

武汉科技大学

研究生课程论文封面

课程名称： 计算机图形学与虚拟现实

论文题目： 基于 Java3D 的真实感树叶建模

学生班级： 计算机学院 2014 级研究生

学生姓名： 李 霄 剑（201413704061）

任课教师： 陈 建 勋

学位类别： 专业硕士

评分标准 及分值	选题与参阅资料 (分值)	论文内容 (分值)	论文表述 (分值)	创新性 (分值)
评分				
论文评语：				
总 评 分		评阅教师：		评阅时间 年 月 日

注：此表为每个学生的论文封面，请任课教师填写分项分值

基于 Java3D 的真实感树叶建模

摘要 利用 Java3D 技术对真实的树叶进行建模研究。首先,要从真实的树叶图片中获取建模树叶所必需的叶茎和叶子轮廓数据,并将数据组织好存放在本地磁盘。然后,利用取得的数据将叶茎分割成许多小段的圆台并将之映射为 Java3D 对象,叶子轮廓映射为 B 样条曲线,利用 Java3D 技术构造场景图,进行几何建模、坐标变换、添加对叶子的缩放、移动、旋转操作,最终构造出可以交互的真实感树叶模型。

引言

由于树叶结构的复杂性和不规则性,一直以来树叶的建模难度较大,一般都是采用分形算法对树叶模型建模,这样做可以构造出不错的效果,但是毕竟不是依据真实的数据来进行建模的,缺乏真实感。

本文是通过对真实树叶取数据,利用 Java3D 技术对树叶进行真实感建模,以树叶为例子探讨真实感图形学的建模技术和手段,从而验证该方法的可行性与适用性。

Java3D 是 Java2SDK 的标准扩展,它对底层的图形库 OpenGL 和 DirectX 进行了封装,其意义非同寻常,Java 本身所具有的强大的网络功能再添加上 OpenGL 和 DirectX 强大的三维图像功能,使得原来只能在单台计算机上看到的三维图形可以通过 Internet 显示在用户的浏览器上。目前 Java3D 已经广泛应用于科学计算可视化、教育、机械设计、地理信息、动画、医学等诸多领域。

Java3D 主要特性有:

1. 以树状场景图作为编程结构,非常适合于描述虚拟空间。
2. 提供多种类库支持三维实体建模及运动控制(平移、旋转、缩放,碰撞检测等)。
3. 以 Java2 类库作为支持平台,运算功能强大。
4. 可扩展性好,适用于许多网络应用程序的开发。
5. 与其他三维环境(包括 3DS、VRML 等)保持兼容性。
6. 与计算机平台无关,可移置性好。适应各种显示环境和输入设备。

基于上述特性,有理由相信可以将 Java3D 作为设计平台,开展真实感树叶建模的研究。进而利用 Java3D 的网络应用优势,拓宽远程虚拟控制研究的新领域。

1 系统架构设计

如下图 1 所示,系统实现为典型的 MVC 结构,其中 MainFram 为控制层, PicPane 和 ThreeDPanel 为视图层, DataCollector 为模型层。系统主要包括 3 个部分,其中 MainFrame 做为总控制器,协调各部分工作。PicPane 实现树叶图像加载和用户从图像取数据的交互, DataCollector 实现树叶数据的存取为 PicPane 和 ThreeDPanel 提供数据支持, ThreeDPanel 从 DataCollector 获取树叶数据从而构造三维场景和几何模型展示三维树叶。

