

嵌入式三维地形可视化技术的研究与实现

武志强¹, 康利刚²

(1. 解放军信息工程大学测绘学院, 郑州 450052; 2. 解放军信息工程大学电子技术学院, 郑州 450004)

摘要: 介绍嵌入式系统和 OpenGL ES 的基本概念以及以 WINCE 为平台进行 OpenGL ES 编程的方法。对嵌入式地形可视化进行总体设计, 并结合 OpenGL ES 图形库, 以 Embedded Visual C++ 4.0(EVC)为应用程序开发平台模拟三维地形显示, 实现嵌入式系统上三维地形的快速建模以及动态显示, 通过漫游功能、纹理映射和立方体天空盒增强地形的三维真实感。

关键词: OpenGL ES 图形库; 三维地形; 可视化

Research and Implementation of Embedded 3D Terrain Visualization Technology

WU Zhi-qiang¹, KANG Li-gang²

(1. Institute of Surveying and Mapping, PLA Information Engineering University, Zhengzhou 450052;

2. Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004)

【Abstract】 The paper introduces the embedded system, the OpenGL ES basic concept and OpenGL ES programming method based on WinCE operating system. A terrain visualization system is designed, and the system is achieved by using OpenGL ES and Embedded Visual C++ 4.0(EVC). The system is a 3D terrain visualization in embedded system. The system carries out modeling and the dynamic manifestation of terrain, and simultaneously through the roaming function, the texture mapping and the cube sky-box realize the 3D terrain with reality.

【Key words】 OpenGL ES; 3D terrain; visualization

随着微电子技术、光电子技术的发展, 计算机发展已经进入移动时代, 因此嵌入式系统的应用领域越来越广泛并且已经成为科学技术发展的新热点。嵌入式三维地理信息系统是运行在嵌入式硬件平台上的地理信息系统, 而地形的三维可视化是三维地形信息系统中最重要的部分。OpenGL Embedded Subset(OpenGL ES)^[1]的出现为在嵌入式系统上绘制具有真实感的三维地形提供了极大的便利。

1 嵌入式系统与 OpenGL ES

1.1 嵌入式系统

嵌入式系统(embedded system)是指以应用为中心, 计算机技术为基础, 并且软硬件可裁剪, 适用于应用系统对功能、可靠性、成本、体积、功耗等有严格要求的专用计算机系统, 主要由硬件环境、嵌入式操作系统及应用软件系统等组成^[2]。嵌入式系统是将先进的计算机技术、半导体技术和电子技术和各个行业的具体应用相结合的产物。

嵌入式操作系统是一种实时、支持嵌入式系统应用的操作软件, 核心通常要求很小, 因为硬件的 ROM 容量有限。一般情况下, 它可以分成 2 类: 一类是面向控制、通信等领域的实时操作系统; 另一类是面向消费电子产品的非实时操作系统, 这类产品包括个人数据助理(PDA)、移动电话、机顶盒等, 如 Windows CE、Palm OS、嵌入式 Linux 和 EPOC 等。本文以 Windows CE 为基础, 结合嵌入式系统 3D 图形绘制编程接口 OpenGL ES 进行研究。

1.2 OpenGL ES

OpenGL ES 是为嵌入式系统而开发的 3D 图形绘制编程接口, 是在 OpenGL 上发展出来的, 针对嵌入式系统所制定

的 3D 绘图 API, 能够使 3D 绘图在不同的移动设备或是嵌入式系统上方便地使用。

OpenGL ES 主要功能包括: 模型绘制, 模型变换, 颜色模式, 光照和材质设置, 纹理映射, 双缓冲动画等^[3]。

2 地形三维可视化基础

2.1 数字高程模型

地形三维可视化过程中最重要的工作就是获取地形的三维坐标数据信息, 并通过透视投影变换, 将其转换为屏幕二维坐标, 进而显示在屏幕上。目前描述地形一般通过数字高程模型(Digital Elevation Model, DEM)来实现。从数学的角度, 高程模型是高程 Z 关于平面坐标 X, Y 2 个自变量的连续函数, 数字高程模型只是它的一个有限的离散表示。DEM 网格是最常见的数字地形模型形式, 其数据组织可分为 2 种类型: 规则格网(GRID)和不规则三角网(TIN)。GRID 型 DEM 由于是按固定采样间隔、按矩阵排列的数据形式, 其主体数据只需记录或存储每个节点的高程即可, 存储结构简单, 数据处理速度快, 因此 GRID 数据结构适用于嵌入式系统。而不规则三角网 TIN 不仅要存储每个节点的高程, 还要存储其平面的坐标、网点的拓扑关系、三角形及邻接三角形等信息, 数据结构比较复杂, 处理速度比较慢, 不适合在嵌入式系统中应用。

