

文章编号：1005-8451（2013）04-0015-04

高速列车驾驶台多功能显示器虚拟仿真

唐 晴¹, 何 润², 江永全³, 李 柯²,

(1.西南交通大学 学报编辑部, 成都 610031;

2.西南交通大学 信息科学与技术学院, 成都 610031;

3.西南交通大学 牵引动力国家重点实验室, 成都 610031)

摘要：本文研究了基于XNA平台的高速列车驾驶台虚拟仿真技术，着重介绍了虚拟驾驶台中多功能显示器在虚拟三维场景中的实现方法。主要涉及虚拟显示器的创建、驱动及交互等功能的实现方法，采用基于XNA平台的离屏渲染技术、动态参数驱动MFD显示和模型拾取算法，逼真地实现了实时显示高速列车驾驶台运行信息和人机交互，提高了虚拟场景的真实感、沉浸感。

关键词：虚拟仿真；XNA平台；多功能显示器（MFD）；离屏渲染；模型拾取

中图分类号：U266 : TP39 文献标识码：A

Simulation of multi-function displays in high-speed train cab

TANG Qing¹, HE Run², JIANG Yongquan³, LI Ke²

(1. Editorial Department of Journal, Southwest Jiaotong University, Chengdu 610031, China;

2. School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China;

3. Traction Power State Key Laboratory, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: This paper studied on the virtual simulation technology of high-speed train cab based on XNA platform, especially the implementation of the multi-function displays (MFD) of the virtual cab in the virtual three-dimensional scene. It was mainly involved the creation, driving and interaction of the virtual monitor. The off-screen rendering technology was used which was based on XNA platform, dynamic parameter-driven MFD and model picking up algorithm. These methods could vividly implement the real-time display of the high-speed train cab operation information and the man-machine interactive control, make the virtual scene much more lifelike and immersive.

Key words: virtual simulation; XNA platform; multi-function display(MFD); off-screen rendering; model picking

随着高铁技术的发展和列车驾驶台显示设备的不断更新，功能强大的多功能显示器（Multi-Function Displays, MFD）已逐步取代了传统的机电显示设备，列车各系统综合信息都集成显示在MFD上，成为与列车驾驶员交互的重要设备。

运用虚拟现实技术对驾驶台MFD进行模拟仿真，是构建高速列车虚拟驾驶的重要组成部分。在虚拟驾驶台仿真环境中，列车操纵设备、内部结构均与真车保持一致，使处于仿真环境中的操作者有置身真实驾驶环境的感觉，操作人员可与虚拟系统进行交互。在这种场合中，逼真的虚拟MFD不仅能提高场景的沉浸感，对于列车的舒适性研究、列车模拟驾驶等应用也具有重要作用。

多功能显示器的仿真是高速列车仿真中的一

个关键问题。每一个MFD由一定数量的多功能周边键组成，通常一个MFD包括多个甚至数十个页面，每个页面还包含多个功能区域。现有文献多采用OpenGL三维图形开发语言结合第三方软件（如Vega仪表开发组件、DISTI公司开发的GL Studio^[1]等仪表开发软件）来实现^[2-6]。但这些方法对于界面繁多、显示格式复杂且数据交互要求高的多功能显示器，不仅开发成本大、开发周期长，且仿真效果欠佳。

XNA (DirectX and Xbox Next-generation Architecture) 是由微软公司为游戏开发人员研发的便捷、高效的下一代游戏开发平台^[7]。它以DirectX为底层支持，提供了诸如内容导入、数据库、输入获取、音视频播放、网络连接等游戏开发中的常用工具。本文研究了一种基于XNA平台的离屏渲染技术，先把虚拟场景中要显示到MFD

的图像信息绘制到内存位图 Buffer 中，然后再将其作为动态纹理映射到 MFD 模型平面上。该方法不仅对 MFD 模型的位置、大小没有任何限制，而且减少了系统各部件之间的通信量，提高了虚拟场景的仿真效率，降低了开发难度、开发成本，缩短了开发周期。

