

学号	姓名	论文规范性 (10)	问题分析与调研 (30)	方案创新性 (20)	实验结果分析与讨论 (40)	结课论文总成绩 (100)
21301043	刘麦	6	21	17	36	80

存在格式问题；缺乏相关工作梳理

北京交通大学

本科计算机图形学设计（论文）

基于 OpenGL 的布料模拟

Cloth simulation based on OpenGL

学 院： 软件学院

专 业： 软件工程

学生姓名： 刘麦

学 号： 21301043

指导教师： 吴雨婷

北京交通大学

2024 年 6 月

# 摘要

在计算机图形学和物理仿真领域，布料模拟是一个重要的研究方向。随着虚拟现实和计算机动画技术的发展，布料的实时运动模拟变得越来越重要。但由于布料具有高度变形的内在特性，在计算机上很难有效和逼真地模拟布料的运动。因此，研究布料的计算机模拟具有较高的理论意义。本文的主要研究内容及成果包括以下几个方面：本项目是一个使用 OpenGL 和 GLM 库以及已有的质点-弹簧模型<sup>[1]</sup>进行实时渲染的布料模拟。在这个模拟中，布料被抽象成一个由粒子组成的网格，粒子之间通过弹簧相互连接，以此来模拟布料的结构特性。同时，除了对于布料本身的考虑，模拟中也考虑了重力、阻力以及粒子间的相互作用力。同时，为了进一步增加了布料的质感的表现，本实验还通过与一个球体的碰撞检测<sup>[2]</sup>，让布料产生形变，实验中还加入了光照来进一步提高布料形态的展示。模拟系统还包含了一个交互式摄像头，可以通过它从不同位置来对布料模拟的结果有更细致的观察和评价。

关键词：布料模拟，质点-弹簧模型，实时渲染，OpenGL，GLM

# 目录

- 基于 OpenGL 的布料模拟 ..... 1
- 摘要 ..... 2
- 目录 ..... 3
- 1 引言 ..... 4
- 2 相关工作介绍 ..... 4
- 3 基于 OpenGL 的布料模拟 ..... 4
  - 3.1 系统架构 ..... 5
  - 3.2 质点-弹簧模型 ..... 5
  - 3.3 实验过程 ..... 5
    - 3.3.1 质点的属性 ..... 5
    - 3.3.2 力的计算 ..... 5
    - 3.3.3 数值积分 ..... 6
    - 3.3.4 碰撞检测与处理 ..... 7
    - 3.3.5 可视化与渲染 ..... 7
    - 3.3.6 主函数实现流程 ..... 8
- 4 实验结果分析 ..... 9
  - 4.1 实验结果 ..... 9
  - 4.2 实验结论 ..... 9
  - 4.3 实验不足 ..... 10
- 5 实验环境 ..... 10
- 6 论文引用 ..... 10
- 致谢 ..... 10

# 1 引言

随着计算机图形学和物理引擎技术的不断发展，布料模拟已经成为虚拟现实、游戏开发和动画制作等领域的重要组成部分。它能够为虚拟世界带来更多的真实感和动态变化，丰富了用户的体验。本项目采用粒子系统和弹簧动力学的方法，通过 OpenGL 和 GLM 库进行实时渲染，实现了一个布料模拟。用户可以通过交互式摄像头，从不同的角度观察布料的状态和形态，深入了解布料的动态特性。

## 2 相关工作介绍

在布料模拟领域，物理引擎和计算机图形学的结合是实现逼真模拟的重要基础。传统的布料模拟方法多采用粒子系统和弹簧-质点模型，通过模拟布料中各质点之间的相互作用力来实现布料的物理行为。

在实验中，我基于 C++ 和 OpenGL 构建了一个实时布料模拟系统。系统的核心是通过粒子和弹簧的物理模型模拟布料的变形和运动。布料被离散为一个由质点组成的网格，每个质点具有位置、速度、受力等物理属性。质点之间通过三种类型的弹簧连接：结构弹簧、剪切弹簧和弯曲弹簧，这些弹簧用以模拟布料在拉伸、剪切和弯曲过程中的行为。

