

# 智能弹药的实时仿真可视化系统设计

金龙<sup>1</sup>, 樊永生<sup>1</sup>, 余红英<sup>1</sup>, 杨臻<sup>2</sup>

(1. 中北大学 计算机与控制工程学院, 太原 030051

2. 中北大学 机电工程学院, 太原 030051)

**摘要:** 针对仿真可视化系统开发时间长、开发费用高的问题, 提出了一种利用 Unity3D 引擎进行实时仿真可视化系统开发的模块式设计方法; 首先, 对武器仿真可视化系统进行了分析, 提出了智能弹药实时仿真可视化系统的五大模块; 然后, 通过在 3ds Max 下建立智能弹药模型和 Unity3D 下的二次开发, 构建了智能弹药仿真可视化系统的仿真环境, 以 C 为编程工具, 设计了仿真驱动、串口数据通讯和仿真输出模块; 最终, 实现了智能弹药实时仿真可视化系统; 实验证明, 该系统可以有效地在监视器上实时显示出智能弹药的飞行姿态、飞行参数和飞行轨迹。

**关键词:** 智能弹药; 实时仿真; 可视化; 串口通讯

## Real-time and Visual Simulation System Design for Intelligent Munition

Jin Long<sup>1</sup>, Fan Yongsheng<sup>1</sup>, Yu Hongying<sup>1</sup>, Yang Zhen<sup>2</sup>

(1. School of Computer and Control Engineering, North University of China, Taiyuan 030051, China

2. College of Mechatronic Engineering, North University of China, Taiyuan 030051, China)

**Abstract:** In order to reduce the cost and time of the development of visual simulation system, a modular design method for real-time and visual simulation system was developed by using Unity3D engine. Firstly it analyzed the visual simulation system for weapon, proposed five major module of the real-time and visual simulation system design for intelligent munition. And then a simulation environment of real-time and visual simulation system for intelligent munition is established through build intelligent munition model on 3ds Max and further development of Unity3d, designed the simulation driver, serial data communication and simulation output with C. Finally realized the real-time and visual simulation system for intelligent munition. The experimental results show that this system can effectively display the flight attitude, flight parameters and the flight path in real-time simulation.

**Keywords:** intelligent munition; real-time simulation; visual simulation; serial communication

## 0 引言

随着计算机技术的飞速发展, 计算机仿真技术因其有效性、经济性、安全性、直观性等特点, 已广泛用于虚拟现实、模拟驾驶、场景再现和城市规划等领域, 成为社会各个应用领域中不可或缺的技术手段。

仿真技术在军事领域的应用, 近年来得到了快速发展<sup>[1]</sup>。采用 OpenGL 图形应用程序接口, 以 Visual C++ 作为编程工具, 实现了无人机三维可视化仿真的设计<sup>[2]</sup>。提出了一种利用 Simulink/Matlab 和 VR 技术进行可视化仿真的方法, 再现了末敏弹的减速减旋段运动过程<sup>[3]</sup>。设计了基于 MFC 的 Vega 程序设计, 实现了激光制导武器作战的可视化仿真。以上仿真可视化系统的开发都存在“建立仿真应用的时间太长、费用太高”的问题<sup>[4]</sup>。

为了解决以上仿真可视化系统开发中普遍存在的问题, 本文提出了一种利用 Unity3D 专业游戏引擎进行实时仿真可视化系统开发的新方法。该方法具有仿真效果好、可视化程度

高、开发周期短、开发效率高等优点, 有较高的工程应用价值。

## 1 实时仿真可视化系统的模块式设计方法

### 1.1 模块式设计方法的提出

在武器的系统设计和试验测试中, 需要全面、形象、直观地显示其形状、航迹、姿态以及动态飞行过程。模型的可视化不仅能实时的显示武器飞行的过程, 而且可以洞察变化的性态, 甚至可以达到虚拟现实的效果。

