计算机图形学 考试大作业报告

郑博洋

519021911086

1. **项目概述**
   1. **功能概述**
      * 1. 玩家自己通过编辑轮廓曲线生成茶具（支持建模包括三才杯、茶杯， 桌凳，茶盘）。
        2. 玩家可选择添加贴图和环境光照，并能将自己设计的茶具模型存储。
        3. 玩家可将自己设计建造的3D模放置在3D场景中。
        4. 系统可提供相机动画实现场景的不同角度观察。
        5. 系统自动在场景中添加粒子特效（例如桃花飞落、落叶飘零）。
   2. **操作概述**
      * 1. 键盘交互功能：

键盘交互主要为调整参数以及页面切换的快捷键。

Tab为二维编辑页面和三位编辑页面的切换。

X/Y/Z 为三维页面中，选中物体进行X、Y、Z轴的正向移动。

Alt + X/Y/Z 为三维页面中，选中物体进行X、Y、Z轴的逆向移动。

Q 为三维页面中，环境光的逆时针移动。

E 为三维页面中，环境光的顺时针移动。

R/G/B 为三维页面中，环境光r、g、b值的正向调整。

Alt + R/G/B 为三维页面中，环境光r、g、b值的负向调整。

S 为三维页面中，选中物体的放大。

Alt + S 为三维页面中，选中物体的缩小。

D 为三维页面中，删除选中物体。

* + - 1. 鼠标交互功能：

在二维页面中，点击并拖动控制点可以编辑曲线。

在二维页面中，滑动滚轮可以缩放页面。

在三维页面中，点击物体可以选中物体。

在三维页面中，右键可以呼出菜单。

在三维页面中，滑动滚轮可以调整视角距离，Alt + 鼠标滚轮可以修改光源高度

* + - 1. 右键菜单交互功能；

模型功能。包括导入导出以及导入预设。

纹理功能。包括使用预设纹理，以及导入并使用纹理。

粒子系统功能。包括粒子系统切换和开关。

天空盒功能。包括切换天空盒。

1. **模型实现**
   1. **曲线绘制实现**

项目中主要使用以三次贝塞尔曲线为主的样条曲线进行曲线绘制和控制。

使用Opengl提供的二维glMap1\*求值器对曲线进行计算。

由玩家指定一组控制点后，通过glMap指定并输出曲线，供编辑页面显示使用。

* 1. **曲面绘制实现**

考虑到需要支持的建模类型以旋转体为主，本项目中的样条曲面基于玩家编辑后的贝塞尔样条曲线进行动态生成。

使用Opengl提供的二维glMap2\*求值器对曲线进行计算。

基于玩家在曲线编辑时编辑的贝塞尔样条曲线的控制参数，将其作为立体旋转体结构的轮廓曲线，通过多个曲面的拼接生成一个完整的三维物体。

考虑到使用一个贝塞尔曲面绘制完整封闭曲面存在困难，因此使用**拆分法**实现

**拆分法**：

定义一个模型的精度R，将一个三维的旋转体拆分为R片曲面（例如R=360，即将圆周拆分为360片，每片的圆心角为1°）。对于其中的每一片来绘制一个贝塞尔曲面，R越高则图形越精确圆润（设定为360时正常使用基本看不出棱角）。

如此可以避开绘制封闭曲面的问题，同时提供了一个动态调整渲染精度的方法，即通过降低R来降低模型精度，提高渲染效率。

在使用拆分法的情况下，发现在场景中同时渲染多个物体时会出现帧率下降，卡顿的情况，因此对拆分法进行了进一步优化（详见（五）模型渲染性能优化）。

* 1. **模型结构实现**

考虑到项目中存在在同一场景中使用多个物体的需求，同时一个物体很可能需要包含复数个曲面，并且会有更多的输入如光照，贴图等加入。因此将上述的曲面组织为更高一级的数据结构，即物体(代码中将其命名为“MyObject”)。