具体实现方面，初始化布料网格的质点位置，并为每个质点计算受到的重力和弹簧力。在每一时间步内，系统首先计算每个质点所受的重力、弹簧力和阻力，然后通过数值积分方法更新质点的速度和位置。为了直观的展示布料模拟的效果，我添加了一个球体，让布料落在球体上来表现布料的质感，通过碰撞检测和响应算法处理质点与球体之间的碰撞，使质点在碰撞后调整其位置并停止其速度。为了增强视觉效果，系统采用 OpenGL 进行布料和对接触的球体的渲染，通过设置光照模型和相机视角，使布料的运动和形变能够直观地展现在屏幕上。

## 3 基于 OpenGL 的布料模拟

料仿真的常用方法就是将布料表达为三维网格，然后通过弹力和外力进行布料仿真。它在游戏中有着广泛的应用，如衣袖的摆动，旗帜的飘动

## 3.1 系统架构

系统的核心包括质点-弹簧模型、力的计算、数值积分以及碰撞检测与处理。利用 OpenGL 进行布料和环境的渲染，实现实时可视化。

## 3.2 质点-弹簧模型

质点-弹簧模型是布料模拟中常用的方法。布料被离散为一个由质点组成的网格，每个质点具有位置、速度和受力等物理属性。质点之间通过弹簧连接，弹簧用来模拟布料的拉伸、剪切和弯曲行为。具体来说，系统包含三种类型的弹簧：结构弹簧、剪切弹簧和弯曲弹簧。

质点： 布料被表示为一个质点网格，每个质点有位置、速度、加速度等属性。

弹簧： 弹簧用于连接质点，模拟弹性行为。常见的弹簧类型有：结构弹簧： 连接相邻质点，保持布料的结构。

剪切弹簧： 连接对角线方向上的质点，防止布料剪切变形。

弯曲弹簧： 连接间隔一个质点的质点，模拟布料的弯曲刚度。

## 3.3 实验过程

### 3.3.1 质点的属性

每个质点的结构如下：

```
struct Particle {  
    glm::vec3 position;  
    glm::vec3 velocity;  
    glm::vec3 force;  
    bool isFixed;  
};
```

1. position: 质点的位置
2. velocity: 质点的速度
3. force: 作用在质点上的力
4. isFixed: 是否固定

### 3.3.2 力的计算

布料中的每个质点都受重力和弹簧力的作用。重力通过简单的常量向量施加，而弹簧力则根据胡克定律计算。

## 重力

重力的计算公式： $F_g = mg$

$m$  是质点的质量， $g$  是重力加速度。

## 弹簧力

弹簧力的计算公式为： $F = k(|L| - L_0) L / |L|$

$k$  是弹簧刚度， $L$  是弹簧的当前长度向量， $L_0$  是弹簧的自然长度。

弹簧力计算代码如下：

```
auto applySpringForce = [](Particle& p1, Particle& p2, float restLength) {
    glm::vec3 delta = p2.position - p1.position;
    float dist = glm::length(delta);
    float displacement = dist - restLength;
    glm::vec3 springForce = springStiffness * displacement * glm::normalize(delta);
    p1.force += springForce;
    p2.force -= springForce;
};
```

## 阻尼力

为了防止系统出现无限振荡，加入阻尼力： $F = -dv$

$d$  是阻尼系数， $v$  是质点速度。

### 3.3.3 数值积分

使用了显式欧拉-拉格朗日方法。这种方法在每个时间步长中，首先计算加速度，然后用加速度乘时间步长来得到速度的增量，最后将这个速度增量加到粒子的当前速度上，从而得到新的速度。接着，将新的速度乘以时间步长来得到位置的增量，并与当前位置相加，得到新的位置。

代码实现如下：

```

void integrate(float dt) {
    for (auto& p : particles) {
        if (!p.isFixed) {
            p.velocity += (p.force / particleMass) * dt;
            p.position += p.velocity * dt;
        }
    }
}

