**深 圳 大 学 实 验 报 告**

**课程名称：­ 计算机图形学**

**实验项目名称： 实验二 三维模型读取与控制**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师： 徐鹏飞**

**报告人： 郑雨婷 学号： 2021150122 班级： 高性能**

**实验时间：2023年 11月3日 -- 2023年 11月19日**

**实验报告提交时间： 2023年 11月13日**

**教务部制**

|  |
| --- |
| 实验目的与要求：   1. 熟悉OpenGL 三维模型的读取与处理；理解三维模型的基本变换操作；掌握鼠标键盘交互控制逻辑；掌握着色器中uniform关键字的使用以及数据传输方法。 2. OFF格式三维模型文件的读取：完成对OFF格式三维模型文件的读取与显示，可改变物体的显示颜色。 3. 三维模型的旋转动画：结合模型进行旋转变换的过程，为模型添加自动的旋转动画。 4. 键盘鼠标的交互：通过键盘设定选择绕x、y、z轴进行旋转，鼠标左右键控制动画的开始与暂停。 |
| 实验过程及内容：  1. OFF格式三维模型文件的读取  参考上机实验2.2的内容，完成对OFF格式三维模型文件的读取与显示，可改变物体的显示颜色，尽量特别，但不要太难看。  2. 三维模型的旋转动画  参考实验2.1中动画的生成方式，并结合实验2.3中对模型进行旋转变换的过程，生成旋转动画。  3. 键盘鼠标的交互  参考实验2.1中鼠标与键盘的交互，通过键盘设定选择绕x、y、z轴进行旋转，通过鼠标左右键控制动画的开始与暂停。  具体实现过程：  1. OFF格式三维模型文件的读取  首先查看cow.off文件。可以看到，第一行一个OFF，表示这是一个OFF文件；第二行是三个数字，分别表示顶点数、面片数、边数。因此这个模型有2904个点，5804个片面，接下来的2904行是2904个点的坐标。再之后的5804行，第一个数字表示片面由几个点构成，剩下的数字表示组成面片的顶点序号。    读取OFF文件的函数是read\_off()，其中最核心的部分如下。首先将首部的一行“OFF”读出。然后读取第二行的顶点数、面片数、边数。    接下来的第一个循环，根据顶点数，循环读取每个顶点坐标，此时顺便利用坐标的三个xyz值来作为颜色rgb的值，这样每个点有一个颜色，可以使得之后生成的模型颜色过渡自然。为了让颜色独特且鲜明，我在添加颜色时并没有之间用位置坐标，而是r+0.5，g+0.2，b+0.3，并且设置这三个都不小于0.1，否则 与背景黑色太过接近不易观察。  第二个循环根据面片数，循环读取每个面片的信息，并用构建的vec3结构体faces保存。    在read\_off函数的最后调用storeFacesPoints函数数在points和colors容器中存储每个三角面片的各个点和颜色信息。Points和colors是要传入GPU是数据。    在init函数中调用read\_off文件，将参数设为”./Models/cow.off”（即该文件的相对位置），运行程序可以得到如下窗口。    2. 三维模型的旋转动画  要生成绕x轴旋转的动画，整体思路是在 main 函数中，while主循环内每过一定的时间，就旋转一定角度。  首先实现绕轴旋转。调用updateDelta函数更新参数数值。例如，updateDelta（X\_AXIS, 1）表示将物体沿x轴在旋转一次，至于一次旋转的角度是多少，取决于rotateDelta的大小，rotateDelta的值越大，说明每次的旋转的角度越大。    在display 函数中，初始化一个变换矩阵m，然后根据 rotateTheta 的三个分量分别绕 x、y、z 轴进行旋转，得到最终的变换矩阵。右手坐标系下相对坐标原点绕坐标轴旋转变换，glm库中rotate函数可以直接得到缩放变化矩阵。这里要注意旋转顺序的影响，即先绕 x 轴、再绕 y 轴、最后绕 z 轴。  向着色器中传递uniform的方法是要先获取这个变量在着色器的位置，使用函数为： glGetUniformLocation，然后再向该位置的变量传递数据，函数名字为：glUniform??。函数名字后面的问号内容与你要传递的数据类型有关。比如你传递的是3个浮点数，那函数名字为：glUniform3f。我们要传递一个4阶的变换矩阵，因此使用的函数名字为glUniformMatrix4fv.    这样，变量 rotateTheta一旦发生变化，物体的变换矩阵m就会发生变化，就实现了旋转。想要生成动画，每经过1秒，调用updateDelta函数增加一次rotateTheta的某个分量。这里，记录开始时间start和当前时间end，若两者差值大于1，就调用updateDelta（X\_AXIS, 1）,并且将start变为end，这样一直循环，即可实现自动旋转。    此时的运行结果如下：    3. 键盘鼠标的交互  参考实验2.1中鼠标与键盘的交互，通过键盘设定选择绕x、y、z轴进行旋转，通过鼠标左右键控制动画的开始与暂停。  ①实现键盘旋转绕x、y、z轴进行旋转  updateDelta（X\_AXIS, 1）是绕x轴旋转，updateDelta（Y\_AXIS, 1）是绕y轴旋转，  updateDelta（Z\_AXIS, 1）是绕z轴旋转。可以新增一个全局变量AXIS，默认值为X\_AXIS,也就是默认按绕x轴旋转。当按下或重复按下 X 键时，设置全局变量 AXIS 为 X\_AXIS，指示在后续操作中应该绕 x 轴进行变换。当按下或重复按下 Y 键时，设置 AXIS 为 Y\_AXIS，指示绕 y 轴进行变换。当按下或重复按下 Z 键时，设置 AXIS 为 Z\_AXIS，指示绕 z 轴进行变换。  在key\_callback函数中用switch语句设置：    并且在main函数中的循环中将updateDelta（X\_AXIS, 1）改为updateDelta（AXIS, 1）。    此时的运行结果如下，当键盘按下x,y,z时，所绕的旋转轴会发生变化：    ②鼠标左右键控制动画的开始与暂停  首先定义一个全局bool变量pause，用来记录动画是否停止，pause==false说明物体可以旋转，pause==true说明物体不能旋转动画停止。  参考实验2.1，实现一个mouse\_button\_callback函数，用来监听鼠标左右键的监听函数，当按键是鼠标左键时，pause=false，说明我们可以让物体旋转动画开始，当按键是鼠标右键时，pause-true，说明动画暂停，物体不能旋转。在main函数中，调用一个系统函数glfwSetMouseButtonCallback(window, mouse\_button\_callback)，该函数可以监听鼠标事件，并调用我们刚刚实现的mouse\_button\_callback函数功能。    以上就已经实现了通过鼠标左右键来改变全局变量pause的值了，但是还需要通过pause作用到控制键盘按键的监听函数中。在主函数的贞循环的绘制中，在旋转时间大于1s的基础上，添加另外一个条件，那就是如果pause==false才调用updateTheta(AXIS, offset)旋转。    运行结果如下：    4.补充与完善  ①退出与重置  增添两个键盘事件，当按下 Escape 键时，如果 action 是 GLFW\_PRESS，则调用 glfwSetWindowShouldClose 函数将 GLFW 窗口的关闭标志设置为 GL\_TRUE，以便退出程序。当按下t键时，如果 action 是 GLFW\_PRESS，则调用 resetTheta 函数。这可能是用于重置某些旋转角度或变换的函数。    resetTheta 函数进行复位操作，使xyz各个方向的旋转角度都置0，并且将pause重置为true,这表示使得物体不旋转。    Reset功能的演示如下：    ②新增提示信息  新增函数使用 printf 函数按行输出键盘选项的说明。用户可以通过按相应的键来执行不同的操作，如按 'e' 键退出程序，按 'x' 键绕 x 轴旋转，以此类推。还提供了鼠标操作的说明，左键点击开始，右键点击暂停。这样的帮助信息对于用户了解程序的交互方式非常有用。 |

