

# 基于 OGRE 的双目立体显示研究与实现<sup>①</sup>

王良燕, 张佑生, 高敏

(安徽三联学院 计算机科学与技术系, 合肥 230601)

**摘要:** 双目立体显示是计算机图形学中的一项前沿技术, 在虚拟现实领域发挥着重要作用. 分析了双目立体显示的原理, 根据平行双目投影模型, 提出一种基于 OGRE(面向对象图形渲染引擎)的双目立体显示方案. 该方案充分利用 OGRE 中的虚拟摄像机和视口合成技术, 并在底层结合 CG 脚本, 保证了渲染的效率. 结果表明, 对于正负两种视差, 该方案均有明显的立体效果, 并能灵活漫游及更换不同的场景, 非常适合实际应用.

**关键词:** 双目立体显示; 平行双目投影; 面向对象图形渲染引擎

## Research and Implementation of Binocular Stereoscopic Display Based on OGRE

WANG Liang-Yan, ZHANG You-Sheng, GAO Min

(Computer Science and Technology, Anhui Sanlian University, Hefei 230601, China)

**Abstract:** Binocular stereoscopic display is a cutting-edge technology in computer graphics which plays an important role in virtual reality. The thesis analyzes the principle of binocular stereoscopic display and a solution of binocular stereoscopic display based on OGRE (Object-Oriented Graphics Rendering Engine) and parallel binocular projection model is proposed. The solution takes full advantage of virtual camera in OGRE and viewport synthesis technology, leveraging CG script at the lower layer which brings the highest rendering efficiency. The results indicate that the solution has a good binocular stereoscopic effect with flexibility of easily loading different scene models, which is very suitable for practical applications.

**Key words:** binocular stereoscopic display; parallel binocular projection; OGRE

立体显示技术是虚拟现实领域的一项重要技术, 对提高虚拟现实系统的真实感具有非常重要的作用. 通过计算机模拟立体显示一般基于双目视差原理实现. 由于人的双眼有一定间距, 人在观看三维物体时, 它在双眼分别投影的图像会稍有差别, 所以人脑中会产生立体感. 通过计算机实现立体效果需要模拟人的双眼, 从两个不同的角度去获取三维物体的投影, 然后借助某种设备将两幅投影图像分别送到人的左右眼, 从而使人能看到具有立体感的物体.

目前计算机模拟立体显示多通过 OpenGL 或 DirectX 图形引擎实现<sup>[1-4]</sup>. 但这两种引擎比较底层, 一般只适用于简单三维场景的立体显示. 本文分析了双目立体显示的投影原理, 提出一种通过 OGRE 三维引擎实现立体显示的方案. OGRE 是一款开源三维图形引擎, 在游戏开

发、虚拟仿真等领域具有非常广泛的应用<sup>[5,6]</sup>. 通过 OGRE 实现的立体显示系统不仅可以获得良好的视觉效果, 还能灵活加载各种场景模型并进行三维漫游, 从而使普通的三维漫游获得更高的沉浸感.

### 1 OGRE 三维场景建立

通过 OGRE 实现双目立体显示, 首先需要建立一个三维场景模型. OGRE 擅长处理各种三维空间场景. 它有一个 SceneManager, 对所有三维场景按照树状层次结构进行管理. 建立一个基本三维场景的大致步骤为: 先取得场景空间树的 root 节点, 并创建 SceneManager, 接着载入资源, 然后创建相机、视口以及帧监听器, 最后进行渲染显示. 这样创建的场景中可以实现最基本的三维漫游.

<sup>①</sup> 基金项目:安徽三联学院院级项目(2012013)

收稿时间:2012-10-11;收到修改稿时间:2012-11-16

OGRE 程序中的资源主要是各种三维模型、贴图等。模型可以在程序中创建,也可以用 3DSMax、Maya 等建模软件建立。为了让程序可以灵活加载各种三维模型,可以采用建模和配置文件的方法实现:

首先通过建模软件建立好三维场景模型,然后通过第三方插件导出成配置文件包。如果采用 3DSMax 软件和 oFusion 插件,导出的文件主要包括 Scene.osm、Scene.material 以及一些.mesh 网格文件和材质图片文件,其中,Scene.osm 里是所有模型资源的列表,Scene.material 是用到的所有材质的列表。将配置文件包放在指定的位置,在 OGRE 程序中只要解析 Scene.osm 和 Scene.material 文件即可得到每个元素的详细参数以及实体的材质。如果要更换其它场景,只需更换配置文件包即可。通过以上方法可以实现加载任意复杂的场景模型。

