



在虚拟环境中,虚拟对象是主要的元素。它的虚拟再现是通过建模来实现的。建模是对现实对象或环境的逼真仿真。一般情况,对象具有静态特征,包括位置、方向、材料、属性等特征,还具有运动特征,它反映对象的运动、行为、约束条件(如碰撞检测与响应)以及力的作用等。虚拟对象的建模意味着对象的静态特征和运动特征各个方面的建模,也就是对形状、外观、运动学约束、智能行为和物理特性等方面的建模,如图 3-1 所示。建模时,首先获取实际三维环境的三维数据,并根据其应用的需要,建立相应的虚拟环境模型。设计出的虚拟模型能否真实反映研究的对象,直接决定了能否满足虚拟现实的三大特征:沉浸性、想象性和交互性,以及整个系统的可信度。



图 3-1 虚拟场景的建模技术



3.1 对象建模

对象建模是虚拟现实研究的重点,是使用户沉浸的首要条件。通常,对象建模主要研究对象的形状和外观的仿真,其过程主要包括建模和视觉外观的设计。

1. 建模

用一定的方式对对象进行直接的描述。它们的描述直接影响图形的复杂性和图形绘制的计算消耗。其建模方法一般包括几何建模、图像建模、几何与图像相结合的建模三种方法。

2. 视觉外观的设计

为场景添加光照和纹理映射。即根据基于光照模型和纹理映射,计算物体可见面投影到观察者眼中的光亮度大小和颜色分量,并将它转换为适合图形设备的颜色值,从而确定投影面上每一像素的颜色,最终生成真实感图形。

3.1.1 几何建模

几何建模是指对虚拟环境中物体的几何网络特性等信息的表示,主要包括几何模型的拓扑信息和几何信息。采用几何建模方法对物体对象虚拟主要是对物体几何信息的表示和

处理,描述虚拟对象的几何模型,例如多边形、三角形、顶点和样条等,即用一定的数学方法对三维对象的几何模型的描述。

目前,几何建模软件越来越多,建模方法也越来越多。但总体而言,可归纳为三大类:多边形(Polygon)、非统一有理 B 样条(NURBS)和构造立体几何(CSG)。但无论采用何种建模软件,同类的建模方法其数学原理大致相同。

1. 多边形

多边形建模是在三维制作软件中最先发展的建模方式。多边形建模是将点、边、面组成一系列线段和平面,通过把线段和平面嵌入到物体中生成一个多边形网格,然后网格逼近来生成模型。对模型的修改是通过对点、边、面三个元素的修改来完成。任何形状的物体,都可以用足够多的多边形勾画出来。

三维物体对象的显示处理过程包括各种坐标系的变换、可见面识别与显示方式等。这些处理需要有关物体单个表面部分的空间方向信息。这一信息源于顶点坐标值和多边形所在的平面方程。

多边形建模方法比较容易理解,并且在建模的过程中,使用者可以根据自己的想象对模型进行修改。但是,随着多边形数目的增加系统的性能会下降。目前,多边形建模在许多三维软件中应用广泛。例如在 3ds Max 中,用户利用多边形构造原始简单的模型,然后通过增减点、面数或调整点、面的位置生成所需要的模型。选择不同的命令,实现对多边形不同操作效果。例如,选择“挤出”命令实现多边形的拉伸和挤入;选择“轮廓”命令实现拉出面的缩放;选择“倒角”命令实现物体面的拉伸、挤入,然后再缩放的操作。图 3-2 所示为采用 3ds Max 软件多边形方法制作的大象模型。图 3-3 所示为采用 Maya 软件的多边形方法制作的轮胎实体模型。

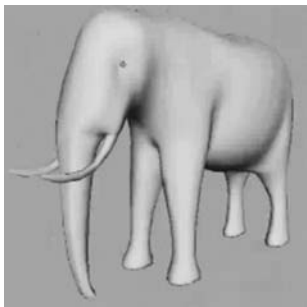


图 3-2 多边形方法制作的大象模型

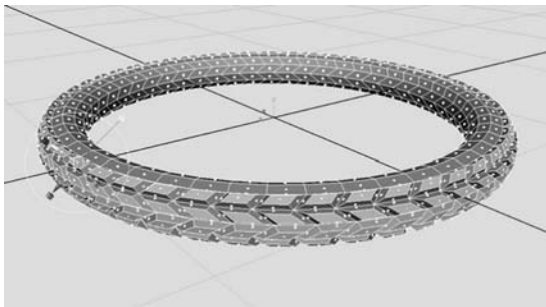


图 3-3 多边形方法制作的轮胎实体模型

2. 非统一有理 B 样条

NURBS 是一种非常优秀的建模方式,在高级三维软件中,例如 SoftImage、3ds Max 和 Maya 软件,都支持这种建模方式。NURBS 是 Non-Uniform Rational B-Splines 的缩写,即非统一有理 B 样条。具体解释为:

- Non-Uniform(非统一): 是指一个控制顶点的范围能够改变,用来创建不规则曲面。
- Rational(有理): 是指每个 NURBS 模型都可以用数学表达式来定义,也就意味着用以表示曲线或曲面的有理方程式给一些重要的曲线和曲面提供了更好的模型,特别是圆锥截面、球体等。

- B-Spline(B样条): 是一种在三个或者更多点之间进行插补的构建曲线的方法。

简单地说,NURBS 是在 3D 建模的内部空间用曲线和曲面来表现物体轮廓和外形,即用曲线和曲面来构造曲面物体。

度数(Degree)是 NURBS 的一个重要的参数,用于表现所使用的方程式中的最高指数。一个直线的度数是 1,一个二次等式度数为 2。NURBS 曲线通常由立方体方程式表示,其度数为 3。度数可以设置更高些,曲线更圆滑,但计算时间也越长,所以,通常不必这样做。

连续性(Continuity)是 NURBS 的另一个重要参数。连续的曲线是未断裂的,有不同级别的连续性,如图 3-4 所示。一条曲线有一个角度或尖端,则它具有 C0 连续性,如图 3-4(a)所示,角位于曲线顶部。一条曲线没有尖端,但曲率不断变化,则连续性为 C1,如图 3-4(b)所示。一个半圆形连接具有较小半径的半圆形,如果一条曲线是连续的,曲率恒定不改变,则连续性为 C2,如图 3-4(c)所示,右侧不是半圆形,与左侧混合。

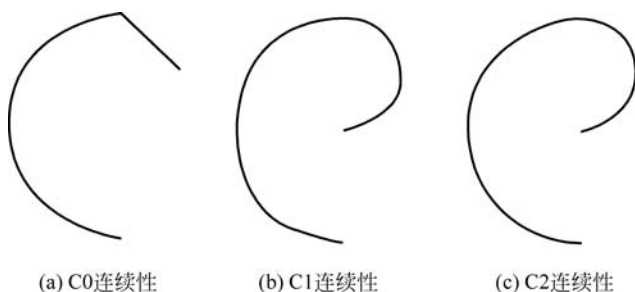
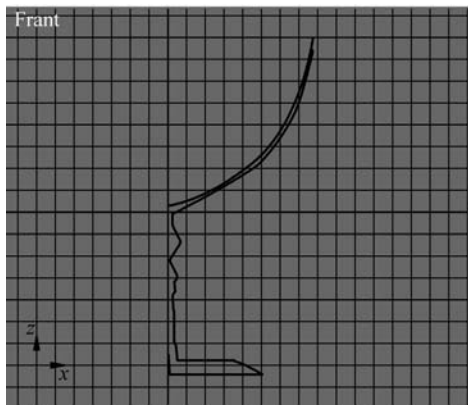


图 3-4 NURBS 的连续性

NURBS 构建几何对象时,首先建立简单的物体作为 NURBS 的起始物体,然后通过修改曲线的度数、连续性和可控点个数等参数来定义形状、制作各种复杂的曲面造型和特殊的效果。图 3-5 所示为采用 3ds Max 软件中的 NURBS 构建的酒杯。图 3-5(a)为使用任意一种 NURBS 曲线绘制的酒杯截面造型。图 3-5(b)为修改参数后得到的酒杯造型。NURBS 比传统的网格建模方式更好地控制物体表面的曲线度,从而能够创建出更逼真、生动的造型,通常用于描述汽车、人的皮肤、面貌等复杂的曲面对象。



(a) 酒杯截面造型



(b) 酒杯造型

图 3-5 酒杯截面造型

3. 构造立体几何(CSG)

CSG 是 Constructive Solid Geometry 的缩写,即构造立体几何,又称为布尔模型,它是一种通过布尔运算(并、交、差)将一些简单的三维基本体素(如球体、圆柱体、立方体等)拼合成复杂的三维模型实体的描述方法,就像搭建积木一样。例如,一张桌子可以由五个六面体组成,其中四个用作桌腿,一个用作桌面。

CSG 方法对物体模型的描述与该物体的生成顺序密切相关,即存储的主要是物体的生成过程。其数据结构为树状结构。树叶为基本体素或变换矩阵,节点为运算,最上面的节点是被建模的物体,如图 3-6 所示,E 物体是通过不同的基本体素——长方体 A 和 B、圆柱体 D,经过布尔运算——并和差,最后生成的。

CSG 方法的优点是简单,生成速度快,处理方便,易于控制存储的信息量,无冗余信息,而且能够详细地记录构成实体的原始特征参数,甚至在必要时可修改体素参数或附加体素对模型进行局部修改。图 3-7 所示为在物体上倒圆的过程。其缺点是由于信息简单,可用于产生和修改实体的算法有限,并且数据结构无法存储物体最终的详细信息,例如边界、顶点的信息等。

