

文章编号: 1005-8451 (2013) 08-0001-04

基于OpenFlight API 的城轨电子沙盘建模方法研究

杨飞龙, 郎诚廉

(同济大学 电气工程系, 上海 201804)

摘要: 针对不同轨道线路的线形特点, 通过线路中心线数据计算轨道截面关键点的空间信息, 以OpenFlight API为基础编程, 实现轨道线路三维模型的自动生成。建立地形模型并在轨线合适位置放置车站、信号机及编组机车, 沿线布置树木和建筑模型, 搭建成逼真的城轨电子沙盘。

关键词: 城轨电子沙盘; OpenFlight API; 自动建模; 道岔

中图分类号: U231.2 : TP39 **文献标识码:** A

Study on modeling method of electronic sand table for Urban Transit based on OpenFlight API

YANG Feilong, LANG Chenglian

(Department of Electrical Engineering, Tongji University, Shanghai 201804, China)

Abstract: This paper calculated spatial information of the key points on the track cross-section based on the centerline data firstly according to the shape characteristics of the different rail lines, then created the models of track and turnout automatically by programming in the basis of the relevant spatial information and OpenFlight API. The electronic sand table for the Urban Transit could be implemented by creating the terrain model, placing the stations, semaphores, train at the appropriate position, laying some trees and buildings along the line.

Key words: electronic sand table for Urban Transit; OpenFlight API; modeling automatically; turnout

城轨电子沙盘通过构建逼真的线路及沿线景物的三维模型, 可以让观众生动形象地了解轨道交通系统的组成及运行概况。但在城轨电子沙盘中, 建立轨道线路模型工作量大且不易人工进行修改。因此, 开发一种线路模型自动生成的方法, 在此基础上构建电子沙盘, 可以减轻劳动强度, 有效提高工作质量及效率。

1 城轨电子沙盘建模方案

电子沙盘比传统物理沙盘能更直观和形象地展示现实场景且互动性好, 因此得到了广泛应用。随着轨道交通的发展, 城轨电子沙盘成为近年的研究热点。构建城轨电子沙盘三维模型的方法有很多种, 可以通过建模软件来完成, 如Multigen Creator、3ds MAX 等就是典型的代表,

或通过建模的一些应用程序接口来编程实现, 如OpenFlight API、OSG 等。OSG 是一个基于工业图形标准的高层次图像开发 API 接口, 即三维渲染引擎。Openflight API 是一个包含头文件和链接库的 C 语言库, 它提供了访问 OpenFlight 数据库和 Creator 模型系统的接口方法, 通过 API 可以进行 OpenFlight 模型的实时模拟仿真、自动建模以及通过插件的形式对 Creator 进行功能扩展。

OSG 中创建几何体通常有 3 种方法: 使用几何图元、使用 OSG 中的预定义几何体、读取文件导入场景模型。OSG 将几何图元的类别分为点、线段、三角形、四边形条带以及多变形等 10 种, 利用这 10 种图元构成任意复杂程度的物体。OpenFlight 格式的模型是通过几何图元 (geometry)、继承 (hierarchy) 和属性 (attributes) 3 种元素定义的。几何图元包含多边形、边和顶点, 通过几何图元可构建出平面, 它通过继承关系进行组织。属性用来提供一些附加特性, 包

收稿日期: 2013-01-12

作者简介: 杨飞龙, 在读硕士研究生; 郎诚廉, 高级工程师。

括颜色属性、纹理属性，以获得理想效果。基于 OpenFlight API 的建模方法就是从这 3 方面出发，计算坐标构成面，再由多边形围成模型轮廓，然后给各个面添加纹理，令模型有较高的真实感。相比较而言，针对轨道线路的特点调用 OpenFlight API 编程建模较为简单，故本文选取以 OpenFlight API 为基础的建模方法来完成城轨电子沙盘轨道三维模型的创建。

2 轨道三维模型建立

2.1 线路数据计算

建立轨道模型的前提是要从图纸上读出描述线路几何位形的中心线数据。分别读取线路中心线的横断面和纵断面数据，然后统一起来。再用多条线段构成的折线来近似表示线路中心线的曲线部分，由图纸数据信息计算组成中心线的所有线段的偏角及端点坐标，可得到描述线路中心线的完整数据，如表 1 所示。