## 2 基于 OGRE 的立体显示

建立好基本的三维场景后,需要实现双目立体显示功能。

### 2.1 双目投影模型

要实现双目立体显示,关键要利用摄像机模拟双眼并获取三维场景在任意时刻的两幅投影图像。由于双眼有一定的间距,因此需要两台摄像机分别模拟左眼和右眼。根据三维模型与摄像机的位置关系,会产生不同的立体视差。如图 1 所示,左右两台摄像机视线位于同一平面内并相交于目标点,A、B、C 为三个模型,A 位于目标点平面内侧,这时产生的视差为负视差,人眼会感觉物体浮出屏幕之外;C 位于目标点平面外侧,这时产生的视差为正视差,人眼会感觉物体在屏幕内部;B 一部分在目标点平面内侧,一部分在外侧,人眼会感觉物体基本就在屏幕上,这种视差称为零视差。观看正视差和负视差一般会有较明显的立体效果。

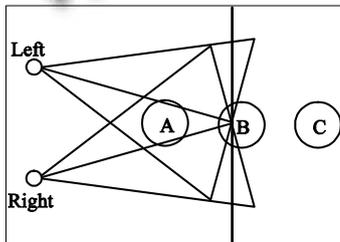


图 1 三种视差

两台摄像机视线相交的投影模型称为汇聚双目投

影模型,但这种模型会引入垂直视差,观看时会感觉不适,影响立体效果<sup>[7]</sup>。为了避免这种缺陷,可以采用平行双目投影模型,这种模型可以使左右眼图像更好地融合<sup>[8]</sup>。平行双目投影模型的投影原理如图 2 所示。如果双目间距为  $d$ ,通过左右两个摄像机模拟双眼,采用透视投影,视轴与投影平面垂直,双目视轴相互平行,双目的视锥体重叠部分是非对称的。

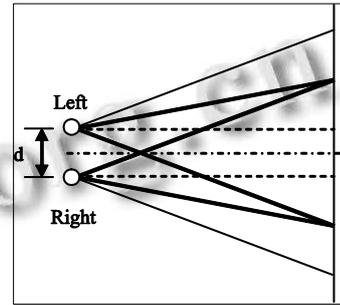


图 2 平行双目投影模型

### 2.2 相机控制

由于平行双目投影模型不会产生垂直视差,因此更适合在立体显示中应用。实现立体显示,关键要模拟出投影模型中的左右两台相机,分别实时获取三维物体的两幅投影图像,从而进行合成显示。

#### 2.2.1 相机参数设置

相机的建立需要根据它的观察空间。如图 3 所示,采用透视投影的相机观察空间是一个平截头体,相机的可视范围在前截面和后截面之间,表示这个空间的参数包括前后截面的距离、前截面宽高比以及 YZ 平面的可视角度。观察空间内的三维物体最终要投影到屏幕上,视口是屏幕上呈现给用户看的窗口,它的宽高比等于前截面的宽高比。相机到投影平面的距离为焦距  $f$ 。通过调节相机的焦距,可以改变投影面相对于前后截面的位置,从而也会改变物体相对于投影面的位置。根据物体相对于投影面所处的位置,会分别产生正视差、负视差或零视差。

OGRE 中有一个相机平截头体类 Frustum,可以用来设置相机投影参数。setNearClipDistance 和 setFarClipDistance 函数可以分别设置前后截面的距离。焦距  $f$  通过 setFocalLength 函数设置。

setAspectRatio 函数用来设置前截面宽高比 AspectRatio,由图 2 比例关系可知:

$$\text{AspectRatio} = \text{Width} / \text{Height} \quad (1)$$

YZ 平面的的可视角度满足下式:

$$\tan \alpha / 2 = \text{Height} / 2f \quad (2)$$

在已知屏幕宽度、宽高比 *AspectRatio* 和焦距 *f* 的情况下, 由(1)(2)式可以得出:

$$\begin{aligned} \alpha &= 2 * \arctan(\text{Height} / 2f) \\ &= 2 * \arctan(\text{Width} / (2f * \text{AspectRatio})) \end{aligned} \quad (3)$$

角度  $\alpha$  可以通过 `setFOVy` 函数设置.

