

虚拟现实软件 编程指南

The Virtual Reality Programmer's KIT

一套完整的开发虚拟
境界的工具



本书附带一张软盘

- 支持调制解调器的REND386
- 支持各种虚拟现实外设的软件
- 三个多用户虚拟境界

〔美〕 Joe Gradecki 著

王涌天 何定

王红涛 徐彤

译

电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

URL: <http://www.phei.co.cn>



WILEY



电子工业出版社

71317
G35

410424

虚拟现实软件编程指南

附赠光盘
[美] Joe Gradecki 著

战守义 校
王涌天 何 定 译
王红涛 徐 彤

电子工业出版社
Publishing House of Electronics Industry

35.06/13

AUTHORIZED TRANSLATION OF THE EDITION PUBLISHED BY JOHN WILEY & SONS NEW YORK, CHICHESTER, BRISBANE, SINGAPORE AND TORONTO. No Part of this book may be reproduced in any form without the written permission of John Wiley & Sons Inc.

Copyright ©1994 by Jee Gradecki.

Published by John Wiley & Sons, Inc.

All right reserved.

John Wiley & Sons 公司已将本书的中文版授权予电子工业出版社。未经允许,不得以任何形式和手段复制或抄袭本书的内容。

书 名:虚拟现实软件编程指南

著 者:[美]Joe Gradecki

译 者:王涌天 何定 王红涛 徐彤

审 校 者:战守义

责任编辑:秦 梅

特约编辑:辛再甫

排版制作:志达公司

印 刷 者:北京科技大学印刷厂

出版发行:电子工业出版社出版、发行 URL:<http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编 100036 发行部电话:68214070

经 销:各地新华书店经销

开 本:787×1092 1/16 印张:17.5 字数:448千字

版 次:1998年1月第一版 1998年1月第一次印刷

书 号: ISBN 7-5053-4363-7
TP·2002

定 价:30.00元

著作权合同登记号 图字:01-97-0825

凡购买电子工业出版社的图书,如有缺页、倒页、脱页者,本社发行部负责调换
版权所有·翻印必究

目 录

引言	(1)
第一章 虚拟现实中的一般性术语和表达	(3)
本书中的虚拟现实	(3)
虚拟境界的一次旅行	(4)
输入设备	(11)
输出设备	(14)
小结	(16)
第二章 创建 PLG 物体	(17)
REND386 的历史	(17)
REND386 的安装	(17)
理解三维坐标	(17)
创建物体	(18)
创建一个立方体	(28)
创建货车物体	(33)
设计高效的物体	(35)
第三章 使用 REND386	(37)
在 REND386 中载入一个物体	(37)
在物体周围漫游	(37)
其他运动控制和视场控制	(48)
屏上漫游辅助工具	(50)
REND386 菜单	(51)
命令行选项	(64)
小结	(66)
第四章 创建形体文件和境界文件	(67)
形体文件	(67)
载入及操纵形体	(77)
建立境界文件	(81)
在 REND386 中载入境界文件	(91)
虚拟境界的几个高级问题	(91)
其他选项 loadpath	(93)
小结	(94)
第五章 如何设计引人入胜的虚拟境界	(95)
创建虚拟境界很容易	(95)
创建引人入胜的虚拟境界的几条方针	(95)
小结	(99)
第六章 性能	(101)
速度	(101)
所需的性能	(102)

硬件与性能的关系	(103)
编程语言	(108)
如何提高速度	(108)
小结	(111)
第七章 REND386 内幕	(113)
物体读入和表示	(113)
产生视点	(115)
绘制	(116)
小结	(119)
第八章 编译 REND386 和 C 代码	(121)
REND386 的文件	(121)
编译	(125)
拓展 REND386 系统	(128)
利用 REND386 骨架程序编写自己的虚拟现实应用程序	(128)
小结	(148)
第九章 对虚拟现实的硬件编程	(149)
最基本的硬件支持:键盘和鼠标	(149)
对操纵杆编程	(158)
Global 3D 控制器与 Cyberman	(160)
对魔力手套编程	(166)
用于选择输入设备的命令行参数	(174)
小结	(175)
第十章 用户与虚拟境界的交互	(177)
到底什么是交互	(177)
能够进行交互的虚拟手	(177)
修改境界文件以实现更好的交互	(187)
创造一个虚拟砖块境界	(189)
小结	(192)
第十一章 动画	(193)
REND386 演示程序	(193)
定义动画任务	(193)
复杂的动画	(197)
小结	(206)
第十二章 开发共享的虚拟境界:通过串行口连接	(207)
为什么要共享虚拟境界	(207)
计算机连接	(207)
使两个虚拟境界保持同步的技巧	(213)
解决传输过程中丢失数据的问题	(219)
小结	(222)
第十三章 把调制解调器加到虚拟境界中	(223)