表 1 描述线路中心线的数据

Km (m)	X	Y	Z	H	P	R
0	10.960	135.380	0	6.31	0	0
1	10.843	153.370	0	7.62	0	0

其中 X、Y、Z 用来确定道路模型在场景中的位置，代表道路模型中心点在场景中的坐标。H、P、R 用来确定道路模型的朝向，代表道路模型绕 3 个坐标轴的旋转角度。

2.2 轨道截面分析

钢轨和道床是轨道的重要组成部分，先对其几何尺寸进行分析，然后计算关键点坐标。钢轨截面如图 1 所示。但为了减少模型的面，建模过程中把钢轨简化成上下底分别为轨头和轨底宽度的梯形。道床以长方体代替，宽度 2.36 m，高度 $h=0.314$ m，轨道模型的截面如图 2 所示。

$A(x, y, z)$ 就是线路中心线在此截面上的点（即构成中心线某条线段的端点）。由上述数据及轨道偏转角度可求得截面上各点坐标，设 d_0 为 0 点至 A 点距离，则 0 点的坐标计算如下：

$$x_0=x-d_0\cdot\cosh, y_0=y-d_0\cdot\sinh, z_0=z \tag{1}$$

同理可算得其他点坐标。由此可见，只要求得钢轨截面上的点距中心线距离，即可求得该点坐标。

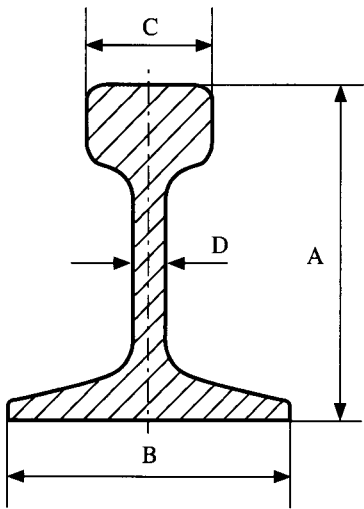


图 1 钢轨截面图

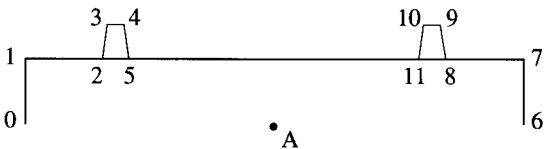


图 2 轨道模型截面图

2.3 轨道模型创建

由 2.1 节中所述的方法，根据线路中心线上一系列点的空间信息可求出各段轨道首尾处截面上的点，然后通过调用 OpenFlight API 编程实现连点成面，并组成模型。部分程序如下：

```
mgrec *prec; mgrec *vrec; mgrec *v[4];
vrec = mgNewRec (fltVertex); mgAppend
(prec, vrec);
mgSetCoord3d (vrec, fltCoord3d, x, y, z);
v[0] = mgGetChild (prec);v[1] = mgGet-
Next (v[0]);
v[2] = mgGetNext (v[1]);v[3] = mgGetNext
(v[2]);
mgSetAttList (v[0], fltVU, 0.0, fltVV, l[k],
MG_NULL);
mgSetAttList (v[1], fltVU, 1.0, fltVV,l[k],
MG_NULL);
mgSetAttList (v[2], fltVU, 1.0, fltVV,
l[k+1], MG_NULL);
mgSetAttList (v[3], fltVU, 0.0, fltVV,
l[k+1], MG_NULL);
```

程序实现的功能就是把 4 个点添加到面节点 prec 上围成一个面，然后为这个面添加纹理并设

置合适的UV坐标。由图2知每段轨道模型由围成道床、钢轨的9个面形成，同理编程建立其余的面，则可组成完整的轨道模型。沿着中心线上的点从开始计算到终点，一条完整的轨道就完成了。可见，只要根据中心线上一系列线段端点的空间信息计算得到相应轨道截面上各点的坐标，就可以通过程序快速高效地建立轨道三维模型，且线路平滑视觉效果良好。