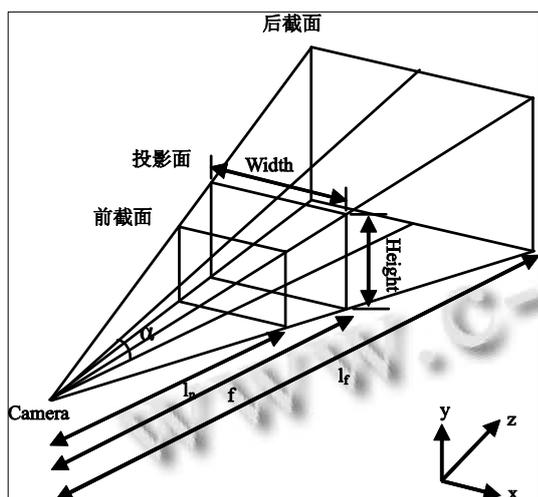


图3 相机投影空间模型

### 2.2.2 相机位置设置

根据平行双目投影模型, 实现立体显示需要设置左右两个平行放置的相机, 分别用于获取左右两个视口, 然后将其合成渲染输出. 由于渲染帧率很高, 可以只用一个相机在每帧渲染前左右平移实现. 相机平移时, 平截头体也必须平移. 在世界坐标系中, 设相机和平截头体的初始位置为 *OldCameraPos* 和 *OldFrustumOffset*, 双眼间距为 *d*, 如果当前这一帧要获取左视口图像, 需将相机和平截头体沿 *x* 轴负方向平移 *d/2*. 渲染后将相机和平截头体复位. 下一帧应获取右视口图像, 则将相机和平截头体沿 *x* 轴正方向平移 *d/2*. 关键实现代码如下:

```
offset = d/2;
if(IsLeftEye)    offset = -offset;
OldCameraPos = Camera->getPosition();
OldCameraPos = Camera->getPosition();
Vector3 pos = OldCameraPos +
              offset * Camera->getRight();
Camera->setPosition(pos);
OldFrustumOffset = Camera->getFrustumOffset();
Camera->setFrustumOffset(OldFrustumOffset
```

+Vector2(offset,0));

### 2.3 视口合成

相机分别获取的左右视口后需要进行合成后显示. 视口合成可以利用 OGRE 中的合成器 `compositor` 实现. `compositor` 一般定义成一个单独的脚本文件, 用于对视口进行后期渲染处理. `compositor` 中通过 `technique` 定义渲染所采用的技术, `technique` 下可以包含多个渲染目标 `target` 以及一个 `target_output`. `target_output` 是最终渲染的输出, 一般将其它已渲染好的 `target` 作为其输入, 用它来合成最后的效果.

通过合成器实现左右视口的合成, 需要定义两个 `target`, 分别为 *Left* 和 *Right*, 最后合成渲染输出到 `target_output` 中. 合成通过定义在另一个脚本文件中的材质 `material` 实现. 立体显示中所用的材质不是某种具体的材质, 而是相机从两个位置捕捉到的画面本身, 不需要在材质脚本中专门指定, 但对这两个 `target` 进行合成时, 需要分别经过某种通道过滤, 因此需要为 `target_output` 定义通道过滤的材质. 通道过滤需要针对每个像素进行, 为了提高渲染效率, 可以添加一段 CG 程序实现, CG 程序能直接作用于硬件<sup>[9]</sup>, 因而可以保证渲染的实时性. CG 函数关键代码如下:

```
float4 LeftPix = tex2D(LeftTex, TextureCoord);
float4 RightPix = tex2D(RightTex, TextureCoord);
return LeftPix * float4(LEFT_FILTER) +
       RightPix * float4(RIGHT_FILTER);
```

*LeftTex* 和 *RightTex* 是分别从左右视口中采样的二维纹理单元, 根据坐标 *TextureCoord* 获取对应的像素值 *LeftPix* 和 *RightPix*, 每个像素包括红、绿、蓝和 alpha 四个值. 然后对每个像素进行通道过滤, *LEFT\_FILTER* 和 *RIGHT\_FILTER* 是通道过滤值. 比如进行红青通道过滤, *LEFT\_FILTER* 的取值为 1,0,0,1, 表示只过滤出红色; *RIGHT\_FILTER* 的取值为 0,1,1,1, 表示只过滤出绿色和蓝色.