调制解调器的限制.....	(223)
对调制解调器编程.....	(223)
RENDMODM.C	(230)
小结.....	(231)
第十四章 抢占游戏	(233)
抢占游戏介绍.....	(233)
如何编制这个游戏.....	(234)
创建并放置旗帜.....	(237)
确定谁是获胜者.....	(243)
小结.....	(243)
第十五章 砖块游戏	(245)
砖块游戏介绍.....	(245)
如何编制这个游戏.....	(246)
小结.....	(252)
第十六章 袭击游戏	(253)
袭击游戏介绍.....	(253)
如何编制这个游戏.....	(254)
小结.....	(264)
附录 A 软盘中软件的安装和使用	(265)
快速安装.....	(265)
定制安装.....	(265)
各个子目录.....	(267)
附录 B REND386 颜色参考	(273)

引 言

自从我的第一本书《虚拟现实系统制作指南》推出之后,读者一直希望有一本书能够解释如何创建能用于第一本书中所制作的硬件的虚拟现实境界,大家还想知道如何创建能通过串行电缆和调制解调器共享的多用户虚拟境界。本书给出了在您的微机上创建虚拟境界所需要的所有工具,以满足读者们的以上要求。

不管您在编程方面是新手还是老手,您都能学会如何创建新的境界。使用一个叫 REND386 的软件系统,您就能够直接运行本书附带软盘中的虚拟境界。通过 DOS 编辑器,您可以改变 ASCII 文件以修改您的虚拟境界。

好,现在打开您的微机,插入本书所附带的软盘,从第一章开始经历完全的虚拟现实感受。

软盘内容

本书所附带软盘给出了创建读者自己的虚拟现实境界和应用程序所需要的所有软件。其中还有四个能通过调制解调器共享的多用户境界——它们都能直接运行!本书要解释的主要的软件系统叫 REND386,一个免费的虚拟现实绘制系统。通过 REND386,您将能够用普通的 DOS 编辑器创建您自己的虚拟境界。

盘中还包括四个完整的虚拟现实应用程序:

- * 射击者——一个单用户游戏,您将面对一个计算机控制的机器人。
- * 抢占游戏——一个抢占旗帜的双人游戏。
- * 砖块游戏——和朋友一起建造房屋。
- * 袭击游戏——将对手置于死地。

本书结构

本书的主题是用附带软盘所提供的软件创建虚拟境界。本书将循序渐进地教用户使用和操作该软件。

- 第一章 解释了用于描述虚拟现实技术的一些术语,使读者初步了解这项技术。本章通过一个具体的虚拟现实系统,介绍了所用软硬件的各个方面。
- 第二章 讨论了一个虚拟现实绘制系统 REND386 以及如何用这个系统创建虚拟境界中的物体。
- 第三章 进一步阐述 REND386,介绍了如何用这个软件包在虚拟境界中漫游并操纵其中的物体。
- 第四章 在前两章介绍的 PLG 物体的基础上,讨论了 FIGURE 文件的创建和操纵,这种高级的物体,功能多样,能用来创建虚拟境界的主体物体。
- 第五章 讨论了一个虚拟现实编程新手应注意的问题。创建虚拟境界并不总是很容易的事。
- 第六章 重点讨论的虚拟境界运行的性能。人们总是认为在微机上运行的虚拟现实是很慢的,本章提出了 PC 虚拟现实最需要什么样的计算机。

- 第七章 简要的剖析了 REND386 软件包——它是如何创建虚拟境界及其物体的。
- 第八章 抛开基本的 REND386 系统,讨论了如何用本书附来磁盘中的库,来创建读者自己的应用程序。
- 第九章 提供了 REND386 与许多普通计算机外设接口的原代码。
- 第十章 讨论了如何用 REND386 以及输入设备(如鼠标)与虚拟境界进行交互。
- 第十一章 创建了虚拟境界中的动画。采用一些简单技术,就可以使虚拟境界活蹦乱跳。
- 第十二章 开始讨论共享式虚拟现实,本章详细阐述了利用串行连接两台计算机的细节,使两个用户可以在地球的两端浸没在同一个虚拟境界中。
- 第十四章 讨论了用前一章中描述的技术创建的第一个虚拟现实应用程序——抢占游戏,主题是两个用户抢占五面旗帜。
- 第十五章 讨论了另外一个共享式应用程序——砖块游戏,其中用户可以用虚拟手移动其中的砖块。
- 第十六章 最后讨论了一个两人玩的射击游戏——袭击。读者在本章中将学会如何在虚拟境界中让用户自由移动。
- 附录 A 讨论了附带软件的安装。
- 附录 B 列出了 REND386 中涉及到的各种颜色。

约定

下面的约定在整本书中都通用。

命令以及用户直接键入的都是五号黑体字,例如:

DEMO ROOM. WLD

所有的文件名和目录都是大写字母,例如:

C:\ZW(RKIT1\CT(APTURE. EXE

所有用户将在显示器上看到的 C 代码以小五号等线细黑类型列出,例如:

delay value=0;