2.2 地形三维可视化及实时显示

在这个过程中, 首先需要对地形数据进行投影变换, 为

作者简介: 武志强(1971 -), 男, 博士研究生, 主研方向: 虚拟地形环境, 战场可视化; 康利刚, 硕士研究生

收稿日期: 2007-05-10 **E-mail:** klgkg01@163.com

了逼真地反映地形表面明暗和颜色的变换,需要设置场景的光源位置和光照方向,并且计算节点或者表面的法向,据此来获得屏幕上每个像素的颜色和灰度。此外,还需要对场景进行消隐和裁剪。消隐是为了能够真实表现物体的深度关系;裁剪则是让当前视窗之外的物体不予显示,提高程序的运行效率。

在嵌入式地形三维实时显示过程中,为了进一步提高运行效率,往往需要对 DEM 数据进行一定的简化处理,通常采用一种层次细节(Level of Details, LoD)的简化方法。

3 具体实现

3.1 开发环境

开发平台选择 Windows XP 操作系统下的嵌入式集成开发工具 EVC 软件,它包含了在不同硬件平台下使用 SDK 和仿真器(PC 环境下模拟掌上电脑硬件环境的软件),具体设置如下:

- (1)Microsoft ActiveSync;
- (2)Embedded Visual C++ 4.0;
- (3)Embedded Visual C++ 4.0 Service Pack 4;
- (4)Microsoft Pocket PC 2003 SDK;
- (5)Windows Mobile 2003 second edition emulator images for pocket PC;
- (6)OpenGL ES Implementation.

Microsoft ActiveSync 用来实现 PC 机和 Windows Mobile 设备的连接,使 EVC++4.0 编写的应用程序可以同步地在 Windows Mobile 设备上实现。EVC 和 Service Pack 4 是 Microsoft 公司推出的 WINCE 程序可视化开发工具。Microsoft Pocket PC 2003 SDK 是 PC 机上开发 WINCE 应用程序的模拟环境,Windows Mobile 2003 second edition emulator images for pocket PC 为模拟环境的中文环境。

OpenGL ES Implementation 为嵌入式系统上开发三维地形的图形库,需要下载安装,下载解压后为 bin 文件夹,其安装步骤如下:

- (1)找到模拟器安装目录,一般为 \\Windows CE Tools\wce420\POCKET PC 2003\(\简称/sdk/);
- (2)将/bin/include 目录下的文件拷贝到 /sdk/include/Armv4 和 sdk/include/Emulator 2 个目录下;
- (3)将/bin/emu/debug/目录下的除 libGLES_CM.dll 的其他文件拷贝到/sdk/lib/emulator 目录下;
- (4)将 bin/arm/release 目录下的除 libGLES_CM.dll 的其他文件拷贝到/sdk/lib/Armv4 目录下;
- (5)需要将/bin/emu/debug/目录下的 libGLES_CM.dll 拷贝到模拟器的/Windows 目录下;
- (6)第(4)步只是在模拟器的 debug 环境下,若要在真机上需要装/bin/arm/release/目录下的 libGLES_CM_CM.dll 拷贝到真机上。

3.2 总体设计

使用 WINCE 应用程序开发工具 EVC++4.0 结合 OpenGL ES 嵌入式图形开发库,开发移动平台三维地形的具体设计流程如图 1 所示,其设计思路如下:

- (1)调入(.DEM)文件格式的网格数据;
- (2)利用网格数据生成三维地形模型,并对其进行简化,简化的同时可以实时显示简化后的地形模型,且自动存储简化后的数据文件,以便以后调用;
- (3)对其设置网格、分层设色、投影变换和视口变换进行

三维地形显示;

(4)采用三维图形处理技术对其设置消隐、光照、纹理贴图、天空盒以达到真实感效果;

