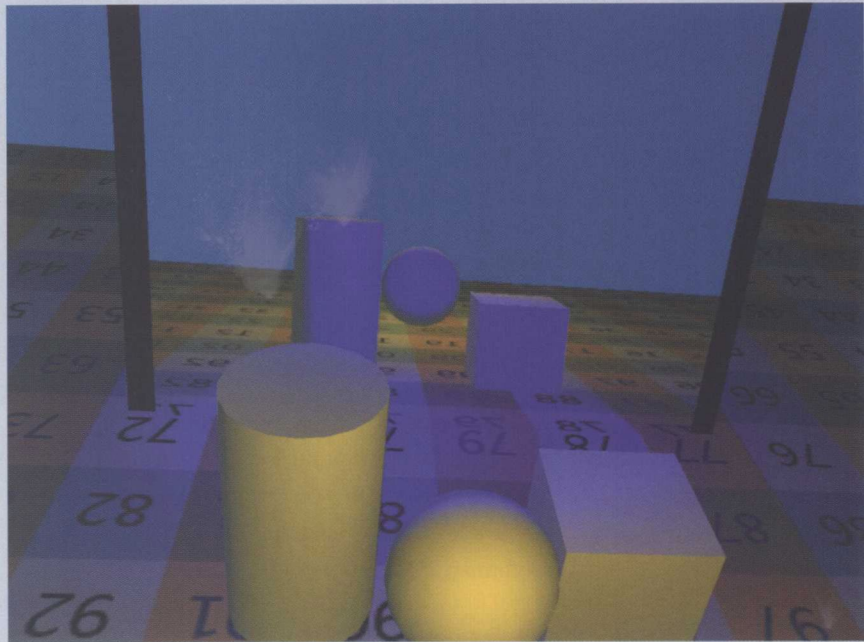
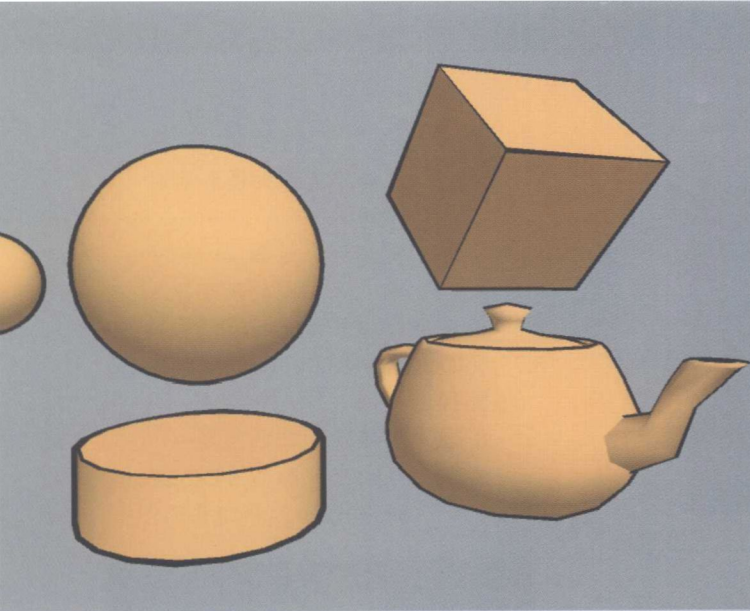
- 8.1 单光照贴图和 VertexLit 渲染路径.....54
  - 8.1.1 测试烘焙的场景.....54
  - 8.1.2 烘焙场景中使用的材质.....55
  - 8.1.3 烘焙的前提：静态物体.....55
  - 8.1.4 如何在烘焙中使用自发光材质.....55
  - 8.1.5 烘焙之后静态物体和非静态物体的实时照明.....57
  - 8.1.6 应用光照贴图到 VertexLit 渲染路径下的材质中.....57
  - 8.1.7 通过自己的材质改变实时光源对烘焙后物体的照明.....59
- 8.2 在效果和性能间进行权衡.....60
  - 8.2.1 影响全局的 Resolution 选项.....60
  - 8.2.2 影响单个物体的 Scale In Lightmap 选项.....61
- 8.3 单光照贴图和 Forward 渲染路径.....62
  - 8.3.1 单光照贴图在 VertexLit 和 Forward 下面的不同表现.....62
  - 8.3.2 准备可应用于烘焙的自发光材质.....62
  - 8.3.3 在 ForwardBase 内计算光照贴图.....64
  - 8.3.4 Forward 渲染路径下烘焙之后的实时照明.....65
- 8.4 单光照贴图在 Deferred 渲染路径下的实时阴影.....67
- 8.5 双光照贴图和 Deferred 渲染路径.....67
  - 8.5.1 全局 GI、间接照明以及双光照贴图.....67

