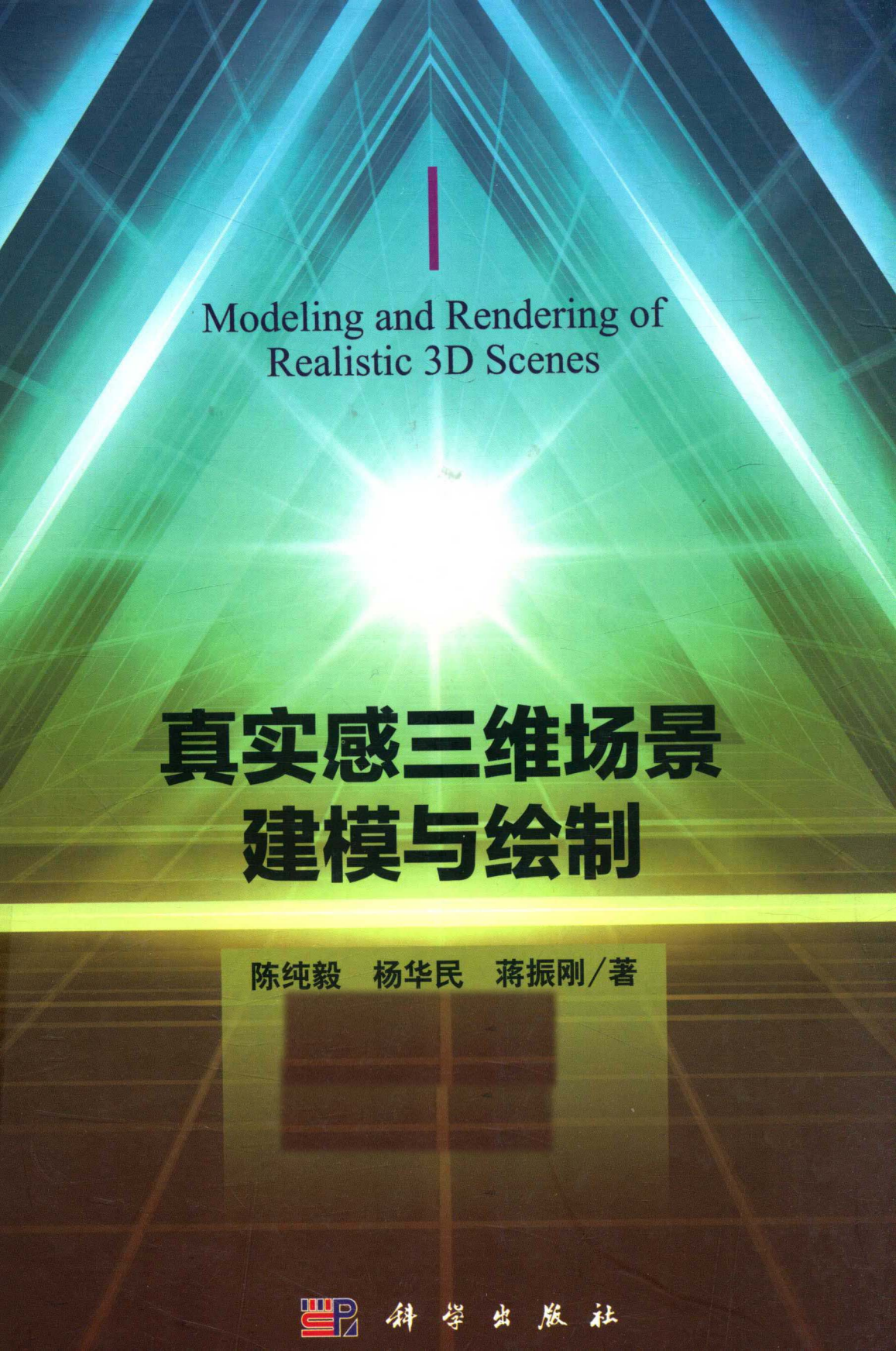




Modeling and Rendering of
Realistic 3D Scenes

真实感三维场景 建模与绘制

陈纯毅 杨华民 蒋振刚/著



科学出版社

真实感三维场景建

陈纯毅 杨华民 蒋振刚 著



科学出版社

北 京

内 容 简 介

当前,动画与影视特效制作、计算机三维游戏、虚拟现实、增强现实等行业都广泛使用计算机来根据虚拟三维场景模型生成逼真的图像画面。真实感三维场景建模的关键是,利用数学模型来描述三维场景中的几何物体及其与光照之间的交互过程。真实感三维场景绘制的关键是,计算从光源发出的沿不同路径传播到达视点的光照值。本书首先讨论真实感三维场景建模、光照传输、绘制方程、面光源可见性计算等问题,然后重点论述基于光线跟踪的真实感三维场景绘制、基于路径跟踪的真实感三维场景绘制、基于虚拟点光源的真实感三维场景绘制等全局光照效果绘制方法。