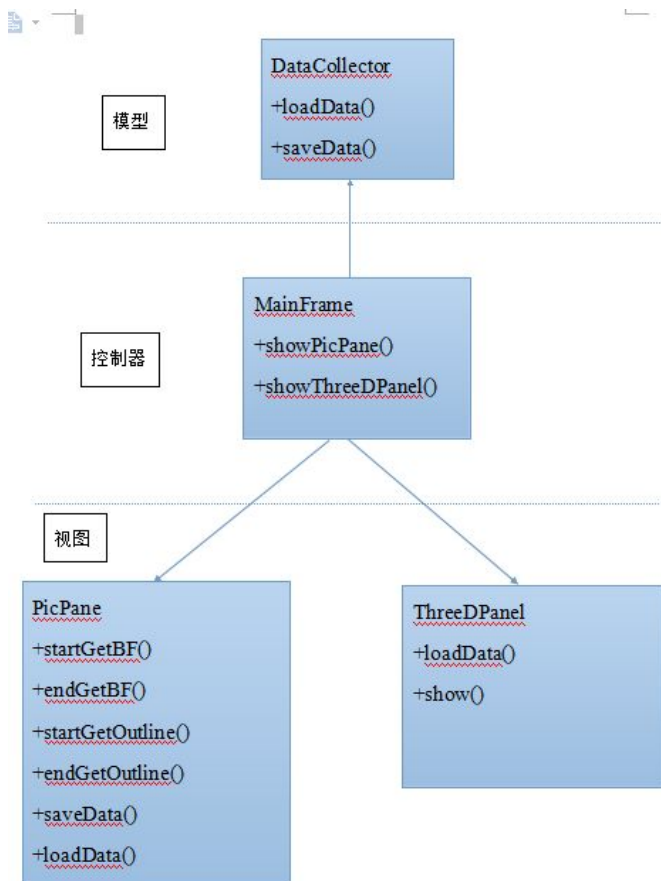


图 1

2 树叶数据的获取

真实树叶数据获取的思想是先加载树叶的图像，然后通过鼠标在图像上通过画线的方式获得每段叶茎的位置信息和叶茎横截面直径，通过点取的方式获得叶子的轮廓信息。

2.1 叶茎信息的获取

由于树叶上面存在许多叶茎，叶茎信息是分段获取的，对于每条叶茎在分叉处和两头端点处取得叶茎横截面的直径，这样就可以获得每条叶茎从上到下不同部分的横截面直径。再分别对其它的叶茎运用相同的做法，就可以获得树叶上所有叶茎的信息。

具体的做法是，如下图 2.1 所示，用鼠标在叶茎处画直线，这样就获得了直线起始点和终点两个坐标信息，通过计算两点之间的距离作为叶茎横截面的直径。



图 2.1

叶茎数据的存储格式如下图 2.2 所示：

```
叶茎数目：n  
第一条叶茎取得很截面直径直线数目：count  
    第一条直线的两个端点：p1,p2  
    .....  
    第count条直线的两个端点：p1,p2  
    .....  
    .....  
第n条叶茎取得很截面直径直线数目：count
```

图 2.2

2.2 叶子轮廓信息的获取

叶子的轮廓是由复杂的曲线构成的，这里我们选择用 B 样条曲线来逼近叶子的轮廓，所以我们要取相应的控制点。具体的做法是在树叶图片上点选，如下图 2.3 所示，红色矩形框即为选取的控制点。



图 2.3

树叶轮廓数据的存储格式如下图 2.4 所示：

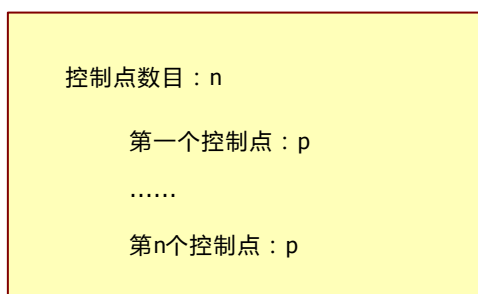


图 2.4

3 树叶三维模型的构建

在建立树叶三维模型之前，首先要把之前取得的树叶数据转化为三维数据，根据数据建立几何子模型，然后再对子模型进行装配成一个完整的大模型，这个处理过程也称为几何模型的链接过程。在此过程中，我们把模型作为一个动态坐标系的子结点，利用层次化的结构，由低到高，从一段段圆台到一条叶茎形成一个控制链，再从一条条叶茎到树叶的整个叶茎结构形成一个控制链，再利用 B 样条曲线形成树叶轮廓，将整个叶茎结构和树叶轮廓结合形成一个控制链，这样就构造了完整的树叶模型。如下图 3 所示，是完整的场景图结构，其中 BaseOfLeaf 是叶茎结点，LeafOutline 为树叶轮廓结点，BaseOfLeaf 是利用 TRIANGLE_STRIP_ARRAY 构建起来的，LeafOutline 是用 B 样条曲线构建起来的。