Unity3D 作为一个全面整合的专业游戏引擎, 拥有对 Direct 和 OpenGL 图形高度优化的渲染通道, 大大提高了仿真的效果; 其丰富的插件资源和革命性的多平台支持, 大大降低了开发的周期; 多种脚本编程方法的支持, 大大降低了开发的难度。因此, 选择通过 Unity3D 引擎进行仿真可视化系统的二次开发, 不失为了解决仿真开发周期长、费用高、难度大问题的一条途经, 同时, Unity3D 出色的粒子系统, 也必将给仿真效果带来质的提升。

仿真技术主要包括仿真环境制作和仿真驱动<sup>[5]</sup>, 仿真环境制作主要包括模型设计、场景构造、纹理设计制作、特效设计等, 要求构造逼真的三维模型和制作逼真的纹理和特效等; 仿真驱动主要包括场景驱动模型调动处理、分布交互、地形处理等, 要求做到逼真高速的再现仿真环境、实时响应交互操作等。

实时可视化仿真, 除需要实现仿真外, 还需要实现数据的

收稿日期: 2014-05-21; 修回日期: 2014-07-08;

基金项目: 山西省自然科学基金(2013011017-8)。

作者简介: 金龙(1988-), 男, 山东省滕州市人, 硕士研究生, 主要从事视觉仿真技术、嵌入式技术方向的研究。

樊永生(1965-), 男, 山西省运城人, 教授, 硕士研究生导师, 主要从事信号与信息处理、多媒体信息处理与计算机仿真、计算机控制技术方向的研究。

实时传输和可视化输出。基于此,提出了利用 Unity3D 引擎进行实时仿真可视化系统开发的模块式设计方法,即由模型的设计、环境的制作、仿真驱动、实时传输和仿真输出 5 个模块搭建仿真系统。

### 1.2 模块式设计方法的应用

在提出模块式实时仿真可视化系统设计方法的基础上,分析了智能弹药实时仿真可视化系统的构成,包括仿真模型的建立、仿真场景的制作、仿真驱动、串口数据通讯、仿真输出五大模块。

以下是各个模块的简单介绍:

1) 仿真模型的建立:在模型设计软件 3DS Max 中,完成智能弹药及飞机模型的设计。

2) 仿真环境的制作:包括界面的设计、场景的构造、纹理设计的制作、发射和命中的特效设计等,仿真环境的可视化程度直接影响仿真的逼真程度。

3) 仿真驱动:通过脚本控制仿真系统,通过数据交互进行驱动。在建立仿真环境的基础上,通过数据驱动,显示出整个仿真的动态过程,达到用户清楚的观察试验物动态性能的效果。

4) 串口数据通讯:要求能够实时的接收弹上数据,并按照通讯协议对数据进行解算,解算后的数据一方面用于仿真驱动,另一方面用于数据的存储。

5) 仿真输出:把众多的数据变化用直观的方式显示给用户,方便用户观察、理解以及做出判断,主要采用轨迹图和数据显示两种方式,此外还实现了仿真数据输出到 Excel 表格,达到存储仿真数据的功能。

## 2 智能弹药实时仿真可视化系统模块的实现

### 2.1 仿真模型的建立

本模块采用 3D Studio Max (3DS max) 绘制。由于 Unity3D 引擎支持多种模型的导入,而 3ds max 在模型的设计方面优势突出,且符合 Unity 引擎的导入要求,因此,选择在 3ds max 中建立模型,并以 .fbx 格式导出后供 Unity 引擎使用。根据仿真的需求和模型的分析,分别设计了飞机和智能弹药的六自由度仿真模型。其中智能弹药的模型如图 1 所示。

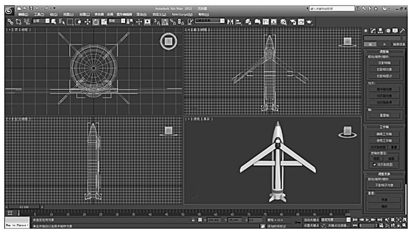


图 1 智能弹药的六自由度仿真模型

在仿真的开始阶段,飞机模型携带智能弹药模型飞行,二者构成整体接受飞行数据,在仿真空间内完成飞行动作。当飞机执行抛弹动作后,智能弹药的主体、舵与尾翼以整体的形式接收六自由度的飞行数据,由数据驱动智能弹药整体在仿真空间内飞行,同时,舵和尾翼又相对独立的接收各自的数据,实现张舵和展翼的效果,从而反映出模型在虚拟场景中的位置变化、姿态变换。