本书可作为从事三维图形绘制、游戏设计、虚拟现实与增强现实等研究的科研工作者、大学教师等的参考用书,也可以作为相关专业的大学本科和硕士研究生课程的教学参考书。

图书在版编目(CIP)数据

真实感三维场景建模与绘制/陈纯毅,杨华民,蒋振刚著. —北京:科学出版社, 2017.2

ISBN 978-7-03-051762-3

I. ①真… II. ①陈… ②杨… ③蒋… III. ①计算机辅助设计—应用软件 IV. ①TP391.72

中国版本图书馆 CIP 数据核字(2017)第 027047 号

责任编辑:张震姜红 / 责任校对:李影

责任印制:张倩 / 封面设计:无极书装

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

中国科学院印刷厂印刷

科学出版社发行 各地新华书店经销

*

2017 年 2 月第 一 版 开本:720×1000 1/16

2017 年 2 月第一次印刷 印张:12 插页:1

字数:245 000

定价:79.00 元

(如有印装质量问题,我社负责调换)

前 言

真实感三维场景建模与绘制是计算机图形学的一个重要研究方向,已经广泛应用在动画与影视特效制作、计算机三维游戏、虚拟现实、增强现实等领域之中。在强烈的实际应用需求推动下,当前学术界和产业界对真实感三维场景建模与绘制技术的研究非常活跃,例如,在 SIGGRAPH 国际学术会议上就经常有像皮克斯动画工作室和华特·迪士尼动画工作室这样的国际知名企业发布技术研究与应用报告。学术界在真实感三维场景建模与绘制方面已经开展了大量研究工作,取得了丰硕的研究成果。在此基础上,不少公司也发布了许多优秀的三维场景绘制软件,并在相关行业中推广使用。随着计算机硬件性能的不不断提升,目前,人们已经能够根据真实感三维场景模型绘制出非常逼真的画面。尽管如此,人们对真实感三维场景建模与绘制技术的求真和求快热情并未消退。相反,由于近年来相关行业对真实感三维场景绘制技术的依赖性越来越高,人们投入了更多精力来探索和设计新的绘制算法,以使用更短的时间绘制出更逼真的画面。

作者从 2007 年开始从事特种电影、增强现实影视摄制、交互式三维场景绘制等方面的研究工作,曾为云南普洱茶博物馆和长影世纪城电影主题公园建成大型正交多幕全景立体特种影院并制作出特种电影样片。在开展项目技术攻关的过程中,作者逐渐对真实感三维场景建模与绘制技术有了深入的认识和理解,也陆续开展了相关学术研究工作,这是撰写本书的基础。

本书主要讨论真实感三维场景的建模与绘制问题。全书内容总共分成 7 章论述,除第 1 章绪论外,其余 6 章涉及真实感三维场景建模、光照传输与绘制方程、面光源可见性计算与柔和阴影绘制、基于光线跟踪的真实感三维场景绘制、基于路径跟踪的真实感三维场景绘制、基于虚拟点光源的真实感三维场景绘制等内容。

第 2 章涉及的真实感三维场景建模部分,首先讨论三维几何对象的描述和建模方法;然后进一步介绍物体材质特性建模理论,为三维物体指定材质模型是真实感三维场景建模中的关键工作;最后详细讨论三维场景中的光源建模问题,分别给出理想点光源、方向性光源和面光源的描述方法,并论述不同类型的光源对三维场景画面的影响。

第 3 章涉及的光照传输与绘制方程部分,首先讨论光与物体之间的交互形式,给出光反射、光折射、光照遮挡等的数学描述模型;然后论述三维场景绘制方程的导出过程,并详细讨论绘制方程的近似求解方法;最后对三维场景中的光照传播路径分类方法进行介绍,并分析不同三维场景光照效果所对应的光照传播路径类型。

第4章涉及的面光源可见性计算与柔和阴影绘制部分,首先介绍硬边阴影与柔和阴影形成原因;然后详细讨论面光源可见性函数积分的近似求解问题;接着分别讨论基于蒙特卡罗光源采样的柔和阴影绘制方法、基于环境遮挡掩码的柔和阴影绘制方法以及近似柔和阴影绘制方法,并在论述的过程中分析相关方法的优缺点。

第5章涉及的基于光线跟踪的真实感三维场景绘制部分,首先介绍光线跟踪的基本概念与光线跟踪抗失真原理;然后重点讨论光线跟踪的并行化设计方法,其中包括基于单个众核处理器的并行化设计和基于计算机集群的并行化设计,也讨论光线跟踪阴影测试与柔和阴影绘制加速方法;最后简要地介绍英伟达(NVIDIA)公司发布的OptiX光线跟踪引擎及其使用方法。

第6章涉及的基于路径跟踪的真实感三维场景绘制部分,首先介绍路径跟踪的基本思想,论述基本路径跟踪和双向路径跟踪的原理;然后给出基本路径跟踪的算法实现流程;接着重点讨论光线传播路径采样中的重要性采样理论,并给出多重重要性采样的实现方法;最后介绍光子映射算法原理。