- 8.5.2 混合双光照贴图和实时照明.....67
- 8.5.3 观察混合过程.....69
- 8.5.4 双光照贴图的使用限制.....70
- 8.6 双光照贴图和 Forward 渲染路径.....71
- 8.7 方向光照贴图和 Forward 渲染路径.....71
  - 8.7.1 烘焙后的凹凸问题.....71
  - 8.7.2 方向光照贴图 (Direction Lightmaps) 和凹凸贴图.....72



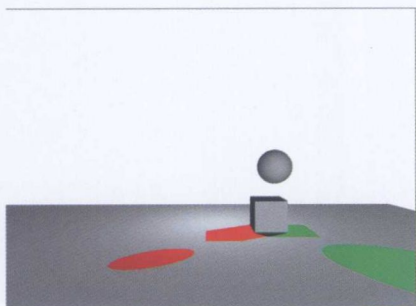
## 第 9 章 基于 LightProbes 的照明.....74

- 9.1 初识 LightProbes.....74
  - 9.1.1 LightProbes 照明的优点.....74
  - 9.1.2 检测 LightProbes 照明的场景.....74
  - 9.1.3 使用 Light Probe Group 进行管理.....76
  - 9.1.4 烘焙场景光照信息到 LightProbes 中.....76
  - 9.1.5 对比 Light Probes 照明和实时照明.....77
- 9.2 放置 LightProbes 的注意事项.....77
  - 9.2.1 必须形成一个体积.....77
  - 9.2.2 单个 Light Probe 必须处于采样光源的照射范围.....77
- 9.3 动态更新 LightProbes.....78
  - 9.3.1 跟新数据的注意事项.....78
  - 9.3.2 更改不同通道的 Coefficient.....78
- 9.4 照明采样的 Anchor Override.....79



9.4.1	基于线性插值的采样	79
9.4.2	改变默认的插值位置	79
9.5	LightProbes 照明和阴影	80
9.5.1	LightProbes 和光照贴图 的异同	80
9.5.2	烘焙阴影时可能会犯 的错误	80
9.5.3	将静态物体的阴影烘焙到 Light Probe 上	81
9.5.4	LightProbes 照明和实时 阴影的混合	82
9.6	烘焙一个色彩丰富的场景	82
9.7	在自己的材质中使用 LightProbes	83
9.7.1	为 Forward 渲染路径的材质 计算 LightProbes	84
9.7.2	使用 ShadeSH9 函数	84
9.7.3	在一个 Surface Shader 中 进行计算	85

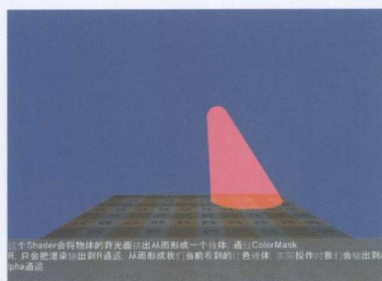
### 第 3 篇 使应用更逼真—— Shadows (阴影)



第 10 章	平面阴影	88
10.1	平行光对平面的投影	88
10.1.1	对平行光投影的考虑	88
10.1.2	进出阴影接受平面 的矩阵	88
10.1.3	使用三角形相似 计算阴影	89
10.2	点光源对平面的投影	90
10.3	阴影的淡出	91
10.3.1	有效利用计算平面阴影 过程中的数据	91
10.3.2	潜在的问题	91



