

文章编号: 1671- 251X(2010)12- 0017- 04

基于 MFC 和 OpenGL 的喷砼机器人 三维仿真平台的设计

曹燕杰, 陈斌

(山东科技大学机器人研究中心, 山东 青岛 266510)

摘要: 以 PJR- 2X 喷砼机器人为例, 介绍了 1 种基于 VC+ + 的 MFC 和 OpenGL 的人机交互式三维仿真平台的设计方法。该方法采用 Solidworks 软件对该机器人建模, 并将三维模型保存为 .STL 文件, 然后导入 3DSMax 中确立纹理坐标, 输出的 3DS 文件再导入 Deep Exploration 中进行三维模型材质、纹理的二次处理, 最后通过 MFC 导入到以 OpenGL 为标准建立的工作场景中进行场景渲染, 从而得到逼真的机器人三维虚拟仿真平台。实际应用表明, 该仿真平台能精确表达机器人的正逆运动特性, 并能示教演示。

关键词: 喷砼机器人; 三维运动仿真; 三维模型; 人机交互; 仿真平台; MFC; OpenGL

中图分类号: TP242/391.9 **文献标识码:** B

Design of 3D Simulation Platform of Concrete Spraying Robot Based on MFC and OpenGL

CAO Yanjie, CHEN Bin

(Robot Research Center of Shandong University of Science and Technology, Qingdao 266510, China)

Abstract: Taking PJR- 2X concrete spraying robot as example, the paper introduced a design method of human machine interactive 3D simulation platform based on MFC of VC+ + and OpenGL. The method uses Solidworks software to build model of the robot, storages 3D model as .STL file which is introduced into 3DSMax to establish texture coordinates, and output 3DS file is introduced into Deep Exploration to make secondary processing of material and texture of 3D model. Finally, the file is introduced into working scene which is built by OpenGL to make scene rendering through MFC, so as to get lively 3D virtual simulation platform of the robot. The actual application showed that the simulation platform can express characteristics of observe and inverse motion of the robot accurately and can make demonstration.

Key words: concrete spraying robot, 3D motion simulation, 3D model, human machine interaction, simulation platform, MFC, OpenGL

0 引言

传统的工程机械开发常常要经历方案设计 \rightarrow 图样设计 \rightarrow 样机制造 \rightarrow 工艺设计(必要时伴以工艺试验) \rightarrow 批量生产这些步骤, 如果在其中任意一个步骤

中发现问题, 常常要回退到前面的某一步骤重新循环。此外, 实体样机的制作、工艺试验本身也要花费很多时间, 因而造成工程机械开发周期长的问题。仿真模拟技术的作用已经深入到工程机械设计的许多方面, 是实现制造技术的计算机化、信息化、智能化和自动化的关键技术与核心技术。应用模拟仿真技术可以大大缩短工程机械开发周期。

机器人三维运动仿真更是机器人各项仿真中的重要组成部分, 它对于验证机器人工作原理、工作空间及进行碰撞检测等都具有非常重要的指导意义。PJR- 2X 喷砼机器人是山东科技大学设计制造的我国第一台煤矿机器人, 它工作可靠性高、结构紧

收稿日期: 2010- 08- 05

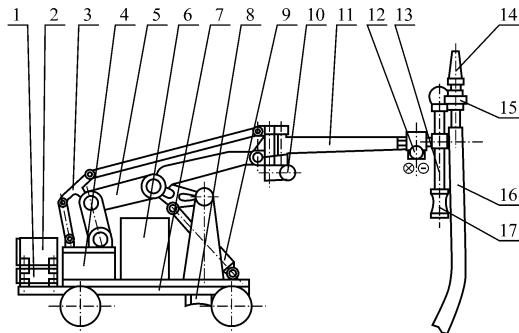
基金项目: 山东省研究生创新基金项目(SDYY06052), 山东省科技计划资助项目(2008GG30004010)

作者简介: 曹燕杰(1982-), 男, 山东聊城人, 山东科技大学机械电子工程专业在读硕士研究生, 研究方向为机器人技术、虚拟仿真设计。E-mail: caoyanjie0707@163.com

凑、操作简便, 可作为煤矿巷道或交通工程隧道的专用混凝土喷射设备。本文以该机器人为例, 简单介绍其基本结构, 着重介绍以 VC++ MFC (Microsoft Foundation Classes) 程序和 OpenGL (Open Graphics Library) 相结合设计实现人机交互式三维仿真平台的一种方法。