第7章涉及的基于虚拟点光源的真实感三维场景绘制部分,首先介绍虚拟点光源的概念及其与间接光照计算的关系;然后分别讨论基于反射阴影图的间接光照计算和基于帧间虚拟点光源重用的动态场景间接光照计算问题;最后简要介绍基于重要性缓存策略的虚拟点光源光照贡献求解方法和基于矩阵行列采样的虚拟点光源光照贡献求解方法。

作者在依托特种电影技术及装备国家地方联合工程研究中心承担一系列科研项目的过程中逐步积累出本书内容的核心素材。作者曾先后得到国家科技支撑计划(项目编号:2009BAE69B01、2009BAE69B02)、吉林省科技发展计划(项目编号:20076010、2012ZDGG004、20140204009GX、20150312029ZX)、长春市科技攻关计划(项目编号:14KG008)等的经费支持,在此表示由衷的感谢。作者指导的研究生在参与相关项目研究的过程中也有一定贡献。本书的部分内容来源于第一作者在吉林大学做博士后研究期间取得的成果,因此也特别感谢吉林大学对第一作者在做博士后研究期间给予的支持。

本书是作者对近年来的研究工作进行系统整理的初步尝试。由于作者学识水平有限,本书可能存在各种不足之处,希望广大同行批评、指正。

作者

2016年9月

目 录

前言

第 1 章 绪论	1
1.1 三维场景建模	1
1.2 光照与人眼对物体的视觉感知	4
1.3 三维场景绘制	6
参考文献	9
第 2 章 真实感三维场景建模	11
2.1 三维几何对象描述与建模	11
2.2 物体材质建模	14
2.2.1 经验材质模型	16
2.2.2 基于物理的材质模型	18
2.3 光源建模	22
2.3.1 理想点光源	22
2.3.2 方向性光源	25
2.3.3 面光源	26
参考文献	28
第 3 章 光照传输与绘制方程	30
3.1 光与物体之间的交互	30
3.1.1 反射	30
3.1.2 折射	31
3.1.3 遮挡	32
3.2 三维场景的绘制方程	34
3.2.1 通用绘制方程的导出	34
3.2.2 绘制方程的近似求解	36
3.3 光照传播路径的分类	41
参考文献	43

第4章 面光源可见性计算与柔和阴影绘制	45
4.1 硬边阴影与柔和阴影	45
4.2 面光源可见性函数积分的近似求解	48
4.3 物理正确柔和阴影绘制算法	51
4.3.1 基于蒙特卡罗光源采样的柔和阴影绘制	51
4.3.2 基于环境遮挡掩码的柔和阴影绘制	55
4.4 近似柔和阴影绘制算法	64
4.4.1 基于对象的近似柔和阴影绘制算法	65
4.4.2 基于图像的近似柔和阴影绘制算法	68
参考文献	73
第5章 基于光线跟踪的真实感三维场景绘制	75
5.1 光线跟踪基本概念	75
5.1.1 光线跟踪基本流程	75
5.1.2 光线与几何图元间的求交	78
5.1.3 加速光线跟踪计算的思路	83
5.2 光线跟踪抗失真	85
5.2.1 产生光线跟踪失真的原因	85
5.2.2 超采样	87
5.2.3 随机光线跟踪	87
5.3 光线跟踪并行化设计	88
5.3.1 线索化 BVH 场景加速结构的 GPU 并行创建	88
5.3.2 线索化 BVH 的遍历算法设计	104
5.3.3 阴影计算加速设计	105
5.3.4 动态三维场景的光线跟踪	112
5.3.5 基于集群的并行化光线跟踪	117
5.4 OptiX 光线跟踪引擎	125
5.4.1 OptiX 光线跟踪引擎简介	125
5.4.2 OptiX 光线跟踪引擎的使用	129
参考文献	131
第6章 基于路径跟踪的真实感三维场景绘制	134
6.1 路径跟踪的基本思想	134
6.2 基本路径跟踪流程	136
6.3 光线传播路径采样	140

6.3.1 基本的重要性采样	140
6.3.2 多重重要性采样	143
6.4 路径跟踪算法扩展	144
参考文献	149
第7章 基于虚拟点光源的真实感三维场景绘制	150
7.1 虚拟点光源与间接光照近似计算	150
7.2 基于反射阴影图的间接光照计算	153
7.2.1 算法总体描述	153
7.2.2 虚拟点光源的采样及间接光照计算	155
7.3 基于帧间虚拟点光源重用的动态场景间接光照计算	161
7.3.1 虚拟点光源的创建	161
7.3.2 可视场景点的间接光照计算	163
7.3.3 虚拟点光源的动态更新与重用	168
7.4 基于重要性缓存策略的虚拟点光源光照贡献求解	173
7.5 基于矩阵行列采样的虚拟点光源光照贡献求解	177
参考文献	182

