

# 基于 OGRE 中间件的装备虚拟维修系统

杨作宾, 肖大海, 刘养科, 刘建强

(防空兵学院, 郑州 450052)

**摘要:** 为解决目前在部队和院校的装备和维修训练存在的训练装备数量少和参训人员多、训练效率低、成本高、局限性大以及易损坏装备等问题, 本文通过分析某装备虚拟维修系统的功能需求, 提出了一个基于 OGRE 的装备虚拟维修系统的设计与实现方法, 该系统以 OGRE 为中间件, 在整体软件系统结构上采用 MVC 模式, 大大简化了设计工作, 增强了系统的可靠性和可扩展性, 降低了软件开发的难度, 对武器装备的虚拟维修系统的设计与开发, 具有一定的指导意义和参考价值。

**关键词:** OGRE 中间件 虚拟维修系统

## 0 引言

目前, 在部队和院校的装备维修训练一般都依赖实装, 存在装备数量少和训练人员多、训练效率低、成本高、局限性大和易损坏装备等问题。因此有必要研究武器装备的虚拟维修系统, 这样不需要借助实装就可在计算机上进行正常的教学、训练, 费用低廉而且安全, 在身临其境的人机交互中, 原本复杂深奥的装置构造、原理、拆装、维修知识变得直观易懂, 这样就极大地提高了教学效率和参训人员的学习热情。

## 1 OGRE 简介

OGRE(Object-Oriented Graphics Rendering Engine)即: 面向对象图形渲染引擎, 是一个用 C++开发的面向场景、非常灵活的 3D 引擎, 诞生于 1999 年。它旨在让开发人员更容易、更直接地利用硬件加速的 3D 图形系统开发应用。这个类库隐藏了底层系统库 Direct3D 和 OpenGL 的所有细节, 并支持多种高级特性, 提供了一个基于现实世界对象和其他直观类的接口。OGRE 具有良好扩展性, 插件机制让用户方便开发专用模块并集成于 OGRE 中。

OGRE 通过面向对象的方法把一个基本几何体渲染到目标区域(如 CRT 或 LCD 显示设备的屏幕缓存)的接口。采用这种设计方法的好处就是将渲染几何体的具体操作从几何体级别的处理操作中抽离出来, 转而处理具体的场景和在场景中的物体。比如: 可活动的物体、静态物体组成的场景本身、灯光、摄像机等。只需简单的把物体放到场景之中, 它们会自动完成杂乱的几何渲染处理, 从而脱离对调用 API 的依赖。

OGRE 将需要在场景中进行渲染的物体称为实体, 一般实体就是一个 3D 模型, 而灯光、摄像机和粒子系统等则不是实体。在 OGRE 中不能直接将一个实体放入到场景中, 而是将实体与场景节点绑定在一起, 实体的位置信息就在场景节点进行记录, 当用户创建一个实体时, 它直到与一个场景节点绑定后才会被渲染。同样, 一个场景节点也不能单独的在屏幕上显示出来, 只有与一个实体绑定后才能在屏幕上显示。场景节点可以绑定多个实体。例如在屏幕上用户希望创建一个车辆, 并且希望这个车辆产生发光效果, 那么在实现时就需要创建一个场景节点, 然后再创建一个车辆的实体并与场景节点绑定在一起, 之后还需要创建一个光照模型也与这个场景节点绑定在一起。场景节点同样可以与其它场景节点绑定以描述更完整的对象。在场景中, 场景节点的位置总是与它的父节点相关。每一个场景管理器都包含一个根节点。