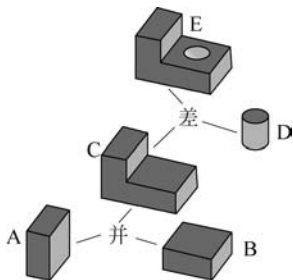


图 3-6 CSG 构造的几何模型

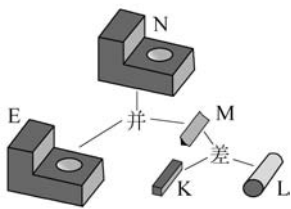


图 3-7 CSG 对模型的局部修改

3.1.2 图像建模

图像建模和绘制技术(Image Base Modeling and Rendering, IBMR)是指用预先获取的一系列图像(合成的或真实的)来表示场景的形状和外观,新图像的合成是通过适当的组合和处理原有的一系列图像来实现。与基于几何的建模和绘制模式相比,IBMR 有以下突出的优点。

(1) 建模容易。拍照容易,照片细节精细,不仅能直接体现真实景物的外观和细节,而且能从照片抽取对象的几何特征、运动特征等。IBMR 是把不同视线方向、不同位置的照片组织起来表现场景,如全景图像(Panoramic Image)和光场(Light Field)。

(2) 真实感强。图像既可以是计算机合成的,也可以是实际拍摄的画面缝合而成,两者可以混合使用,能较真实地表现景物的形状和丰富的明暗、材料及纹理细节,可获得较强的真实感。

(3) 绘制速度快。IBMR 方法只需要离散的图片采样,绘制时只对当前视点相邻的图像进行处理,其绘制的计算量不取决于场景复杂性,而仅仅与生成画面所需的图像分辨率相关。所以,绘制图形对计算资源的要求不高,仅仅需要较小的计算开销,有助于提高系统的运行效率。

(4) 交互性好。由于具有照片级的真实感,并且以位图存储。所以,在实时交互时,模型的真实感强,显示速度快,易实现实时性。

基于图像的绘制技术是基于一些预先生成的场景画面,对接近于视点或视线方向的画面进行交换、插值与变形,从而快速得到当前视点处的场景画面。基于图像的建模方法(IBR)的相关技术主要有两种:全景图建模技术和图像插值及视图变换技术。

1. 全景图建模技术

全景图建模技术指在一个场景中选择一个观察点,固定广角照相机或摄像头,然后在水平方向每旋转固定大小的角度(满足相邻照片的重叠部分达到 20% 以上)拍摄得到一组照片(通常 12 张以上),再采用特殊拼图工具软件拼接成一个全景图像,如图 3-8 所示。图 3-8(a)为采用广角照相机拍摄的 4 张照片,图 3-8(b)为采用全景图制作软件合成的室内全景图。图 3-9 所示为数字航空航天博物馆的全景图。全景图所生成的数据较小,对计算机要求低,适用于桌面型虚拟现实系统中,建模速度快。但照相机的位置被固定在一个很小的范围内拍摄,所以,观察的视点固定不动,视线做上下、左右及前后任意转动,交互性较差。详细内容参见第 4 章。



(a) 4张照片



(b) 合成的全景图

图 3-8 室内全景图



图 3-9 数字航空航天博物馆的全景图

2. 图像插值及视图变换技术

图像插值及视图变换技术是根据在不同观察点所拍摄的图像,以相邻的两个参考图像所决定的直线为基准,交互地给出或自动得到相邻两个图像之间对应点,采用插值和视图变换的方法求出对应于其他点的图像,生成新的视图。

图像插值及视图变化技术包括两个关键问题:一个是图像变换(Image Warping),即从已知图像的对应特征(点或线)推演出一组相应的变换函数(Warp Function),也称为传递函数(Transition Function)。在图像变形过程中,一组传递函数使源图像沿着目标图像的方向扭曲,如图 3-10 所示, WS_1 、 WS_i 、 WS_n 等都为中间图像。同时,另一组传递函数又使目标图像沿相反方向扭曲变形,如图 3-10 中的 WE_1 、 WE_i 、 WE_n 都为逆中间图像。这两列中间图像形成了两个相对的时间序列。色彩变换是另一个关键问题,与图像变化相反,它只改变像素的色彩,而不改变其坐标。色彩变换将两个图像序列中位于同一时刻的两幅变形中间图像融合成该时刻的一个中间图像,如图 3-10 所示的 I_1 、 I_i 、 I_n 中间图像分别为序列 WS_1 、 WS_i 、 WS_n 和序列 WE_1 、 WE_i 、 WE_n 融合而成的。

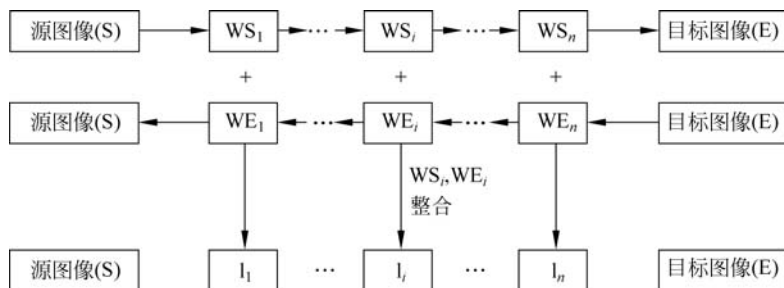


图 3-10 图像变换过程

对于图像变换方法的算法很多,最流行的方法有基于网格的图像变换算法、基于域的图像变换算法和小波变换算法。基于网格的图像变换算法是首先在源图像和目标图像中指定一组对应网格点,并利用网格点拟合样条形成一对可视的样条网格。把网格看作坐标系,则图像的变换就可以看作一组网格内一个坐标系向另一个坐标系的变换。基于域的图像变换算法是首先利用源图像和目标图像中对应的位置求得几何特征线段集,然后根据每个点的移动都会受到多条线段的影响,通过加权平均每个特征线段对该点位置的改变来计算每个点在变换时的位置。基于网格和基于域的图像变换算法有一个共同特点是非常耗时,计算时间取决于图像分辨率和特征数目。因此,提出了小波变换算法。关于小波变换算法可以查看相关的图形学书籍。

对于采用图像插值与视图变换技术进行对象建模,其步骤可分为以下几步。

(1) 采样：使用照相机或者摄像头等光捕捉设备，从不同的角度对物体进行拍摄，获得所需的照片样本。

(2) 立体匹配：获得两幅图像之间的对应关系，这是最困难的。由对应点构造从第一幅图像到第二幅图像之间的映射函数，这样第一幅图像中其余的点可以根据这个映射函数，在第二幅图像中找到各自的对应点。

(3) 插值与视图变换：利用插值与视图变换算法生成中间图像，这些中间图像感觉像是虚拟照相机所拍摄的，如图 3-11 所示。

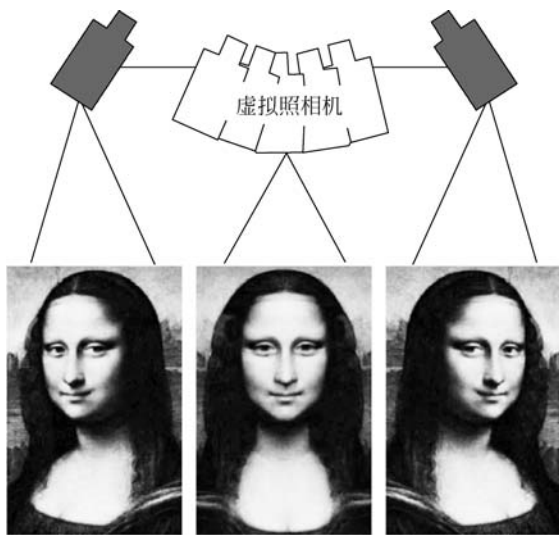


图 3-11 图像插值及视图变换的效果图

(4) 优化处理：目的是使图像边缘的表现更完美。

3.1.3 图像与几何相结合的建模

几何建模的优点是交互性好，用户可以随意更改虚拟环境的观察点和观察方向，实现实时交互，如移动或旋转虚拟物体等。其缺点是所建构的对象模型都是由多边形组成，数据量较大、难以达到较强的真实感，并且建模过程也较复杂等。图像建模的优点是虚拟环境渲染质量高，有照片质感，且绘制速度快。缺点是交互能力有限，只能虚拟浏览，用户不是参与者，更像一个旁观者。另外，众所周知，物体是由物体的形状和外观组成的。物体的形状由构成物体的多边形和顶点来确定，外观是由物体的表面纹理、颜色以及光照来确定。因此，借助两种建模技术的优势，提出了图像与几何相结合的建模技术，图像建模用于物体外观的设计，几何建模用于物体外形的制作。

根据不同视角的被建模物体的照片，通过建模软件多视图的点、线位置采样，然后分区块构建模型。这种方式建模是使图像与几何结合的建模技术尽最大限度地挖掘建模技术的潜力，把高仿真度的图像映射于简单的对象模型，在几乎不牺牲三维模型真实度的情况下，可以极大地减少模型的网格数量。图 3-12 所示为使用 3ds Max 软件构建的建筑物。其步骤为 4 步。