1 喷砼机器人工作原理

喷砼机器人有 6 个自由度, 即大臂俯仰、小臂摆动、手腕转动、喷枪摆动、喷枪转动和整车行走, 其机械结构如图 1 所示。大臂俯仰油缸 9 驱动大臂 5 做升降运动, 使大臂末端沿铅垂线运动, 从而保证小臂平行四连杆 11 做垂直起落运动; 液压马达 10 驱动小臂平行四连杆 11 做水平摆动, 使喷枪头 14 相对巷道壁面的姿态保持不变; 液压马达 12 驱动喷枪杆 13 做 $\pm 100^\circ$ 转动; 液压马达 15 驱动喷枪头 14 做旋转运动, 以达到喷浆工艺要求; 液压马达 17 驱动喷枪头 14 做 $\pm 50^\circ$ 的调姿摆动; 液压马达 8 驱动车体做前后往复运动^[1]。



1- 控制器; 2- 电源箱; 3- 平衡拉杆; 4- 油箱; 5- 大臂;
6- 液压控制箱; 7- 车盘; 8- 液压马达(驱动车体);
9- 大臂俯仰油缸; 10- 液压马达(小臂摆动);
11- 小臂平行四连杆; 12- 液压马达(喷枪杆旋转);
13- 喷枪杆; 14- 喷枪头; 15- 液压马达(喷枪头旋转);
16- 送料管; 17- 液压马达(喷枪头摆动)

图 1 喷砼机器人机械结构

2 三维仿真平台设计思路

VC++ 与 OpenGL 的结合在建立三维仿真平台方面具有独特的优越性: VC++ 的对话框能简便地编写出处理能力强大、操作简单的人机交互式平台; OpenGL 是 SGI 公司开发的一种跨平台的国际计算机标准图形应用程序接口, 可在不同的操作系统下运行工作, 具有良好的可移植性、强大的图形处理能力, 并且采用双缓存技术, 即前台缓存和后台缓存, 可以实现平滑的动画效果, 适于再现机器人的

机械结构并进行运动仿真^[2]; VC++ 与 OpenGL 之间接口方便、运行精确, 只要包含相应的库文件和必要的程序接口, 就可以利用 OpenGL 在 Windows 窗口界面中进行设计。

VC++ 中的辅助软件开发包 MFC 是一个建立在 Windows API 之上的 C++ 类库 (C++ Class Library), 这使得开发 Windows 应用程序变得简单而高效; 它提供复杂的资源编辑器、窗口程序的界面和基本输入输出, 可以编辑对话框、菜单、工具栏、图像和其它许多 Windows 应用程序的组成元素。Microsoft 公司自 Windows NT 和 Windows 95 开始提供对 OpenGL 的支持, 因此, 可以利用 MFC 为 OpenGL 运行环境进行软件界面 (菜单、对话框) 和人机交互的设计和编程^[2-5]。

但是, OpenGL 只能提供基本几何元素的造型函数, 没有提供三维造型的复杂命令, 使得 OpenGL 在建立复杂模型时具有一定的困难。Solidworks 软件能方便地进行三维实体建模, 但在物体渲染和程序可移植性上不尽如人意。

基于此, 在对 PJR 2X 喷砼机器人进行三维建模时, 采用 Solidworks 软件将三维模型保存为 .STL 文件导入 3DSMax 中确立纹理坐标, 输出的 3DS 文件再导入 Deep Exploration 中进行三维模型材质、纹理的二次处理, 最后通过 MFC 导入到以 OpenGL 为标准建立的工作场景中进行场景渲染, 得到逼真的三维虚拟仿真平台。该机器人摆柄体部分绘制处理过程如图 2 所示。