1 关键技术

1.1 离屏渲染

在三维视景仿真中，场景的每一帧画面通常都是绘制到显示屏幕上供用户观看的。目前显示硬件不仅能够把场景渲染到屏幕，还能渲染到内存位图中，通过纹理映射技术就可以把位图再次赋到场景中的物体上去。这种在后台生成显示数据，在场景更新时再显示到屏幕的技术被称为离屏渲染（Off-Screen Rendering）。

1.2 纹理映射

纹理映射（Texture Mapping）是一种将二维图像中的像素映射到三维几何形状表面的过程，是使其产生特殊效果或真实感的一种技术。使用纹理映射技术可以避免对场景的每个细节都使用多边形来表示，进而可以大大减少场景模型的多边形数目，提高图形显示速度^[8]。

在二维图像的平面区域内，每点都可用数学函数表达，从而可以离散地分离出每点的灰度值和颜色值，这个平面区域称为纹理空间。一般将纹理空间的平面区域定义在 $[0,1]^2$ 二维区间上。纹理映射就是确定物体表面一点 P 在纹理空间的对应点 (u,v) ，而纹理空间的点 (u,v) 处的值就是物体表面 P 的纹理属性^[9]。纹理空间到物体空间的映射可表示为：

$$[x \ y \ z] = [u \ v \ 1] H \quad (1)$$

式中： $0 \leq u \leq 1, 0 \leq v \leq 1$ ；H 为系数矩阵。

由于透视变换，纹理映射提供了良好的三维线索，因此纹理面被映射到一个几何模型的三维 (x,y,z) 坐标上。当 3D 模型转换和投影到屏幕时，映射后的纹理也就被旋转和改变大小，并在屏幕上绘制出来，就好似位于模型的表面上。

1.3 模型拾取

模型拾取（Model-Picking）是指通过鼠标在屏幕上单击来选中某个 3D 模型，然后对该模型

进行操作。在 XNA 中，拾取算法基本思想如下：

得到鼠标点击处的屏幕坐标，通过投影矩阵和观察矩阵把该坐标转换为通过视点和鼠标点击点的一条射入 3D 场景的光线，如图 1 所示，该光线可能与场景中多个模型相交，但在大多数应用中只需找出离相机最近的那个对象，因为这个对象才占据屏幕的像素。

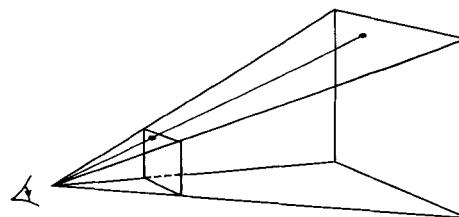


图1 屏幕2D光标对应3D空间的一条射线

2 MFD的仿真与实现

2.1 MFD模型创建

列车驾驶台采用传统建模工具 3DSMax。对于不同应用场合，3DSMax 软件可导出多种不同格式的三维模型，根据需要再从模型中提取有用的模型信息。

XNA 平台通常使用 .FBX 和 .X 格式的模型文件，专门负责模型文件导入和管理的内容管道，可以方便、快速地将这两种格式的三维模型加载到项目中。FBX 和 X 模型的数据组织结构为层级关系，可看作是树型结构，这种格式虽然结构复杂，对模型内部数据的操作非常方便，不但可以比较容易地获取子模型数据，甚至可以精确地定位到模型的任意三角面片，并对其进行各种操作。

在本文中，为了使用方便，使用了自定义的模型格式，相当于是抽取出 FBX 模型中有用的数据再重新组合而成的数据结构。

多功能显示屏模型作为高速列车模型的一个子模型，也叫基元（primitive），在建模时只需要由 2 个三角形拼接成一个显示平面，设置好贴图、纹理坐标，就可以正确地渲染在场景中。MFD 的纹理分为静态纹理和动态纹理两部分。显示器背景为静态纹理；指针、曲线、动态文本等变化的部分作为动态纹理，需要由外部数据实时驱动。建模时，需要把制作的 MFD 背景纹理作为贴图