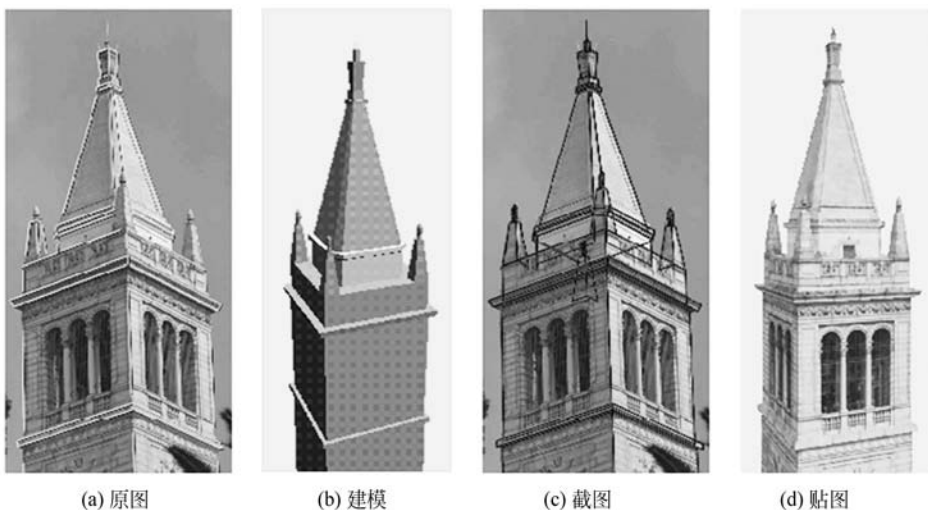


图 3-12 模型+贴图建构的建筑物效果图

1. 准备工作

基于图像与几何相结合的建筑物建模是利用照相机从不同的角度对建筑物进行拍照,通常为前、后、左、右、顶方向,然后使用建筑照片重新进行空间位置和形状上的还原,形成三维的建筑物模型。因此,建筑物各视图图片的采集或拍摄非常关键。图 3.12(a)为建筑物的前左方的照片。当然如果没有合适的图片,可以利用高精度的建筑物实物模型导入三维建模软件进行各视图的采集。

2. 利用三维空间信息创建建筑物外形

建筑物的外轮廓线的创建是建模的关键步骤。轮廓线必须与建筑物的结构有关,通常为每个相邻面之间的分界线。从多张照片中创建视图的轮廓线,如图 3.12(a)图中的边界线即为轮廓线。

3. 构造三维模型

运用 3ds Max 软件边界线造型命令,根据所画的轮廓线依次创建三维曲面,在保证建筑物外形的情况下,作最大限度的优化,利用立体视觉算法精化模型,使所有建筑物面浑然一体,以便于图像的拟合,如图 3-12(b)所示。

4. 贴图

模型表面的纹理和质地是贴图实现的,即由图像代替了几何建模,较真实地再现了物体的细节,并减少了系统的运行时间。图 3-12(c)是某角度建筑物表面图像的截图,图 3-12(d)是将图 3-12(c)截取的表面图像纹理贴图到图 3-12(b)模型后的效果图。当然在贴图时,必须采用相应的方法产生逼真的效果。例如,采用遮罩通道,让需要镂空或透明的地方产生类似效果。

该方法简单快捷,仅仅通过拍摄几张照片即可合成逼真的新视图。但是该方法较适用于普通建筑物等外形较规整的实物。图 3-13 所示为采用“模型+贴图”形式实现的室内设计,其中图 3-13(a)为模型图,图 3-13(b)为贴图后的效果图。

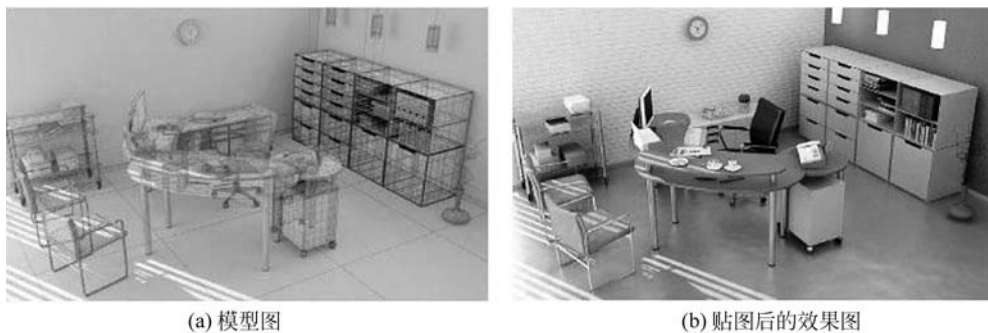


图 3-13 图像和几何相结合的室内效果图

3.1.4 视觉外观

要达到生动逼真的虚拟场景,对虚拟对象的视觉外观的修饰是必不可少的。场景光照和纹理映射可以实现场景的复杂度和真实感。

1. 光照

场景光照决定了对象表面的光强度,可分为局部光照和整体光照两类。在局部光照模型中,通过明暗处理等算法计算光照对象某一点的亮度时,仅考虑虚拟场景中所有预定义的光源对象,并孤立地处理对象和光源之间的交互,忽略对象之间的相互依赖关系。例如,Unity 中的点光源、聚光灯等都属于局部光源,如图 3-14 和图 3-15 所示。

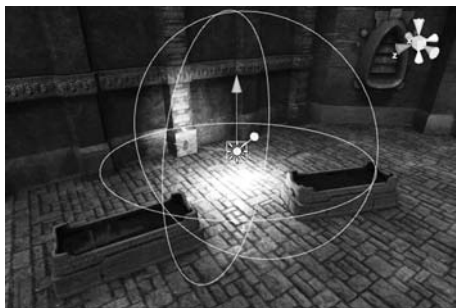


图 3-14 点光源

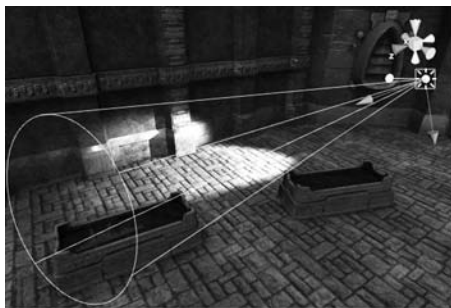


图 3-15 聚光灯

整体光照模型将整个环境作为光源,不仅考虑场景中的光源对被绘制对象的直接影响,还考虑光线经反射、折射或散射后对对象产生的间接影响。通常采用光线跟踪算法和辐射度算法从全局完成光照绘制,使绘制结果真实感大大增强。因为其计算量较大,所以整体光照是目前光照计算研究的重点。经典的整体光照现象包括颜色渗透、阴影/柔和阴影、焦散/光谱焦散、次表面散射等,对于其中任何一种现象的模拟再现都可以显著提高绘制效果。

在 Unity 中,由于实时照明对硬件要求较高,为提高场景加载的速度,通常采用预先计算好的烘焙光照实现整体光照绘制。即根据光的影响,将场景内的静态对象计算出多张贴图,并附在场景对象上建立照明效果。这些贴图可以包含场景内投射到物体表面的直接光源,也包括不同物体间反射的间接光源。

在三维建模开发平台中也集成了多种光照算法,可以进行光照效果的设计。图 3-16 所示为 3ds Max 制作的房间模型,图 3-16(a)为没有添加光照的效果图,图 3-16(b)为 3ds Max 中添加了光照(IES 阳光命令、泛光灯命令)后的效果图。

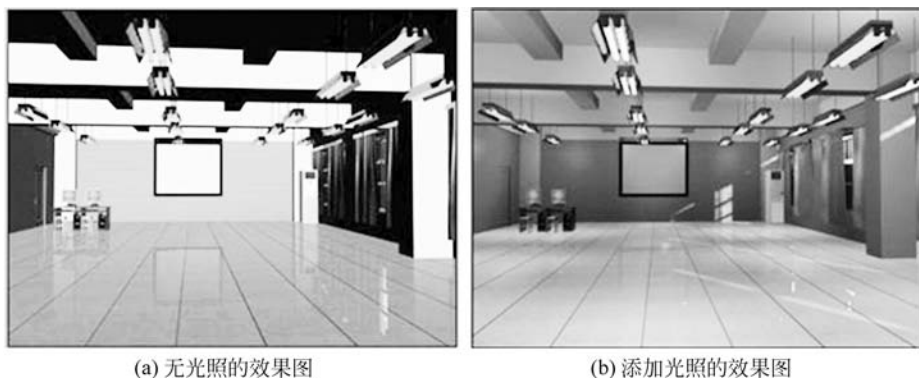


图 3-16 添加光照前后效果对比

2. 纹理映射

纹理映射是在不增加表面多边形数目的情况下提高图像真实感的一种有效方法。它是一种为了显示表面几何无法表示的细节特征,而逐渐改变表面属性的方法。其目的是更改对象模型的表面属性,例如颜色、漫反射和像素法向量等。图 3-17 为纹理映射的效果图,茶壶映射不同的图像,会得到不同的效果。

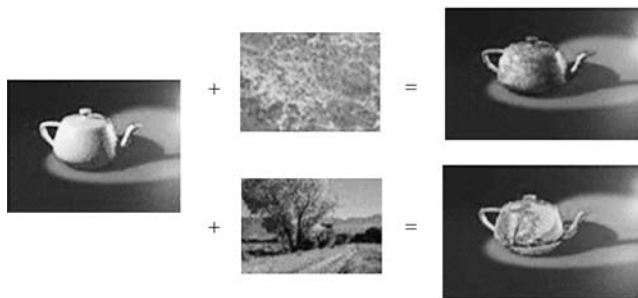


图 3-17 纹理映射效果图

纹理映射原理是使用一个对应函数把对象(屏幕像素)的参数坐标映射成纹理空间中的坐标,纹理坐标是坐标数组的索引。纹理数组的大小取决于操作系统的要求。当光栅化程序检索到对应的纹理像素的颜色后,用它来改变明暗模型中的像素颜色,这个过程称为调制,用纹理颜色乘以几何处理引擎输出的表面颜色,如图 3-18 所示。