左右眼分别经过不同颜色的通道过滤后, 再经过合成器合成, 即可通过分色立体眼镜形成立体显示的效果.

### 3 运行效果

本文在 Windows 平台下基于 OGRE1.7.2 实现了立体显示模拟程序. 实验基于的硬件平台为 Intel Core 2 双核 CPU2.53GHz, 内存 2GB, 显卡为 NVIDIA

GeForce 9600 GSO.



图 4 正视差立体效果

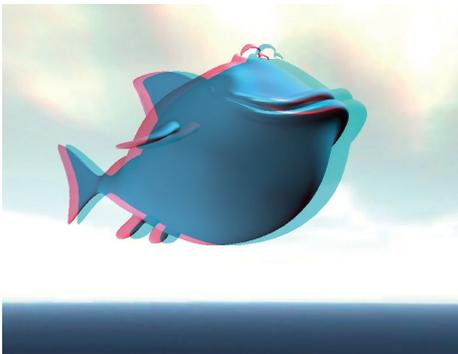


图 5 负视差立体效果



图 6 包含三种视差的立体效果

图 4、图 5 和图 6 为程序运行时漫游过程中的效果图。所有场景模型均采用 3DSMax 软件创建, oFusion 插件导出。双目过滤采用的是红青通道, 通过佩戴红青立体眼镜即可看到立体效果。实验中, 相机左右间距设为 0.5。图 4 和图 5 焦距设为 15, 图 4 中鱼模型中心点距离相机 38.239, 生成的立体效果为正视差, 模型在屏幕之内。图 5 中鱼模型中心点距离相机 8.117, 生成的的立体效果为负视差, 模型会浮出屏幕之外。图 6 中包含正视差、负视差和零视差三种效果, 其中焦距设为 20, 最前面的树距离相机 8.887, 效果为负视差, 后面的建筑物距离相机 30.095, 效果为正视

差, 中间的树距离相机 20.364, 效果基本为零视差。

实验结果表明, 通过 OGRE 建立双目立体显示可以灵活加载场景模型, 实现三维漫游, 并具有良好的视觉效果。

#### 4 结论

立体显示技术在虚拟现实中具有非常关键的作用, 可以极大地提高真实感, 从而改善视觉效果。本文提出的这种基于 OGRE 的立体显示方案, 不仅可以实现各种视差的立体效果, 还可以灵活加载各种复杂的场景模型, 同时由于 OGRE 本身就具有开发三维系统的优势, 因此该方案很适合在实际中应用。

#### 参考文献

- 1 Boev A, Raunio K, Georgiev M, G otchev A, Egiazarian K. Opengl-based control of semi-active 3D display. 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2008: 125-128.
- 2 Li Y, Li DX, Wang LH, Zhang M. Real time stereo rendering for augmented reality on 3DTV system. Systems and Informatics(ICSAI), International Conference, 2012: 2125-2129.
- 3 Yi PY, Zhao YJ. 3D landscape development based on the high-resolution satellite image. Remote Sensing, Environment and Transportation Engineering(RSETE), Int. Conference, 2011: 8011-8014.
- 4 吕成宾, 鲍苏苏. 在虚拟现实快速应用立体显示. 计算机系统应用, 2011, 20(10): 193-197.
- 5 Cai L, Chen Z. Design and implementation of OGRE-based game scene editor software. Computational Intelligence and Software Engineering(CiSE), International Conference, 2010: 1-4.
- 6 Gao PQ, Yang DT, Shen M, Guo XZ, Yu HH, Zhao Y, Xu ZY. Space situation simulation and visualization based on OGRE. Fuzzy Systems and Knowledge Discovery(FSKD), Eighth International Conference, 2011: 2566-2569.
- 7 张宇辉, 吕国强, 胡跃辉, 刘文文. 立体显示的双目模型算法及实现. 计算机工程与应用, 2006, 35: 65-67.
- 8 赵猛, 金一丞, 尹勇. 立体显示中双目视差模型和深度感知研究. 计算机工程, 2011, 37(17): 271-273.
- 9 Randima Fernando, Mark J. Kilgard. Cg 教程——可编程实时图形权威指南. 北京: 人民邮电出版社, 2004. 10-19.