绑定到 MFD 模型上。

2.2 MFD 动态参数驱动显示

XNA 应用程序模型设定了场景更新 (Update) 频率和绘制 (Draw) 频率。在执行每一帧的 Draw 方法之前, Update 线程将被调用, 它将更新时间、输入和其它一切随时间变化的场景事物。应用程序模型结构如图 2 所示。

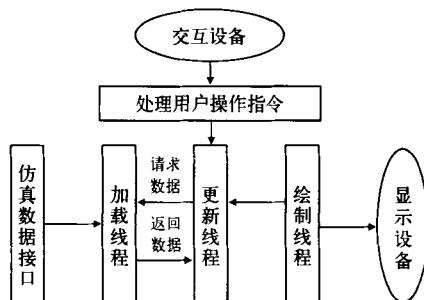


图2 应用程序模型结构

MFD 显示内容的更新也是在 Update 线程中完成, 实际上就是更新 MFD 所在平面模型所绑定的贴图纹理, 具体的实现流程如图 3 所示。

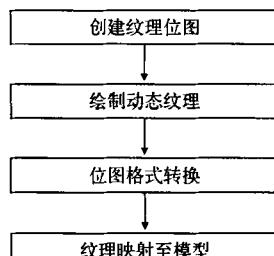


图3 MFD界面实现的流程图

(1) DMI 界面的动态纹理需要由外部数据实时驱动。为了减少后台生成静态纹理部分的工作量, 可事先利用 PhotoShop 等图像处理工具制作简单的背景纹理位图, 由程序直接读入内存。首先获取 MFD 初始的静态纹理图像, 将其拷贝至内存, 保存为位图图像。

(2) 从拷贝的初始位图创建一个图形设备接口 Graphics, 通过该绘图接口就可以在二维位图平面绘制 MFD 动态部分纹理, 此时需要从其他仿真数据接口读取驱动 MFD 的动态参数。

(3) XNA 中通常把需要渲染的图像经过内容管道的处理, 都转换为 Texture2D 格式。

(4) 把在后台生成的 Texture2D 图像通过纹理映射至 MFD 模型, 在执行 Draw 方法时, MFD 将按照新的纹理进行渲染。DMI 显示屏模型由 2 个三角形拼接成一个显示平面, 设置的初始纹理

及纹理坐标如图 4 所示。

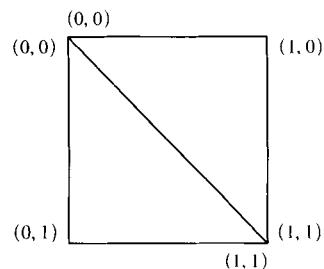


图4 显示屏纹理坐标示意图

2.3 MFD 的交互

通过按压 MFD 周边按键, MFD 可以显示不同的子页面。在 XNA 中, 实现虚拟 MFD 的交互有两种方法: (1) 键盘作为交互设备, 通过不同的组合键实现 MFD 的按键功能, 程序中不停的监听键盘输入并做出相应的响应, 这种方法势必占用大量键盘资源, 也不容易记忆, 不便于操作。(2) 鼠标作为交互设备, 在虚拟场景中, 直接使用鼠标点击按钮, 程序中监听鼠标动作并做出响应, 此方法更直观, 易于操作。

本文采用鼠标实现 MFD 页面的切换。当 MFD 按键模型接收到鼠标点击事件时, 立即切换对应的贴图纹理, 具体实现流程如图 5 所示。

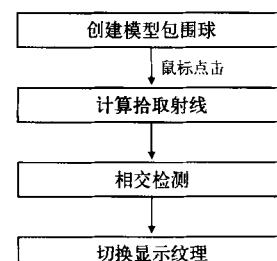


图5 MFD界面交互流程

2.3.1 创建按键模型的包围球集合

每个 3D 模型都拥有自己的最小包围球, 模型的所有顶点都位于该包围球内部。包围球有两个主要参数, 球半径和模型坐标系下的球心坐标。在列车驾驶台模拟仿真中, 多功能显示器按键模型的包围球是在载入系统时, 通过遍历模型的顶点, 生成各自的包围球, 并为每个按键包围球设定标识用于识别, 最后将按键模型的包围球存入一个包围球数组中。

