

基于视线的智能碰撞检测方法

王雪光, 付新良

(河北工程大学信息与电气工程学院, 河北 邯郸 056038)

摘 要: 针对在虚拟漫游系统中运动物体的碰撞检测问题, 提出一种基于视线的智能碰撞检测方法。分析虚拟现实漫游环境中碰撞检测的基本原理和基于视线的碰撞检测, 采用模糊控制隶属函数提高碰撞检测效率。实验结果表明, 该方法的运算效率和检测精度有所提高, 能避开墙体和障碍物。

关键词: 智能碰撞检测; 虚拟现实; 虚拟漫游; 视线检测; 模糊控制; 包围盒

Intelligent Collision Detection Method Based on Line of Sight

WANG Xue-guang, FU Xin-liang

(College of Information and Electrical Engineering, Hebei University of Engineering, Handan 056038, China)

【Abstract】 In this paper, in order to solve the collision detection in the virtual roaming system of moving object, a collision detection method based on line of sight is proposed. It analyzes the basic principle of collision detection and collision detection based on line of sight in Virtual Reality(VR) roaming environment. This paper adopts the membership function of fuzzy control method to improve the efficiency of collision detection. Experimental results show that this method can improve the calculation efficiency, ensure the accuracy of detection, and can avoid the wall and obstacles.

【Key words】 intelligent collision detection; Virtual Reality(VR); virtual roaming; line of sight detection; fuzzy control; bounding box

DOI: 10.3969/j.issn.1000-3428.2012.18.048

1 概述

虚拟现实(Virtual Reality, VR)技术是一项综合集成技术, 以视觉和听觉数据库为基础, 涉及计算机图形学、人机交互技术、传感技术、自动控制、计算机通信、人工智能等领域, 运用计算机技术生成虚拟环境, 能让使用者有身临其境的感觉, 并可通过 VR 工具实现与该环境交互的一种技术^[1]。从本质上来说, 虚拟现实是计算机用户接口, VR 通过给用户提供视觉、听觉、触觉等各种直观自然的实时交互感知手段, 最大限度方便用户的操作。它的特性包括: 交互性, 想像性, 沉浸性^[2]。

在虚拟现实环境中, 通常存在形状复杂的物体间的碰撞行为, 而碰撞检测是虚拟漫游系统中不可或缺的重要功能, 对虚拟场景中的碰撞现象予以测试并进行处理, 如人是否碰撞物体、汽车是否撞击到墙壁等。由于虚拟漫游系统对于实时性要求很高, 因此精确、高效的碰撞检测算法及其实施已经成为虚拟仿真技术研究的热点。碰撞检测技术是在虚拟现实系统中表现场景真实感的必要条件之一, 它使虚拟对象能够真实地反映现实中的对象属性, 如人不能穿过墙壁行走、小车不能穿过设备等。这就要求虚拟现实系统必须能够提供在虚拟环境或虚拟漫游的过程中具

有碰撞检测功能。鉴于碰撞检测算法的时间复杂性较高, 而基于视线的碰撞检测效率又低, 结合两者, 本文提出一种基于视线的智能碰撞检测方法。

2 虚拟环境中碰撞检测的基本原理

从计算机图形学的角度来说, 虚拟环境中的碰撞检测可以归结为组成虚拟环境中的实体对象的多边形相交性检测问题^[3]。对于虚拟现实系统, 人们对碰撞检测的研究主要针对如何更好地解决碰撞检测的实时性与精确性的矛盾, 而不同的应用场合, 对实时性和精确性的要求不尽相同。

碰撞检测的定义如下: 设三维空间 R 用三维几何坐标系 F_w 表示, 其中, 有 N 个运动模型, 它们的空间位置和姿态随着时间而改变, F_i 表示第 i 个模型所占的空间。 F_w 随着时间的变化形成四维坐标系 C_w , 模型 F_i 沿着一的轨迹运动形成四维坐标系 C_i , 碰撞检测就是判断 $C_1 \cap C_2 \cap \dots \cap C_n = \emptyset$ 是否成立。碰撞检测问题的算法可以通过伪代码表示:

```
Begin
For i=0 to tend, 步长为  $\Delta t$ 
    for 对象  $A_i \in \{A_1, A_2, \dots, A_N\}$ 
```

