山东大学 计算机科学与技术 学院

计算机图形学 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：202200130119 | 姓名：于斐 | | 班级：学堂计机22 |
| 实验题目：平滑旋转 | | | |
| 实验学时： 4 | | 实验日期： 2024年11月23日 | |
| 实验目的：  通过编程实现基于四元数的3D模型平滑旋转，掌握四元数在3D计算中的优势及其在姿态插值中的应用。通过设置模型的初始和终止姿态，并沿给定路径实现平滑的位姿过渡，深化对3D空间变换、插值计算及其视觉效果的理解。结合交互界面的开发，培养对复杂3D模型控制的综合设计能力，为计算机图形学领域的学习奠定技术基础。 | | | |
| 实验步骤与内容：  实验环境：OpenGL 4.6 及 GLFW, GLM 等附属库。  实验步骤：   1. 与实验1完全相同：项目使用 CMake 管理，原则上任何支持CMake的编辑器均可使用。所有未包含在项目文件中的库均使用 CMake 的FetchContent导入，不需额外手动安装任何库。 程序自动生成到 dist 目录下，但运行时pwd需包含assets。因此需要在根目录运行 dist/Renderer 或将 assets 拷贝到 dist 下。    1. 建立OpenGL窗体及 argument parser, config parser。前者代码位于 Renderer::init()中，在 src/renderer/renderer.h, renderer.cpp 下可以找到。后者代码位于 src/utils 中。使用 ImGUI 构建交互选单，用于控制程序运行时的行为。相关代码位于 src/gui/ 下。    2. 构建 Model 类用于从 .obj 中导入模型。构建Shader类用于加载 Shader Program。构建 Camera 类，用于变换相机。 2. 构建物体变换方式。项目中构建了Motion类，支持vec3形式的 translation, scale 和 quaternion 形式的 rotation。   相关数据存储为 Motion 类。定义可在src/animation/animation.h 中找到。     1. 构建 Animation 类，其中主要包括 std::vector<std::pair<Motion, float>> 用于存储关键帧。同时实现一个计时函数和插值函数。给定当前时间，可以通过相邻的两个关键帧插值出当前结果。     对计时函数，renderer 在处理输入时有使用 glfwGetTime() 计算 delta time between frames，因此直接使用 update 函数将其传入 Animation 类即可。  对插值函数，程序首先找到当前时间所处在的两个相邻关键帧。 对 translation 和 scale，使用线性插值方法，直接数值计算比例并混合两个关键帧对应的 motion。 对 rotation，使用球面线性插值方法 slerp。对于两个 rotation quaternion ，相对时间 ，球面线性插值函数可以被如下定义：  其中 。  三者混合后即可得到平滑结果。   1. Demo | | | |
| 实验总结：  通过本次实验，成功实现了3D模型的平滑旋转和姿态过渡，深入理解了四元数在3D图形学中的重要性。实验过程中，通过正确定义模型的起始和终止姿态，掌握了姿态表示的方法及其与四元数的关系。进一步，通过实现沿路径的平滑旋转和平移，体会到四元数在插值计算中的平滑性和高效性，解决了传统欧拉角表示中的万向节锁问题。结合交互界面的设计与实现，增强了对用户友好操作的理解和实践能力。 | | | |