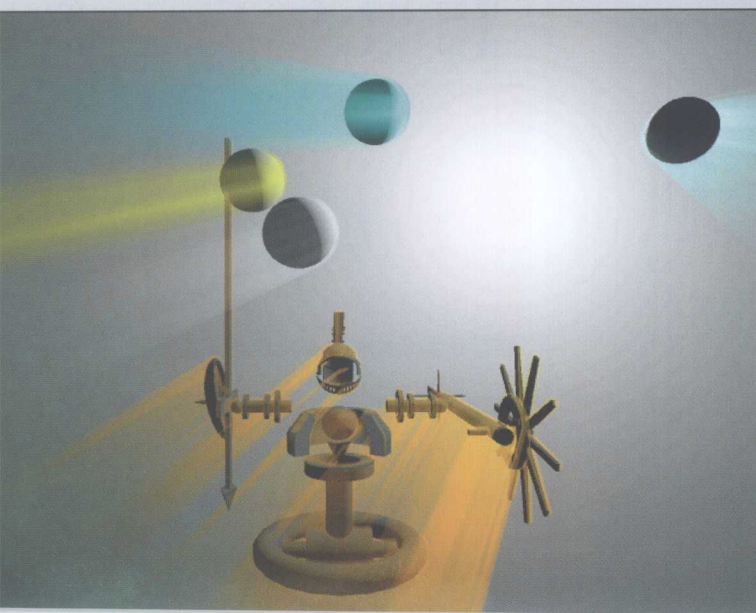
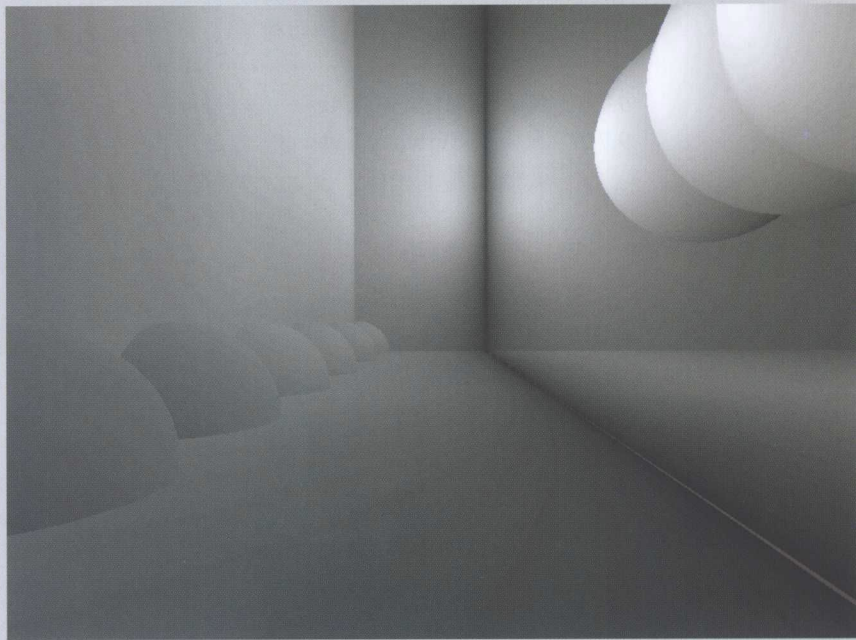
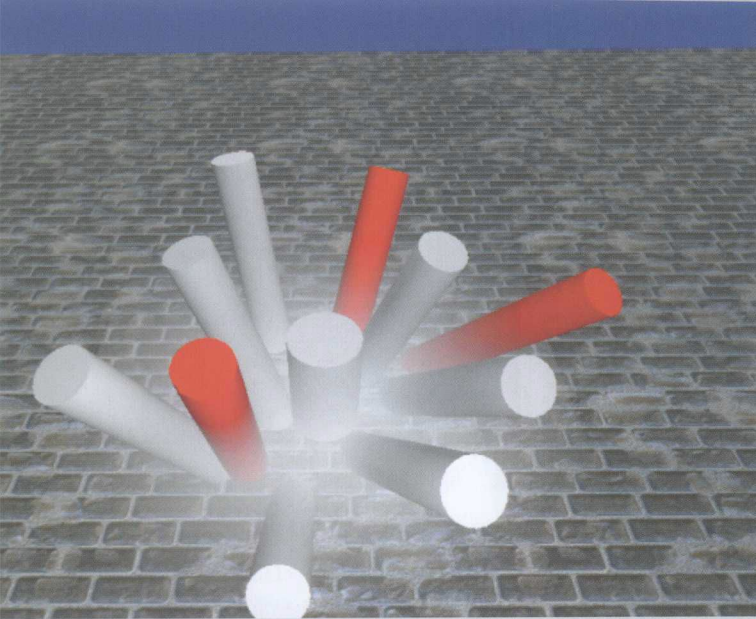
第 11 章	球体阴影	92
11.1	平行光对球体的投影	92
11.1.1	投影球体的信息	92
11.1.2	使用相似三角形计算投影	92
11.2	阴影的淡入/淡出	93
11.3	点光源对球体的投影	94