第1章 绪 论

真实感三维场景建模与绘制技术广泛应用在动画影视制作、虚拟现实、电脑游戏等领域中,近几十年来在实际应用需求的驱动下不断向前发展,已经成为计算机图形学中的一个非常重要的研究方向。美国的皮克斯动画工作室和华特·迪士尼动画工作室多年来一直利用真实感三维绘制技术来制作动画影片^[1,2],在商业上取得了巨大的成功。为了根据三维场景模型绘制出具有照片级真实度的图像画面,首先要求建立符合真实物理特性的三维场景模型,其次要求在绘制三维场景时能收集到不同类型的光线传播路径携带的光能量。真实感三维场景建模的一个关键任务是,为三维几何对象指定合适的材质模型,以便正确地描述光照与物体之间的交互过程。在计算机图形学中,材质用于建模光照与几何对象之间的交互,常见的材质模型包括经验材质模型和基于物理的材质模型两类。不同材质的几何对象与光照发生不同类型的交互,引起具有不同特征的光散射,使光照在三维场景中沿不同类型的路径传播。各种光线传播路径携带的光能量在最终绘制的图像画面中产生不同的视觉效果。照片级真实感画面绘制的计算开销通常非常大,多年来该领域研究的主要目标是,如何在保证画面绘制真实度的条件下,设计新的算法来减小绘制计算开销、提高绘制效率。在实际应用中受硬件计算能力的限制,往往需要在画面真实度和绘制效率之间进行折中。然而,学术界和产业界对更高画面质量和更快绘制速度的永恒追求,为真实感三维场景建模与绘制技术研究提供了不竭的动力源泉。

1.1 三维场景建模

从总体上说,三维场景模型由几何数据、材质数据、纹理数据等组成。几何数据用于描述三维场景中的几何对象,例如,办公室场景中包含的桌椅、电脑、文件柜等几何物体。顶点是最基本的可用来描述几何对象的数据,而每个顶点又可以用笛卡儿坐标系中的 x 、 y 、 z 三个分量来描述。因此,在计算机中用三个浮点数就可以表示一个三维顶点。通过为一系列顶点指定拓扑结构,可以构造出组成几何对象的基本多边形面片,由面片可以进一步构造出几何对象,如图 1-1 所

示。在建模时可为每个顶点指定法向量,面片内部某个位置的法向量可根据所在面片的各顶点的法向量插值计算得到。几何对象与光照之间的交互通过材质模型来描述。在建模时,也可以为顶点指定纹理坐标,以便在需要的时候根据纹理坐标读取纹理数据。

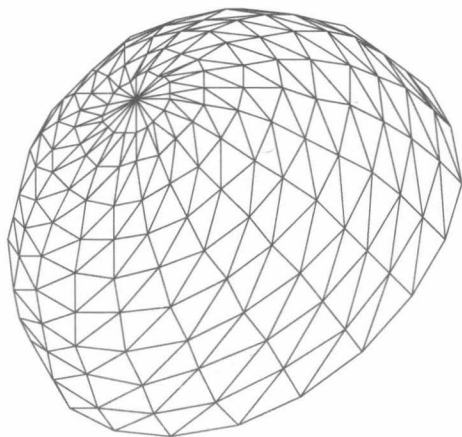


图 1-1 由三角形面片组成的半球面

在实际中,有多种创建三维场景模型的方法。一方面,可以利用三维扫描技术来对真实世界中的物体进行扫描,然后再根据扫描得到的点云数据重构多边形网格模型。例如,斯坦福三维扫描库(Stanford 3D Scanning Repository)中的三维模型就是通过三维扫描技术构建的。另一方面,也可以使用三维建模软件来创建三维场景模型。例如,使用 3DS Max、Maya、Blender、DAZ Studio 等软件就可以很方便地创建三维场景模型。当然,要想创建出复杂而精美的三维场景模型,要求三维场景构建人熟练地掌握各种三维建模技巧。

目前,常见的三维模型文件格式有 3DS、OBJ、POV、PLY 等。3DS 格式的三维模型可以使用 3DS Max 软件来创建,该软件由 AUTODESK 公司研发,可以提供图形化的界面来辅助用户完成三维建模工作。3DS 格式文件的基本组成单位是块,一个块由块索引、块数据、下一个块的位置等内容组成^[3]。OBJ 格式由 Alias|Wavefront 公司定义,也是一种常见的三维模型格式,可以被大多数三维建模软件导入和导出。OBJ 三维模型可以被保存在 ASCII 码格式的文本文件中,其使用关键字来说明数据的含义,例如,用“v”表示几何顶点,用“vt”表示顶点的纹理坐标,用“vn”表示顶点的法向量,用“f”表示多边形面片。POV 格式是 POV-Ray 光线跟踪绘制软件使用的三维场景模型描述格式。POV-Ray 是一个开源的光线跟踪绘制软件^[4],可以绘制出非常精美的三维场景画面。以下是一个完整的 POV 模型文件包含的代码,其对应的三维场景画面如图 1-2 所示。

```
global_settings { assumed_gamma 2.2 }
background {color red 0.3 green 0.3 blue 0.5}
camera {
    location <0, 0, -8>
    direction <0, 0, 1.2071>
    look_at <0, 0, 0>
}
sphere { <0.0, 0.0, 0.0>, 1.5
    finish {
        ambient 0.2
        diffuse 0.6
        phong 1
    }
    pigment { color red 1 green 0 blue 0 }
}
box { <-1.0, -0.2, -2.0>, <1.0, 0.2, 2.0>
    finish {
        ambient 0.2
        diffuse 0.9
        phong 1
    }
    pigment { color red 0.8 green 0.8 blue 1 }
    rotate <-20, 30, 0>
}
light_source { <-10, 3, -20> color red 1 green 1 blue 1 }
```