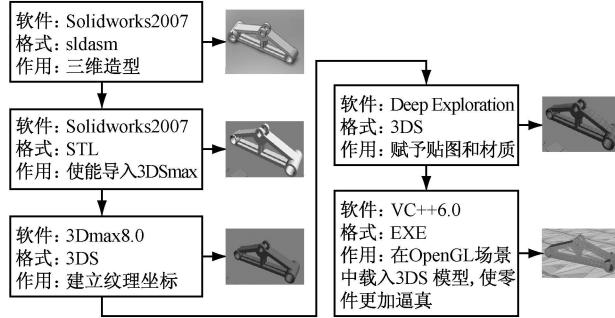


图 2 喷砼机器人摆柄体部分绘制处理过程

3 仿真平台的程序实现

该机器人机械结构最重要的部分是大臂, 它在垂直平面上做升降运动, 初始位姿如图 3 所示。

3.1 重要参数的确定

由图 3 可知, 当液压油缸和活塞杆总长度为 L 、摆柄边长 $BC = 247.542$ mm 时有

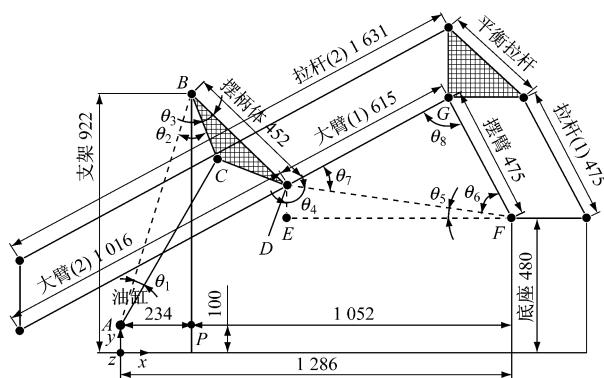


图 3 喷砂机器人大臂结构

$$\theta_1 = \arccos \frac{(\sqrt{234^2 + 822^2})^2 + L^2 - 247.542^2}{2L \sqrt{234^2 + 822^2}}$$

$$\theta_2 = \arccos \frac{(\sqrt{234^2 + 822^2})^2 + 247.542^2 - L^2}{2 \sqrt{234^2 + 822^2} \times 247.542}$$

L 的初始值为 670 mm, 则 θ_1 的初始值为 12.507° , θ_2 的初始值为 35.885° , θ_3 为 24.080° , $\angle ABP = 15.89^\circ$ 。摆柄体与大臂交点 D 的坐标为 (a, b) , 其中:

$$a |_{\text{mm}} = 234 + 452 \sin(24.080^\circ + 35.885^\circ - 15.89^\circ) = 548.409$$

$$b |_{\text{mm}} = 922 - 452 \cos(24.080^\circ + 35.885^\circ - 15.89^\circ) = 597.270$$

则

$$DE |_{\text{mm}} = 597.270 - 480 = 117.270$$

$$EF |_{\text{mm}} = 1286 - 548.409 = 737.591$$

$$DF |_{\text{mm}} = \sqrt{DE^2 + EF^2} = 746.856$$

$$\theta_4 |_{(\circ)} = \arcsin \frac{EF}{DF} = 80.966$$

$$\theta_5 |_{(\circ)} = 90 - 80.966 = 9.034$$

已知 $DG = 615$ mm, $FG = 475$ mm, 则

$$\theta_6 |_{(\circ)} = \arccos \frac{746.855^2 + 475^2 - 615^2}{2 \times 746.855 \times 475} = 55.174$$

$$\theta_7 |_{(\circ)} = \arccos \frac{746.855^2 + 615^2 - 475^2}{2 \times 746.855 \times 615} = 39.347$$