一个Object包含以下信息：

1. Model Type: 物体可能是玩家从预设类型（茶杯、三才杯等）中选择的物体。
2. Num Of Surface: 包含的曲面数量。
3. Faces: 包含的曲面所拥有的具体参数。
4. Position: 在场景中的三维坐标。
5. Scale：渲染时采用的大小比例。
6. Textures: 使用的贴图序号。

提供以下方法：

1. 三维绘制方法，将其渲染在三维场景中。
2. 二维绘制方法，将其控制轮廓曲线绘制在二位场景中。
3. 导入、导出方法，将模型数据导出为自定义的格式，将已有的数据导入场景。
4. 移动等交互编辑方法。
   1. **模型读入读出实现**

调用读入读出时，将打开一个文件（夹）选择窗口供使用。

模型导出：依照(三)中阐述的Object信息。将信息和数据直接保存在以自定义的格式保存在文件中。

模型读入：依照(三)中阐述的Object信息。根据选取的文件生成三维对象并将其初始化生成在场景中。

* 1. **模型渲染性能优化**

优化一：

考虑到使用求值器进行计算时，其计算过程占了整个绘制过程的大部分，因此若是对拆分法中的每一片曲面都通过输入控制点让其计算的方式生成，在追求精度提高R的情况下可能会导致画面的不流畅等问题。

因此优化考虑每次只使用求值器计算一次，而其余的曲面片则通过旋转第一片的方式依次生成，从而降低计算消耗。

优化二：

拆分法主要目的通过足够多的面片来近似得到一个圆形。

同样使用近似法，可以用三次贝塞尔曲面模拟四分之一圆周的方式进行，使用预先计算的魔数为曲面补足控制点，使得同样使用求值器计算的情况下，一个旋转体仅需进行四次计算即可完成。

1. **贴图实现**
   1. **贴图读取实现**

本项目使用第三方库stb\_image来处理导入的jpg格式图片。

通过使用stb\_image，将jpg文件读入并对应生成纹理。

将生成的纹理以数组方式储存其绑定的序号以供后续绘制使用。

* 1. **贴图使用实现**

对于基本图形进行贴图（如天空盒，粒子系统等），在绘制时对四角直接进行纹理坐标的设定，并在绘制时调出预先生成的纹理。

* 1. **贴图对曲面使用优化**

对于使用求值器计算的贝塞尔曲面，直接绑定顶点的方法并不适用，因此在求值器计算贝塞尔曲面的同时，需要额外适用一个求值器对纹理坐标进行计算，从而将纹理坐标分配到生成的曲面上，完成曲面贴图。

而在使用拆分法生成模型的情形下，无法直接为整个曲面进行纹理映射。因此在绘制每个面片的同时，需要维护其角度信息用于计算它在整个曲面上的相对位置，并计算出其应有的纹理信息。从而使得纹理贴图能够跨越复数个曲面片在整个曲面上呈现完整图案。

1. **场景实现**
   1. **天空盒实现**

本项目使用方形天空盒。

使用上述贴图实现中的方法，分别读取立方体的六面贴图，将其坐标设定为以原点为中心的立方体，将场景笼罩在这个立方体中。从而显示出立体背景的效果。

* 1. **天空盒效果优化**

直接将天空盒作为一个普通物体的方式在场景中渲染，会出现以下问题：

* + - 1. 光照问题：如果天空盒与普通立方体一样接收环境光照，那么每个面上将会有显著的明暗不同问题，导致分界出能够明显察觉不同，影响整体效果。

优化方案：在绘制天空盒时为其关闭光照计算，在绘制结束后重新开启。

* + - 1. 接缝问题：以立方体形式绘制的天空盒在接缝处因渲染模式的问题可能会出现黑边，影响整体效果。

优化方案：修改贴图的方案为边缘截取（GL\_CLAMP\_TO\_EDGE），从而将靠近边缘的单元也用作纹理计算，从而将缝隙闭合。又因为Opengl使用的版本问题，无法直接使用（GL\_CLAMP\_TO\_EDGE）的宏定义，改为其定义的数值0x812F，同样解决问题。