第一章 虚拟现实中的一般性术语和表达

在开始开发虚拟现实应用程序之前,首先不要被各种媒介和业界广告所迷惑,先来对虚拟现实的基本组成部分作一些规定。对于虚拟现实的准确定义,似乎每个人都有自己的看法。如果读过 Francis Hamit 写的《Virtual Reality and the Exploration of Cyberspace》,读者将看到虚拟现实被定义为“一种依赖于空间成象以及在计算机生成环境中形成的错觉的人机界面”。而根据 Ken Pimental 和 Kavin Teixeira 的《Virtual Reality——through the new looking glass》,“至少,需要使用一副虚拟现实眼镜和一台计算机来创建一个三维的人工环境,在其中,用户有一种亲临其境的感觉。他能够到处观看,到处移动,确实感到亲临其境”。这两个不同的定义表明为该技术创建专门的术语确实是件困难的事情。

本书中的虚拟现实

给出我对虚拟现实的定义,以使读者能够理解我以后对虚拟环境编程的进一步讨论。我的观点是,虚拟现实必须用两个专门的概念来定义。第一个概念涉及用户如何观看虚拟境界:虚拟现实必须允许观察者(用户)能从任何视点,以任何角度观看虚拟环境。

这一点很重要,因为在真实世界中,我们能够自由地按自己的意愿移动,我们应该把这种自由扩展到我们要进入的虚拟环境中去。图 1.1 所示的是从某个视点观看的虚拟境界,用户将注意到远处有几个离开地面的高台。虚拟现实将允许观察者将自己移动到某一个高台上去,以得到另一个视点,新的视点如图 1.2 所示,现在我们正在从图 1.1 所示的几个远处的高台之中的一个上观看。

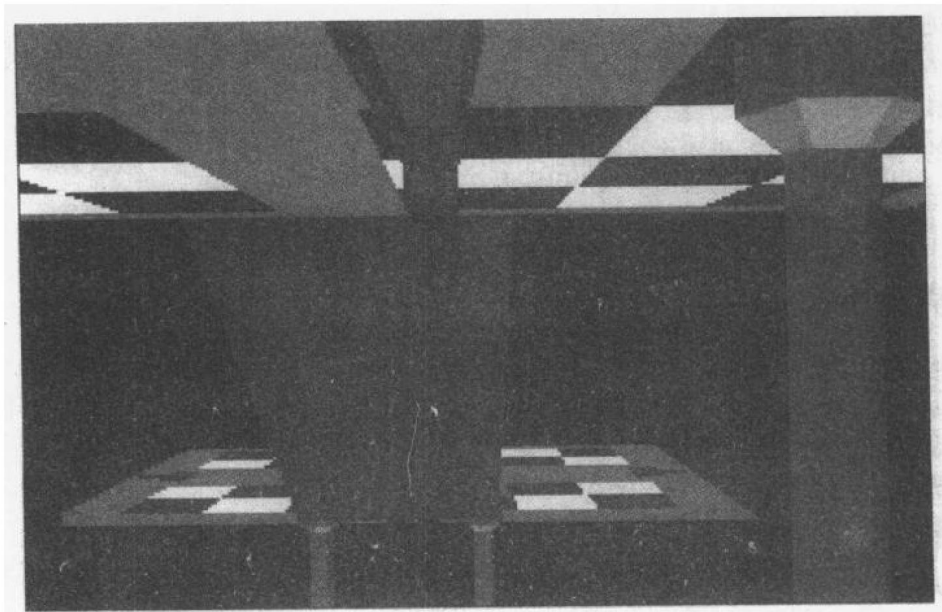


图 1.1 虚拟环境的一个视图

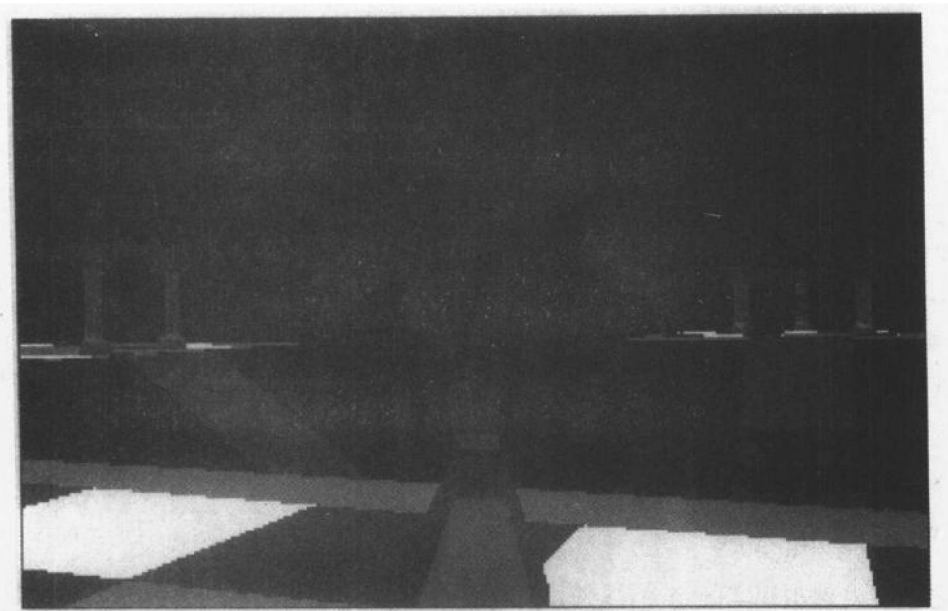


图 1.2 同一虚拟环境的另一个视图

虚拟现实的定义所包含的第二个概念是交互性。上面提到的两本书给出的定义都未提及交互性,而这可能是虚拟现实的最重要的概念:

虚拟现实必须允许用户与该环境中的物体能够进行交互。

换句话说,如果在虚拟环境中有一张桌子,上面放着一个杯子,那么,用户应该能够抓住并移动这个杯子。这种交互根据需要可以简单也可以复杂,也不必遵循真实世界的规律。事实上,虚拟环境越具有创造性,它就越有趣。记住这两个概念,我们就能够得出虚拟现实的定义了:

虚拟现实是这样一种技术,它允许用户从任何视点和角度来观看虚拟环境,并和组成该环境的物体进行交互。

在这个定义中,我们省去了几个在其他的虚拟现实定义中经常出现的观点。其中最重要的观点是计算机生成。大多数人想到虚拟现实时,就会想到在附近有一台计算机。当然,一般来说是这样的。但是,我们不想把虚拟现实限制为计算机生成的应用,因为我们无法知道未来会是什么样子的。

虚拟境界的一次旅行

全书将主要讨论如何用软件开发工具开发虚拟境界。为了理解如何开发虚拟境界,读者需要熟悉许多术语。我们不是把一长串术语匆匆地讲一遍,而是进行一次虚拟境界的旅行,同时讨论我们所看到的情景和所进行的行动。

冒险首先从眼睛开始。大多数情况下,我们观看的是计算机的显示器,有时候我们将在头上戴上头盔显示器(HMD)(见图 1.3)。

头盔显示器是一种视觉设备,在眼睛前面有一或两块液晶显示屏(LCD),并且还有一组光学镜头使观察者能够聚焦在显示屏上。头盔显示器的目的是使观察者浸没到虚拟环境中,浸没感的意思是观察者感觉到自己是环境中的一部分,它不仅仅是一种深度感。为了创造真正的



图 1.3 PCVR 杂志中提到的一种头盔显示器

深度感,我们应该能够使我们的的大脑信服这个虚拟环境就是我们实际的环境。为了使我们的的大脑相信环境的真正性,除了提供深度感之外,还应使观察者看到自己身体的某些部分。想象一下这种感觉:当用户在虚拟环境中往下看时,看到自己的脚站在地面上。

当观察者在虚拟环境中观看显示器或头盔显示器时,他还能用虚拟手捡起或移动物体。

物体和多边形

虚拟环境中所有的“东西”都由物体组成。树是一个物体,观察者的手和脚的虚拟代表也是物体。这些物体都由多边形构成,图 1.4 所示的是几种不同类型的多边形。

所有的多边形,都是平面的(没有深度),由至少三个点组成,组成多边形的“角”的每一点都叫做顶点。当我们在虚拟环境中活动时,我们所面临的是三维空间,所以每个顶点的准确位置由一组三维坐标标定,在这里,采用的是笛卡尔坐标系统(见图 1.5)。

三维空间中的任一顶点都可以沿三根分离的轴绘出:X 轴(定义该点向左或向右多远),Y 轴(定义该点向上或向下多远),Z 轴(定义该点向里或向外多远)。定义好各个顶点后,再画线把它们连接起来,就得到该多边形。把各个多边形按一定位置放置,就能得到复杂的物体。例如,把六个多边形放在一起,就能形成一个看上去象箱子一样的物体(见图 1.6);把一系列箱子放在一起,就能形成一个机器人(见图 1.7)。