2.3.2 鼠标点击, 计算拾取射线

系统获取到鼠标点击事件后, 记录下鼠标在屏幕上点击的位置, 并计算该点射向场景中的一条射线, 如图 1 所示。只要知道射线上的两个点,

就可以创建这条射线。第1个点是射线与近裁平面的交点，第2个点是与远裁平面的交点。

2.3.3 射线与模型相交检测

在模型众多的三维场景中，判断与拾取射线相交的模型需要遍历整个场景中的模型，这将带来繁重的计算量。为了简化起见，只遍历场景中的按键模型，因为按键模型之间没有遮挡，与拾取射线相交的按键一定是鼠标点击的对象。

当检测到鼠标所点击的按键模型后，再根据按键标识切换不同的显示纹理，采用纹理生成方法产生不同纹理显示的效果。

2.4 模拟运行效果

图6为本文实现的高速列车模拟运行效果图，其中MFD显示内容是实时变化的，将相机镜头向驾驶台拉近时，可以清晰地观看MFD显示情况，如图7(a)~7(c)所示。逼真地显示了高速列车驾驶台运行的实时信息和人机交互，提高了虚拟场景的真实感和沉浸感。



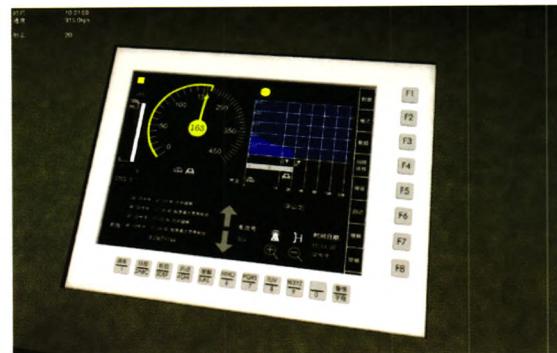
图6 高速列车驾驶台MFD效果图

3 结束语

本文详细介绍了基于XNA平台开发虚拟MFD的关键技术及实现方法，应用于高速列车模拟仿真中，交互性好、实时性高、性能稳定、界面美观，提高了仿真的逼真度，对相关视景仿真和虚拟运行系统的研究具有一定的参考价值。

参考文献：

- [1] 乐建炜, 向复生, 戴树岭. 虚拟座舱中多功能显示器的仿真 [J]. 系统仿真学报, 2005, 17 (12): 3046-3049.
- [2] 陈磊, 王海丽, 任萱. 面向对象的实时三维视景可视化实现 [J]. 计算机工程与应用, 2000, 36 (9): 40-42.
- [3] 刘东鑫, 欧阳中辉, 王东. 虚拟座舱中虚拟仪表的设计与实现 [J]. 计算机与现代化, 2009 (6): 44-47.



(a) 列控



(b) 车辆动力



(c) 调度信息

图7 高速列车驾驶台MFD细节图

- [4] 马俊, 朱衡军. 利用P-Buffer模拟虚拟场景的后视镜和鸟瞰图 [J]. 计算机辅助设计与图形学学报, 2006, 18 (5): 748-752.
- [5] 颜声远, 于晓洋, 张志俭, 等. 多仪表综合显示系统人机界面的虚拟评价研究 [J]. 系统仿真学报, 2007, 19 (12): 2720-2722, 2726.
- [6] 相海东, 沈为群, 宋子善. 航空通用虚拟仪表生成系统的设计与实现 [J]. 计算机仿真, 2003, 20 (1): 70-72.
- [7] Riemer Grootjans. XNA 3.0 Game Programming Recipes [M]. USA: Apress, 2009.
- [8] 段小娟, 张茂军, 李国辉, 等. 虚拟实景空间的纹理映射技术研究 [J]. 计算机工程, 2001, 27 (5): 17-18.
- [9] 袁慧, 曾黄麟. 有真实感图形的纹理映射算法 [J]. 计算机应用, 1998, 18 (10): 29-31.

责任编辑 徐侃春