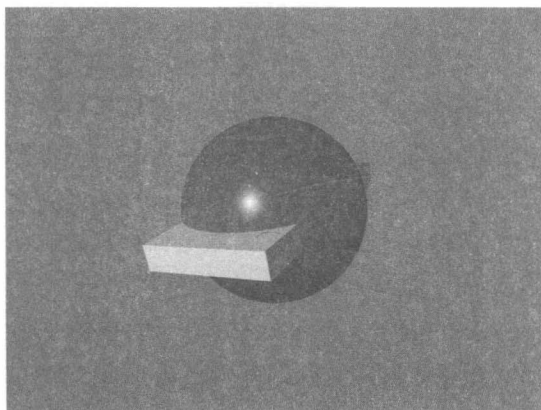


图 1-2 包含一个球和一个立方体的三维场景

POV 格式的三维模型描述文件类似于用脚本编写的代码。关键字“background”用来定义背景色，颜色通过红（red）、绿（green）、蓝（blue）三个分量来描述。关键字“camera”用来指定相机参数，包括相机位置（location）、

相机方向 (direction) 和相机正前方观察点 (look_at) 等属性。关键字 “sphere” 用来定义球, 包括球心位置、球半径、材质等属性。关键字 “box” 用来定义立方体, 包括立方体的体对角线上的两个顶点位置及材质、旋转参数等属性, 关键字 “light_source” 用来定义光源, 包括位置、发光颜色等属性。

PLY 格式由斯坦福大学设计, 是斯坦福三维扫描库中的模型存储格式, 该格式可以用来定义三维网格数据。在许多图形学研究论文中使用的牛、兔子、龙等三维模型都是斯坦福三维扫描库中的模型。与 OBJ 格式类似, PLY 格式模型也允许用 ASCII 码格式的文本文件来存储。相对于其他三维模型描述格式, PLY 格式的结构非常简单, 它只用来对单个三维几何对象的网格数据进行描述, 不能描述多个几何对象之间的组合关系。尽管如此, PLY 格式还是在各种图形应用中被广泛使用, 很多三维图形库都提供加载 PLY 格式的三维模型的辅助函数。

1.2 光照与人眼对物体的视觉感知

人眼能观察到真实世界中的各种物体的根本原因是, 这些物体散射 (或者从这些物体主动发射) 的光进入了人眼。人眼通过对光照的感知来识别各种物体的色彩、质感和几何特征, 进而辨别出各种物体之间的视觉差异。人眼对光照的感知结果直接表现为在现实世界中看到的五颜六色及其对应的亮度。在日常生活场景中, 大多数物体都不主动向外发光, 只是对入射到物体表面的光产生散射作用。在本书中, 术语 “散射” 既可以指 “反射” 也可以指 “折射”, 其用于表述光与物体之间的交互作用, 如图 1-3 所示。人们在生活中观察到的许多现象都和光散射有关。例如, 在晚上人眼能看见月亮, 是因为被月球散射的太阳光进入了人眼, 此时的散射就是反射; 人眼透过眼镜的镜片能看见在公路上行驶的汽车, 是因为入射到汽车上的光被汽车散射后再被镜片散射进了人眼, 其中, 汽车对光的散射属于反射, 镜片对光的散射则属于折射。焦散是一种常见的因光被光滑物体散射而形成的视觉效果, 如图 1-4 所示, 三维场景包含一条游动的鱼, 其材质被设置为玻璃材质, 光源位于鱼的上方, 在地面上形成了鱼的影子和焦散斑。

光是一种电磁波, 大多数情况下人眼可以感知 380~780 nm 波段的电磁波^[5]。换句话说, 在真实世界中, 人眼可以观察到的物体必须向外发射或者散射 380~780 nm 波长的电磁波。真实感三维场景绘制的核心思想是, 模拟光照在三维场景中的传播, 并计算最终进入视点的光照值。根据前面的叙述可知, 在三维场景绘制中只需考虑 380~780 nm 波段的光传播。人眼大约能区分 1000 万种不同的颜色, 在视网膜上有三类不同的视锥细胞能对各种波长的光产生不同的感应, 然而实际