$$\theta_8 |_{(\circ)} = 180 - \theta_6 - \theta_7 = 85.479$$

摆臂与大臂(1)的夹角为 θ_8 , 液压油缸和活塞杆总长度为 $(670 + X)$ mm, X 为活塞的行程, 修改该变量即可控制大臂俯仰^[1]。

3.2 模型重构的 VC 程序

模型重构的 VC 程序部分代码如下:

```
# include<windows.h>; // Windows 的头文件
# include<glew.h>; // 包含最新的 gl.h, glu.h 库
# include<glut.h>; // 包含 OpenGL 实用库
...
g_eye[0] = MAP; // 初始观测点
```

```
g_eye[2] = 0; // 终止观测点
g_Angle = 180; // 场景方位角
g_elev = 0; // 场景俯仰角
g_text = gluNewQuadric(); // 处理纹理
LoadT8("地面.bmp", g_cactus[0]); // 载入地面贴图
m_3ds = new CLoad3DS(); // 创建 3DS 类
load3dobj("Model/", "摆臂.3DS", 1); // 导入摆臂的 3ds 文件
load3dobj("Model/", "大臂.3DS", 2); // 导入大臂的 3ds 文件
load3dobj("Model/", "平衡拉杆.3DS", 3); // 导入平衡拉杆的 3ds 文件
...
glBegin(g_ViewMode); // 开始以 g_ViewMode 模式绘制
for(int j = 0; j < pObject->numOfFaces; j++) // 扫描所有的面
{
    for(int tex = 0; tex < 3; tex++) // 扫描所有三角形的点
    {
        int index = pObject->pFaces[j].vertIndex[tex];
        // 获得面对每个点的索引
        glNormal3f(pObject->pNormals[index].x, pObject->pNormals[index].y, pObject->pNormals[index].z); // 给出法向量
        if(pObject->bHasTexture); // 如果对象具有纹理
        { if(pObject->pTexVerts); // 确定是否有 UVW 纹理坐标
            glTexCoord2f(pObject->pTexVerts[index].x, pObject->pTexVerts[index].y);
        }
    }
    glEnd(); // 绘制结束
}
}

以大臂结构的点 F 为坐标原点, 将摆臂、大臂、平衡拉杆依次安装的程序代码如下[6]:
// 定义初始参数:
float AB = 854.658f;
float AC = 670.0f; // 变量 X 为活塞的行程
float BC = 247.542f;
float DF = sqrt(DE * DE + EF * EF);
float DG = 615.0f;
float FG = 475.0f;
float X1 = (AB * AB + AC * AC - BC * BC) / (2 * AB * AC);
float X2 = (AB * AB + BC * BC - AC * AC) / (2 * AB * BC);
float X6 = (DF * DF + FG * FG - DG * DG) / (2 * DF * FG);
float X7 = (DF * DF + DG * DG - FG * FG) / (2 * DF * DG);
// 求 θi (i = 1, 2, 3)
θ1 = arccos(X1) * 180 / 3.1416; // 求 θ1
θ2 = arccos(X2) * 180 / 3.1416; // 求 θ2
θ3 = 24.080; // 求 θ3
// 摆柄体与大臂交点 D 的坐标为 (a, b)
a = 234.0f + 452.0f * sin[(θ2 + θ3 - 15.89) * 180 / 3.1416];
b = 922.0f - 452.0f * cos[(θ2 + θ3 - 15.89) * 180 / 3.1416];
DE = b - 480.0f;
EF = 1286.0f - a;
θ4 = arcsin(EF / DF) * 180 / 3.1416; // 求 θ4
θ5 = 90 - θ4; // 求 θ5
θ6 = arccos(X6) * 180 / 3.1416; // 求 θ6
θ7 = arccos(X7) * 180 / 3.1416; // 求 θ7
```

```

 $\theta_7 = \arccos(X7) * 180 / 3.1416; // 求 \theta_7$ 
// 摆臂部分
glRotatef(-\theta_5 - \theta_6, 1.0, 0.0, 0.0); // 部件绕 X 轴逆时针旋转
(\theta_5 + \theta_6)°
m_drawGL->Scene(1, 0, 0, 0, 0, 0.005f); // 1 为摆臂的代号
// 显示摆臂
glTranslatef(0.0f, 0.0f, 0.27f);
// 坐标系原点沿 Z 轴正方向移动 54 mm, 使大臂转轴中心与摆臂末端转轴中心重合
glRotatef(-\theta_7 - \theta_6, 1.0, 0.0, 0.0); // 摆臂绕 X 轴逆时针旋转
(\theta_6 + \theta_7)°
glTranslatef(0.0f, 0.0f, -3.075f); // 坐标系原点沿 Z 轴负方向移动 615 mm, 到达点 G
m_drawGL->Scene(2, 0, 0, 0, 0, 0.005f); // 2 为大臂的代号
// 显示大臂
glTranslatef(0.0f, 0.0f, 3.075f); // 坐标系原点沿 Z 轴正方向移动 615 mm, 回到点 F
glRotatef(90 - \theta_4 - \theta_7, 1.0, 0.0, 0.0); // 大臂绕 X 轴顺时针旋转