目前,在三维建模软件中,存在多种纹理映射。将具有颜色信息的图像附着在几何对象的表面,在不增加对象几何细节的情况下,提高绘制效果的真实感的纹理映射称为传统的纹理映射。随着绘制技术的不断发展,复杂的纹理映射,例如浮雕映射,应运而生。纹理映射方法越来越多,给虚拟现实仿真带来了许多好处。首先,增强了场景的细节等级和真实度。其次,基于透视变换提供了较好的三维空间线索。最后,纹理的使用极大地减少了场景中多边形的数量,可以提高帧刷新率。

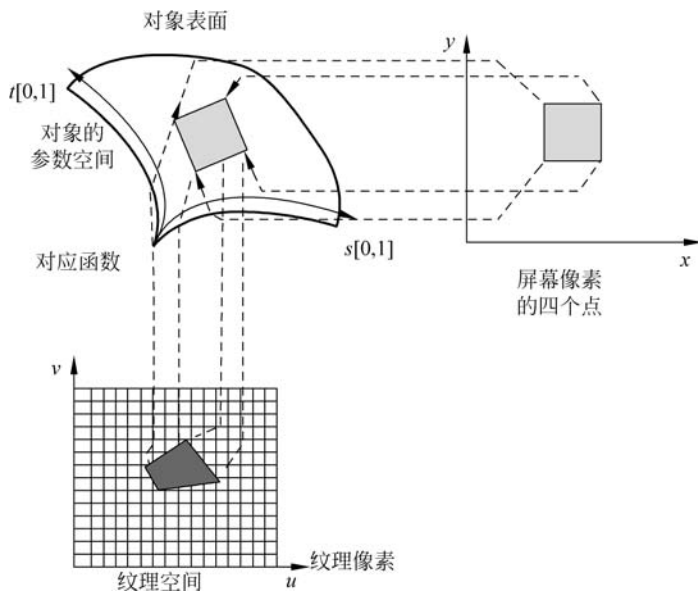


图 3-18 纹理映射原理图

3.1.5 常用的对象建模工具

虚拟场景建模是虚拟现实技术的核心,确定了场景的逼真度。目前,用于对象建模的软件有很多。本节简单介绍几款广泛应用的 3D 建模软件,分别是 3ds Max、Maya、Blender、XSI。它们都可以用来对虚拟环境进行建模,每个软件都有各自的特点及应用范围。

1. 3ds Max

3ds Max 是由 Autodesk 公司的 Discreet 子公司推出的三维动画制作软件。3ds Max 系列产品有着悠久的历史,在 DOS 时代 3D Studio 就拥有庞大的用户群体,1996 年发布的 Windows 平台下的 3D Studio Max1.0,该软件在 3D Studio 的基础上有了质的飞跃,成为集建模、渲染和动画为一体的、突破性的三维动画制作软件。伴随着计算机相关产业发展,3ds Max 相继推出了很多版本,自 2007 年以后,几乎年年有升级版本推出。目前 3ds Max 已成为世界知名的一款集三维建模、材质调制、灯光设置、摄像机布局、光影粒子特效、动画表现以及渲染于一体的 3D 建模工具软件,具有以下特点:

(1) 简单易用,兼容性好。3ds Max 具有人性化的友好工作界面,建模制作流程简洁高效,易学易用。并可以发布成各种格式的文件,满足多种不同虚拟现实引擎的软件的需要,大大方便了 VR 场景的构造。

(2) 功能强大,扩展性好。3ds Max 提供了多边形建模、放样、片面建模、NURBS 建模等多种建模工具,建模方法和方式快捷、高效。并具有非常好的开放性和扩展性,拥有最多的第三方软件开发商,具有成百上千种插件,极大地扩展了该软件的功能。

目前,3ds Max 在国内外拥有众多的用户,在使用率上占据绝对的优势。随着 VR 技术的发展,以及仿真技术在科学与工程上的应用,快速实时、交互性强、操作方便的 3ds Max 软件具有广阔的发展前景。

2. Maya

Maya 是美国 Autodesk 公司出品的世界顶级的三维动画软件,以建模功能强大著称。Maya 的操作界面及流程与 3ds Max 比较类似,是目前世界上最为优秀的三维动画制作软件之一,最早是美国的 Alias Wavefront 公司于 1998 年开发的三维动画制作软件。虽然在此之前已经出现了很多三维制作软件,但 Maya 凭借其强大的功能、友好的用户界面和丰富的视觉效果,一举引起了动画和影视界的广泛关注,成为顶级的三维动画制作软件。

Maya 不仅包括一般三维建模和视觉效果制作的功能,而且还与最先进的建模、数字化布料模拟、毛发渲染、运动匹配技术相结合,极大地提高了制作效率和品质,而且工作方式灵活,渲染效果真实感极强,是众多制作者梦寐以求的视频特效制作软件。自诞生之日起,就参与了多部国际大片的制作,从早期的《玩具总动员》《精灵鼠小弟》《金刚》到《汽车总动员》等众多知名影视作品的动画和特性,都是由 Maya 参与制作完成的。

Maya 软件的发展速度惊人,庞大的开发群体使其每年更新一个版本,并已经广泛应用于电视广告、计算机游戏制作、角色动画和电影特技等方面。

3. Blender

Blender 是由 NeoGeo 公司于 1995 年创作的 3D 创造套件,是一款开源的跨平台全能动画制作软件。Blender 集建模、雕刻、绑定、粒子、动力学、动画、交互、材质、渲染、音频处理、视频剪辑以及运动跟踪、后期合成等的一系列动画短片制作解决方案,并以 Python 为内建脚本,拥有方便在不同工作条件下使用的多种用户界面,内置绿屏抠像、摄像机反向追踪、遮罩处理、后期节点合成等高级影视解决方法。同时还内置有卡通描边和基于 GPU 技术的 Cycles 渲染器,同时支持多种第三方渲染器,商业创作永久免费。

Blender 也提供了跨平台支持,它基于 OpenGL 的图形界面可以工作在所有主流的 Windows、Linux、OS X 等众多操作系统上。Blender 小巧的体积、便利的发布和高质量的 3D 架构带来了快速高效的创作流程,也吸引了大量的充实粉丝,每次版本发布都会在全球超过 20 万的下载次数。

4. XSI

XSI 原名 SoftImage 3D,是 SoftImage 公司推出的三维动画软件,以其杰出的动作控制技术,成为众多专业动画师强大的三维动画制作工具。它的功能完全涵盖整个动画制作过程,包括交互独立建模和动画制作工具、SDK 和游戏开发工具、具有业界领先水平的 Mental Ray 生成工具等。

为了体现软件的兼容性和交互性,最终以 SoftImage 公司在全球知名的数据交换格式 XSI 命名。XSI 以其先进的工作流程、无缝的动画制作,以及领先业内的非线性动画编辑系统,出现在世人的面前。动画合成器功能更是可以将任何动作进行混合,以达到自然过渡的效果。灯光、材料和渲染已经达到了一个较高的境界,系统提供的十几种光斑特效可以延伸出千万种变化。

XSI 的自由建模能力很强,拥有世界上最快速的细分优化建模功能,以及直觉的创造工具,让 3D 建模感觉就像在做真实的模型雕塑一般。而且,它的渲染和动画功能也非常好。

3.2 物理建模

物理建模是对虚拟环境中的物体的质量、惯性、表面纹理(光滑或粗糙)、硬度、变形模式(弹性或塑性)等物体属性特征的建模,其主要作用是使得虚拟世界中的物体具备和现实世界类似的物理特效。典型的物理建模方法有用于场景搭建的分形技术和粒子系统,以及物体受力作用碰撞响应的物理规律的模拟。

3.2.1 分形技术

众所周知,自然界中的树,当忽略树叶个体区别时,从一定距离观察中间树枝,每一根树枝也像一棵大树。这种现象就是自相似性,是客观世界中许多实物具备的特性。分形就是事物整体与局部之间相似的结构,具有自相似性和迭代性。分形技术是利用实物整体与局部之间的相似结构,采用迭代方法实现复杂不规则外形物体的建模,如图 3-19 所示。分形技术绘制的弯弯曲曲的海岸线、起伏不平的山脉、粗糙不堪的断面、变幻无常的浮云、九曲回肠的河流、纵横交错的血管、令人眼花缭乱的满天繁星、粒子运动的轨迹以及树冠和花朵等,都栩栩如生、惟妙惟肖。图 3-20 为 Mandelbrot 在 1980 年发现的整个宇宙以一种出人意料的方式构成自相似的结构图。图 3-20(a)为原始图 Mandelbrot 集合,图 3-20(b)是将图 3-20(a)中的矩形框区域放大后的图形。

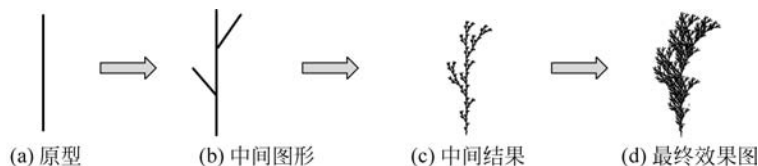


图 3-19 分形技术产生的图形

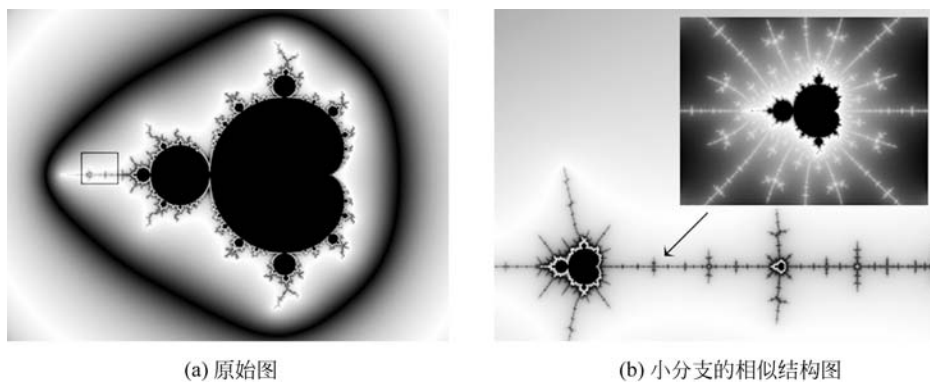


图 3-20 Mandelbrot 集合

分形技术通常用于对复杂的不规则外形物体的建模,建模过程分为:

(1) H 分形,它是简单二叉树的推广,对物体进行分形,寻找树的树梢。

(2) 迭代函数系统(Iterated Function Systems, IFS),是分形绘制的一种重要方法,基本思想是选定若干仿射变换,将整体形态变换到局部,这一过程可一直持续下去,直到得到

满意的结果。也就是说,对第一步得到的树梢,选用迭代算法,绘制完成完整的一棵树。

分形技术使用数学原理实现艺术创造,使人们觉悟到科学与艺术的融合、数学与艺术审美上的统一,搭起了科学与艺术的桥梁。它的出现不仅影响了数学、理化、生物、大气、海洋以至社会学等学科,并在音乐、美术中也产生了很大的影响。目前,分形技术在各个行业都有所使用,例如印染业、纺织业、装饰以及艺术创作等。图 3-21 为地质结构。图 3-22 为山水画。

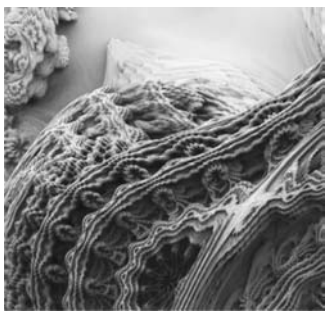


图 3-21 地质结构



图 3-22 山水画

分形技术的优点是用简单的迭代算法就可以完成复杂的不规则物体建模。缺点是迭代运算量过大,不利于实时显示,在虚拟现实系统中一般仅用于静态远景建模。

3.2.2 粒子系统

粒子系统是一种典型的物理建模系统,主要用来解决由大量按一定规则运动(变化)的微小物质(粒子)组成的物质在计算机上的生成与显示的问题。它用来模拟一些特定的模糊现象,例如火、爆炸、烟、水流、火花、落叶、云、雾、雪、灰尘、流星尾迹或者像发光轨迹这样的抽象视觉效果等。这些现象用其他传统的渲染技术难以实现其真实感。

粒子系统是一个动态系统,可以生长和消亡。也就是说,每个粒子除了具有位置、速度、颜色、加速度等属性外,还有生命周期属性,即每个粒子都有着自己的生命值,随着时间的推移,粒子的生命值不断减小,直到粒子死亡(生命值为 0)。一个生命周期结束时,另一个生命周期随即开始。除此之外,为了增加物理现象的真实性,粒子系统通过空间扭曲控制粒子的行为,对粒子流造成引力、阻挡、风力等影响。

典型的粒子系统循环更新的基本步骤为以下四步:

- (1) 加入新的粒子到系统中,并赋予每一新粒子一定的属性;
- (2) 删除那些超过其生命周期的例子;
- (3) 根据粒子的动态属性对粒子添加外力作用,如重力(Gravity)、风力(Wind)等空间扭曲,实现对粒子进行随机移动和变换;
- (4) 绘制并显示所有生命周期内的粒子组成的图形。

通常粒子系统在三维空间中的位置与运动是由发射器控制的。发射器主要由一组粒子行为参数以及在三维空间中的位置所表示。粒子行为参数包括粒子生成速度(单位时间粒子生成的数目)、粒子初始速度向量(什么时候向什么方向运动)、粒子寿命(经过多长时间粒子湮灭)、粒子颜色、在粒子生命周期中的变化以及其他参数等,值的确定是在遵循自然规律

的基础上,确定其变化范围,然后在该范围内随机地确定它的值。

为了得到较为逼真的景象,需对粒子进行纹理贴图。例如雪的基本形状都是六角形,但由于表面曲率不等(有凹面、平面、凸面),各面上的饱和水汽压力也不同,因此产生了千变万化的六角形雪片。所以,制作时,雪粒子采用圆形粒子,然后进行纹理贴图,制作出不同形状的多种雪花图样,如图 3-23 所示。



图 3-23 纹理贴图形成的雪花造型

目前,粒子系统在许多三维建模软件及渲染包就可以直接创建,例如 3ds Max、Maya 等。这些编辑程序能够立即显示特定的特性或者规则下的粒子系统。图 3-24 为使用 3ds Max 制作的深水中气泡,采用了 SuperSpray 粒子系统,水泡粒子的“模样”是通过纹理贴图实现的。图 3-25 是 DirectX3D 软件制作的粒子系统——雪。图 3-26 为粒子系统制作的火焰。图 3-27 为粒子系统制作的星系。



图 3-24 深水中气泡



图 3-25 雪

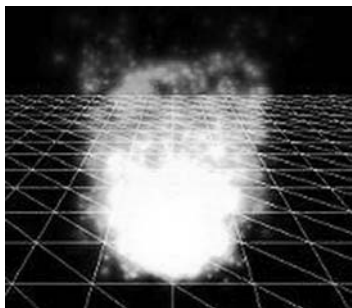


图 3-26 火焰

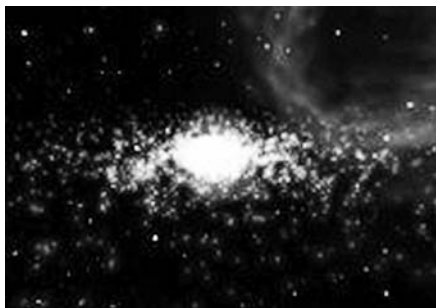


图 3-27 星系

3.2.3 碰撞响应

真实世界中的物体在运动过程中很有可能与周围环境发生碰撞、接触及其他形式的相

互作用。为了使虚拟世界“看起来更真”，虚拟物体之间必须能够实时的、无缝的、可靠的检测相互碰撞并做出恰当的响应。而碰撞响应即是对物体间的相互作用的检测与响应，主要包括两部分：碰撞检测与碰撞响应。碰撞检测是研究物体能否发生碰撞，以及发生碰撞的时间与位置。碰撞响应是研究物体之间发生碰撞后，物体发生的形变或运动变化，并以符合真实世界中物体的动态效果实时显示。

1. 碰撞检测

碰撞检测是检测两个(或多个)物体是否互相接触。为了保证虚拟世界的真实性，碰撞检测需具有较高的实时性和精确性。而通常情况下，实时性和精确性是相互矛盾的。所以，在满足场景碰撞检测的精确性的基础上，如何有效和合理地降低碰撞检测的时间复杂性，提高碰撞检测速度，是虚拟环境中碰撞检测的一个重要研究方向。目前，用于碰撞检测的方法有很多，如直接检测法、包围盒检测法、空间分割法等。应用广泛的方法为包围盒检测法。

包围盒检测法是使用比被检测物体体积略大、几何特性简单、包围被检测三维物体的三维包围盒来进行检测的。通过对包围盒的检测来粗略确定是否发生碰撞，当两个物体的包围盒相交时其物体才有可能相交；若包围盒不相交其物体一定不相交。利用包围盒法可以排除大量不可能相交的物体和物体的局部，从而快速找到相交的部位。包围盒是包围给定三维对象所有顶点的棱柱或球，如图 3-28 所示，圆柱体为被检测物体，外部的框体为包围盒。根据形状的不同，包围盒分为沿坐标轴的包围盒(Axis-Aligned Bounding Boxes, AABB)，如图 3-28(a)所示；球形包围盒，如图 3-28(b)所示；任意方向的包围盒(Oriented Bounding Box, OBB)，如图 3-28(c)所示；及上述方法的扩展或变异等。

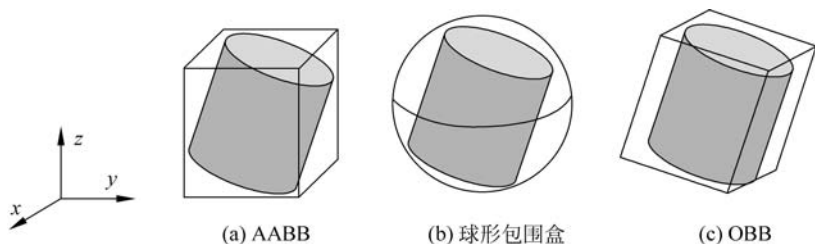


图 3-28 包围盒的类型

1) 沿坐标轴的包围盒 AABB

是沿着世界坐标轴方向的棱柱，包含几何对象且各边平行于坐标轴的最小六面体[图 3-28(a)]。构造时根据物体的形状和状态取得坐标 x, y, z 方向上的最大最小值就能确定包围盒最高和最低的边界点。

AABB 包围盒的边界总是与坐标轴平行，它的平面与其相应的坐标平面相平行。一个 AABB 包围盒通常可以用其向 3 个坐标轴的投影的最大最小值来表示，也可以用物体中心和 3 个方向上的跨度来表示。前一种表示方法在两包围盒进行相交测试时比第二种的运算量要少一些。检测两个 AABB 包围盒是否相交最简单的方法是利用投影，如图 3-29 所示，比较两物体在某轴上投影的最大值和最小值。两个 AABB 包围盒相交当且仅当它们 3 个坐标轴上的投影均重叠。只要存在一个方向上的投影不重叠，那么它们就不相交。

AABB 包围盒具有建构简单快速、相交测试简单、内存开销少的特点，能较好地适应可变形物体实时更新层次树的需要，可用于进行可变形物体之间的相交检测。AABB 的缺点