## 2 某型虚拟维修系统的功能需求

某型装备的虚拟维修系统的功能主要包括开关机训练操作和虚拟维修训练操作两个主要部分, 另外还实现了虚拟仪表的测试、视点的漫游等功能。

### 1. 开关机训练操作

针对该装备的开关机操作过程进行建模, 实现了虚拟开关机操作功能, 完整描述了开关机过程中装备

的各种现象变化过程。

## 2. 维修训练操作

针对维修案例专家故障信息，对维修过程进行建模，实现了装备的虚拟维修训练操作功能，包括对故障现象的描述、装备的拆卸，装配、故障点的测试等。

## 3. 虚拟测试

在虚拟维修训练过程中，通过对虚拟万用表、示波器等建模。实现了对故障点电阻、电压等故障信息的测试功能。

## 4. 视点漫游

采用了视点自由漫游方式，在虚拟维修过程中可方便的实现对所有故障点的观测。

## 5. 操作错误提示

若操作者在操作过程中执行了错误的操作，系统会给出“操作非法！”的提示，并给出正确的操作名称，提示操作者进行正确的操作。

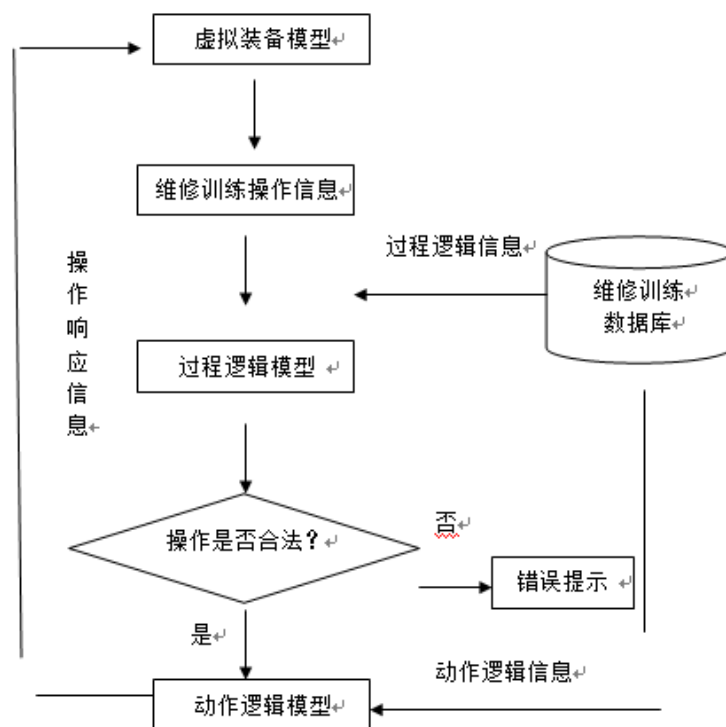


图1 虚拟维修训练系统工作流程

## 3 软件体系结构

基于对某型虚拟维修系统的功能需求分析可以得出，需要为虚拟维修系统设计一个具有高可扩展性和灵活的人机交互能力的体系结构框架。结合本虚拟维修训练系统的特性，我们选择了 MVC 模式作为开发虚拟维修训练系统的解决方案。然而，侧重于视景展示的仿真系统和传统的以数据处理为核心的程序有着显著的差异。前者 Model 组件的构造较为简单，多为 3D 模型的构建和封装，而针对模型进行展示的 View 组件和对模型进行控制的 Controller 组件的设计是一件非常困难的事情。如在本系统中，View 组件要解决 mesh 格式模型的读入与展示等问题，Controller 组件要解决如何对模型进行旋转、移动等问题。为此，我们采用第三方图形渲染引擎(OGRE)作为 View 组件和 Model 组件、Controller 组件和 Model 组件之间的中间件。如图 2 所示，这样不但使 3D 模型的读入、展示变得非常简单，对模型的控制异常灵活，还大大简化了 View 组件和 Controller 组件的设计工作，较强地增强了系统的可靠性和可扩展性。

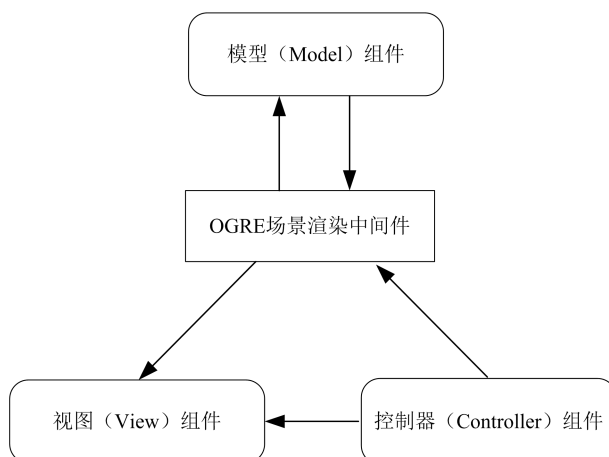


图 2 基于 OGRE 中间件和 MVC 模式的场景渲染框架

### 3.1 Model (模型) 组件

Model(模型)组件是应用程序的主体部分，它不仅包含了应用问题的核心数据、逻辑关系和计算功能，还封装了相应的数据并输出执行特定应用程序处理的过程。人机交互时，用户通过控制器调用这些过程。同时，模型也提供访问其内部数据成员的接口函数，这些数据由获得待显示数据的视图组件使用。

以装备拆装模块为例，模型主要是装备整体及其所包含的主要设备和元器件。首先使用 3D Max 进行模型的三维建模，接着使用专用插进将模型导出以 “.mesh” 为后缀名的模型文件，同时也将模型的材质数据导出。在此之后，将与本模块相关的仿真资源，包括 mesh 模型、材质数据、器件波形等，封装成 Model 组件。