```

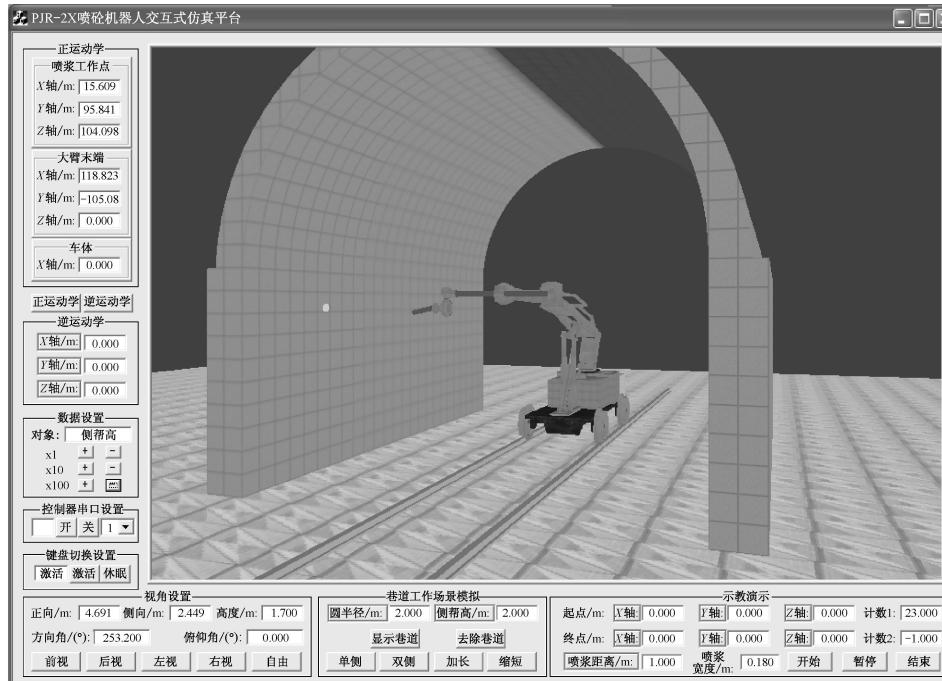


图 4 基于 MFC 和 OpenGL 的喷枪机器人交互式三维仿真平台

5 结语

实际应用表明, 基于 MFC 和 OpenGL 的喷枪机器人交互式三维仿真平台能精确表达机器人的正逆运动特性, 并能进行示教演示; 在选择最佳的机构运行方案、调整机器人设计、制造过程中的不合理问题、避免制造物理样机的浪费、缩短制造周期等方面起到了重要作用。

参考文献:

- [1] 樊炳辉, 苏学成, 李贻斌, 等. 喷浆机器人大臂结构的计算机仿真设计[J]. 矿山机械, 1994(12): 26-29.

(90 - \theta_4 - \theta_7)°, 使平衡拉杆始终处于垂直状态

m_drawGL->Scene(3, 0, 0, 0, 0, 0.005f); // 3 为平衡拉杆代号
// 显示平衡拉杆

其它零部件的导入方法与上面类似, 不再赘述, 导入完毕就建立了虚拟工作场景中的虚拟样机。

4 仿真平台界面

基于 MFC 和 OpenGL 的喷枪机器人交互式三维仿真平台带有正运动学区、逆运动学区、演示区、视角切换区和场景模拟区等操作模块, 其界面如图 4 所示, 通过正、逆运动学区能实时准确地显示工作点的运动学参数, 如果在演示区输入数值, 机器人的各关节就会摆出相应的姿态, 从而达到喷枪的工作位置。另外, 该平台不仅能由键盘控制, 还设计了串口通信, 即可用外部控制器控制。

- [2] 汪日伟, 韩其睿, 吴晓青. 如何在 MFC 中利用 OpenGL 实现三维图形的绘制[J]. 天津工业大学学报, 2004, 23(3): 65-67.
- [3] 向世明. OpenGL 编程与实例[M]. 北京: 电子工业出版社, 1999.
- [4] SHREINER D, MASON W, JACKIE N, et al. OpenGL 编程指南[M]. 徐波, 译. 北京: 机械工业出版社, 2009: 63-285.
- [5] 和平鸽工作室. OpenGL 高级编程与可视化系统开发(高级编程篇)[M]. 北京: 中国水利水电出版社, 2003.
- [6] 陈斌, 樊炳辉, 王传江, 等. 基于 OpenGL 的喷浆机器人虚拟样机设计与实现[J]. 矿山机械, 2010(9): 36-38.