是包围物体不够紧密,在一些情况将出现较大的空隙,如图 3-30 所示,会增加许多不必要的检测,反而影响算法效率。

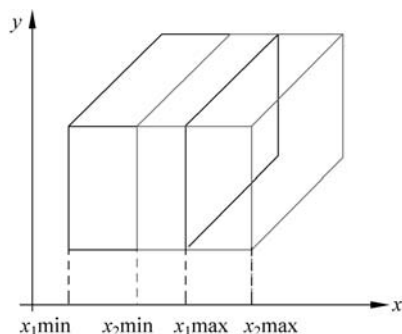


图 3-29 AAB 的投影方法



图 3-30 包围不够紧密的 AAB 包围盒

2) 球形的包围盒

利用检测物体的中心为球心,以物体边界点到中心最大的距离为半径所组成的球体[图 3-28(b)].构造时仅需 2 个标量,即球心和半径。计算给定对象 SE 的 Sphere 包围球,首先需通过 SE 中各个元素顶点的 x 坐标、 y 坐标和 z 坐标的值确定包围球的球心 c ,再由球心与 3 个最大值坐标所确定的点间的距离计算半径 r 。

使用球形包围盒进行相交检测相对比较简单。对于两个包围球 (c_1, r_1) 和 (c_2, r_2) ,如果球心距离小于半径之和,如图 3-31 所示,即 $|c_1 - c_2| < r_1 + r_2$,则两包围球相交,可进一步简化为判断 $(c_1 - c_2) \cdot (c_1 - c_2) \leq (r_1 + r_2)^2$ 。当对象发生旋转运动时,包围球不需要做任何更新,这是包围球比较友好的一个特性,当几何对象进行频繁的旋转运动时,采用包围球可取得较好的结果。当对象发生变形时,很难从子节点的包围球合成父节点的包围球,只能重新计算。

此方法非常适用于需要快速检测、不需要精确碰撞检测的应用中。执行速度相对较快,不会给 CPU 带来过大的计算负担。但球体碰撞的一个缺点是只适用于近似球形物体,如图 3-32 所示,会有很大面积的空隙,影响检测精度。

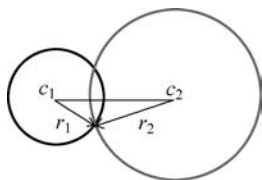


图 3-31 球形包围盒



图 3-32 包围不紧密的球形包围盒

3) 任意方向的包围盒 OBB

OBB 定义为包含该对象且相对于坐标轴方向任意的最小长方体。它是根据物体本身的几何形状来决定盒子的大小和方向,盒子无须和世界坐标轴垂直,而是一个沿着物体主轴方向的最紧凑最适合物体的六面盒子[图 3-28(c)].与 AAB 相比,OBB 的最大特点就是方向的任意性,这使得它可以根据对象的几何特点尽可能紧密地包围对象,但同时也使得它的相交测试变得复杂。

OBB 的计算相对复杂,关键在于寻找最佳方向,并确定在该方向上包围盒的最小尺寸。OBB 间的相交测试基于分离轴理论,主要是确定包围盒是否有重叠,并不要求确定具体的接触位置和接触深度。如果空间中存在一个向量,使两个 OBB 在该向量上的投影不重叠(相交),则这个向量即为一根分离轴(不一定是坐标轴)。如果两个 OBB 在一条轴线上的投影不重叠,则可以判定这两个 OBB 不相交,其原理如图 3-33 所示。

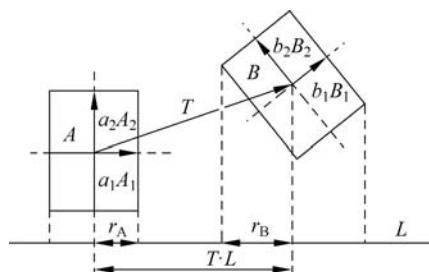


图 3-33 基于分离轴理论的
OBB 包围盒相交测试

图 3-33 中, A 、 B 是两个 OBB 包围盒, 向量 T 是 A 的中心到 B 的中心的距离。 L 是 A 和 B 的一个分离轴, 因为 A 和 B 在 L 上的投影是不相交的。两个投影区间的中心距离是 $|T \cdot L|$ 。 a_i 和 b_i 表示 A 和 B 的各轴向的半径, A_i 和 B_i 表示 A 和 B 的轴向, 其中 $i=1, 2, 3$ 。

r_A 、 r_B 分别是 A 、 B 在 L 上的投影区间半径, $|T \cdot L| > r_A + r_B$ 即为投影区间不重叠的充分必要条件。

对任何两个不相交的凸三维多面体(长方体), 其分离轴要么与任一多面体的某一个面正交, 要么同时垂直于两个多面体的某一条边。因此, 对于一对 OBB, 只需测试 15 条可能是分离轴的轴(两个 OBB 的 3 个面方向的轴和由 3 个边方向的轴与另 3 个边方向的轴两两组合的 9 个轴), 只要能找到一个这样的分离轴, 就可以判定两个 OBB 是不相交的; 若 15 条轴都不能将两个 OBB 分离, 则它们是相交的。

尽管 OBB 间相交测试的计算量比较大, 但它的紧密性最好, 可以成倍地减少参与相交测试的包围盒的数目和基本几何元素的数目, 在大多情况下其总体性能要优于 AABB 和包围球。当几何对象发生旋转运动后, 只要对 OBB 的基底进行同样的旋转即可。因此, 对于刚体间的碰撞检测, OBB 是一种较好的选择。但重新计算每个节点的 OBB 的代价太大, 因此 OBB 不适用于软体对象环境中的碰撞检测。

4) 离散方向多面体

离散方向多面体(k -Discrete Orientation Polytope, k -DOPs)是在分析以往采用的层次包围盒的缺点后提出的。一个物体的 k -DOPs 定义为包含该对象, 且它的所有面的法向量均来自一个固定的方向(k 个向量)集合的凸包。其中的方向向量为共线且方向相反的向量对, 术语上称为 FDH(Fixed Direction Hull)。如图 3-34 所示, 同一物体的 3 种包围盒对比, (a)为 AABB, (b)为 OBB, (c)为 6-DOPs。 k 为可选的固定方向的向量, 每个面一个。其最简单的是固定方向集中包含坐标轴方向, 即 $k=6$ 时, 便成为了 AABB。因此, k -DOPs 也可看作是 AABB 的扩展。另外, 当 k 的值无限大时, 它就成为被测对象的凸包。因此, 它不但继承了凸包紧密性好的优点, 同时也继承了 AABB 简单性好的优点。

2. 碰撞响应

物体碰撞以后需要做出一些反应, 比如产生反冲力反弹出去, 或者停下来, 或者让阻挡物体飞出去等, 这都属于碰撞响应。碰撞响应是当检测到虚拟环境中发生碰撞时, 修改发生碰撞的物体的运动表示, 即修改物体的运动方程, 确定物体的损坏和变形, 实现碰撞对物体运动的影响。

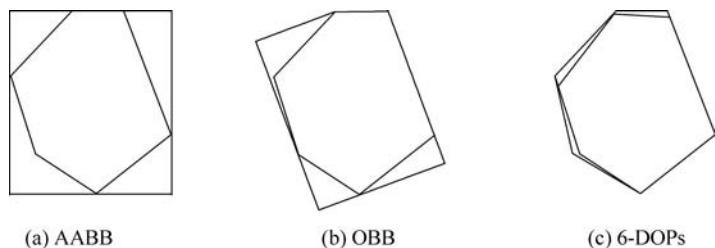


图 3-34 同一物体的 3 种包围盒对比

碰撞响应是由发生碰撞的虚拟对象的自身特性以及具体应用要求决定的。如果发生碰撞的对象是弹性物体,物体弹性形变后反弹出去,物体恢复原来的几何形状。如果是塑性物体,物体发生表面变形后不反弹。如果是刚性物体,物体被强有力的反弹回去。弹性物体、塑性物体与刚性物体的区别:刚性物体的运动仅仅是位置、方向和大小的改变,而弹性和塑性物体则额外增加了变形属性。因此,碰撞响应通常分两种情况:表面变形和力的反弹。不同的碰撞响应采用不同数学模型和物理属性计算实现,这也是当前研究的热点之一。

3.3 运动建模

“动起来真实”是虚拟现实逼真度的一个要求,需要考虑虚拟场景中对象的具体位置、平移、旋转、缩放等变化效果。这些内容统称为对象的运动属性,是通过对象的运动建模实现的。运动建模主要用于确定三维对象在世界坐标系中的位置,以及它们在虚拟世界中的运动。其目的是对运动物体实现服从客观世界的运动规律的建模,其核心内容包括虚拟摄像机、对象坐标和对象层次。

3.3.1 虚拟摄像机

三维世界通常采用摄像机的坐标系来观察。摄像机坐标系在固定的世界坐标系中的位置和方向称为观察变换,即在观察虚拟对象时通过摄像机窗口来观察。在图形实时绘制时,需要根据摄像机的坐标实时绘制对象,仅需绘制摄像机看到的部分场景内容,这部分场景用一个称为视口的名词定义,如图 3-35 所示。眼睛的图标代表摄像机的位置,指向右侧的标记为 x 轴的向量代表摄像机的指向。中间的方块为 3D 场景中的物体。左侧的矩形和右侧的矩形分别代表近裁剪面(近切面)和远裁剪面(远切面),这两个平面决定了 3D 空间的子集的边界,称为视锥体或视平锥体。视口的内容是最终在屏幕上看到的渲染的 2D 图像。视锥体内 x 值较大的对象,离摄像机较远,如果被 x 值较小的对象遮挡住,则不需被绘制出来。只有处于视锥体内部的物体才可以被渲染到屏幕上,其他的对象则被裁剪掉。