* + - 1. 透视问题：在场景漫游时若调整摄像头位置到较高或较低坐标，在侧边的接缝处很可能出现透视问题，影响整体效果。

优化方案：扩大天空盒尺寸，降低摄像头移动时相对角度的变化。或通过让天空盒随摄像头一起移动从而保持摄像头始终位于天空和的中心来避免透视角度问题。

1. **光照实现**
   1. **日光光源实现**

本项目引入光源作为日光，可以由玩家来设定其位置、色彩、强度。

光源参数分为环境光、漫反射光、镜面光开放给玩家可以进行调整。默认状况下将环境光参数置为0，漫反射光和镜面光设置为全白光。

光源位置模拟日光以坐标原点为中心，使用极坐标的方式进行调整。

1. **相机实现**
   1. **相机视角实现**

为便于对3D对象进行观察和编辑，相机视角动画采用以观察对象为中心的视角，使用极坐标的参数计算方式。使用Opengl提供的gluLookAt函数，在每次渲染时刷新相机位置和目标位置来实现相机动画效果。

1. **粒子系统实现**
   1. **模拟运动实现**

本项目中的粒子系统参考平抛运动的物理规律进行实现，使用带有加速度的模型来使得落叶/桃花飘落的效果更加理想。为便于管理每一个粒子，将速度、数量、生命周期等参数组织为高一级的数据结构，即粒子(代码中将其命名为”Paticle”)

一个粒子维护以下信息：

* + - 1. 存活状态：超出生命周期的粒子将进入回收阶段，每次只渲染存活的粒子。
      2. 生命周期：生命周期随时间衰减。
      3. 坐标：粒子在三维空间的坐标。
      4. 旋转角：粒子的旋转角度。
      5. 颜色：渲染颜色。
      6. 运动方向：记录三维空间中的速度方向矢量。
      7. 外力影响：记录为其设定的受力方向。
      8. 外形参数：记录该粒子的外形。
  1. **随机化实现**

大量的粒子如果形状或运动轨迹上过于统一，会使得表现效果缺乏美感。因此需要在不影响性能的前提下，实现一定程度上的随机化。

生命周期随机化：所有粒子不应在同一时间消失或出现，否则可能出现断层或断续的情况，因此在初始化粒子时，对其生命周期进行随机化，使其在下落不同高度后消失。

起始坐标随机化：本项目目的在于模仿花瓣或树叶飘落效果而非水流效果，因此粒子不应从一个定点上冒出。对粒子的起始坐标随机化，使其在一定范围内冒出。

旋转角随机化：粒子不应以相同的角度下落，对旋转角进行随机化能够消除整齐感。

外形随机化：为模仿花瓣或树叶，粒子大致上是又两个三角形拼接的细长四边形，在此基础上对其宽度和角度随机化。

运动方向随机化：花瓣和树叶不会像雨滴一样垂直下落，往往会出现打转的效果，因次对其初始的运动方向进行随机化，制造纷乱的感觉。

* 1. **使用对象池的性能优化**

考虑到粒子系统会需要大量的粒子不断出现和消失，如果将每一个粒子在生命周期结束后都进行销毁再重新生成，会极大地影响运行效率。因此用一个容器作为对象池来管理粒子。每当粒子生命周期结束后，对其所有参数进行重设和随机化，再使其重新进入生命周期，如此往复来搭建一个高效的粒子系统。