### 2.2 仿真环境的制作

仿真环境的制作,直接影响可视化仿真的逼真程度。通过 Unity3D 的内置 GUI 和插件 NGUI<sup>[6]</sup> 的使用,完成仿真系统所需的界面设计;通过插件地形系统,构造了平原、山地、沙漠 3 种场景供仿真使用;通过引擎粒子系统的特效设计<sup>[7]</sup> 的设计,构造出了云、尾烟、爆炸等粒子效果。同时,添加天空盒子,构造天空,使地形边缘更接近于现实。

通过以上设计的实现,在 Unity3D 中导入飞机和智能弹药的仿真模型,搭建出仿真可视化系统的仿真环境,仿真开始阶段,空中挂弹飞行效果如图 2 所示,图中箭头所指的圈出部分为智能弹药。



图 2 模型在仿真环境中的初始状态

### 2.3 仿真驱动

Unity3D 的脚本作为仿真驱动,既能实现仿真界面的布局与渲染,又能控制对象的状态和变换。前面构建的仿真环境,模型处于静态,仿真驱动的主要作用就是让模型动态的运动,达到仿真过程可视化的目的。在智能弹药实时仿真可视化系统中,考虑到串口数据通讯的实现,仿真驱动以 C 作为编程工具,在 Microsoft Visual Studio 下完成 Unity3D 的脚本编写。

仿真驱动脚本函数的编写,包括系统进入界面、仿真设置界面和仿真界面的脚本编写,前两个界面的驱动脚本编写在这里不再赘述,仅以仿真界面下的驱动脚本为例,介绍 C. NET 下的 Unity3D 驱动脚本编写的主要过程:

1) 添加命名空间,包括 System 命名空间(系统命名空间)、UnityEngine 命名空间(Unity 的命名空间)、System.IO.Ports 命名空间(包含控制串行端口的类)等;

2) 对脚本变量声明,对模型进行定义,声明串口通信端口等;

3) 添加 Start() 函数。添加 GameObject.Find() 函数唤醒模型、粒子效果,锁定视角位置,串口设置部分也在 Start() 函数中进行;

4) 添加 FixedUpdate() 函数。用于定时更新模型的位置和对串口通讯的线程检测。

5) 添加 OnGUI() 函数。通过 GUI.Label() 和 GUI.Button() 函数完成仿真界面的数据实时显示和仿真控制等。

5) 添加 OnCollisionEnter() 碰撞检测函数。用于检测碰撞完成命中目标爆炸的粒子效果。

此外,脚本中还包括 ReceiveData() 串口数据接收函数,DealData() 串口数据处理函数,Check() 提示函数,DrawLine() 轨迹函数,SaveData() 数据保存函数,这里不再一一说明。

### 2.4 串口数据通讯

本系仿真统中,先由主仿真机采用无线传输方式实时接收

弹上传下来的数据, 再通过 RS-422 标准实现主仿真机与实时显示计算机之间的数据通讯。通讯双方遵守以下约定:

1) 字符格式: 即一个字符包括多少数据位、停止位以及采用何种校验格式。

2) 数据传送速率即波特率。

主仿真机通过 COM 口按节拍 (20 ms) 发送智能弹药姿态角、坐标、舵偏角以及动作标志等信息。每帧数据格式如图 3 所示, 共 49 字节。通讯采用 19 200 bps, 1 个起始位, 8 个数据位, 1 个停止位, 无奇偶校验位, 数据传送周期为 20 ms。校验数据为从 3 到 48 所有字节累加之和。