基金项目: 河北省自然科学基金资助项目(F2009000853)

作者简介: 王雪光(1974—), 男, 副教授, 主研方向: 智能检测, 虚拟现实; 付新良, 硕士研究生

收稿日期: 2011-11-14 **修回日期:** 2012-01-12 **E-mail:** xueguang_wang@126.com

```

移动对象  $A_i$  到  $t$  时刻的位置
for 对象  $A_j \in \{A_{i+1}, A_{i+2}, \dots, A_N\}$ 
{移动  $A_j$  到  $t$  时刻的位置
for 对象  $A_i$  中的每一个多边形  $P_i$ 
for 对象  $A_j$  中的每一个多边形  $P_j$ 

```

如果 P_i 和 P_j 相交

则在 t 时刻发生了碰撞

系统进入: 碰撞响应

伪代码所示的算法含有 3 层循环: 首先是最外层的 for 循环, 该循环中时间 t 以步长 Δt 递增。步长 Δt 越大, 检测速度越快, 但检测精度相应下降, 且这种固定步长的检测算法不能根据具体情况调整步长和精度。第 2 层、第 3 层 for 循环, 它们要对所有“对象”(A_i 和 A_j) 进行检测, 使问题的时间复杂度变为 $O(N_2)$ 。再就是第 4 层、第 5 层表示的最内层的 for 循环, 即对组成多面体的基本几何元素进行相交测试。如果虚拟场景中有 N 个多面体, 每个多面体有 M 个顶点, 那么碰撞检测的时间复杂度为 $O(N_2 M_2)$ 。上述碰撞检测算法需要对整个场景中的所有基本几何元素进行遍历, 这是最基本也是速度最慢的碰撞检测算法。在实际应用中, 为了提高检测速度, 人们对其进行了相关的简化, 大体上分为空间分解法和层次包围盒方法 2 类^[4]。

空间分解法是将虚拟空间分解为体积相等的小单元格, 只对占据同一单元格或相邻单元格的几何对象进行相交测试。典型的空分法有八叉树法和二叉空间剖分法。

层次包围盒方法是利用体积略大而形状简单的包围盒, 把复杂的几何对象包裹起来。在进行碰撞检测时, 首先进行包围盒之间的相交测试, 如果包围盒相交, 再进行几何对象之间精确的碰撞检测。层次包围盒方法中由于包围盒的不同又分为沿坐标轴的轴向包围盒检测算法、方向包围盒检测算法、离散方向多面体检测算法^[5]、时空包围盒检测算法等, 其中, 前 3 种检测算法一般应用较多。

3 虚拟漫游中基于视线的碰撞检测原理

漫游系统的碰撞检测技术主要涉及视点与虚拟实体对象之间的碰撞检测, 以及碰撞发生后视点运动状态的调整, 目的在于防止漫游过程中视点“穿墙(设备)而过”现象的发生。在漫游过程中的视点碰撞检测, 通常基于射线与平面的相交性检测, 通过判断射线与组成实体三角形面片或包围盒是否相交, 及当前视点与相交点的距离来实现视点的碰撞检测。

在漫游过程中, 由于每一帧系统都要调用漫游任务函数, 因此每一帧针对当前视点位置和方向都会进行相应的碰撞检测, 其主要原理如下: 首先以视点的位置坐标作为空间三维场景中的检测射线原点, 沿视点坐标系的 Z 轴正方向(视线的方向)生成检测射线, 射线与窗口屏幕剪裁面相交, 如果交点位于模型在剪裁面上的投影区, 那么认为视线与模型相交, 然后得到视点到当前相交点的距离 D ,

与此同时, 通过遍历场景树查找相交点所在的模型最前端面片所属的模型节点指针, 并得到当前碰撞节点。如果视点到相交点的距离 D 小于某一设定值且模型节点不为空, 则认为当前视点与模型产生了碰撞, 点碰撞检测示意图如图 1 所示。