3 道岔模型建立

本文所述道岔建模方法的主要思路就是根据道岔不同部分的构造特点，将主线和侧线的中心线数据分段，然后依据关键点与中心线间几何关系分别求各段截面上关键点空间信息，最后调用OpenFlight API 实现模型建立。

3.1 单开道岔数据分段

道岔的基本功能是实现线路的连接和交叉。其中以单开道岔在城市轨道交通中的应用最为广泛，如图3所示。

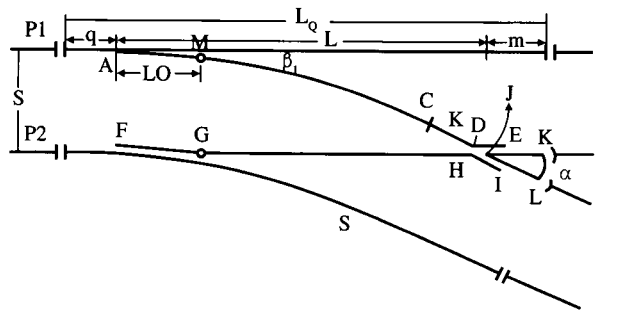


图3 单开道岔示意图

本文以在地铁中应用广泛的9号右开道岔为例，说明自动建立道岔模型的方法。9号道岔的数据信息如表2所示。

表2 9号道岔数据信息

道岔号数	辙岔角	钢轨类型(Kg/m)	辙岔角度
9	6° 20' 25"	50	0° 59' 26.2"
尖轨长度(m)	辙岔趾距(m)	辙岔跟距(m)	基本轨长(m)
6.45	1.849	2.708	2.7

尖轨后导曲线半径为180.717 5 m，CE段距离为K=2.176 7 m，CD=1.474 7 m，GH=18.374 6 m。为应用方便，特将道岔模型做了适当延长，点E距模型主线和侧线终点的距离均为10 m。相关轨

道工程类书籍已对道岔主要尺寸计算做过详细分析，本文不再重复每个数据的计算过程。基本轨的建模与普通线路相同。主线数据可分为5段，第1段左侧为普通钢轨，右侧为尖轨，第2段两侧均是普通钢轨，第3段左侧为普通钢轨，右侧为有害空间，第4段左侧为普通钢轨，右侧为辙岔前段，第5段左侧为普通钢轨，右侧为辙岔后段及普通钢轨。侧线同理可分成7段，分段信息见表3。线路分段后，同2.1节中所述，将中心线曲线部分用折线近似表示并求出各线段端点的空间信息，由此构成主线侧线中心线详细数据。

表3 侧线中心线分段数据

线形	NO	X	Y	Z	H	P	R	L	R1	R2
直线	0	0	0	0	-0.990 611	0	0	4	0	0
直线	0	0	0	0	-0.990 611	0	0	2.45	0	0
圆弧	0	0	0	0	-0.990 611	0	0	16.874 6	180.717 5	0
直线	0	0	0	0	-6.340 278	0	0	1.474 7	0	0
直线	0	0	0	0	-6.340 278	0	0	0.802	0	0
直线	0	0	0	0	-6.340 278	0	0	1.16376 5	0	0
直线	0	0	0	0	-6.340 278	0	0	8.736 235	0	0

3.2 构建道岔模型

3.2.1 钢轨关键点求取

道岔模型的建立是由中心线上各点的空间信息求得该点所在道岔截面上关键点的坐标，然后由多个点围成面，组成道岔模型。但由于道岔有尖轨、辙岔等特殊元件，因此部分关键点距线路中心线的距离d与普通的轨道不同，如尖轨宽度较窄，此段钢轨内侧距中心线的距离d就大一些，可见要根据元件的具体形状分析其关键点距中心线的距离，然后再依照公式(1)求解。

3.2.2 道床及DOF节点

上文提到普通轨道道床的模型是依靠中心线求得相应截面上0、1、6、7点(见图2)坐标建立的。但是在道岔处若仍依照此法建模，则正线和侧线道床将会有大面积重叠。为简单计算，建模时可先画钢轨，然后根据道岔的形状再建立道床，同时沿着钢轨建2个高度不同的辅助面并贴上扣件纹理，便不会发生面重叠，达到更好的显示效果。道岔通过转动尖轨来完成引导列车进入另一股道的功能，因此把围成尖轨的面组织在DOF节点下，然后设置DOF节点参数控制尖轨转动，建成的道岔模型如图4所示。