图 3 数据帧格式

串口通信的编程, 通过 Serial Port 类实现, 考虑到 Unity 不支持 C 的命名空间, 加之 Unity 的底层 MonoDevelop 脚本编译器不支持事件触发方式<sup>[8]</sup>, 程序采用轮询的方式, 每隔 20ms 查看一次串口缓冲区, 并对符合要求的数据进行处理。串口数据通讯工作流程如图 4 所示。

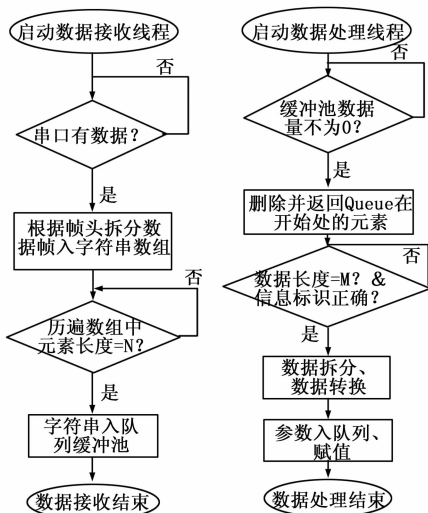


图 4 串口数据通讯线程工作流程图

通讯模块程序采用多线程机制, 实现数据的接收和处理; 通过对 C 中的字符串处理类<sup>[9]</sup>中函数的操作, 实现数据的处理; 通过操作先进先出 (FIFO) 方式的队列, 实现数据存储。具体步骤包括:

1) 声明实现串口读写基本功能的 SerialPort 对象成员 SDB\_Port, 声明线程 ReceivePort 和 DealPort 分别用于接收和处理串口数据, 声明字符型队列 QueueDataPool;

2) 在 Start () 函数中, 打开串口, 接收串口设置参数, 并对串口进行相应的设置, 通过 ReceivePort.Start () 和 DealPort.Start () 开启线程;

3) 添加 ReceiveData () 函数中, 设置串口缓存大小, 添加 SDB\_Port.Read () 查看串口缓存区, 添加 String 方法对数据进行判别, 符合要求的字符串通过 QueueDataPool.Enqueue () 函数, 以字符串形式入队列 QueueDataPool;

4) 添加 DealData () 函数, 通过 QueueDataPool.Dequeue ()

函数, 读取第一个进入的字符串, 通过 Regex.Split () 方法对字符串按照通讯协议进行拆分, 再通过移位、显式转换后将各种参数分别赋值给模型, 并存入各种队列中。

### 2.5 仿真输出

轨迹图的绘制, 使用底层的 GL 图像库, 添加 DrawLine 函数用于飞行轨迹线的绘制。具体步骤为: 先通过计算将数据坐标转化为屏幕坐标, 再调用 GL.Vertex 函数, 在屏幕上指定区域内绘制 2D 几何图形。由于使用 GL 图形库绘制的图形永远显示在屏幕中, 这样每帧进行图形的绘制就形成了轨迹线。

数据显示包括坐标、姿态、舵偏角、仿真时间, 通过脚本内调用 GUI 函数完成对数据显示界面的调用和数据的显示, 实现可视化仿真实时显示飞行数据的功能。

Unity3D 与数据库通信<sup>[10]</sup>的实现, 为 Unity3D 的数据存储问题提供了可行的实现方法, 考虑到本系统的应用背景, 设计了将实时仿真数据存储到 Excel 电子表格中的数据存储方式, 方便数据表的读取。

### 3 仿真结果

本文采用了 Unity3D 4.3.3 版本作为开发引擎, 结合 3ds Max 完成模型设计, 基于 Window XP、Microsoft Visual Studio 2010 环境完成了智能弹药的实时仿真可视化系统的开发。

本系统开发的智能弹药实时仿真可视化系统, 经过某实验室进行的实时仿真测试, 验证了实时仿真可视化系统各个功能模块的完善性, 同时验证了利用 Unity3D 进行实时仿真可视化系统开发的模块式设计方法的可行性。