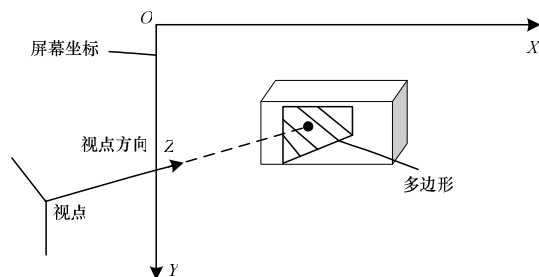


图1 视点碰撞检测示意图

漫游任务函数程序流程, 如图 2 所示。

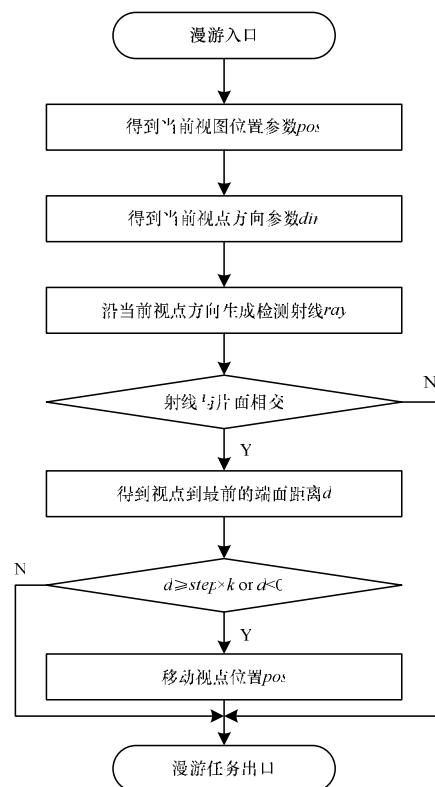


图2 漫游任务函数流程

4 虚拟漫游中基于视线的智能碰撞检测方法

对于大多数虚拟现实仿真系统而言, 最重要的是精确及时地完成观察和监控任务。保证生产的质量和安, 提高工作效率。要实现及时, 就应该让视点快速移近需要监控的设备, 而对不需要监控的设备快速浏览和及时避让。要实现精确控制就要在接近设备时, 根据与设备的距离, 使视点以适当的速度移动, 并在适当位置自动停步。这样才能使操作者在移动过程中, 很容易地、正确地调整视野的角度, 使视野中央正对要观察或控制的设备, 在适当位置停步, 避免穿过物体的现象。

使用基于碰撞检测相交距离智能控制视点移动速度

的方法有效地解决了上述问题。该方法的主要思想是：采用基于视线的相交性碰撞检测，通过设置与监控设备的较小距离而产生碰撞响应，保证视点自动停步并通过判断视点与发生碰撞响应距离的大小智能调节视点移动的步长，保证及时性和准确性。并根据速度变化情况，提醒操作者前方的设备是否需要控制，从而能及时改变方向，提高工作效率。

4.1 方法设计

从虚拟漫游中基于视线的碰撞检测原理可以看出，每一帧系统都要比较 D 与 d 的距离判断是否发生碰撞响应。而在研究中发现，计算机对用户的操作指令的理解和执行是机械化和精确化的，而作为输入设备的鼠标对用户在三维环境中操作指令的表达又非常有限，造成视点的每一次移动值都是固定的。这样比较 D 与 d 的次数，根据距离的变化情况调节移动速度时更多，影响了浏览速度。另外，对于需要监控的设备以固定速度移动，如果设定的固定步长小，移动就慢。设定的步长大，在接近设备时，很难调整使得视野中央正对监控设备，这样容易造成操作上的错误，给生产和安全带来隐患。

