

文章编号: 1005-8451 (2019) 12-0062-05

现代有轨电车中乘客行为的仿真研究

李卓真, 郎诚廉

(同济大学 电子与信息工程学院, 上海 200331)

摘要: 有轨电车上乘客行为仿真是研究现代有轨电车模拟器视景及乘客信息系统仿真的重要组成部分。利用3ds MAX建模软件构建了三维虚拟列车模型、人物的三维模型及基本动作, 以Unity3D为应用平台, 以VC#2017为编译器实现对三维模型的驱动控制, 从而实现对人物基本动作的仿真。对有轨电车上乘客的行为仿真, 可给现代有轨电车模拟系统提供必要的场景支持, 可以广泛应用于对列车驾驶员的培训当中。

关键词: 有轨电车; 乘客信息系统; Unity 3D

中图分类号: U231.6 : TP39 **文献标识码:** A

Simulation of passenger behavior on modern tram

LI Zhuozhen, LANG Chenglian

(College of Electrical and Information Engineering, Tongji University, Shanghai 200331, China)

Abstract: The simulation of the passenger behavior on the tram is an important part of the research on the scene of the modern tram simulator and the simulation of the passenger information system. This paper used 3ds MAX modeling software to build three-dimensional virtual train model, three-dimensional model and basic action of Artificial Intelligence(AI) characters. Unity3D was used as application platform, and VC# 2017 was used as compiler to drive and control the three-dimensional model, so as to implement the simulation of AI characters. The behavior simulation of passengers on the tram can provide necessary scene support for the modern tram simulation system, and can be widely used in the training of train drivers.

Keywords: tram; passenger information system(PIS); Unity 3D

由于我国目前城市化的快速发展, 道路的压力越来越明显^[1]。现代有轨电车是在传统有轨电车的基础上发展起来的先进公共交通制式, 与传统公共汽车相比, 具有客运能力大、速度快、弹性灵活、舒适新颖、绿色节能等特点^[2]。随着现代有轨电车的蓬勃发展, 对驾驶员的培训日益提上议程, 如何对驾驶员进行安全高效的培训是未来城市交通体系中不可缺少的组成部分^[3]。

在列车上, 司机的主要职责是驾驶列车、到站后进行站台操作、监控列车运行状态及保证乘客安全。司机通过安装在列车上的监控设备, 对乘客动态进行实时监控, 以避免发生意外事故^[4]。在列车驾驶模拟器中, 也需要对这些监控系统进行模拟^[4]。这其中, 列车乘客的动作仿真研究显得非常重要^[5]。

三维视景仿真技术是计算机技术的重要分支,

是计算机技术、图形图像处理技术、多媒体技术、信息合成技术以及显示技术等诸多高新技术的综合运用。我国在上个世纪 80 年代末 90 年代初将三维视景仿真技术应用于城轨仿真中, 主要是为司机培训提供一种模拟驾驶平台, 平台的三维视景通过地铁列车运行时从司机室实景拍摄得到, 通过叠加关键技术生成。比较有代表性的系统有: 文献[6]着重从场景模型的生成、动态场景的控制、视点变换、人物仿真等几个方面研究了虚拟仿真技术与地铁运行仿真系统的结合; 文献[7]就列车模拟驾驶器视景生成技术、计算机成像(CGI)视景系统功能、系统关键技术等作简要阐述; 文献[8]从提高动态仿真效率和渲染速度的角度出发, 采用基于图形与图像的混合建模方法和细节层次(LOD)与纹理映射技术, 实现了列车运行的三维动态视景仿真。本文在前人的基础上, 对人物进行场景上的支持, 利用 Character Studio 进行人物动画创建, 使用 AStar 算法进行场景内的自动

收稿日期: 2019-01-15

作者简介: 李卓真, 在读硕士研究生; 郎诚廉, 高级工程师。

寻路功能, 规避障碍及各种险情的发生, 以实现对驾驶员的培训。

本文利用 3ds MAX 建模软件构建了三维虚拟列车模型、人物的三维模型及基本动作模型, 以 Unity3D 为应用平台, 以 VC#2017 为编译器实现对三维模型的驱动控制, 从而为现代有轨电车模拟驾驶提供场景支持。

1 三维模型的建模

本文以南京有轨电车麒麟线为模型, 在此基础上对车站、列车等场景进行建模仿真。选用 3ds MAX 建模软件构建了三维虚拟列车模型、人物三维模型及基本动作模型, 3ds MAX 是一款三维建模与渲染的软件, 主要包含帧动画制作、多边形建模、曲面建模、材质制作、模型优化等功能。较其他建模软件相比, 3ds MAX 具有操作简单、渲染高端、三维动画逼真、互操作性灵活等优点。3ds MAX 的模型导出格式很多, 本文中, 主要将 max 文件导出成 FBX 格式文件, 然后导入到 Unity3D 中进行控制^[9]。