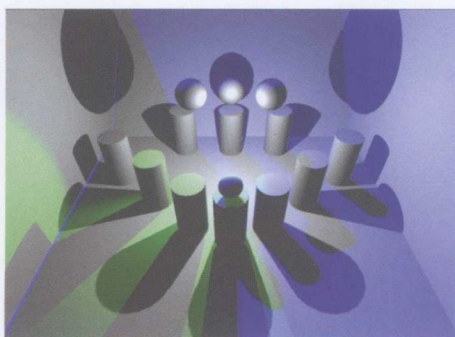
第 12 章	体积阴影	95
12.1	将顶点沿某一方向挤出	95
12.1.1	在 Vertex 函数中操作	95
12.1.2	判断顶点是向光还是背光	95
12.2	从 Volumes 中找到阴影区域	96
12.2.1	两次挤出	96
12.2.2	计算出阴影区域	97
12.2.3	渲染阴影	97
12.2.4	需要注意的问题	98



第 13 章	阴影映射	99
13.1	灯光空间和相机空间	99



13.1.1	观察两个空间	99
13.1.2	两个视角的 Z 深度	99
13.1.3	渲染 Z 深度的材质	99
13.2	投射 Z 深度	100
13.2.1	准备灯光视角的 投影矩阵	100
13.2.2	在材质中计算投影后 的 Z 深度	101
13.3	比较 Z 深度	103
13.3.1	比较 Z 深度的材质	103
13.3.2	Z 精度引起的问题	104
13.3.3	增加 Z 的精度	104
13.3.4	对 Z 值进行偏移	105



## 第 14 章 内置的阴影 ..... 107

14.1	投射阴影	107
14.1.1	使用 ShadowCaster 投射阴影	107
14.1.2	ShadowCaster 里都做 了什么	108
14.1.3	写一个自己的 ShadowCaster	108
14.1.4	改变 ShadowCaster 的行为	109
14.1.5	阴影和 FallBack 机制	110
14.2	接受阴影	111
14.3	Surface Shader 和阴影	112
14.3.1	Surface Shader 的阴影 和 Fallback	112
14.3.2	Surface Shader 里的灯光 参数和阴影	112
14.3.3	Surface Shader 对 Forward 渲染路径下 阴影的支持	112

## 第 4 篇 Unity 中的各种 Shader



## 第 15 章 Pass 的通用指令开关 ..... 116

15.1	使用 LOD 在运行时决定材质	116
15.1.1	材质的 LOD	116
15.1.2	运行时设定单个材质 的 LOD	116
15.1.3	设定全局所有材质 的 LOD	117
15.1.4	Unity 内置的 LOD 层级	118
15.2	渲染队列	118
15.2.1	标签队列和渲染顺序	118
15.2.2	渲染队列和 ZTest 判断	120
15.2.3	Unity 中内置的渲染 队列	120
15.3	透明的产生	120
15.3.1	Alpha 检测和 8 种 比较条件	120
15.3.2	动态生成 AlphaTest 的材质	120
15.3.3	动态生成 Shader 的内容	121
15.3.4	结合 AlphaTest 和 Blend 操作	122
15.4	混合操作	123
15.4.1	什么是混合 (Blend) 操作	123
15.4.2	动态生成测试用的材质	123
15.4.3	生成 Shader 的代码	124
15.4.4	检测不同的混合操作	125
15.4.5	BlendOp 选项	126
15.4.6	动态生成带 BlendOp 选项的材质	126
15.4.7	生成 Shader 的代码	127