在平原场景下, 进行智能弹药的可视化实时仿真中, 系统运行正常, 模型动态仿真效果出色, 其中抛弹过程如图 5 所示, 巡航过程如图 6 所示。界面左侧是实时仿真过程中的智能弹药, 其动作依靠串口实时传输的数据驱动完成; 界面右侧为仿真输出的数据显示和轨迹图, 上部是智能弹药飞行的各种实时数据显示区域, 下部是智能弹药的飞行轨迹示意图。

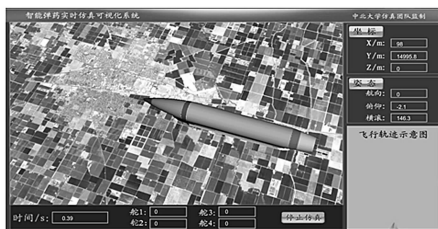


图 5 智能弹药仿真中的抛弹过程



图 6 智能弹药仿真中的巡航过程

从仿真效果可以看出, 该智能弹药实时仿真可视化系统满足仿真的需求, 能够实时的虚拟再现智能弹药从飞机携弹, 到抛弹、巡航, 再到俯冲下降, 直至命中目标的全过程, 多种形

式的仿真输出，满足相关人员的分析、研究需要。系统实时化、可视化程度较高，达到非常逼真的仿真效果，给人以强烈的视觉冲击。

### 4 结论

智能弹药的实时仿真可视化系统，以 Unity3D 作为开发平台，通过导入 3DS Max 中建立的仿真模型，共同搭建出智能弹药实时仿真可视化系统的仿真环境，以 C 为编程工具，实现了仿真驱动、串口实时通讯和仿真输出，生动、清晰、直观的显示了智能弹药试验的全过程。该系统可扩展性强，除可用于智能弹药的实时可视化仿真外，经过模型替换等修改，还可应用于其它导弹的实时仿真，为相关人员的分析、研究也提供极大的方便。利用 Unity3D 引擎进行实时仿真可视化系统开发的模块化设计方法，为仿真可视化系统的开发提供了一条新途径。

### 参考文献:

[1] 丁 斌, 祖家奎. 基于虚拟现实的无人机三维可视化仿真 [J]. 计算机测量与控制, 2007, 15 (12): 1769-1771.  
 [2] 邱 荷, 刘荣忠, 郭 锐, 张 俊. 末敏弹的可视化仿真研究 [J].

计算机仿真, 2013, 30 (1): 137-140.  
 [3] 朱思瑾, 张永志, 冷冕冕, 等. 激光制导武器作战的可视化仿真 [J]. 四川兵工学报, 2011, 32 (2): 29-32.  
 [4] 徐庚保, 曾莲芝. 计算机仿真系统述评 [J]. 计算机仿真, 2012, 29 (4): 1-5.  
 [5] 单家元, 孟秀云, 丁 艳, 等. 半实物仿真 [M]. 北京: 国防工业出版社, 2013.  
 [6] 宣雨松. Unity 3D 游戏开发 [M]. 北京: 人民邮电出版社, 2012.  
 [7] 熊 耀. 基于 Unity3D 粒子系统的三维影视特效开发研究 [J]. 软件导刊, 2012, 11 (11): 134-136.  
 [8] McClure W B, Blevins N, Croft IV J J, Jonathan Dick, Chris Hardy. Professional Android Programming with Mono for Android and .NET/C [M]. Indianapolis: John Wiley & Sons, Inc., 2012.  
 [9] Watson K, Nagel C, et al. Beginning Visual C 2010 [M]. Indianapolis: Wiley Publishing, Inc., 2010.  
 [10] 张利利, 李仁义, 李晓京, 等. Unity3D 与数据库通信方法的研究 [J]. 计算机技术与发展, 2014, 24 (3): 229-232.