为了解决碰撞检测的实时性与监控精确性之间的矛盾，可根据 D 与 d 的比较值动态调整视点移动的速度。 D 与 d 的比较值是时变、非线性的，而且带有模糊性的过程，且对视点移动快慢的调节须柔性化、无级化，因此，很难采用误差很小的函数作为表达式建立数学模型，而是要通过模拟人的思维、推理和判断进行控制。针对以上特点，利用模糊技术鲁棒性和，以及在解决不确定性问题上的优势，将模糊逻辑应用于根据碰撞检测相交距离控制视点移动的速度。

4.2 方法实现

为了实现快速浏览和精确监控，在设计时，按照虚拟设备本身的作用和功能，分成需要监控的设备和不需要监控的设备，并在配置表中做上标记，供系统识别。要达到的目的是，通过系统对设备判断是否需要监控，选择不同的模糊控制器，从而实现对碰撞检测中视点移动速度的不同调节，提高浏览和监控的效率。

以系统每一帧检测的沿视点坐标系 Z 轴正方向的射线与窗口屏幕的剪裁面的相交距离为输入变量，把视点移动速度 $V=step \times K$ (其中， $step$ 为固定步长； K 为步长变化因子，通过调节 K 来调节漫游速度 V) 作为输出变量，采用单输入单输出的模糊控制器对视点移动速度进行改善，并将需要监控和不需要监控的设备发生碰撞响应的距离分别设为 $d_1=5$ inch 和 $d_2=10$ inch。本文只是研究对需要监控设备视点移动模糊器的设计。

4.2.1 输入/输出变量语言值的模糊子集设置

设输入/输出变量语言值的模糊子集为：

$L(D-d_1)=\{\text{距离为 } 0, \text{距离很近, 距离较近, 距离为中, 距离远}\}=\{OL, SL, MSL, MBL, BL\}$ 。

$K=\{\text{为 } 0, \text{很小步长, 小步长, 中步长, 大步长}\}=\{KO,$

$KTS, KS, KM, KB\}$ 。

4.2.2 输入/输出变量的论域设置

设输入变量的论域为 X 将距离长短值量化成 14 个等级，即： $X=\{0,1,2,3,4,5,6,7,8,9,10,11,12,13\}$ 。

设输出变量的论域为 Y ，将速度值量化成 12 个等级，即： $Y=\{0,1,2,3,4,5,6,7,8,9,10,11\}$ 。

4.2.3 模糊控制规则的提取

在一般情况下，设精确量 x 的变化范围为 $[a, b]$ ，运用公式：

$$\frac{y}{x - \frac{a+b}{2}} = \frac{2n}{b-a}$$

从而有 $y = \frac{2n}{b-a} \left(x - \frac{a+b}{2} \right)$ ，将其转换为 $[-n, n]$ 区间的离散量 y 即模糊量。

为了设计方便，将输入精确量的变化区间定义为以 0 为中心的对称区间，即 $a=-b$ ，这样，就可以化简为 $Y=MX$ 。其中， M 称为量化因子。

本文的模糊控制隶属函数采用三角形隶属函数^[6]，通过模糊化方法，分别将输出量 Y 和误差 e 规范化到 $[-1, 1]$ 。将区间 $[-1, 1]$ 分割为一定数量的模糊子集，如图 3 所示。

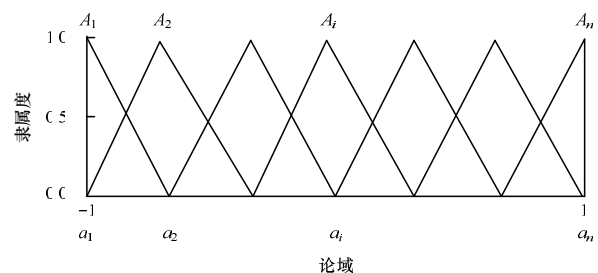


图3 模糊区间分割图

本文将模糊区间分割为等间距的三角形，则每个模糊子集将构成等腰三角形。在模糊集 A_i 中 (其中， μ 为参数)，输入量及输出量的隶属函数如下：

$$\mu_{A_i}(x) = \begin{cases} \frac{x-a_{i-1}}{a_i-a_{i-1}} & a_{i-1} \leq x \leq a_i \\ \frac{-x+a_{i+1}}{a_{i+1}-a_i} & a_i \leq x \leq a_{i+1} \\ 0 & \text{其他} \end{cases} \quad (1)$$