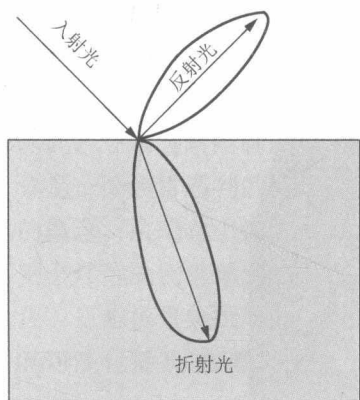


图 1-3 物体对入射光的散射

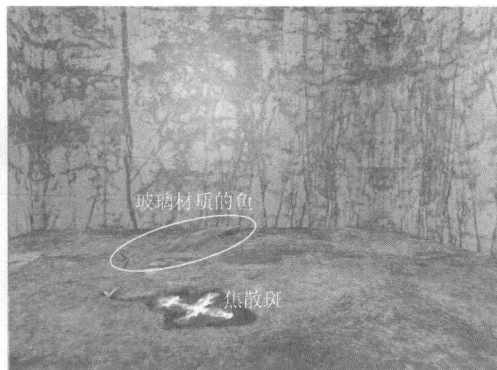


图 1-4 光被玻璃材质的鱼折射形成的焦散斑(见书后彩版)

上人脑只能收到三类不同的颜色感知信号^[5]。因此,在计算科学领域中,经常用三个数字来描述颜色,例如,图像处理中常用的 RGB 颜色空间就使用红(R)、绿(G)、蓝(B)三原色的不同组合来表示其他颜色。在真实感三维场景建模与绘制中,通常也用红、绿、蓝三个颜色分量的不同组合来表示各种颜色。从某种意义上说,颜色并不是物体的固有属性,颜色只是光与物体进行交互后的散射光属性,即物体的颜色取决于入射光的频谱分布及其与物体的交互过程。例如,在日光灯的照射下白墙看起来呈现白色,但是如果换成用发红光的灯照射白墙,则白墙看起来会呈现红色。所以,在绘制三维场景时,从三维场景点散射进入视点的光的颜色由入射光颜色和场景点的材质特性共同决定,这就是物体材质模型对三维场景画面有重要影响的原因。根据日常生活经验可知,太阳发出的白光照射到红色盒子上,人眼可以感知到盒子呈现红色,据此可以推断白光中的其他非红色分量在红色盒子表面上的反射率很低(即非红色光在散射过程中发生严重的衰减),经反射进入人眼的光基本上都是红光。因此,从严格意义上讲,在物体材质建模时,也应该分别为红、绿、蓝三种不同的光分量设置交互模型。当然,在实际三维场景建模时,有时为了简便,也把红、绿、蓝三种光分量与物体的散射交互模型设置为同一个模型,即用相同的双向散射分布函数来描述。

值得注意的是,入射到三维几何对象上的光可能源自其他几何对象的散射光,这些散射光的颜色实际上包含了相应几何对象的材质信息。例如,在发白光的日光灯照射下的办公室中,一个穿红衣服的人站在白色墙边,白色的墙可能会被映成红色,白墙上的红色就体现了红衣服的材质特性。这种一个物体表面映上了其他物体颜色的现象被称为颜色渗透。图 1-5 为一个可以观察到颜色渗透现象的三维场景,其中一个聚光灯光源位于屋顶,聚光灯向下照射,聚光灯发出的白光经鸽子反射后再入射到屋顶上(鸽子和屋顶都为漫反射材质),在屋顶上就会映上鸽

子的颜色。颜色渗透一般发生在从漫反射物体到漫反射物体的光照传输交互过程中, 需要使用特殊设计的算法才能绘制出这一视觉效果。



图 1-5 三维场景中的颜色渗透现象 (见书后彩版)

1.3 三维场景绘制

三维场景绘制就是根据三维场景模型计算出从视点能够看到的所有场景点的颜色值的过程。在计算机图形学文献中, 除了“绘制”这个术语外, 还经常见到用术语“渲染”来表达相关的概念。实际上, 这两个术语在真实感三维图形技术领域中的内涵既有密切联系也有细微差别。中文术语“绘制”对应于英文术语“rendering”, 中文术语“渲染”对应于英文术语“shading”^[6]。“渲染”的含义相对于“绘制”更窄一些, 主要指根据几何面片顶点的光照模型计算结果, 通过插值计算几何面片内部其他点的光照值的过程, 常见的渲染模型有“Gouraud Shading”和“Phong Shading”。如果把用计算机建立的三维场景看成是真实的三维世界, 则可以把绘制操作看成是用相机拍摄三维世界得到一张照片的过程。因此, 要讨论三维场景绘制, 首先必须说明虚拟相机的概念。与用真实相机可以拍摄真实三维场景类似, 虚拟相机可以用来拍摄虚拟三维场景的画面。图 1-6 给出了虚拟针孔相机的定义规范。虚拟针孔相机由相机位置、向前方向、向上方向、水平方向、近裁剪平面、远裁剪平面、水平视场角、竖直视场角等参数定义。虚拟针孔相机只能拍摄到位于近裁剪平面和远裁剪平面之间、水平视场角和竖直视场角范围内的三维几何对象, 这个可以被虚拟针孔相机拍摄到的三维空间范围通常被称为视域体。假设把视域体内的所有几何对象都投影到远裁剪平面上, 且先投影离相机位置较远的几何对象, 后投影离相机位置较近的几何对象, 若多个几何对象投影到远裁剪平面上的相同位置, 则后投影的结果将覆盖先前投影的结