### 3.2 View (视图) 组件

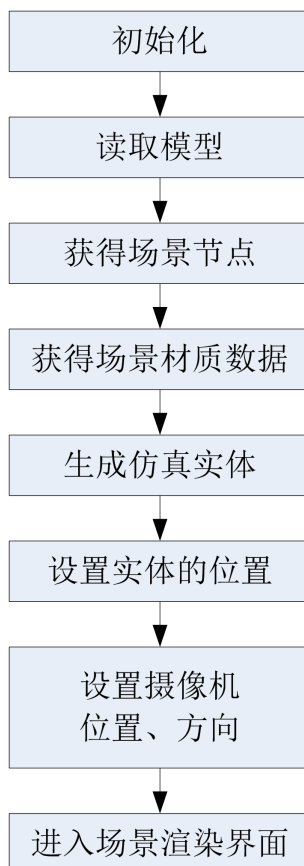


图 3 视图组件通过 OGRE 中间件调用模型组件流程图

View (视图) 组件主要用于向用户显示相关信息，包括模型数据及其逻辑关系、状态信息等。它从模

型组件获得数据, 经过控制器组件的变换, 然后进行信息的输出。不同的视图使用不同的方法呈现模型的信息。每个视图定义了一个被变更—传播机制激活的更新过程。更新过程被调用时, 视图就会恢复从模型组件读取的待显示的当前数据, 并将其显示在屏幕上。

由于本系统的视图组件主要用来完成模型的读取及其相关信息的展示。因此, 它主要是对微软 MFC 库中 View 类的封装, 而其复杂度却远低于传统的 View 组件。

视图组件通过 OGRE 调用模型组件以完成其功能。其基本流程如图 3 所示。

### 3.3 Controller (控制器) 组件

Controller (控制器) 组件主要用于完成人机交互, 它是用户与软件的交互接口。它用于解决用户与系统之间的交互。控制器组件接受输入, 并将鼠标的移动、鼠标按键的动作、键盘按键的动作等事件翻译成对模型或视图的服务请求, 系统捕获这些请求, 调用相应的功能模块, 对用户进行响应。

基于此结构的虚拟维修训练系统在实际工作时, 首先通过 OGRE 中间件对视图组件进行场景渲染, 然后系统通过 OGRE 中间件读取封装成模型组件的视景仿真资源并在场景中进行渲染。其中, 视景仿真资源包括三维模型、材质脚本数据、纹理图片等。在进行用户交互时分两种情况, 一种是需要对场景中的场景元素进行交互, 此时控制器组件调用 OGRE 中间件对模型数据进行相应操作, 然后由 OGRE 中间件对视图更新以完成交互过程; 另一种是不涉及场景元素的交互, 如更改系统的快捷键、工具栏的等动作, 直接由控制器组件对视图组件进行操作。因此, 基于该体系结构模式构建的装备虚拟维修训练系统具有较强的人机交互能力。

### 3.4 OGRE 中间件

OGRE 中间件主要用于解决视景场景中视景模型的读取、显示与操控等问题。其具体功能包括:

负责模型的读取。在本系统中, 主要用于对封装后的虚拟设备模型进行读取, 以供视图组件显示使用。

负责仿真场景的渲染。如场景的背景颜色、光源的类型及位置、光线强度等场景元素的渲染。

负责场景元素的控制。根据用户交互命令, 对场景元素进行相应的控制。如控制模型的移动、转动等。

负责场景的清理。在系统结束时, 负责场景模型的销毁等动作, 并进行内存管理, 回收内存。

## 4 结论

随着装备技术的发展和装备虚拟维修系统的复杂性进一步增强, 建立一个可扩展、可移植、通用性好的装备虚拟维修系统具有重要的现实意义。本文针对某型装备虚拟维修系统的功能需求, 提出了一种基于 OGRE 中间件的装备虚拟维修系统的体系架构, 该软件体系架构主要采用了 MVC 设计模式, 将系统中的模型、视图和控制器分离, 解决了系统的可视化渲染与数据模型的扩展以及系统移植的问题, 增强了系统的可靠性和可扩展性, 降低了软件开发的难度, 对武器装备的虚拟维修系统的设计与开发, 具有一定的指导意义和参考价值。

### 参考文献:

- [1]. E.Gamma, Richard Helm, Design Patterns Elements of Reusable Object—Oriented Software. Addison—Wesley, 1995, 10—20
- [2]. 杨宇航, 李志忠, 傅煜, 等. 基于虚拟现实的导弹维修训练系统. 兵工学报. 2006, 27(2): 297-300
- [3]. 邸锐. OGRE 3D 游戏框架指南[M]. 电子工业出版社, 2010.
- [4]. 邸锐, 李旭东译. Pro OGRE 3D Programming, 2006.