

# 基于 VR 的二维触发式碰撞检测算法<sup>①</sup>

王 坤

(四川信息职业技术学院 信息工程系, 广元 628017)

**摘 要:** 在虚拟场景中经常会用到虚拟现实技术设置某精灵的移动范围和区域, 物体与物体之间碰撞检测的算法成为技术关键, 常用的基于图形图像的算法虽然很有效应用范围广, 但是无事件处理能力, 并且执行效率低和占用大量的系统资源. 本文介绍一种利用二维数组高效的碰撞检测算法, 满足执行效率高但精度要求不高的碰撞检测.

**关键词:** 碰撞检测; 虚拟现实技术; 二维空间; 事件处理

## Algorithm of Two-dimensional Triggered Collision Detection Based on VR

WANG Kun

(Information Engineering Department, Sichuan Information Technology College, Guangyuan 628017, China)

**Abstract:** Virtual reality technology is often used in virtual scene to set the movement range of a certain role, where collision detection algorithm between objects has been a crucial technology. Common algorithm based on graphics and image is effective and widely used but is incapable of event handling, unefficient and occupies too many system resources. Therefore, an efficient algorithm for two-dimensional array collision detection is introduced to meet the needs of high efficiency and less precise collision detection.

**Key words:** collision detection; virtual reality technology; two-dimensional space; event handling

虚拟现实(Virtual Reality, 简称 VR)是近年出现的高新技术. 虚拟现实是利用计算机模拟产生一个虚拟世界空间, 提供用户关于视觉、听觉、触觉等感官的模拟, 让用户如同身临其境一般, 可以及时的没有限制的观察虚拟空间内的任何事物.

虚拟现实技术集成了计算机图形(CG)技术、计算机仿真技术、人工智能、传感技术、显示技术、网络并行处理等最新发展的技术成果, 是一种由计算机技术辅助生成的高技术模拟系统.

在虚拟世界中, 如果不加以控制, 运动的物体会随便相互穿过, 而实际上在现实的世界中是不可能办到的, 就需要加一个控制机制算法来模拟真实的现实世界. 碰撞检测就成为模拟现实世界中不可缺少的控制方法.

## 1 理论研究

### 1.1 研究背景

目前碰撞检测算法主要有两种, 一种是转化为判断线面的相交问题, 另一种是建立在包围盒基础上, 并对其不断完善的算法. 这两种算法的基本原理为:

第一种转化为判断线面, 是以视点为起点, 前进步长为长度形成线段, 将此线段与场景中的所有可见面进行相交运算, 如果有交点(相交), 则表示发生了碰撞.

第二种包围盒算法, 如果是物体与场景之间的碰撞检测, 则沿前进方向作一条射线, 射线首先与包围盒检测是否相交, 然后再作多变性相交检测; 如果是三维实体之间的碰撞检测, 给三维对象建立包围盒, 计算包围盒在三维空间的相交情况, 具体做法是先用包围盒(球)进行两两碰撞检测, 如果碰撞到了包围球,

<sup>①</sup> 收稿时间:2013-02-20;收到修改稿时间:2013-03-13

则用多边形碰撞检测方法进行计算,但场景实体很多的话,这个方法的执行效率比较低,影响计算机处理性能,虽能实现,但是不是最优化的算法.

如今,在国外也有学者在此两种算法的基础上改进了一些新的算法. Moore 曾提出了两个有效的碰撞检测算法,其一是用来处理三角剖分过的物体表面,由于任一问题表面均可表示成一系列三角面片,因而该碰撞检测算法具有普遍性,该算法的缺点是当景物为一复杂的雕塑曲面时,三角剖分可能产生大量的三角片,这会大大影响算法的效率. 而另一算法则用来处理多面体环境的碰撞检测: Moore 和 Wilhelems 根据 Cyrus-Beck 裁剪算法提出了一种凸多面体碰撞检测算法,即通过检测多面体顶点是否相互包含来判定它们是否发生碰撞.

在国内,最近也提出了一些算法,就是利用层次有向包围盒,即用若干体积略大而形状简单且规则的几何包围体来代替复杂形状的几何对象,来进行碰撞检测;还有一种基于分离包围盒的碰撞检测的算法,这种算法有效的提高了碰撞检测的效率. 本文所研究的算法是在此算法基础上做了更进一步的改进,使得算法的执行效率更高,同时还兼并事件处理能力.