```

### 3.3.4 碰撞检测与处理

为了模拟布料与地面之间的碰撞，系统采用简单的碰撞检测和响应。若质点进入地面范围，则将其位置调整到地面表面，并将速度设为零。

```

void handleCollision() {
    for (auto& p : particles) {
        glm::vec3 toCenter = p.position - groundCenter;
        float dist = glm::length(toCenter);
        if (dist < groundRadius + clothThickness) {
            p.position = groundCenter + glm::normalize(toCenter) * (groundRadius + clothThickness);
            p.velocity = glm::vec3(0.0f); // 停止粒子的速度
        }
    }
}

```

### 3.3.5 可视化与渲染

系统采用 OpenGL 进行布料和地面的渲染。通过设置光照模型和相机视角，实现布料的动态行为的直观展示。

#### 3.3.5.1 布料渲染

布料通过绘制三角形网格进行渲染，代码如下：

```

void renderCloth() {
    glColor3f(0.6f, 0.6f, 0.8f); // 布料颜色
    glBegin(GL_TRIANGLES);
    for (const auto& index : indices) {
        const Particle& p = particles[index];
        glVertex3f(p.position.x, p.position.y, p.position.z);
    }
    glEnd();
}

```

#### 3.3.5.2 地面渲染

地面通过绘制一个球体进行渲染，代码如下：

```

void renderGround() {
    glColor3f(1.0f, 1.0f, 1.0f); // 地面颜色
    glPushMatrix();
    glTranslatef(groundCenter.x, groundCenter.y, groundCenter.z);
    gluSphere(gluNewQuadric(), groundRadius, 32, 32);
    glPopMatrix();
}

```

### 3.3.5.3 添加光照

光照的实现是通过 OpenGL 的 API 来设置的。

在渲染布料时使用法线，以便光照计算可以正确进行。

本系统启用了颜色材质，并将环境光和散射光颜色与物体的颜色绑定。同时，设置了材质的镜面反射颜色和反光度。在设置了光照和材质属性后，调用 OpenGL 的渲染函数，会根据这些属性和光源的位置来计算每个顶点的光照颜色。

### 3.3.6 主函数实现流程

创建窗口，设置光照参数，初始化布料和弹簧系统，渲染循环包括处理用户输入（对相机的控制），设置相机是叫，透视矩阵，投影矩阵，视图矩阵，计算质点受力，按照时间步长更新指点速度和位置，处理碰撞，渲染球体。

主循环

```

while (!glfwWindowShouldClose(window)) {
    processInput(window);
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glm::mat4 view = glm::lookAt(cameraPos, cameraPos + cameraFront, cameraUp);
    glm::mat4 projection = glm::perspective(glm::radians(45.0f), 800.0f / 600.0f, 0.1f, 100.0f);

    glMatrixMode(GL_PROJECTION);
    glLoadMatrixf(glm::value_ptr(projection));

    glMatrixMode(GL_MODELVIEW);
    glLoadMatrixf(glm::value_ptr(view));

    computeForces();
    integrate(0.01f);
    handleCollision();
    computeNormals(); // 计算法线

    renderCloth();
    renderGround();

    glfwSwapBuffers(window);
    glfwPollEvents();
}

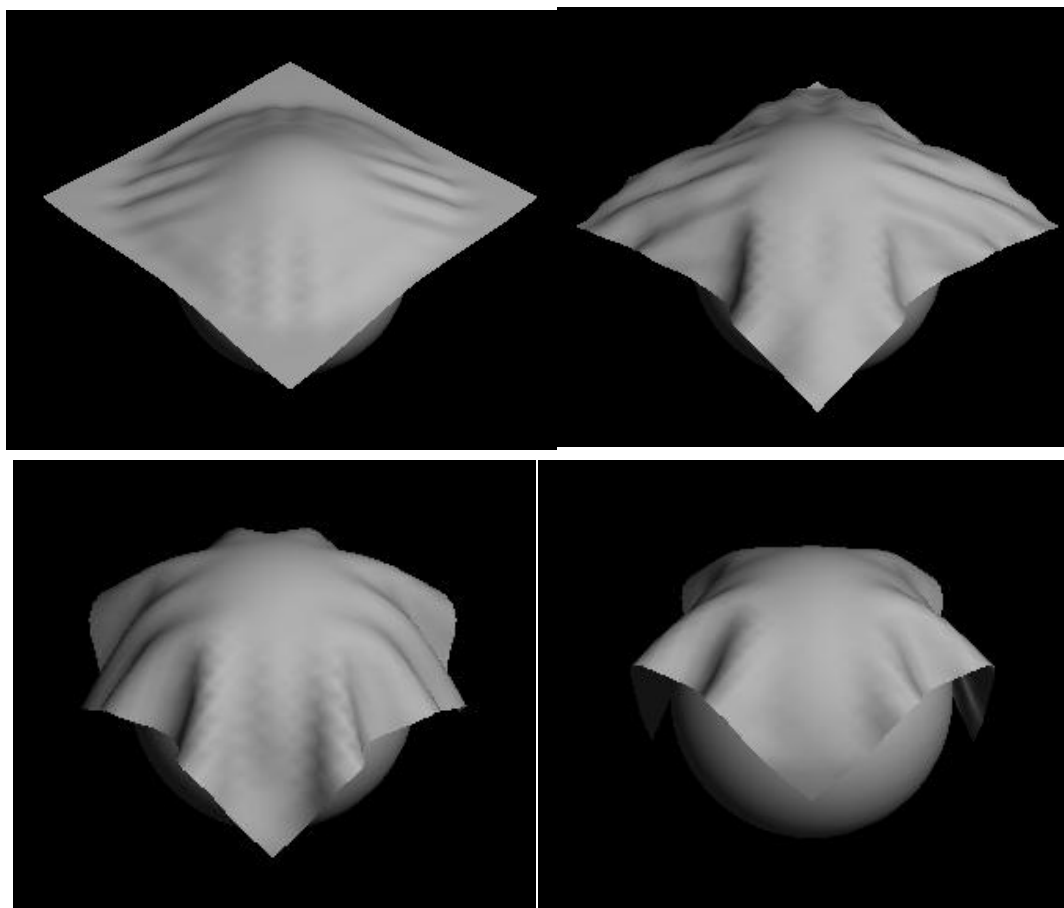
```