境界

我们创建的所有物体都包含于虚拟环境中,这种虚拟环境叫做境界,这正如同真实世界包

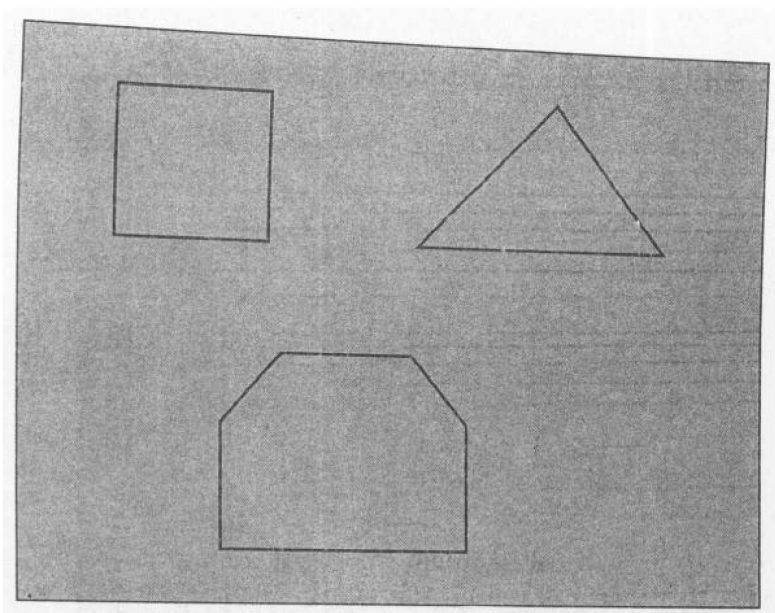


图 1.4 顶点数从 3 到 6 的几种不同类型的多边形

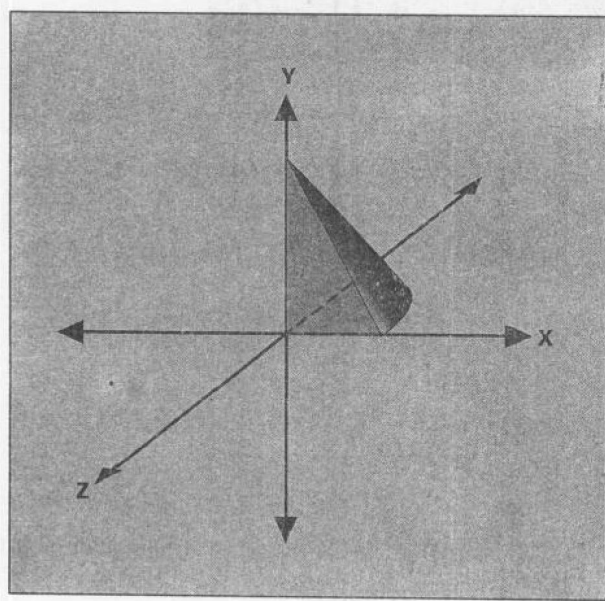


图 1.5 笛卡尔坐标系

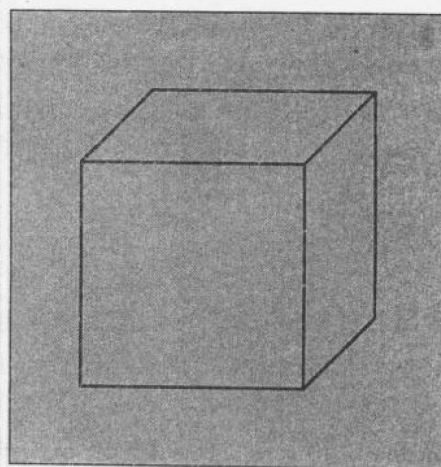


图 1.6 一个立方形的物体

含真实物体一样。除了物体之外,我们还必须把自己放入虚拟境界之中。大多数情况下,这包括两个分离的部分:头部和躯干。虚拟头用于确定视点,视点就是观看虚拟境界的眼睛——观察境界的方向和角度;虚拟躯干用来准确地确定观察者在虚拟境界中的位置以及移动的方向。

视点运动

有了视点之后,还必须能够移动这个视点,共有三种移动和三种转动。

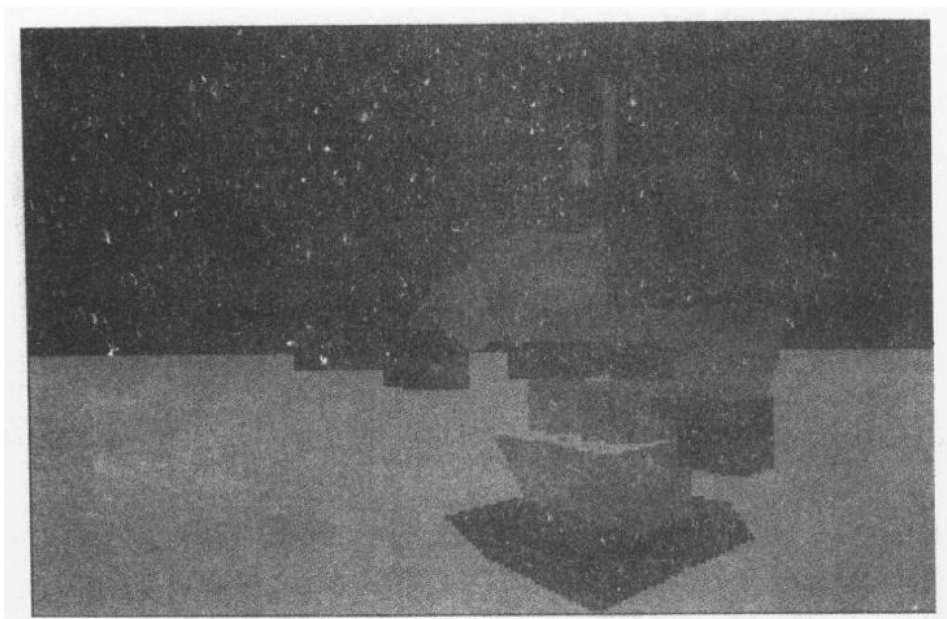


图 1.7 由简单物体组成的一个复杂的机器人

转动之一:摇摆

如果我们想看位于我们左边或右边的东西,我们通常会把头转到该方向上,这种左右转动就叫做摇摆。如果观察者向左摇摆头部,那就相当于整个世界向右移动;如果观察者向右摇摆头部,那就相当于整个世界向左移动,这可能是最常用的视点转动方式。图 1.8 就是摇摆转动的示意图。