### 1.2 触发式检测

触发式检测是以事件为基础,由单个或多个事件组成,事件内包含多个可执行的指令,如当检测到某一事件被触动,那么该事件内的指令按照逻辑顺序自动执行. 事件与事件之间不能有重复性,都应该是相对独立的,一般情况下互不关联的.

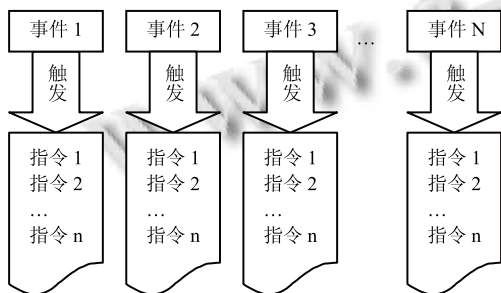


图 1 触发式示意图

### 1.3 逻辑思路

首先把虚拟世界的空间构想成一个平面的二维空间,人物在这二维空间进行移动,现实生活中正常情况下,会遇到不能跨越的障碍,比如:围墙. 那么这些障

碍在虚拟空间内也同样存在,在计算机系统中只能通过数据来表示,加以判断就能完成类似于人脑的思考处理. 因此就必须先把遇到的问题转换成数据模型,再由计算机系统计算出来当前的障碍是否能跨过. 图 2 是一个虚拟人物与障碍的关系位置图,黑色的部分为障碍,“X”为虚拟人物的位置.

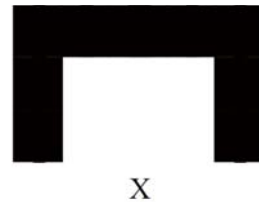


图 2 虚拟人物与障碍物的关系位置图

从平面上分析出关系位置后,需要把平面位置关系转换成的规则矩形块,只有通过规则的划分才能得出二维数组,当然在现实中相同比例的物体或障碍是不可能的,只能通过其他的算法判断来规避这一类问题. 划分好的模块图见图 3.

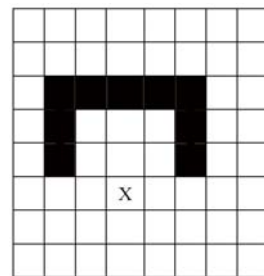


图 3 模块分布图

从模块图上可以清晰看出,虚拟人物只占其中的一个模块,同时也把整个空间划分成了等量的模块. 从图 3 中已经展现了二维数组的直观模型,再把这些位置关系转换成数据见图 4.

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0
0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	0
0	0	0	9	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