果,最后远裁剪平面上的投影结果就包含了虚拟针孔相机拍摄到的几何对象信息。用均匀网格对远裁剪平面进行离散采样,计算出每个采样点的颜色值(指投影结果),就完成了对三维场景的一次绘制。前述离散采样网格可以被看成是像素矩阵。因此,三维场景绘制就是计算像素颜色的过程。在三维场景绘制技术领域,像素是一个经常遇到的概念。三维场景绘制中的像素与显示器上的像素并不是等同的概念。从概念上讲,显示器上的像素可以被简单地理解为一个包含红、绿、蓝三个灯泡的物理单元,通过控制三个灯泡的发光强度可以使像素显示出不同的颜色。三维场景绘制中的像素则是指与可视场景点外观描述相关的一个数据集合,其可能包括 RGB 颜色值、深度值、alpha 透明度等数据。

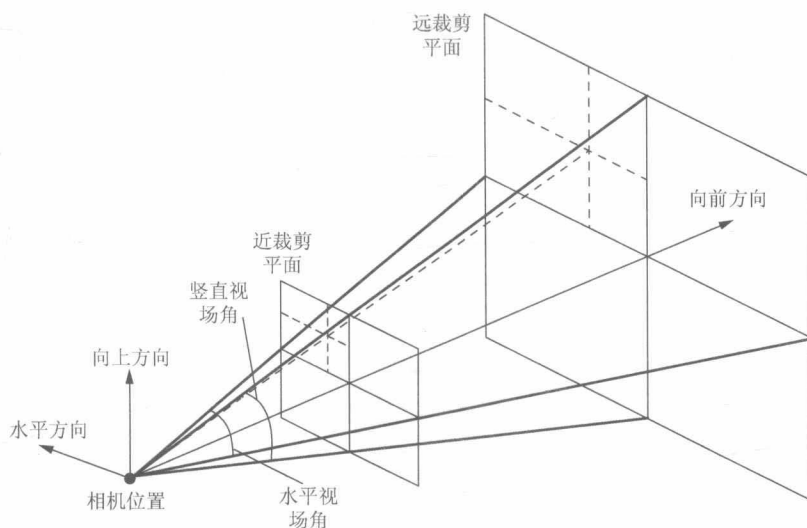


图 1-6 虚拟针孔相机的定义规范

当前,用于绘制三维场景的技术包括光栅化图形绘制管线、光线跟踪(ray tracing)、光能辐射度(radiosity)、路径跟踪(path tracing)等方法。这些方法适用于不同的应用场合。在需要实时交互的应用中,通常使用基于光栅化的图形绘制管线来快速绘制三维场景。在像动画电影场景绘制这样的离线应用中,为了生成逼真度的三维场景画面,可以使用光线跟踪、路径跟踪等绘制方法。OpenGL 和 DirectX 是当今最流行的两种三维图形编程接口,它们对光栅化图形绘制管线进行了实现,能够充分利用现代图形处理器(graphics processing unit, GPU)的并行计算能力完成三维场景的快速绘制。基本的 OpenGL 绘制管线如图 1-7 所示^[7]。在早期,OpenGL 和 DirectX 只支持固定功能的图形绘制管线,后来学术界和工业界一起努力又对固定功能图形绘制管线进行了扩展,通过引入着色器概念,提出了可编程图形绘制管线。可编程图形绘制管线的提出大大增强了图形绘制算法设

计的灵活性,目前人们已经能利用可编程图形绘制管线绘制许多复杂的三维场景视觉效果。例如, Wyman 和 Davis^[8]利用 OpenGL 和着色器实现了近似焦散效果绘制, Nguyen 等^[9]利用 DirectX 和着色器实现了近似柔和阴影绘制。不少三维图形开发引擎库对基本的 OpenGL 接口做了进一步的封装,从而方便程序设计人员开发三维图形应用。例如, G3D 图形引擎^[10]就是一个非常优秀的三维图形引擎库,它预定义了许多类来对 3D 数学库、Windows 窗口系统和图形处理器等进行抽象,向程序员提供 C++ 类型的应用程序接口,从而为三维图形应用开发提供重要的辅助支持。随着硬件计算能力的增强,光线跟踪、路径跟踪等三维场景绘制算法也越来越受到人们的重视,学术界逐渐推出了不少开源的光线跟踪绘制软件,例如, POV-Ray、PBRT 等^[4,11]。不少研究者都直接在相关开源软件的基础上,实现自己设计的真实感三维图形绘制算法^[12,13]。为了充分利用英伟达 GPU 的并行计算能力来加速光线跟踪计算,英伟达公司发布了 OptiX 光线跟踪引擎^[14]。利用该引擎不但可以开发光线跟踪应用,甚至也可以开发其他非光线跟踪应用。在实现新的三维图形绘制算法时,利用已有的开发库可以加快实现算法的进度。因此,熟练地掌握一两个三维图形绘制开发库的使用方法对真实感三维图形绘制技术研究很有帮助。