1.1 虚拟有轨电车建模

虚拟有轨电车的建模有如下几个步骤:

(1) 在 3ds MAX 建模软件中使用立面和平面比照参考, 通过多边形建模方法对列车进行精确建模;

(2) 根据纹理数据和贴图建立对应结构的材质球;

(3) 将材质球赋给对应的结构模型, 主要包括车内客室座椅、玻璃窗、地板、立柱和扶手、照明系统及列车整体。

所建模的列车共 5 节车厢, 其三维模型如图 1 所示。

1.2 人物建模

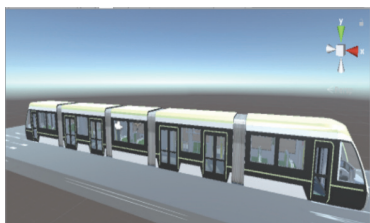


图1 虚拟电车三维模型图

在 3ds MAX 中利用 Character Studio 创建角色动画。对骨骼加载设定好的基本动作, 然后将骨骼

和人物模型绑定, 可看到骨骼驱动 3ds MAX 模型的运动, 这样就可以创建一个有设定动作的人物模型。

1.2.1 创建人物动作

正常情况下, 人物在列车内产生的行为基本可以分为: 行走 (walk)、站立 (stand)、坐下 (sit down)、起立 (stand up) 等^[10]。本文使用 3ds MAX 中的关键帧动画功能创建了这几个基本动作, 并将这几个动作加载到虚拟人物模型, 从而可以实现人物的运动。

所谓的关键帧动画, 就是按照自己所想达到的动画效果, 选择不同位置的帧, 设置不同动作的式样, 在短时间内设置尽量多的帧, 然后连续播放。通过采用特殊的计算方法算得这些关键数据, 由此达到非常顺畅的动画情景。

1.2.2 人物文件的导出

本文选择文件导出类型为 Unity3D 所支持的 FBX 格式。FBX 主要有 ASCII 和二进制两种编码格式, ASCII 编码优点是可通过代码调整数据, 缺点是文件所占内存较大, 影响 3D 引擎渲染模型的速率, 在场景模型数目过多的情况下, 渲染速率过慢会严重影响运行的帧数, 视觉效果不好; 二进制编码的模型文件虽然不易修改, 但是文件所占内存较小。因此在本文中导入的模型为二进制编码的 FBX 格式文件。图 2 为导入的人物各个状态。

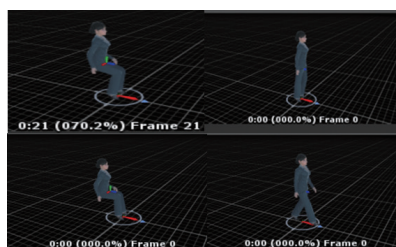


图2 各状态的人物模型

2 人物动作的仿真

在仿真过程中, 使用了导航网格 (NavMesh) 来实现乘客的自动寻路过程。

2.1 AStar算法原理

在 NavMesh 方法中采用了 AStar 算法, 其自动寻路的思路是将寻路区域划分成小方块用以存放在二维数组中。通常将起点所代表的方块记为 A, 终点所代表的方块记为 B, 障碍物则用阴影表示。计算并

比较从起点 A 到终点 B 的成本便可找到一条最优的路径, AStar 算法流程如图 3 所示。其具体方法如下:

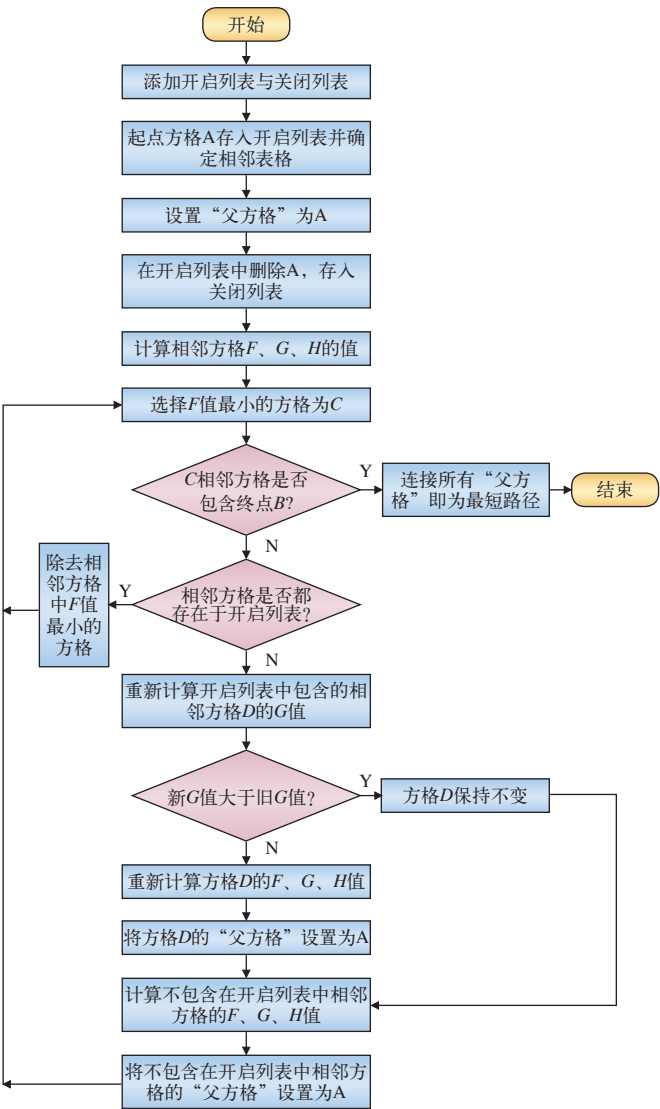


图3 ASrar算法流程图

(1) 从起点 A 开始, 把它作为待处理的方格存入一个“开启列表”, 开启列表就是一个等待检查方格的列表。

(2) 寻找起点 A 周围可以到达的 8 块方格, 将它们放入“开启列表”, 设置它们的“父方格”为 A。

(3) 从“开启列表”中删除起点 A, 并将起点 A 加入“关闭列表”, “关闭列表”中存放的都是不需要再次检查的方格。

在开启列表中计算每个方格到达终点 B 的通过成本, 计算公式为:

$$F=G+H$$

(1)

在公式 (1) 中, G 表示当前方格到起始方格的

估量代价, H 表示当前方格到终点的代价 (H 有很多计算方法, 这里设定为上下左右移动)。

平行移动距离为 1, 斜向移动距离为 $\sqrt{2}$, 为方便计算, 假设上下左右移动一个方格成本为 10, 斜向移动一个方格成本为 14。每个方格的左上角数字表示 F, 左下角数字表示 G, 右下角数字表示 H。

(4)从“开启列表”中选择 F 值最低的方格设为 C, 将其作为新的起点, 并将 C 放入“关闭列表”。

(5) 检查方格 C 相邻的方格, 但不包括障碍物和“开启列表”中的方格。判断“开启列表”中是否存在这些方格, 若不存在, 则将其加入至“开启列表”, 同时计算每个方格的 F、G 的值 (H 值始终为定值, 不用计算), 并将 C 设置为其“父方格”。

(6)若某个相邻方格 D 已经存在于“开启列表”中, 则计算经过 C 到达 D 的路径的 G 值, 若新的 G 值小于旧的 G 值, 将 D 的“父方格”改为方格 C, 然后重新计算 F、G 的值, 反之不用对方格 D 进行任何操作。

(7) 由于方格 C 相邻且可到达的方格除了障碍物之外都在“开启列表”中, 因此可以选择方格 D 作为新的起点并将其设置为父方格, 此种情况默认不允许斜向走至障碍物相邻的方格。