## 4 实验结果分析

### 4.1 实验结果

通过调整系统参数（如弹簧刚度、阻尼系数、重力加速度等），可以观察到不同条件下布料的动态行为。实验结果表明，所实现的系统能够在合理的计算时间内，逼真地模拟布料的动态行为，包括褶皱、拉伸和碰撞等现象。



### 4.2 实验结论

本文通过质点-弹簧模型实现了一个实时布料模拟系统，并通过 OpenGL 进行可视

化。系统的实现展示了质点-弹簧模型在布料模拟中的有效性，并为实际应用提供了基础。

## 4.3 实验不足

1. 碰撞处理简单：当前的碰撞检测和处理较为简单，仅能处理布料与地面的基本碰撞，未考虑布料自碰撞及复杂物体的碰撞。这可能导致布料在复杂环境中的模拟精度不高。
2. 数值稳定性：采用显式欧拉法进行数值积分，在大时间步长下可能会有误差导致不稳定。
3. 计算复杂度较高：当前计算方式未优化，较为原始，当质点数量和弹簧数量增加导致计算量过大时对性能的需求较大。
4. 缺乏物理细节：虽然质点-弹簧模型能够模拟布料的基本行为，但对真实布料的复杂物理特性（如摩擦、空气阻力、织物结构等）模拟不足。

## 5 实验环境

操作系统：windows 10

开发环境：vs 2022

相关库：OpenGL glew glfw glm

## 6 论文引用

[1]赵微巍. 动态布料模拟[D]. 辽宁师范大学, 2010.

[2]陈(日午). 关于真实感服装生成算法的研究[D]. 西北工业大学, 2002.

## 致谢

时间飞逝，计算机图形学课程在课业繁重的大三下学期陪我走过了 16 周的快乐时光。课上老师讲的只狼，古墓丽影，艾尔登法环等游戏中用到的图形学技术让同学们充满了兴趣，用循循善诱的方法，提起同学们的学习热情，而不是一味的只是输出图形学知识。对于计算公式，坐标转换，矩阵计算和公式推导等老师也会在黑板上用数形结合的方式一步一步推导。每当老师提起游戏和电影，本来可能溜号的我就会被重新带入课堂中。感谢老师和组员们，让我在图形学课上收获了知识和快乐。