1. **交互功能实现**
   1. **交互功能概述**

本项目中交互功能主要分为三部分：

* + - 1. 键盘交互功能：

键盘交互主要为调整参数以及页面切换的快捷键。

Tab为二维编辑页面和三位编辑页面的切换。

X/Y/Z 为三维页面中，选中物体进行X、Y、Z轴的正向移动。

Alt + X/Y/Z 为三维页面中，选中物体进行X、Y、Z轴的逆向移动。

Q 为三维页面中，环境光的逆时针移动。

E 为三维页面中，环境光的顺时针移动。

R/G/B 为三维页面中，环境光r、g、b值的正向调整。

Alt + R/G/B 为三维页面中，环境光r、g、b值的负向调整。

S 为三维页面中，选中物体的放大。

Alt + S 为三维页面中，选中物体的缩小。

D 为三维页面中，删除选中物体。

* + - 1. 鼠标交互功能：

在二维页面中，点击并拖动控制点可以编辑曲线。

在二维页面中，滑动滚轮可以缩放页面。

在三维页面中，点击物体可以选中物体。

在三维页面中，右键可以呼出菜单。

在三维页面中，滑动滚轮可以调整视角距离，Alt + 鼠标滚轮可以修改光源高度

* + - 1. 右键菜单交互功能；

模型功能。包括导入导出以及导入预设。

纹理功能。包括使用预设纹理，以及导入并使用纹理。

粒子系统功能。包括粒子系统切换和开关。

天空盒功能。包括切换天空盒。

* 1. **右键菜单功能实现**



Opengl本身UI相关功能较为薄弱。因此在本项目中使用glut提供的菜单功能来代替UI。

通过glutCreateMenu的方式可以构建多层次结构的菜单，足以满足使用需求。通过定义宏命令，将菜单指令与响应函数绑定。同时将菜单绑定在鼠标右键上，方便的同时避免影响编辑功能的使用。

* 1. **模型编辑交互实现**

二位页面中提供鼠标拖动控制点进行曲面编辑的功能。

主要实现方式为获取鼠标的窗口坐标，通过矩阵变换获取在Opengl三维空间中的三维坐标。

之后遍历所有控制点，查找在一定范围内的最邻近点，实时更新该点的坐标使其跟随鼠标移动，从而实现拖动编辑的效果。

注意到该方法中获取的三维坐标为一定范围内的碰撞点。因此如果鼠标移动到空处，其变换得到的三维坐标将与理想位置产生出入，影响使用体验。因此在二维页面中额外绘制一个背景白板，便于实时计算拖动坐标。

* 1. **多模型编辑实现**

在场景中，会出现需要拜访多个物体并实时编辑的需求。因此需要实现在三维场景中的物体选中功能来交互获取玩家当前在编辑的是哪个物体，即确定编辑对象。

为实现这一功能，采用了Opengl中的拾取方法。

在鼠标点击后，初始化一个拾取矩阵。在GL\_SELECT的模式下重新渲染一遍物体，为每个渲染的部分使用glLoadName进行命名，这次渲染不会出现在屏幕上被玩家看到。

注意到此时不是所有场景中渲染的物体都需要被拾取，因此设置部分物体如天空盒和粒子系统不在拾取模式下渲染。此外为使得相机转动后，拾取矩阵“眼中“的场景与玩家眼中试图拾取的物体一致，在拾取模式下也需要调用gluLookAt进行视角的变换。

通过处理拾取矩阵获取的结果，我们可以得到在矩阵范围内出现的物体的命名列表，所有渲染部分以调用glLoadName时的命名来区分彼此，在本项目中为方便确定目标，直接取排在第一位的物体作为编辑对象。

* 1. 贴图选择实现

场景中需要为物体赋予贴图。

在选取了编辑对象的情况下呼出右键菜单。通过使用预设贴图或导入JPG图片的方式均可将贴图赋予物体。贴图将由前文所述的逻辑映射到物体上。

* 1. 天空盒切换实现

本项目提供两套预设的天空盒贴图，实现方式如前文所述，通过右键菜单即可切换。

* 1. 光照参数交互修改实现

光源的坐标以极坐标的方式实现，因此其位置移动范围是以原点为球心的球面。因此通过键盘Q和E可以调整其位置，通过Alt + 鼠标滚轮可以调整其高度。

环境光的色彩和强度可以通过键盘R、G、B以及Alt + R、G、B的方式进行调整。

* 1. 相机控制交互实现

相机的移动方式以及坐标定义与光源相类似，为方便操作，其视角的移动主要由鼠标拖动来实现，此外通过滚轮可以缩放势角距离。

* 1. 粒子系统切换实现

右键呼出菜单即可开关粒子系统。