图 4 数据模块图

假设把人物所在位置以一个数字“9”来表示,障碍

物以“1”来表示, 如有其他特殊的障碍物或事件, 可以按照此规则命名赋值即可。

在二维空间中, 假设有 8 个方向, 在移动的过程中 xy 的值不断的变化, 方便程序在二维数组中做出快捷的判断。

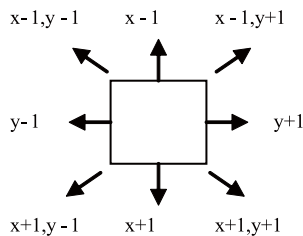


图 5 模块分布图

### 1.4 算法解析

经过分析, 下面的图可以看出算法的逻辑思路, 见图 6。

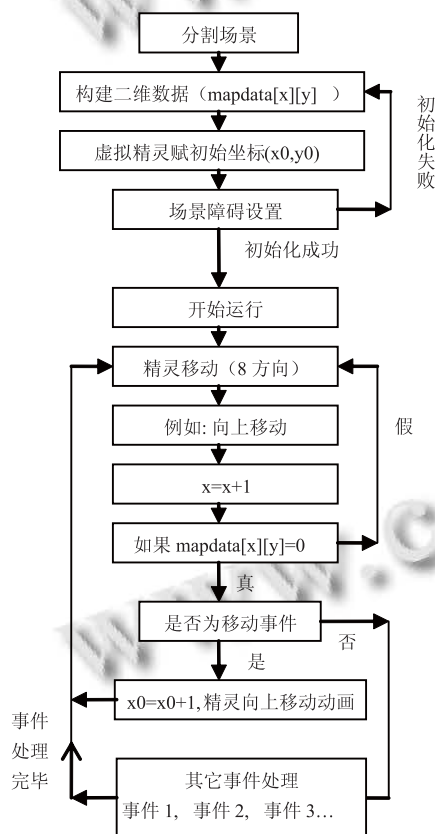


图 6 算法流程图

## 2 程序实现

### 2.1 移动碰撞检测处理

下面以 Lingo 多媒体程序设计语言来实现算法,

该算法只是思路, 可以根据算法思路改写成 C 语言, JAVA 等, 下面程序展现了一个简单的向上移动的算法, 见以下程序函数。

-- rwx, rwy 为精灵的坐标, 全局变量。

on shang

if xzzt=0 then --精灵向上移动动画控制

fangx[dlxz]=1 --移动方向控制

n=rwx\*15+50 --移动像素控制

sprite(n).member=member(rwxian[1]&"上 0")

--判断移动坐标是否超出范围

if rwx-1>0 then

if mapdata[rwx-1][rwy]<>1 then

if mapdata[rwx-1][rwy]<=10 \

or (mapdata[rwx-1][rwy]>100 \

and zb\_yings=1) then

xzzt=1

--调用移动动画函数

timeout().new \

("shangdh", yidongtt, #shangdh)

end if

end if

end if

end if

end

### 2.2 事件触发处理

在虚拟空间内, 不只是需要完成精灵物体的移动和障碍判断, 还需要完成各种各样的事件, 这些事件和现实中都是类似的, 比如: 某人看见地上有垃圾, 于是执行捡起垃圾丢入垃圾桶等事件。事件都需要触发条件, 时间、地点和人物都是触发的条件。本算法除了能判断障碍设置外, 最大的一个应用就是事件触发。根据虚拟空间程序设计, 定义某些代表特别意义的数字, 比如: 数字 11 代表跳转到另外一个场景, 数字 12 代表触发提示对话框等。

如果事件的值大于 10, 表示为其它事件, 前面已经默认事件 1 为障碍设置, 为了程序的可扩展性, 可以预留 9 个移动事件位。

下面程序简单介绍了事件的处理及应用, 加以改进后能使其达到多元化的效果。

on mapgaib

if mapdata[rwx][rwy]>10 then

```

--获得该区域地图模块的数据
txt=member(nam).text.line[mapdata[rwx][rwy]-1]
--解密模块数据
juti=baDecryptText(txt, "wwkk")
--判断该图是否为限制地图
if jinditu (integer(chars(juti.item[2], 4, 6)))=0 then
--存储人物地图数据
rwmap[dlxz]=juti.item[2]
--存储人物地图坐标
rwloc[dlxz]=point(integer(juti.item[3]),
integer(juti.item[4]))
--分类处理地图事件
case mapdata[rwx][rwy] of
11: move(x0, y0) --调用跳转移动函数
12: alert "欢迎来到虚拟世界"--弹出对话框
...
N:
end case
ditusx() --地图画面刷新函数
else
systs("此地图尚未开放") --系统提示函数
end if
end
    
```

除了移动事件之外, 如有其他事件发生, 则应具有相应事件的处理函数, 可自行添加事件处理函数, 并在触发函数中调用即可, 如其中某一事件存在程序逻辑问题, 不会对其它事件构成致命威胁。

### 2.3 试验结果与分析

测试环境: Windows XP SP3 操作系统, PC 机配置见下图 7。

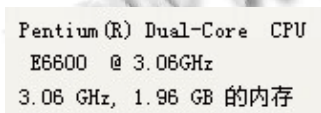


图 7 PC 机配置信息

测试方法: 建立 256Pixel×256Pixel 的地图, 按照等比分割成 8×8 的地图数据模块, 在用户发出指令点到完成碰撞检测间加入检测代码(以毫秒为计算单位), 分别对有障碍和无障碍碰撞检测 5 次, 检测结果如下图 8。