在虚拟现实引擎中,摄像机相对场景有位置和方向,提供了透视、正交两种渲染场景的方式,如图 3-36 所示。透视方式可以模拟人眼观察世界的方式,即近大远小的效果。如图 3-36 的左图所示,视口中的两物体有大小区分,即透视方式有近大远小的效果。正交方式是取消了透视效果,直接将三维的场景渲染成一幅二维的图片投射到相机视口。如图 3-36 的右图所示,视口中的两物体无大小区分。除此之外,视锥体外的物体被裁掉。

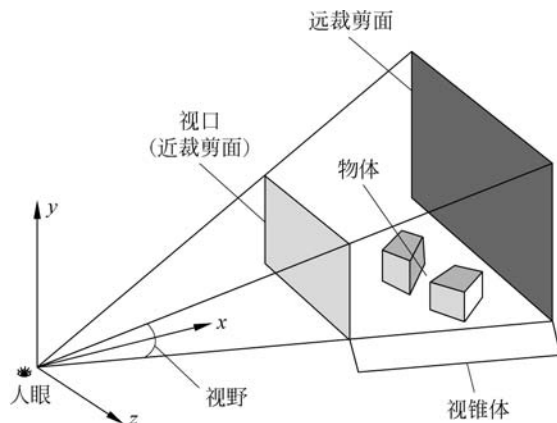


图 3-35 虚拟摄像机

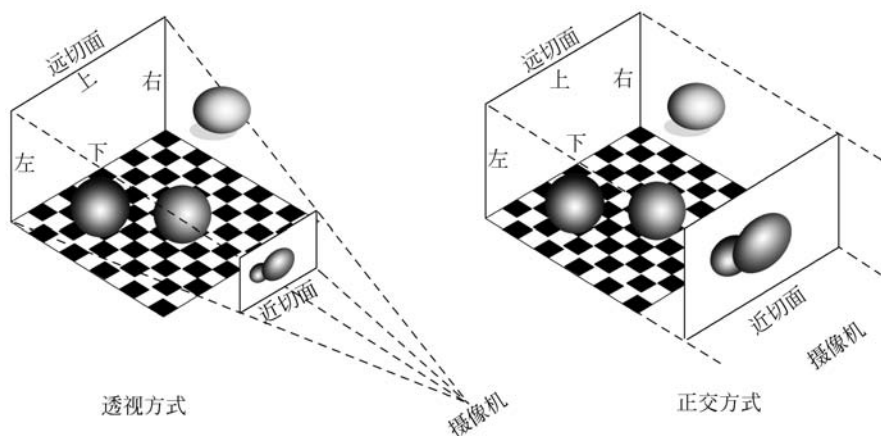


图 3-36 摄像机的两种渲染场景方式

3.3.2 对象位置

在虚拟现实的运动建模中,对象位置通常采用坐标系来表示。对象位置的变化通常是由平移、旋转、比例缩放等几何变换所引起的。在场景创建时,对象的平移、旋转和缩放通常采用齐次变换矩阵来描述。坐标系采用绝对坐标系,即世界坐标系,起着定位每一个物体的作用。而在对象表面建模中,顶点坐标使用的是对象坐标系中的 (x, y, z) ,即每个物体对象定义的坐标系。这个坐标系捆绑在对象身上,通常位于重心处,其方向沿对象的对称轴方向。当对象在虚拟世界中移动时,其对象坐标系位置随着物体一起移动。因此,无论对象在场景中的位置如何变化,在对象坐标系中,对象顶点坐标的位置和方向一直保持不变。只要对象表面不发生变形或切分,就一直保持不变。图 3-37 显示的为一个虚拟长方体的坐标变换。坐标系 (i_1, j_1, k_1) 是虚拟对象的对象坐标系,坐标系 (i_w, j_w, k_w) 是世界坐标系。 P_1 是对象坐标系和世界坐标系进行转换的位置向量。图 3-37(a)为静止状态, P_1 是不变的;图 3-37(b)为运动状态, P_1 成为一个关于时间 t 的函数。

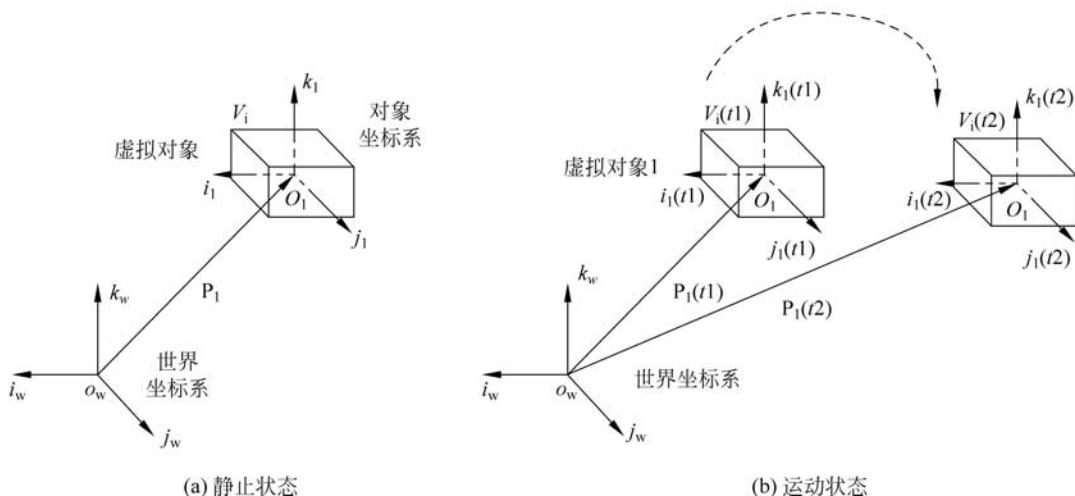


图 3-37 虚拟对象在世界坐标系中的位置

3.3.3 对象层次

对象层次定义了一个作为一个整体一起运动的一组对象,各部分也可以独立运行。假设不考虑对象层次,对象模型是一个整体,运动时只能做整体运动,例如虚拟手,如果没有做层次划分,意味着手指不能单独运动。为了实现手指的运动,必须对手的三维模型进行分段设计,这种分段是虚拟世界中对象层次的基础。

在对象层次中,上一级对象称为父对象,下一级对象称为子对象。根据人身体运动的生理机制,父对象的运动会被所有的子对象复制,而子对象的运动却不会影响父对象的位置。即父对象运动,则子对象会跟随父对象的运动而运动,而子对象可以独立运动,不影响父对象。因此,遵循自然规律,对象的层次采用树图来表示,每个节点的描述采用齐次变换矩阵。树的节点表示对象的分段,分支表示关系。对于虚拟手来说,它的层次结构

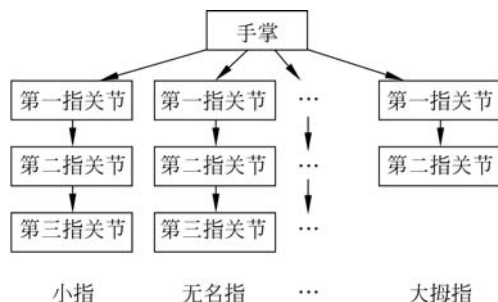


图 3-38 虚拟手的层次结构

构为一个手掌父节点和 5 个手指子节点,如图 3-38 所示。手掌是手指的父对象,当手掌运动时,所有子节点也随之运动。相反,子节点运动,不会影响父对象。在实际应用中,使用来自传感手套的数据改变手指分段的位置,可以模拟出虚拟手的动作,这是通过改变手的树图结构中的各个节点之间的变化矩阵实现的。此原理常用于骨骼动画中,或者运动捕捉等领域。

在具体的实现中,分为两种形式,一种称为正向运动学,又称为前向运动学,由父骨骼的位置、运动(平移、旋转等操作)以及变换带动子骨骼的位置、运动以及变换等操作。而子骨骼的运动不会影响父骨骼的变化。处于节点链末端的节点位置是由此链条上的各个旋转角和相对位移来决定的。正向运动学的优势是计算简单、运算速度快;缺点是需指定每个关

节的角度和位置,很容易产生不自然不协调的动作。

另一种为逆向运动学,又称反向运动学,是以子骨骼的位置和变换逆向推出父骨骼的位置和变换。在物理世界中,偶尔会有反向运动,由子骨骼的运动带动父骨骼的改变。例如,在做引体向上运动时,胳膊的运动带动整个身体的运动。通常,用于游戏人物与周围环境的交互,计算骨骼链中每个骨骼的角度,使得末端骨骼可以达到一个特定位置,从终端节点开始计算,然后逐层往上计算其他祖先节点的信息。反向运动学方法在一定程度上减轻了正向运动学方法的复杂计算,是生成逼真关节运动的最好方法之一。

3.4 声音建模

Jaunt 公司 CEO Jens Christensen 表示,“音乐在 VR 体验的沉浸感和真实感中扮演着非常重要的角色,目前我们认为,它至少占据了整个 VR 体验中的 50% 的分量”。由此可知,在虚拟现实里,声音是非常重要的部分,声音能够直接影响虚拟现实的沉浸感。众所周知,现实生活中的声音是三维立体的,来自四面八方,人们能够利用耳朵的特殊性判断出声音的距离和方位。所以,在虚拟世界中,观者处于场景的中心,不仅在自由地选择观看的方向和角度时能听到来自各个方向的声音,而且当观者需要来回转动头部或者大幅度身体运动时,声音也要实时跟随运动发生变化,这样才能实现虚拟环境的沉浸感。为此,声音建模主要是对虚拟场景中三维立体声音的定位和跟踪,让置身于虚拟世界的人,能实时识别声音的类型和强度,判断出声源的位置。声音建模技术的关键在于声音的录制、合成和重放技术。