(5)利用移动平台上方向键设置旋转、前进、后退等交互功能,或者通过对路径的设置,从而使用户能够在场景中随意漫游,在任意位置、各个角度观察三维地形。

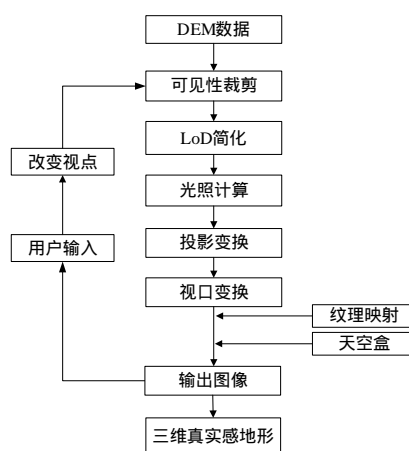


图 1 嵌入式地形三维可视化框图

3.3 地形建模

本文利用 LoD 简化在嵌入式系统上实现地形建模,当观察视点的位置和视向发生变化时,根据变化后的位置和视向对每个地形节点进行可见性判断和绘制分辨率的计算,让进入新的视线范围内的节点地形参与地形绘制,同时让那些不在新的视线范围内的地形节点不再参与三维地形的绘制,这样当观察视点发生变化后,视线内的地形也实时发生变化。

此外,要三维地形适当地显示出来,还需进行投影变换和视口变换。投影变换是生成三维模型的重要基础,一般分为透视投影和正射投影,为增加模型的真实感,采用了透视投影方式。视口是指显示屏幕中的矩形区域,视口变换的目的是将三维空间坐标映射为显示屏幕上的二维平面坐标^[4]。图 2 显示了最后输入的三维地形模型。

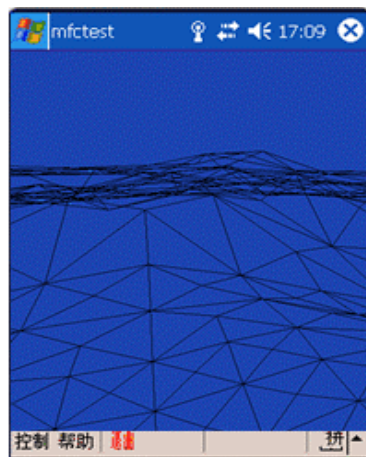


图 2 地形三维可视化模型

3.4 三维地形的真实感

对最后输入的三维地形进行纹理映射和添加天空盒是生成逼真三维地形的必要途径,应用 OpenGL ES 提供的纹理映射技术可把从真实世界中拍摄到的地形的表面细节,采用贴图的方式贴到三维地形的表面,从而使渲染后得到的三维地形模型与真实世界中的地形惟妙惟肖,作为贴图的纹理可以

是正射影像或者数字摄影相片。用 OpenGL ES 进行纹理映射,首先需要定义纹理内容,包括纹理数据指针、纹理尺寸、类别(灰度或彩色)等,然后设置纹理映射到三维表面的方式,如纹理滤波、重复、颜色融合等;最后定义三维模型顶点的纹理坐标与几何坐标,绘制场景。

天空盒的生成采用了立方体天空盒,通过在立方体盒子的 5 个不同面上贴上不同的天空纹理来达到逼真的模拟效果。用于天空背景的图片有特殊的要求。四面图的边与顶图的边相连,四面图前后相连。图 3 显示了纹理映射和天空盒的使用使三维地形更具有真实感。



图 3 三维真实感地形

3.5 漫游功能

三维地形的漫游是通过操作方向键模拟在场景中的行走。实现第一人称的漫游基本上有 2 种方法:移动场景和移动眼坐标(视点坐标)。移动场景相对来说比较简单,只要使用 OpenGL ES 提供的 `glTranslatef` 与 `glRotatef` 方法相配合即可,但一般只在简单场景和单角色的情况下使用,而且角色的各种计算(如实时坐标、碰撞)不好实现,所以在这里没有使用,而是采取移动观察视点坐标的方法。

移动观察视点坐标的方法非常灵活,它对场景和角色的状态未做任何操作,一般只要设置成跟随主角色移动旋转即可实现第一人称视角视觉效果。通过使用 OpenGL ES 提供的 `glLookAt` 方法来实现此功能。其实现代码如下:

首先定义宏:

```
#define KEY_DOWN(vk_code)((GetAsyncKeyState(vk_code) & 0x8000) ? 1 : 0)
```

再加入以下代码:

```
float speed=0.2f;
//向左向右转
```

(上接第 237 页)

- [3] Tourapis H Y C, Tourapis A M. Fast Motion Estimation Within the H.264 Code[C]//Proc. of International Conference on Multimedia and Expo. Maryland, USA: [s. n.], 2003: 517-520.
- [4] Lu Xiao'an, Tourapis A M, Yin Peng, et al. Fast Mode Decision and Motion Estimation for H.264 with a Focus on MPEG-2/H.264 Transcoding[C]//Proc. of IEEE International Symposium on Circuits and Systems Proceedings. Maryland, USA: [s. n.], 2005: 1246-1249.
- [5] Tang Qiang, Ward R K, Nasiopoulos P. An Efficient MPEG2 to

```
if (KEY_DOWN(VK_SHIFT)) speed =speed*4;
if (KEY_DOWN(VK_LEFT)) g_Angle-=speed*2;
if (KEY_DOWN(VK_RIGHT)) g_Angle+=speed*2;
rad_xz = float (3.13149* g_Angle/180.0f);
/*角度转化为弧度值,转化公式为:rad=ang*PI/180.
```

其中 `rad` 表示弧度值, `ang` 表示角度值。 `g_Angle` 即为观察视点的视线角度。*/

```
if (KEY_DOWN(VK_UP))
{ Io.z+=sin(rad_xz)*speed;
  Io.x+=cos(rad_xz)*speed;
} //观察视点前移
if (KEY_DOWN(VK_DOWN))
{ Io.z-=sin(rad_xz)*speed;
  Io.x-=cos(rad_xz)*speed;
} //观察视点后移
//边界检测
if (Io.x<0) Io.x=0;
if (Io.x>MAXX) Io.x=MAXX;
if (Io.z<0) Io.z=0;
if (Io.z>MAXZ) Io.z=MAXZ;
//观察视线的方向
g_look[0] = float(Io.x + 20*cos(rad_xz));
g_look[2] = float(Io.z + 20*sin(rad_xz));
g_look[1] = Io.y;
ugluLookAtf(Io.x,Io.y,Io.z,g_look[0],g_look[1],g_look[2],0.0f,1.0f,0.0f);
```

上述代码主要是通过修改 `ugluLookAtf` 函数的参数来实现漫游功能,通过移动平台的方向键来改变参数的值。

4 结束语

本文借助 OpenGL ES 嵌入式图形开发库,以地形三维可视化为目标,初步实现了嵌入式系统中三维地形建模、可视化及三维地形漫游功能。同时对嵌入式 3D 图形开发所需的环境设置进行说明,但在地形数据量过大时,运行的效率还需要提高,此外,该系统还可以进一步实现三维查询与空间分析功能,提高系统的实用性。

参考文献

- [1] Khronos Groups. OpenGL ES Common/Common-lite Profile Specification[EB/OL]. [2007-03-13]. <http://www.khronos.org/opengles>.
- [2] 金 励. 面向 21 世纪的嵌入式计算机系统及应用[J]. 太原师范学院学报: 自然科学版, 2004, 3(2): 46-49.
- [3] 徐桂林. 移动平台上三维软绘制器的实现、优化和应用[D]. 杭州: 浙江大学, 2006.
- [4] 郭立波, 赵海燕, 王新海. 基于 OpenGL 的三维地形图显示方法及应用[J]. 石油天然气学报, 2005, 27(5): 585-586.
- [5] H.264 Half-pixel Motion Compensation Transcoding[C]//Proc. of IEEE International Conference on Image Processing. Atlanta, USA: [s. n.], 2006: 865-868.
- [6] Puria A, Chenb X, Luthra A. Video Coding Using the H.264/MPEG-4 AVC Compression Standard[J]. Signal Processing: Image Communication, 2004, 19: 793-849.
- [7] JVT Software Implementation Group. JM10.1 Reference Software [EB/OL]. [2007-12-20]. <ftp3.itu.ch/av-arch/jvt-site>.