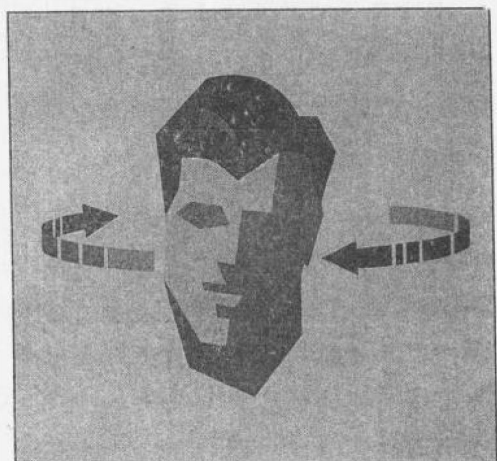


图 1.8 头部的摇摆转动

转动之二:俯仰

第二种视点的转动是俯仰运动,这种运动通过向上看或向下看实现。注意当观察者仰头时,境界向下移动;而观察者低头看时,境界向上移动,在我们后面用到这些概念时这一点很重要。

图 1.9 是俯仰运动的图示。

转动之三:侧倾

最后一种转动是侧倾运动,即将头向左或向右倾斜,这可能是最罕用的转动方式。图 1.10 是侧倾运动的图示。

头部和躯干的关系

利用以上三种转动,在保持身体静止的同时,观察者可以获得虚拟境界的三种不同视图。转动并不是与观察者的身体固连的,而主要是与观察者的视点相固连的,这一点很重要。如果把转动与观察者的身体固连起来,这就意味着如果观察者向右看,则他的整个躯体也将转向右

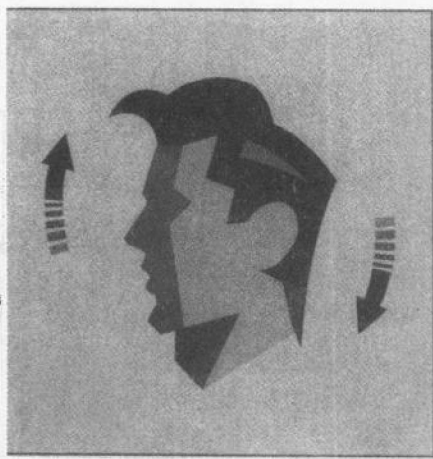


图 1.9 头部的俯仰运动



图 1.10 头部的侧倾运动

边。想象一下如果站在一条狭长的走廊的一头,将视点(头部)转向右边,观察者仍旧能够沿着走廊往下走,而不会撞到墙上;而如果将转动与观察者的身体固连起来,那观察者向右看时,不得不将整个身体也转过去,这时如果观察者再试图往前走,那观察者就会撞到墙上。所以为了更真实地模拟人体,我们使视点的运动与其身体分离。当考虑视点的有向运动时,我们就必须要注意观察者的身体了。

如果将其他物体连到虚拟身体上,那它们就会与身体一起运动,例如观察者的虚拟手和虚拟脚。这些物体叫做子物体,而观察者的躯干是父物体。子物体运动时不影响父物体(手运动时并不影响身体的位置),而父物体运动时子物体将跟着运动(身体运动时手也将跟着运动)。

躯干运动

所有的转动注意的都是观察者的视点,我们还应能够将视点移动到虚拟境界中的各个位置,这可以通过平移观察者的躯干来实现。由于视点与观察者的头部是固连的,而头部又与躯干固连,这样视点将会跟随躯干运动。躯干能在六个方向的任一方向上运动,为了确定躯干在虚拟境界中的位置,我们需要将它存在一个变量中。因为我们处于三维世界中,所以需要三个变量 x, y, z 来确定躯干在 X, Y, Z 方向上的位置。软件将利用这些变量获知观察者躯干及相关视点的位置。

平移:前后运动

增加或减少躯干当前位置的 z 坐标,我们就可以使躯干前后运动, z 坐标增加则向前运动,减少则向后运动。图 1.11 是虚拟境界中躯干前后运动的图示。

平移:左右运动

左右运动则是由 x 坐标来反映当前位置的变化,请注意运动的是观察者的整个身体,这就跟我们迈腿向左或向右运动一样,与我们的头部并没有什么关系。图 1.12 表明了如何实现左右运动。

平移:上下运动

最后一种运动是上下运动。这种运动在真实世界中并没有对应的运动,因为真实世界中有重力的作用,而在虚拟境界中没有。如果增加当前位置的 y 坐标,我们的躯干就会向上运动,减