深圳大学学生实验报告用纸

|  |
| --- |
| 实验结论：  本次实验的目标是实现对OFF格式三维模型文件的读取与显示、三维模型的旋转动画、以及键盘鼠标的交互。  通过参考上机实验2.2的内容，我们成功地实现了对OFF格式的三维模型文件的读取与显示。通过读取文件中的顶点和面片信息，我们能够准确还原原始的三维模型，并通过图形渲染技术将其显示在窗口中。此外，我们通过改变物体的显示颜色，使其更加生动、美观。  参考实验2.3中的模型旋转变换，我们成功地实现了三维模型旋转。通过在每一帧更新旋转角度，我们实现了模型的平滑旋转，并且能够通过按下键盘上的 'x'、'y' 和 'z' 键，我们能够选择绕不同轴进行旋转。通过鼠标左键开始和右键暂停，我们实现了对旋转动画的直观控制。  综上所述，本次实验不仅帮助我们熟悉了OFF格式三维模型文件的读取与显示，还深入了解了三维模型的旋转动画生成和键盘鼠标的交互操作。这为我们在图形学领域的学习和应用提供了重要的基础。  实验心得：  通过完成本次图形学实验，我深入了解了OFF格式三维模型文件的读取与显示、三维模型的旋转动画生成以及键盘鼠标的交互操作。在实现模型加载和渲染的过程中，我通过调整颜色增强了模型的视觉效果。生成三维模型旋转动画的经验使我更加熟悉旋转矩阵和动画生成的原理，而通过键盘和鼠标的交互，我成功地融合了用户友好的操作方式，提升了用户体验。这次实验让我深刻认识到图形学在计算机科学中的实际应用，激发了我对这一领域的兴趣，为今后深入学习和应用图形学技术打下了坚实基础。 |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。