如此循环, 直至“开启列表”中出现了终点方格 B 后, 结束循环。连接所有“父方格”即为从起点 A 到达终点 B 的最低成本路径, 如图 4 所示。

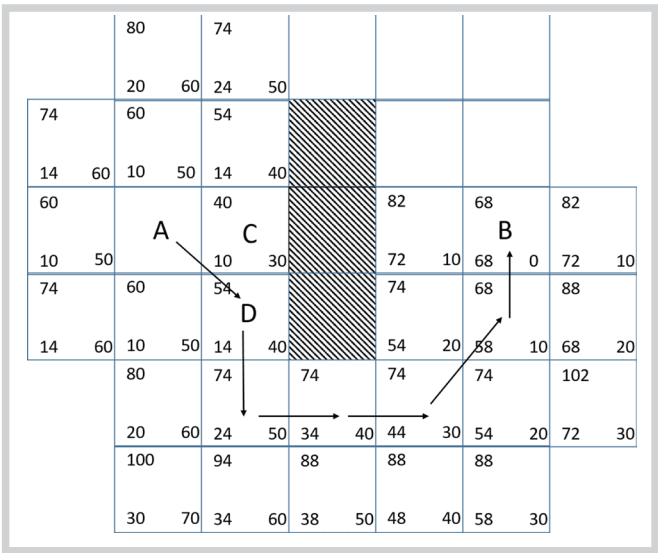


图4 最优路径模型

2.2 实现导航系统

导航系统可以让人物在场景里智能地移动、绕过障碍等，本文使用 NavMesh 来解析环境。

2.2.1 设置NavMesh

在 Inspector 窗口中点击 Static 下拉菜单，勾选静态导航 (Navigation Static)，在 Navigation Area 下拉框里选择可行走 (walkable)，即设置地面为可走区域。同时，乘客在寻路过程中要避免碰撞到柱子、闸机、墙、座椅、楼梯等障碍物，所以在 Navigation 窗口中依次选择这些障碍物，勾选 Navigation Static，设置为不可行走 (Not Walkable)。

当车厢内物体类型设置完毕后，需要对场景进行烘焙。烘焙的目的是保存当前的导航网格设置，并且保存可行走区域和障碍物的信息。烘焙分为 2 种，动态烘焙和静态烘焙。其中动态烘焙会根据当前场景的变化而实时更新导航网格，这种方式适用于障碍物需要移动的场景，比如模拟战争游戏等；静态烘焙是在场景编辑状态下设置好导航网格，在场景运行状态下不再更新，这种方式适用于障碍物以及可行走区域位置和大小不变等场景，比如室内探险、捉迷藏等游戏。因此相对于动态烘焙，静态烘焙消耗的资源更少，场景运行的帧数会更高。由于在场景运行时地铁车站里的柱子、闸机、座椅、楼梯等障碍物的位置和大小不会发生变化，所以本文采用静态烘焙方式，烘焙之前需要根据乘客的大小来调整烘焙的参数。

在开发过程中，乘客模型身高设计为 1.5 m ~ 1.9 m，调整合适的烘焙参数，在 Windows 中打开 Navigation 窗口，将导航代理半径 (Agent Radius) 参数设置为 0.1，单位为 m，因为车厢宽度为 1 m，如若导航代理半径设置的数值太大，为躲避各种障碍物则几乎无可行走区域。最后点击烘焙即可完成导航网格设置，烘焙结果如图 5 所示，地面蓝色为可行走区域。

2.2.2 设置导航代理

导航代理可以理解为去寻路的主体，设置方法为：在人物模型上为其添加 NavMeshAgent 组件，并通过相应的脚本进行驱动。

2.3 设置人物状态动画机

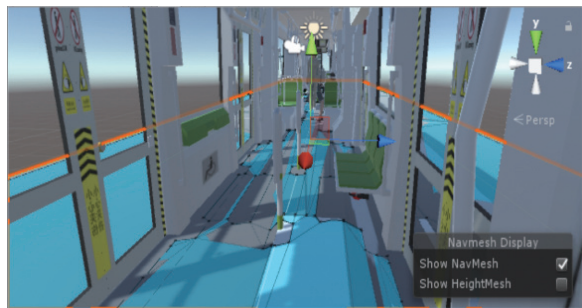


图5 导航网格模型

对于动画的控制可以选择 Animation 或 Animator，由于 Animation 只能单一地控制一个动画的播放，而 Animator 可以在不同动画之间进行切换，设置动画状态机，包括从脚本控制的状态机、混合树和事件来进行设置。所以本文采用 Animator 组件，乘客的动画状态机主要包含 walk、stand 两个动画，动画之间为双向箭头，且参数列表设置了 walk 的 bool 值，方便程序对动画状态机的控制。

使用 NavMesh 结合动画系统可使人物在车厢内实现角色的移动。创建空物体可以给指定人物设定 target，通过调用 agent.destination 函数给人物设定目标位置，使人物进行寻路。当距离终点 0.1 m 时会更新终点的坐标，以维持行走的状态。

3 实现人物上下车

3.1 车门控制

为了实现车门开关的功能，需要对车门模型进行平滑移动，这里选择一种简单轻量的插件 iTween 来实现。iTween 的核心原理是根据不同的需求，通过多种插值方法来自动生成起始点与终点之间的中间值，实现不同的效果。iTween 中的大部分方法除了简单的输入参数外，都会重载一个输入参数为哈希表 (Hashtable) 型变量的方法，通过哈希表可以对方法进行更加详细的控制。

在此处我们需要使车门进行平滑移动，可以利用 iTween 的 MoveTo 方法来实现。MoveTo 方法可以在指定的时间内不断改变物体的位置，直到达到目标位置，定义如下：