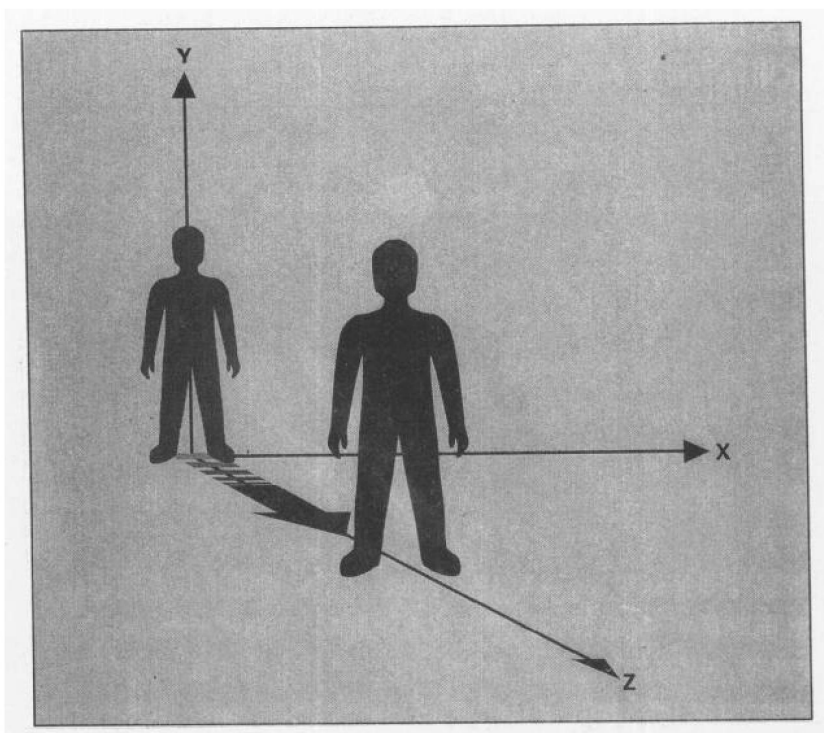


图 1.11 由 z 坐标定义的身体运动

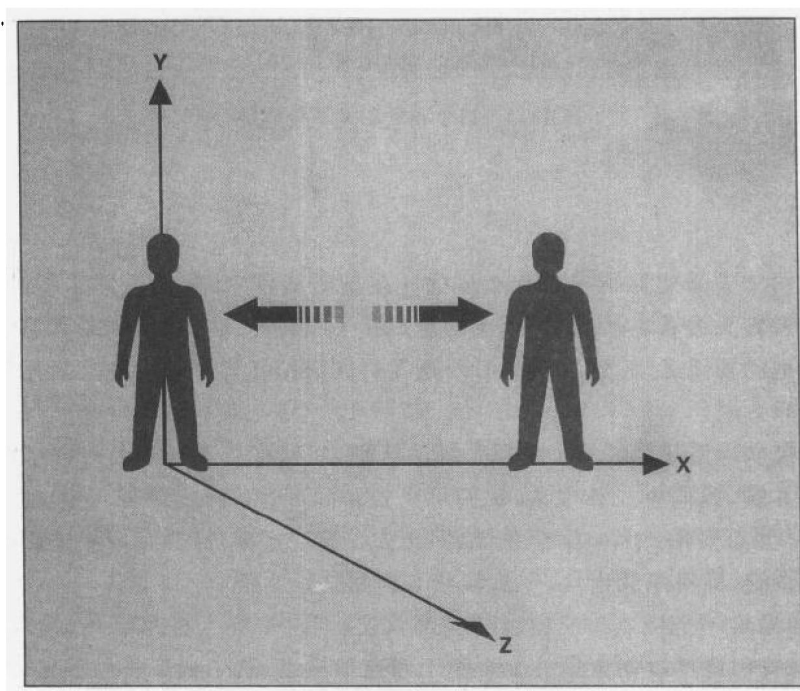


图 1.12 由 x 坐标定义的身体运动

少 y 坐标身体则向下运动, 这样我们就可以成为一只老鼠或一只雄鹰。图 1.13 所示的就是上下运动。

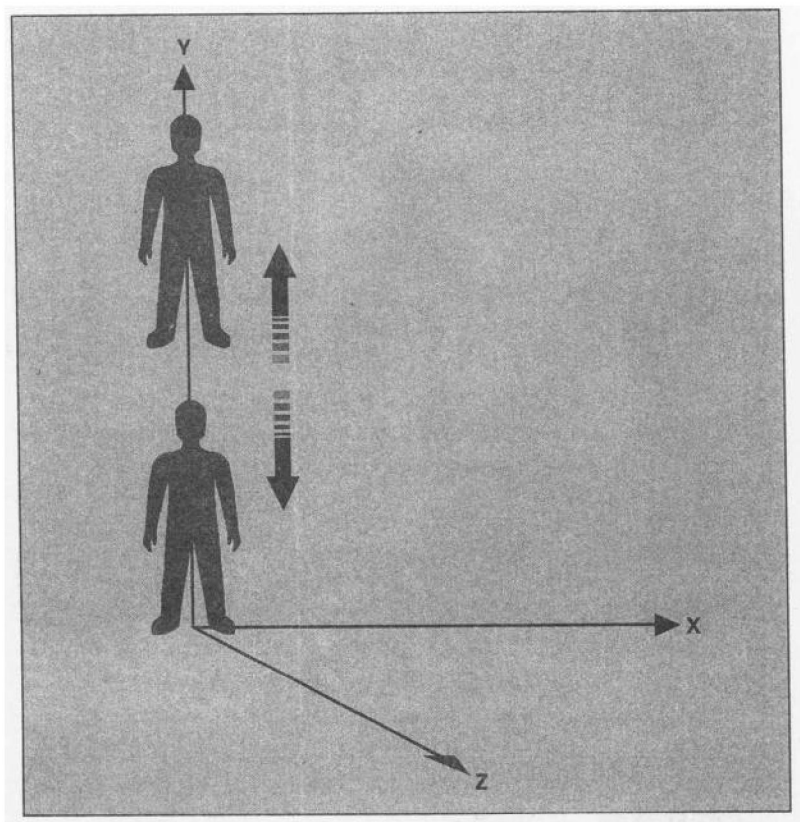


图 1.13 由 y 坐标定义的身体运动

错误的关系

现在再来看看实际情况是什么样的。在虚拟境界中,我们通常并不使视点独立于躯干。如果观察者转头去看右边的东西,他的虚拟躯干也将转向右边。如果观察者向前运动,他的身体将沿着他的视线运动。这样做的原因是在虚拟境界中移动观察者用的是极坐标,而不是 x, y, z 坐标系。

为了搞清极坐标系统和 x, y, z 坐标系统的区别,我们来看看键盘上的数字小键盘。如果我们想从 5 键去 9 键,我们有三种方法:我们可以先移到 8 键或 6 键再去 9 键;也可以直接沿着对角线去 9 键。如果使用 x, y, z 坐标系,即先去 8 键或 6 键,需要走两步才能到达目的地;如果使用极坐标系统,即对角线走法,一步就从 5 键到达了 9 键。

进一步,如果将一个视点与观察者的头部固连,则这个视点将用于转动,观察者能够任意转动他的头部;我们还可以创建另一个与躯干固连的视点,这个视点将用于确定观察者平移的方向。观察者从与头部固连的视点而不是从与躯干固连的视点,观看虚拟境界,这使得观察者在平移身体和观看境界时互相没有联系,就像在真实生活中一样。

输入设备

上面谈论了如何移动观察者的视点和躯体,我们还必须得想办法告诉计算机去移动整个虚拟境界。在本节中我们将讨论输入设备——用某种方法使计算机获得信息的设备。至于把输入设备告诉计算机的东西翻译过来并用于虚拟境界,那就是软件的事情了。能用于虚拟现实的输入设备有好多种:键盘、鼠标(普通的和三维的)、操纵杆、手套、跟踪器(包括超声波的、连杆式的和磁场的)等。

键盘

