

# CG HOMEWORK

董恒硕

Hw8

## 1. Introduction

用 OpenGL 实现纹理映射。

## 2. Method

根据 Tutte 参数化:

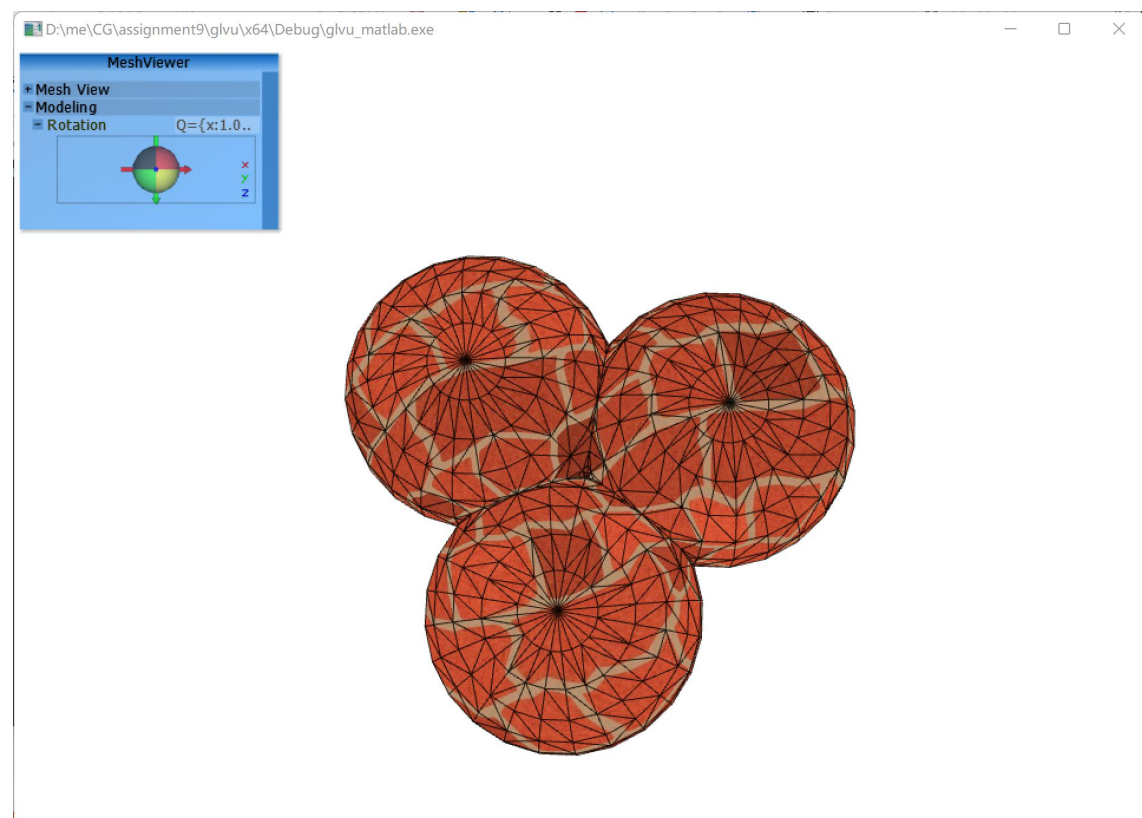
$$p = \sum \frac{1}{d} p_i, \text{ (其中 } p_i \text{ 是 } p \text{ 的邻居)}$$

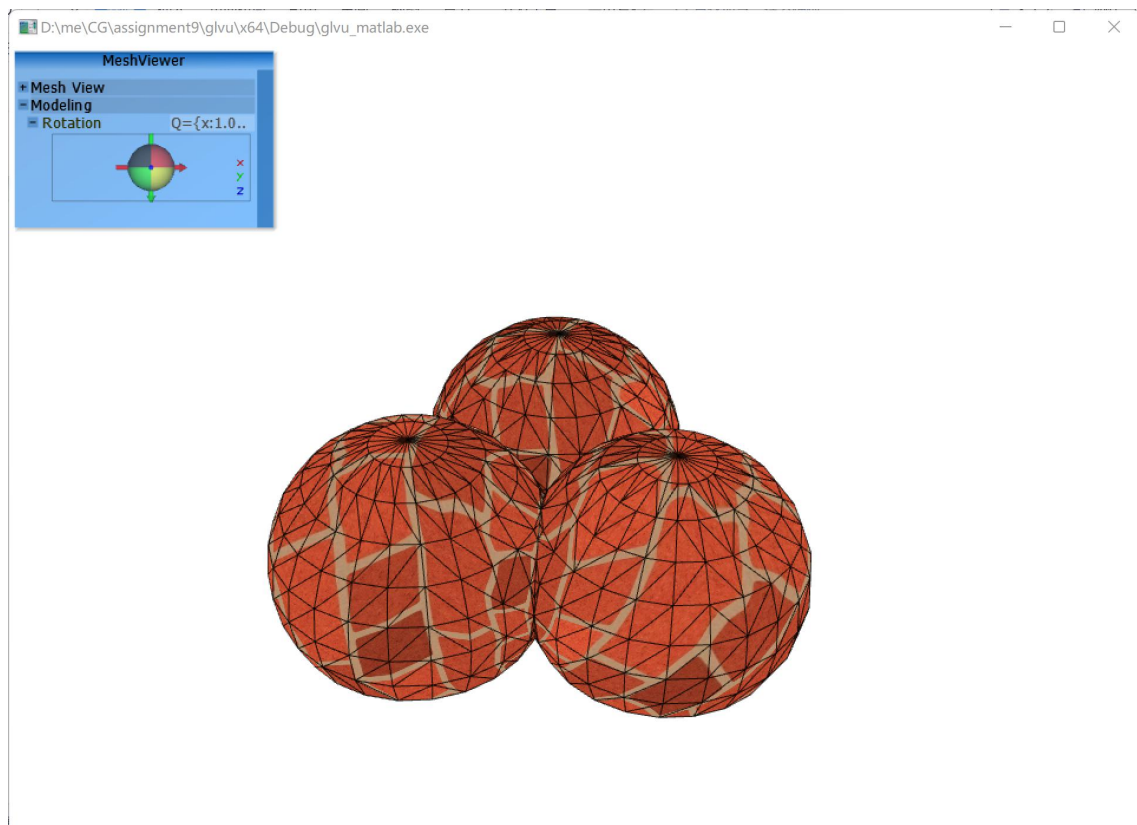
将三维图转为二维图片。

再将纹理图片映射到二维图片上, 得到对应位置的像素值, 最后再将像素值映射回三维图形。

2,

## 3. Result





#### 4. Discussion

从结果上观察可以发现，纹理嵌入效果较好。

#### 5. Compute code

用内置函数读入纹理信息并调用函数 `gettutte()`

```
//homework
M.tex.setImage(MyImage("bricks2.jpg", 3));
std::vector<float> y = gettutte();
M.upload(x.data(), x.size() / 3, f.data(), f.size() / 3, y.data());
M.showTexture = true;
```

用 `tutte` 参数化方法将三维图像参数化并用 `tutte.txt` 记录。

```
std::vector<float> gettutte()
{
    std::ifstream file;
    file.open("tutte.txt");
```

```
if (!file) std::cout << "error" << std::endl;
```

```
float num;
```

```
std::vector<float> texcodd;
```

```
while (file >> num)
```

```
{
```

```
    texcodd.push_back(num);
```

```
}
```

```
return texcodd;
```

```
}
```