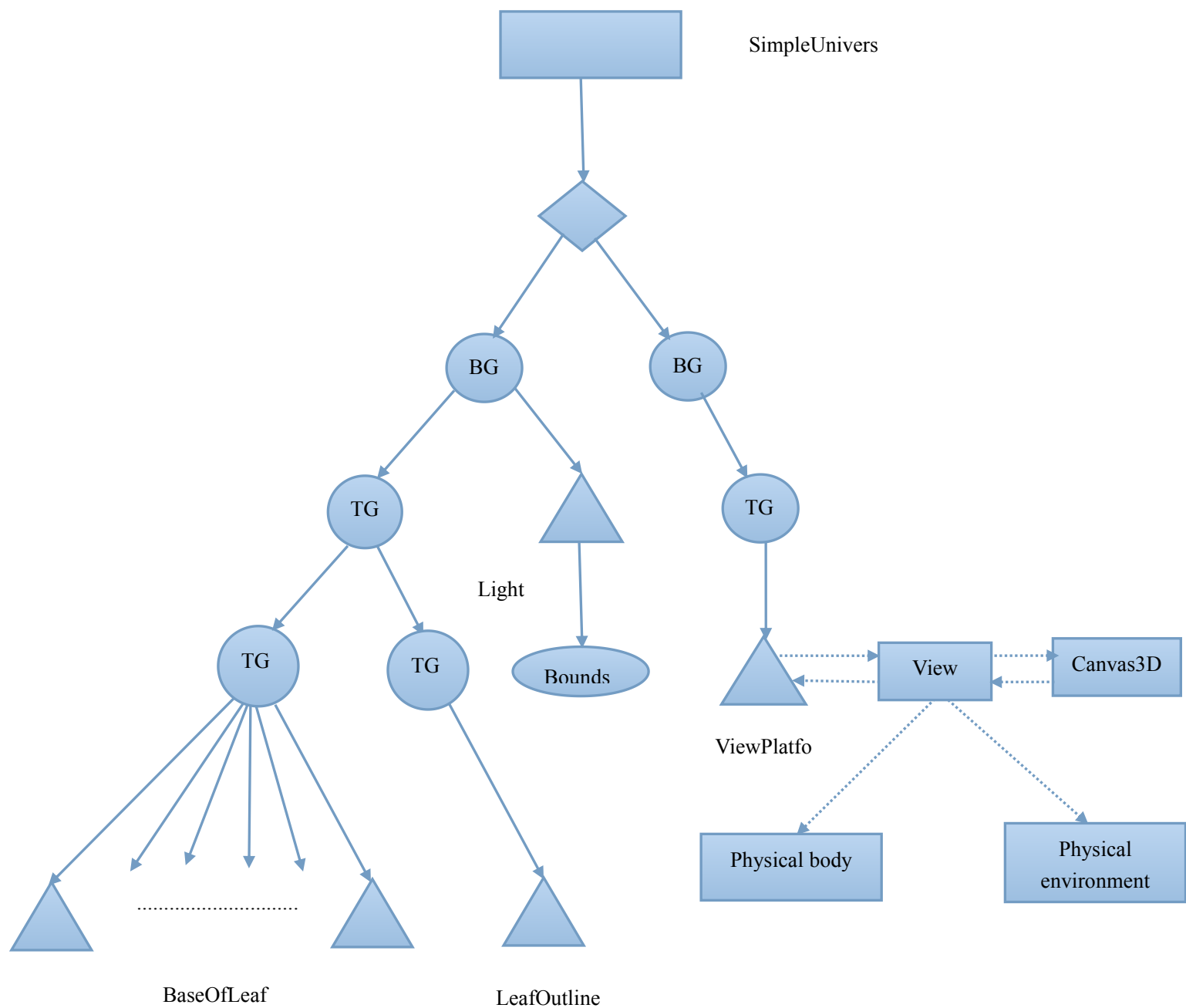


图 3

4 试验结果及分析

程序运行后的界面如图 4.1 和图 4.2 所示，通过取真实树叶的数据构造树叶的三维模型，可以看到实验结果如图 4.3-图 4.6，叶茎的走向和树叶的轮廓都和真实树叶图像很像，达到了不错的效果。