MoveTo (GameObject target, Vector3 position, float time)

MoveTo (GameObject target, Hashtable args)

其中, target 为需要移动的物体, position 为目标位置, time 为经过的时间。为了使车门能匀速移动, 我们通过之前提到的哈希表来将物体的移动方式设定为线性。程序中调用 MoveTo 方法的一段代码如下:

```
iTween.MoveTo (this.gameObject, iTween.Hash  
("x", 1.0f, "z", 0, "time", 3, "islocal", true, "easetype",  
iTween.EaseType.linear));
```

其中, this.gameObject 是所挂在的组件, iTween.Hash 是 iTween 中方便用户创建哈希表的方法。x 表示目标位置的 x 坐标, z 表示目标位置的 z 坐标, time 表示在多久后移动到目标位置, islocal 为采用世界坐标还是与父物体有关本地坐标, easetype 表示物体移动的类型, 这里选择 iTween 内置的线性方式, 即为匀速运动。车门移动如图 6 所示。



图6 模拟车门关闭及开启

3.2 乘客上下车模拟

乘客分为动态乘客和静态乘客。静态乘客主要集中在列车上, 不参与寻路过程。为了模拟乘客上下车, 这里采用一种简单的方式。以每个车厢模型为父物体在车门外创建若干空的子物体, 在开始时将乘客模型生成在车厢内, 需要下车时就让乘客走向这些子物体。当列车未进站时, 动态乘客在路面的导航网格内不断地进行寻路, 当列车进站时, 会选择最近的车门上车, 并跟随列车一起移动^[11]。

本文中使用的乘客模型是已经绑定骨骼以及动画的预制体 (Prefab)。Prefab 是 Unity 中的一种资源类型, 存储在项目资源中的一种可重复使用的对象, 当预制体被添加到场景中, 就创建了他的一个实例。将需要频繁创建的物体设置为预制体可以节省内存, 乘客模型很适合被设置为预制体。

车厢门有左右的区别, 乘客也根据左右分为 2 组控制, 定义 2 个方法用于将模型预制体生成为实

例并置于车厢内。

通过为人物进行脚本编辑来实现人物的上下车。

4 结束语

本文以南京有轨电车麒麟线为模型, 通过 3ds MAX 建模软件完成对虚拟列车场景及人物三维模型的建立, 虚拟了 2 个基本动作 (walk、stand)。在此基础上, 使用 NavMesh 与人物动画状态机的结合, 完成了人物在车厢内的移动, 实现了乘客动作的三维动作仿真。同时为有轨电车模拟驾驶系统提供了部分车站乘客上下车的场景, 具有一定的实用价值。

参考文献

- [1] 康 宏. 城市快速交通枢纽综合体设计研究 [D]. 上海: 同济大学, 2006.
- [2] 周 淮. 我国现代有轨电车发展现状与需要研究的若干问题 [J]. 交通与港航, 2017 (1): 15-20.
- [3] Zhu L, Yu F R, Ning B, et al. Cross-Layer Design for Video Transmissions in Metro Passenger Information Systems[J]. IEEE Transactions on Vehicular Technology, 2011, 60(3): 1171-1181.
- [4] 张博凯. 城市轨道交通车载信息系统方案研究 [J]. 科技与生活, 2011 (8): 209, 199.
- [5] Nelson J D. The potential for realtime passenger information as part of an integrated buscontrol/information system[J]. Journal of Advanced Transportation, 2010, 29(1):13-25.
- [6] 黄友能, 唐 涛, 宋晓伟. 虚拟仿真技术在地铁列车运行仿真系统中的研究 [J]. 系统仿真学报, 2008, 20 (12): 3208-3211.
- [7] 吴海超, 郭四维. CGI 成像技术在列车模拟驾驶器中的应用研究 [J]. 现代城市轨道交通, 2013 (1): 86-88.
- [8] 李碧波, 杨 岳, 易 兵. 列车运行三维动态视景仿真 [J]. 铁路计算机应用, 2017, 26 (5): 64-68.
- [9] 王怀松, 陈荣武. 基于三维视景的列车运行仿真平台的研究与实现 [J]. 系统仿真技术, 2016 (4): 28-32.
- [10] 陈宏川. 虚拟交通环境下的人群仿真 [D]. 成都: 西南交通大, 2013.
- [11] 李斯硕, 阎国强. 现代有轨电车交叉路口视景的仿真 [J]. 铁路计算机应用, 2017, 26 (6): 55-59.

责任编辑 付 思