模糊控制规则的形式如：If e is A then Y is K 。通过计算输入量与输出量所处隶属度最大的模糊集合，可产生如下规则：若 A 和 K 分别对应的模糊集合为 OL 和 VO 依据上述控制规则，可得：If e is OL then Y is VO ，综合模糊隶属函数提取如表 1 所示的模糊控制规则。

表1 模糊控制规则输入与输出的对应关系

输入 ($D-d_1$)	输出 (步长 K)
BL	VB
MBL	VMB
MSL	VMS
SL	VS
$OL(\geq)$	VO

在表1中, BL、MBL、MSL、SL、OL 表示输入的等级代码; VB、VMB、VMS、VS、VO 表示输出的等级代码。

4.2.4 关系矩阵的计算

根据模糊规则表, 利用下式计算模糊关系矩阵:
$$R=R_1 \cup R_2 \cup \cdots \cup R_5 \tag{2}$$

即:

$$R=[(BL \times VB)+(MBL \times VMB)+(MSL \times VMS)+(SL \times VS)+(OL \times VO)]$$

4.2.5 控制查询表的获取

经实验得到模糊控制查询(需要监控设备)输入与输出的对应关系如表2所示。

表2 模糊控制查询(需要监控设备)输入与输出的对应关系

输入	输出	输入	输出
0	0	7	6
1	1	8	7
2	1	9	9
3	2	10	9
4	3	11	10
5	3	12	11
6	5	13	11

用同样的方法设计对不需要监控设备视点移动时采用的模糊控制器, 得到的控制查询(不需要监控设备)输入与输出的对应关系如表3所示。

表3 模糊控制查询(不需要监控设备)输入与输出的对应关系

输入($D-d_2$)	输出(步长 K)	输入($D-d_2$)	输出(步长 K)
0	0	5	4
1	0	6	5
2	1	7	6
3	2	8	8
4	2	9	9

4.3 测试方法对比

为了测试算法的有效性, 请8名同学分别在采用和不采用模糊控制方法的虚拟环境中, 对完成精确控制和浏览场景快速性进行测试和比较。为了使实验结果具有可比性, 要求同学在同一场景下, 对同一设备中的同一个按钮进行控制, 完成精确控制所用的时间如表4所示, 浏览整个场景所用的时间如表5所示。

表4 完成精确控制所用的时间

分类	不使用本文方法 所用时间/s	使用本文方法 所用时间/s	提高比例/(%)
实验人员 1	20.87	12.56	39.82
实验人员 2	18.25	11.14	38.96
实验人员 3	22.12	12.75	42.36
实验人员 4	21.14	13.22	37.46
实验人员 5	19.25	11.48	40.36
实验人员 6	18.65	12.11	35.07
实验人员 7	17.59	10.14	42.35
实验人员 8	20.17	13.87	31.23
平均时间	19.76	12.16	38.45

表5 浏览整个场景所用的时间

分类	不使用本文方法 所用时间/s	使用本文方法 所用时间/s	提高比例/(%)
实验人员 1	48.56	22.45	53.77
实验人员 2	38.12	26.12	31.48
实验人员 3	39.55	24.15	38.94
实验人员 4	41.26	21.19	48.64
实验人员 5	42.73	26.23	38.61
实验人员 6	45.25	25.19	44.33
实验人员 7	46.18	23.86	48.33
实验人员 8	47.12	25.17	46.58
平均时间	43.60	24.30	44.27

4.4 实验结果

本文实验使用中视典公司的 vrp-builder 编辑器作为虚拟现实的操作平台^[7], 模拟了运用视线的碰撞检测的实验过程, 图4为未采用碰撞检测时, 模拟人漫游的过程中“穿墙而过”。



图4 模拟人“穿墙而过”示意图

图5为采用基于视线的碰撞检测后, 模拟人“转身绕行”, 避开墙体及障碍物^[8]。



图5 模拟人“转身绕行”示意图