但是，我们在数据的获取方面还是比较麻烦的，下一步要做的工作就是要通

过图片自动获取树叶的三维数据，并且还要实现对树叶模型的纹理映射。

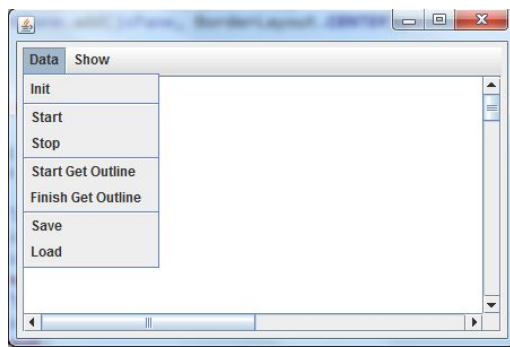


图 4.1

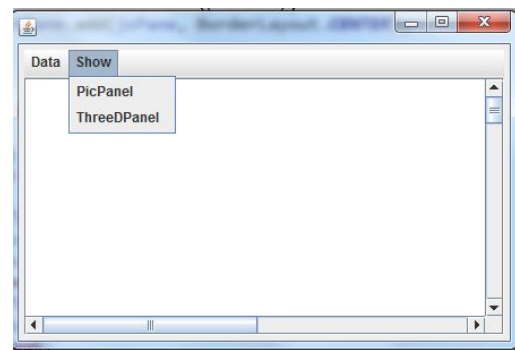


图 4.2

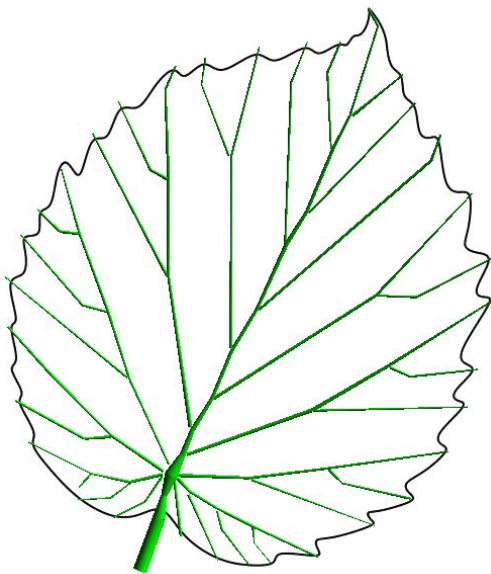


图 4.3

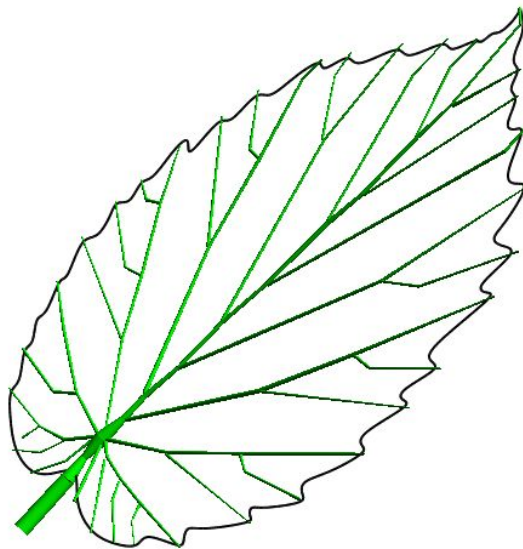


图 4.4

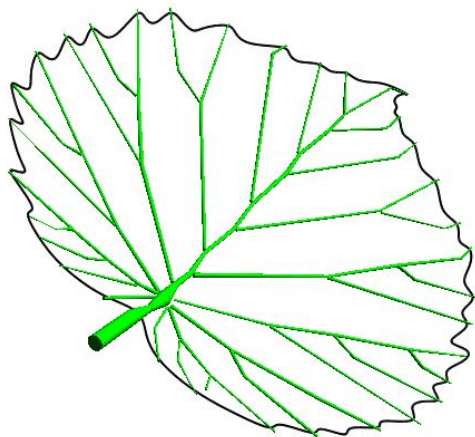


图 4.5

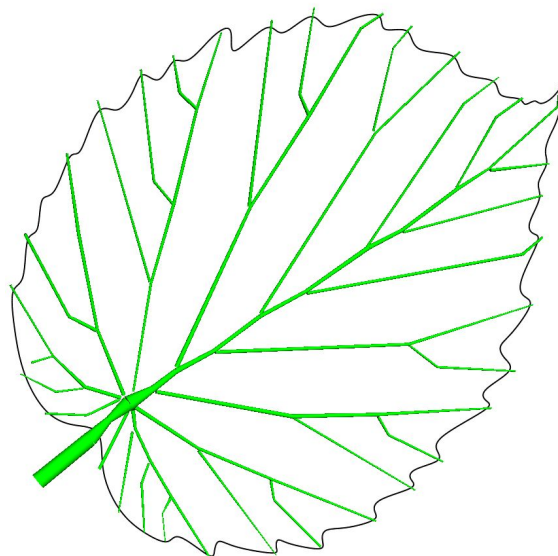


图 4.6