将2付同型号道岔尾部用一段反向曲线连接



图4 道岔模型

起来便构成了渡线,实现列车折返,完善了城轨电子沙盘轨道的功能。

4 沙盘整体模型完善

实现城轨电子沙盘的各项功能需要展示一个完全的轨道交通系统,因此需要在轨道线路的基础上引入其他模型来丰富和完善城轨电子沙盘。这些模型主要包括沿线设置的车站及站内的行人,线路上运行的列车、沿线的信号机、建筑物、树木及马路,轨道所经过的高架及桥梁等。

模型的组合在 Multigen Creator 建模软件中进行,通过外部引用的方法将各模型加以组合。组合上下行线路和渡线构建完整的线路模型,在线路的起点、中点和末端设置3个不同的车站以体现城轨站台的多样性,分别是岛式站后折返站、侧式站、岛式站前折返站。在线路合适位置放入编组列车,分别表示列车的进站、出站以及在线路上正常运行等3种状态。

信号机是城轨交通系统的重要信号设备,根据相关信号设备布置规范,在车站模型的附近位置放置出站信号机和进站信号机,在线路终端处设置终端信号机,在道岔相应位置设置防护信号机表明道岔开放方向。通过定义 Switch 节点可使信号灯显示不同的颜色以代表不同的含义,同时注意道岔防护信号机信号灯的颜色应与尖轨转动的位置相符合。

为了获得更好的视觉效果,需要根据轨线走向建立地形模型,通过截取地形的卫星照片制作地形纹理,可使地形模型形象逼真。在车站等人流集散地放置简单的住宅、商场等建筑及道路模型,在道路旁及轨道沿线引入了大量三维树模型,合理布置并适当调整大小,使整个沙盘看起来更加接近真实场景。图5和图6分别是城轨沙盘全貌和车站内电子沙盘的视景图片,这也体现了城轨电子沙盘的另外一个优点,可以任意移动和缩放整个沙盘,从而能让观众从各个角度去观察城轨电子沙盘以获得更丰富的信息。

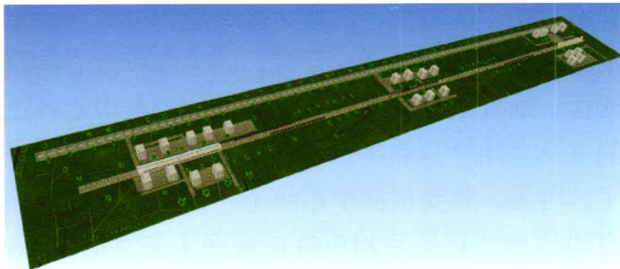


图5 电子沙盘全貌



图6 车站内电子沙盘

5 结束语

本文所述的建模方法提高了城轨电子沙盘轨道线路模型建立的效率,且模型有较好的真实感。建立的城轨电子沙盘模型基本涵盖了城市轨道交通系统的轨线、车站、机车、信号设备等基本要素,同时地形、建筑以及树木等模型的引入提高了电子沙盘的逼真程度,能让观众准确形象地了解到轨道交通系统的组成部分及其作用。此外,轨道线路的中心线数据描述了线路的走向,以中心线数据为基础在 Visual C++ 平台下开发的基于 OSG 的列车控制程序,可以控制列车运动,控制信号灯颜色及道岔开放方向,使其与机车的运行协调一致,从而模拟列车在轨道上的运行过程,在城市轨道交通系统的运作展示及教学实践方面都有着积极的作用。

参考文献:

- [1] OpenFlight_API_Users_Guide(Volume1):Read/Write[Z]. MultiGen-Paradigm, Inc, 2003.
- [2] 练松良. 轨道交通 [M]. 上海: 同济大学出版社, 2006.
- [3] 杨智勋. 三维电子沙盘系统研究与实现 [D]. 长沙: 中南大学, 2011, 5.
- [4] 朱丹. 城市轨道交通概论 [M]. 北京: 人民交通出版社, 2012.

责任编辑 方圆