(上接第 226 页)

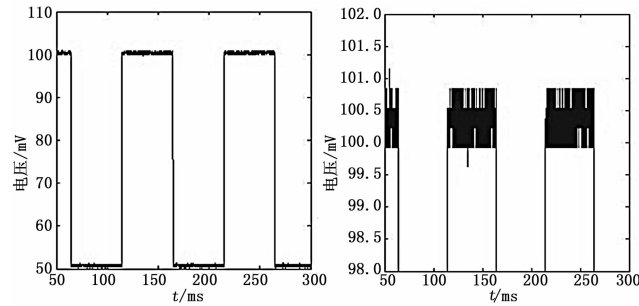


图 5 电压输出方波下所采集的数据

### 3.2 动态试验

将数据采集系统与计算机和前置放大器等其它外围设备连接，随后将待测生物分子加入检测池中，在计算机上设置输出 ±100 mV 的矩形波进行信号记录，试验结果 6 所示，其中图 (a) 表示 1.2 s 内所采集的电流信号变化，图 (b) 表示 200 ms 时刻到 220 ms 时刻内生物分子穿过纳米通道所采样的电流信号变化，图 (c) 表示施加的电压信号。

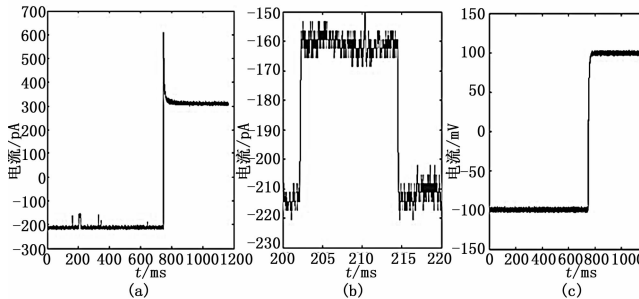


图 6 纳米通道单分子检测实验的电流采集信号

由图 6 中的 (a) 和 (c) 分析可知，电压输出信号和电流采集信号达到点对点对应；从图 (b) 中可以看出，获取的数据可以得到生物分子通过纳米孔的阻断电流的大小及阻断时间，完全满足实际应用的需求。

### 4 结束语

根据纳米通道单分子检测系统的要求，本文基于 FPGA 和 USB2.0，完成了数据采集系统的设计。FPGA 和 USB2.0 技术在该系统中的应用，使其具有高速、方便、易于升级等特点，该系统还采取了低噪声设计，满足微电流测量系统的要求，能够很好地应用于纳米通道单分子检测实验。此外，该数据采集系统还可以根据不同的需求移植到其他类似地应用中，具有广阔的应用前景。

### 参考文献:

[1] Cockroft S, Chu J, Amorin M. A single-molecule nanopore device detects DNA polymerase activity with single-nucleotide resolution [J]. Journal of the American Chemical Society, 2008, 130 (3): 818-820.  
 [2] 应伏伦, 张 星, 刘 钰等. 纳米通道单分子检测 P53 蛋白与 DNA 的弱相互作用 [J]. 化学学报, 2013 (71): 44-50.  
 [3] Analog Devices. AD7865 datasheet [EB/OL]. http://www.analog.com.  
 [4] Uram D, Ke K, Mayer M. Noise and bandwidth of current recordings from submicrometer pores and nanopores [J]. ACS Nano, 2008, 2 (5): 857-872.  
 [5] Rosenstein J K, Wanunu M, Merchant C, et al. Integrated nanopore sensing platform with sub-microsecond temporal resolution [J]. Nature Methods, 2012, 9: 487-492.  
 [6] Rosenstein J K, Ramakrishnan S, Roseman J, et al. Single Ion Channel Recordings with CMOS-Anchored Lipid Membranes [J]. Nano Lett, 2013, 13 (6): 2682-2686.  
 [7] 姚剑清. 运算放大器权威指南 [M]. 北京: 人民邮电出版社, 2010.  
 [8] 蒋 毅, 蒋 明. 并行 D/A 转换器件 AD5547 及其接口设计 [J]. 世界电子器件, 2007 (3): 45-47.  
 [9] 李英伟. USB2.0 原理与工程开发 [M]. 北京: 国防工业出版社, 2007.  
 [10] 向文芳, 邢 涛, 等. 高速数据采集系统的 USB 接口设计 [J]. 计算机测量与控制, 2011, 19 (10): 2567-2570.