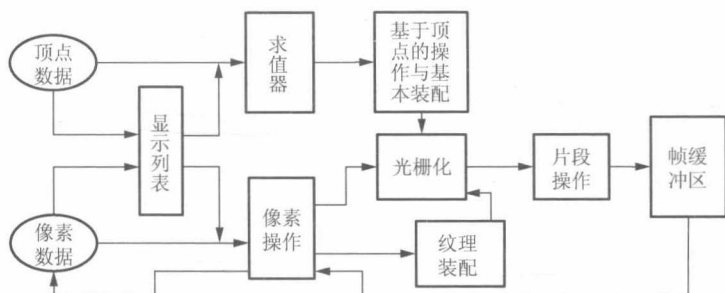


图 1-7 OpenGL 绘制管线

真实感三维场景绘制的关键是计算全局光照。从理论上讲,三维场景的全局光照是指光源发出的光经场景中不同物体多次散射后达到光能辐射平衡状态时的光亮度场。当然,在实际应用中,计算完整的全局光照解是非常困难的,很多情况下只计算局部光照或者全局光照的部分近似结果。在理想的情况下,真实感三维场景绘制算法应当具备以下两种能力。第一,要求能够正确地绘制三维场景中的柔和阴影。真实世界中的绝大多数阴影都是柔和阴影,柔和阴影的基本特征是拥有平滑变化的边缘。三维场景阴影计算与光源可见性计算密切相关。当用点光源照射三维场景时,场景点与光源之间的可见性为二值可见性,即要么为 1 (表示可见),要么为 0 (表示不可见),此时绘制出的阴影具有清晰的边缘,阴影区

与非阴影区之间不存在平滑过渡。当用面光源或者体光源照射三维场景时,场景点与光源之间的可见性不再是二值可见性,有可能光源对场景点部分可见、部分不可见,此时绘制出的阴影通常呈现柔和阴影特征。第二,要求能够正确地计算出经不同路径传播进入视点的光照值,从而绘制出光滑表面、漫反射表面、透明物体等与光交互后形成的视觉效果,其中包括镜面高光、焦散、颜色渗透、镜像等典型视觉效果。光传播路径可以分为光源发出的光经一次散射后进入视点的路径和光源发出的光经多次散射后进入视点的路径,前者对应直接光照(或者称为局部光照),后者对应间接光照。直接光照的计算相对简单,间接光照计算才是真实感三维场景绘制算法的核心问题。在电脑游戏中,经常使用环境遮挡(ambient occlusion)算法^[15,16]来提高三维场景绘制画面的逼真度,它能正确地绘制场景中的“阴暗角落”,是对绘制质量和计算开销两者进行折中后开发的“廉价技术”。在各种研究文献中常见的全局光照绘制算法有光能辐射度算法、光线跟踪算法、路径跟踪算法、光子映射算法(photon mapping)、面发光度缓存算法(irradiance caching)、面发光亮度缓存算法(radiance caching)、瞬态光能辐射度算法(instant radiosity)等^[17-26]。这些算法大多数可计算某些特定类型的光线传播路径携带的光能量,能够绘制出一种或几种全局光照效果。

参 考 文 献

- [1] Hery C, Villemin R. Physically based lighting at Pixar//Physically Based Shading in Theory and Practice. Proceedings of SIGGRAPH 2013 Course, Anaheim, 2013: 22.
- [2] Burley B. Extending the Disney BRDF to a BSDF with integrated subsurface scattering//Physically Based Shading in Theory and Practice. Proceedings of SIGGRAPH 2015 Course, Los Angeles, 2015: 22.
- [3] 胡卓玮, 杨国东, 田静, 等. 3D GIS 中 3DS 数据模型的可视化研究. 吉林大学学报(信息科学版), 2004, 22(1): 87-90.
- [4] Fava A, Fava E, Bertozzi M. MPIPOV: a parallel implementation of POV-Ray based on MPI. Lecture Notes in Computer Science, 2002, 1697: 426-433.
- [5] Akenine-Möller T, Haines E, Hoffman N. Real-Time Rendering. 3th edition. Wellesley: A K Peters Ltd., 2008.
- [6] 何援军. 论计算机图形学的若干问题. 上海交通大学学报, 2008, 42(4): 513-518.
- [7] Shreiner D, The Khronos Open GLARB Working Group. OpenGL 编程指南. 第七版. 李军, 徐波, 等译. 北京: 机械工业出版社, 2009.
- [8] Wyman C, Davis S. Interactive Image-Space Techniques for Approximating Caustics. Proceedings of Symposium on Interactive 3D Graphics, Redwood City, 2006: 153-160.
- [9] Nguyen K T, Jang H, Han J. Layered occlusion map for soft shadow generation. Visual Computer, 2010, 26(12): 1497-1512.
- [10] McGuire M. The G3D graphics engine. Dr. Dobb's Journal. [http://www.drdobbs.com/the-g3d-graphics-engine/184401883\[2004-12-1\]](http://www.drdobbs.com/the-g3d-graphics-engine/184401883[2004-12-1]).
- [11] Pharr M, Humphreys G. Physically Based Rendering: From Theory to Implementation. 2nd edition. Burlington: Morgan Kaufmann Publishers, 2010.