再加入事件处理函数后分别测试 5 次(事件函数

为空语句), 测试结果如下图 9。

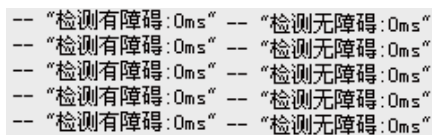


图 8 测试碰撞检测耗时



图 9 测试触发事件处理耗时

由试验结果可以得出在处理碰撞检测时可以完全忽略系统执行的时间, 加入事件处理函数后处理效率是由事件执行效率决定, 表明本算法执行效率高并且稳定。

以上测试的地图数据较小, 如果地图数据庞大, 会占用大量的内存空间(占用的大小和地图大小成正比), 为了合理利用系统资源, 应做如下优化:

地图数据分区载入, 只载入检测点周围的数据, 当检测点移位, 则根据移位量重新载入新的数据。

## 3 算法优势与局限性分析

### 3.1 算法优势分析

如需构建三维虚拟空间, 只需把相应的三维空间转换成三维数组, 在算法上加以改进即可. 该算法主要体现在以下几个方面的优势:

1) 碰撞检测高效, 稳定, 计算机数据处理工作量小, 有效利用了计算机系统资源. 再加上触发式的检测, 缩小了检测范围。

2) 算法结构模块化, 各个事件模块互不关联, 所以事件处理容错能力高. 可扩展性强, 在数组中各个数据元素都可以独立的表示不同的事件, 并可以无限延伸。

3) 很适合应用在坐标概念的虚拟空间. 虚拟人物或物体的移动始终都在一个二维数组里面, 每一次的移动其坐标值就会改变, 这样能够更加快速精准的获得坐标。

4) 算法直观, 容易查错. 此算法结构明了, 整体

都是围绕着二维数组进行的,在数据的检测和程序结构的查错方面都是非常容易的。

5) 在二维数组中可以传递多维的信息,在数据中可以用不同的数字或字符代表不同的指令,通过判断语句赋予逻辑管理,使得虚拟空间更加多样化、多维化和真实化。

### 3.2 算法应用局限性分析

不难发现,此算法存在一定的弊端,应用方向有一定的局限性,不能胜任很多对图形碰撞检测要求较高的处理中。

1) 存在精度的问题,如果对图形碰撞检测精度要求较高的情况下达不到要求,就必须采用基于图形图像碰撞检测算法。

2) 方向的局限性,只有 8 个方位,这在某些多维空间下是难以胜任的。

3) 必须在规则的虚拟空间内才能实现,没有规则的分布模块就不能构建成二维数组,算法也就不能实现。

## 4 结语

碰撞检测的算法有很多,本文的算法只是其中的一种方法,不同的算法适用于不同的需求,只要使用最适合应用体系的算法才是最佳算法。虚拟现实技术尽管存在着很多技术上的问题,但在现实生活中的应用越来越多。交互式多媒体产生以来,一直都朝着虚拟现实的方向发展。虚拟现实技术的发展一直是多媒

体开发工作者的崇高追求。

### 参考文献

- 1 苏诺,季桂树,邓拓.一种基于 GPU 的碰撞检测算法.计算机系统应用,2009,18(9):65-68.
- 2 范昭炜,万华根,高曙明.基于流的实时碰撞检测算法.软件学报,2004,15(10):1505-1514.
- 3 李学庆,孟祥旭,汪嘉业,王文平,Chun GK,Yiu SM.基于启发式搜索分离向量的凸多面体碰撞检测.计算机学报,2003,7(1):837-847.
- 4 纪传舜,刘卉.基于质点的可变形体自碰撞检测.中国图象图形学报,2011,16(3):454-461.
- 5 赵伟,谭睿璞,杨秋娜,丁文保,李文辉.基于着色算法的并行碰撞检测算法.计算机应用研究,2009,26(5):1695-1699.
- 6 范昭炜,万根华,高曙明.基于并行的快速碰撞检测算法.系统仿真学报,2000,12(5):548-552.
- 7 霍滨焱.基于图像空间的碰撞检测算法.哈尔滨:哈尔滨工程大学,2005.
- 8 Nyquist JR, Martin R. Lingo 宝典.北京:电子工业出版社,2001.
- 9 黎成茂.DIRECTOR 虚拟现实设计宝典.合肥:合肥工业大学出版社,2010.
- 10 Lawbros K. A voxel-based parallel collision detection algorithm. Proc of International Conference on Supercomputing. New York: ACM Press, 2002: 285-293.