目前,键盘是计算机最常见的输入设备。大多数较初级的虚拟现实软件都允许使用键盘,因为它是计算机系统现成就包括的,但仅仅就因为需要记住那么多的键及其功能,使得用键盘来操纵虚拟境界成为一件令人筋疲力尽的事情。

对于观察者的视点和躯干的每一种运动,都必须提供一个键。视点有六种不同的运动:左右摇摆、上下俯仰、左右侧倾。在键盘上,控制这些运动最理想的选择就是箭头键,Up 键和 Down 键操纵视点的俯仰,Left 键和 Right 键操纵视点的摇摆,侧倾是用得最少的运动,可用 PageUp 键和 PageDown 键来操纵。

上述规定可以理想地转动视点,但如何移动观察者的躯干还是一个问题。观察者要在虚拟境界中走动就必须移动躯干,用箭头键控制躯干的运动要比控制视点的运动更理想一些。躯干也有六种运动,Up 键和 Down 键可以用来前后移动躯干,Left 键和 Right 键可用来控制身体的左右转动,这样身体的左右移动就需要另外的键了。如果把箭头键用于控制躯干,那就得另寻它键来控制视点了。

所以,键盘可以作为虚拟现实的输入设备,但用各个键来控制躯干和视点的运动绝对不是一种舒适的方法。另外,还必须要给虚拟手的运动分配几个键。

鼠标

另外一种常见的计算机输入设备就是鼠标。鼠标通常用来在屏幕上以二维方式移动小箭头,在虚拟境界中,普通鼠标能提供两个作用。第一个作用是“漫游”,Sense8 的虚拟现实软件 WorldToolKit for Windows 采用了这种技术。当用户启动程序后,屏幕上出现一个光标,可由鼠标控制,光标可在包含虚拟境界的窗口内任意移动。如果用户将光标移至窗口的顶部中央并按鼠标左键,用户的身体将向前移动;如果用户将光标移到窗口的左半部并按鼠标左键,用户的身体将向左转。通过将光标移到屏幕的不同部位并按下鼠标的的一个或两个键,用户可以很容易地控制自己在虚拟境界中的位置,不过,仍然没有办法控制虚拟手的运动。

鼠标的另一个作用是:控制用户的虚拟手,而让其他设备去操纵视点和躯体。此时,鼠标的左右移动控制虚拟手的左右运动;如果前后移动鼠标,虚拟手将沿着 Z 轴移进或移出虚拟环境;如要上下移动虚拟手,用户可在按下鼠标右键的同时前后移动鼠标。

虚拟手还应该可以抓东西。通过编程,鼠标左键可以用来控制“抓”这个动作,当用户按下鼠标左键时,虚拟手作抓的动作,当用户松开鼠标左键时,虚拟手也将松开。