3.4.1 声音的录制

人工头录音(Dummy Head Recording)是一种双路立体声录音(Binaural recording)的录音方式,通过把两个微型全指向性话筒安置在一个仿真人头的耳道内,模拟人耳听到声音的整个过程,如图 3-39 所示。这样两个话筒录制到的信号,就相当于一个在仿真人头所在位置的真人双耳所听到的声音。

另一种声音录制是声场录制机现场录立体声,包括多轨收音器和主机,同时录制多方位的声音,类似于同期声。图 3-40 为时代拓灵的 Twirling720 便携式 VR 声场录制机,可以一键录入全场声场,完成三维环境中所有方向的声音录制,颠覆了传统的沉浸式声音制作。360°声场录制机的声场进行后期的算法处理,不仅可以支持传统立体声耳机和 5.1/7.1 播放系统,还可以灵活地配合虚拟设备,实现沉浸式交互。



图 3-39 人工头仿生耳



图 3-40 Twirling720 便携式 VR 声场录制机

3.4.2 声音的合成

声音的合成称为双耳信号合成(Binaural Synthesis),是把多轨音素材收集起来,按照方位远近用引擎加工。每个声音任何一个时刻在三维空间里的位置是需要精确的。

一个点声源通过人的身体躯干、头部和耳廓等身体部分反射或折射后,进入人的双耳。可以将这一物理过程看作一个线性时不变的声滤波系统。这一物理过程的特性可以由其传输函数——头相关传输函数(Head-related Transfer Function, HRTF)来描述。双耳信号合成一般通过将测量的头相关传输函数与声源信号在频域相乘(或者时域卷积)得到。

3.4.3 声音的重放

声音重放是将 VR 声音打包到播放器或者虚拟现实引擎内播出。根据播放的设备分为耳机重放和扬声器重放两种,采用扬声器重放时会产生交叉串音的干扰(左扬声器的声音不仅传输到左耳朵,而且传输到右耳朵),消除交叉串音的处理比较困难;而耳机提供了完全隔开的通道,其更符合立体声的处理思路。因此,耳机重放在声音重放领域被广泛应用。

3.5 虚拟现实开发引擎

虚拟现实开发引擎是为虚拟现实系统开发提供强有力支持的一种解决方案,用于虚拟现实内容开发的交互平台。此平台包括各种交互硬件接口、图形数据的管理和绘制模块、功能设计模块、消息响应机制、网络接口等功能,如图 3-41 所示。

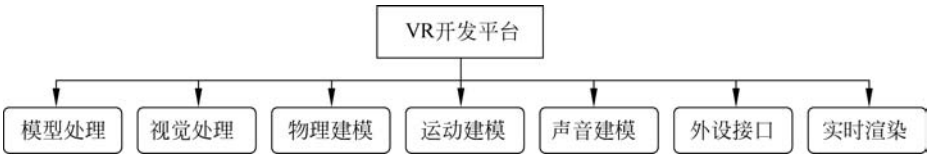


图 3-41 虚拟现实开发引擎的功能

随着虚拟现实技术的普及,虚拟现实开发引擎越来越多,例如,Unity、Unreal、EON 等,都在各自领域、各自优势方向长足地发展。如表 3-1 所示。

表 3-1 常见 VR 开发引擎的比较

VR 开发引擎	学习资源可获得性	需要浏览插件	可操作性及难易	功能及用途说明
Unity	中英文资源丰富	否	可视化操作界面,较易	集代码编写、骨骼动画、声音、光照、物理系统、地形系统、粒子系统等于一体的开发平台,具有跨平台、技术门槛低等优点
Unreal Engine	中英文资源丰富	否	可视化操作界面,较易	集代码编写、骨骼动画、声音、光照、物理系统、地形系统、粒子系统等于一体的开发平台,具有出色的画面效果

续表

VR 开发引擎	学习资源可获得性	需要浏览插件	可操作性及难易	功能及用途说明
EON	中文资料相对较少,外文资源丰富	是	可视化操作界面,较易	EON 是世界上公认整合性、延展性、可操作性最好的虚拟现实开发展示系统之一
VRML	中文资料较多,外文资源丰富	是	要求有一定的编程基础,掌握 VRML 语法,较难	早期桌面虚拟现实技术的代表,可以实现交互功能,已经完成了到 X3D 的转换
Cult3D	中文资料较少,外文资源丰富	是	可视化操作界面,较易	生成文件小、图像质量高,得到世界许多大公司的支持。对场景的表现不尽如人意。Cult3D 的开发环境比 Viewpoint 更具人性化和条理化,开发效率也要高得多
Viewpoint	中文资源难觅,有一定的外文资源	是	可视化操作界面,较易	具有一个纯软件的高质量实时渲染引擎,渲染效果接近真实而不需要任何的硬件加速设备
Java3D	资源较丰富	否	需要一定的编程基础,入门难	Java 3D 是建立在 Java 2 (Java 1. 24) 基础之上的,Java 语言的简单性使 Java 3D 的推广有利

总体来看,一个完善的虚拟现实开发引擎具有以下特点:

- (1) 可视化开发界面,实现“所见即所得”方式开发虚拟场景。
- (2) 支持二次开发能力,允许开发人员能够针对特定需求设计和添加功能模块,实现了虚拟现实场景的可扩展性和开放性。
- (3) 数据兼容性,支持多种媒体的处理操作,如图形、图像、文字、音视频、动画等,提高虚拟现实场景内容的丰富度。
- (4) 支持多种交互方式,丰富虚拟现实场景的交互形式,提升沉浸感。
- (5) 提供快速的数据处理能力,降低数据的传输延迟、提升渲染的速度和响应速度,进一步提升场景的流畅感。

目前,Unity 和 Unreal Engine 是当前开发领域使用最多、跨平台性最好、资源最丰富及支持 VR 外设最全的两款虚拟现实开发引擎,已经囊括了中小型以及大型游戏和虚拟场景的开发。

1. Unity

Unity 是由 Unity Technologies 开发的一个让开发者创建诸如三维视频游戏、建筑可视化、实时三维动画等类型互动内容的多平台综合性游戏开发工具,是一个全面整合的专业引擎。Unity 凭借较低的技术门槛,以及对跨平台的支持,获得了一大批拥护者。目前,更多的 VR 初创企业普遍采用 Unity 引擎。

图 3-42 为可视化的开发界面,界面比较简洁、容易使用。Unity 支持 C#,JavaScript 等

编程语言,上手快,而且文档、视频等多种类型的学习资源丰富,支持多平台开发,所以学习和开发成本都较低。Unity 不仅适合大团队制作,小团队甚至独立制作也可以完成,主要趋向移动平台的开发,尤其是手游,目前 Unity 仍然是手游市场占有率最高的引擎。在 VR 硬件支持度上,Unity 很广泛,包括 Oculus Rift、Samsung Gear VR、PlayStation VR、Microsoft HoloLens、Steam VR/HTC Vive 及 Google Daydream 等。

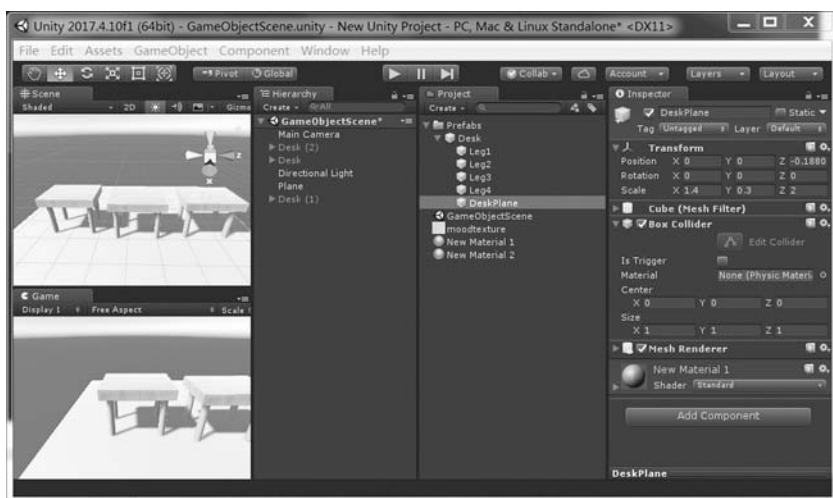


图 3-42 Unity 开发界面

2. Unreal Engine

Unreal Engine 是游戏公司 Epic Game 的杰作,诞生于 1998 年,经历了多个版本后,达到今天所看到的免费、开源的 UE5,图 3-43 为 Unreal Engine4 的开发界面。Unreal Engine 凭借顶级的图形处理能力,包括高级动态光照、新的粒子系统(可同时处理数以百万的粒子)等,具有更加出色的画面效果,为广大 3A 级游戏开发厂商所青睐。Unreal Engine 使用 C++进



图 3-43 Unreal Engine 开发界面

行开发,内含模块功能强大、操作十分复杂。所以,相比较而言,Unreal 的学习难度比较大,一般入门开发者可能需要一年半左右的时间才能上手使用 Unreal 开发。不过,从画面效果来看,Unreal 的画面渲染质量更胜一筹。通常情况,Unreal 更加适合大团队大制作,主要趋向于 PC 平台。Unreal Engine 积极跟进各种 VR 硬件,支持 Oculus Rift、Samsung Gear VR、Playstation VR、Steam VR/HTC Vive、Google Daydream 及 Leap Motion 等。

综上所述,这两个开发引擎各有千秋,从目前的发展来看,如果 VR 项目为移动平台,而且制作预算较低,对画面要求不是特别高,一般首选 Unity;反之,Unreal 会是更加适合的开发工具。