

# 曲面细分上机实验报告

刘知行 12124079

## 1. 简介

本人系浙江大学航空航天学院 2021 级直博生刘知行。本程序是 2021 年秋季应用几何造型基础课程上机作业，实现三种曲面细分算法。采用 c++ 开发，开发环境为 Windows10、Visual Studio 2017。

## 2. 程序包含

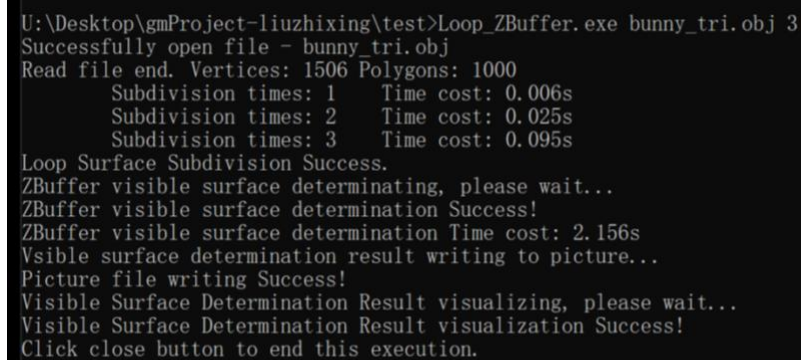
```
\cmakeLists.txt      #用于 cmake 编译
\src                  #源代码
    \main.cpp
    \object.h
    \object.cpp
    \vector.h
    \vector.cpp
\lib                  #第三方库，用于可视化
    \glut
\doc                  #说明文档
    \（本文件）
\cases                #程序用例
    bunny_tri.obj
    spherecylinder_quad.obj
    spring_quad.obj
    torus_filled_quad.obj
    torus_simple_quad.obj
\release              #编译好的可执行文件
    \doosabin.exe
    \catmullclark.exe
    \loop.exe
\ParaView.rar         #具有可视化等功能的查看器，需要解压
```

### 3. 编译

请使用 `cmake` 进行编译。

### 4. 程序使用

在 windows powershell 或 cmd 中使用方法如图 1，命令行需输入需要进行细分的模型文件名和细分次数。读文件成功后将开始曲面细分，细分成功后将展示细分结果（用的是计算机图形学作业涉及到的 Z-Buffer 消隐算法），如图 2 所示，同时生成这一细分结果的 `obj` 文件，用三维模型查看器可以打开该模型进行查看，如图 3 所示。每次细分消耗的时间会在命令行中打印出来。



```
U:\Desktop\gmProject-liuzhixing\test>Loop_ZBuffer.exe bunny_tri.obj 3
Successfully open file - bunny_tri.obj
Read file end. Vertices: 1506 Polygons: 1000
    Subdivision times: 1    Time cost: 0.006s
    Subdivision times: 2    Time cost: 0.025s
    Subdivision times: 3    Time cost: 0.095s
Loop Surface Subdivision Success.
ZBuffer visible surface determinating, please wait...
ZBuffer visible surface determination Success!
ZBuffer visible surface determination Time cost: 2.156s
Vsbile surface determination result writing to picture...
Picture file writing Success!
Visible Surface Determination Result visualizing, please wait...
Visible Surface Determination Result visualization Success!
Click close button to end this execution.
```

图 1

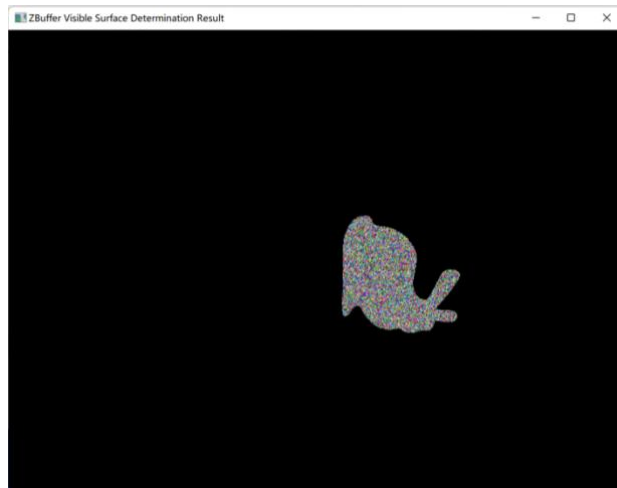


图 2

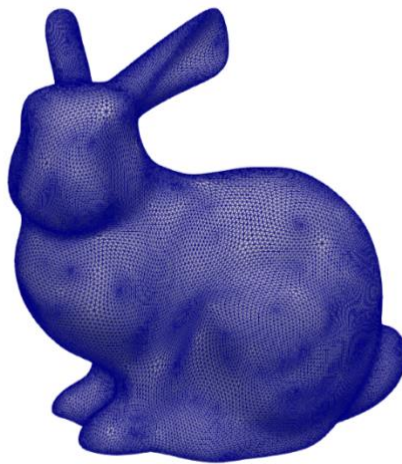


图 3

## 5. 相关说明

- 1) 迟交原因：课题组内项目结点时间与作业 ddl 重合了，实在是忙不过来。希望老师能理解。不过我也明白这完全是我个人的问题，所以因此而扣分我也能够接受。
- 2) \release 文件夹中的可执行程序、上面展示的算例运行图片、后面给出的时间结果均为我放寒假回家后在 macOS 中所装 windows11 虚拟机在 win32 位环境下编译运行的结果。由于我的笔记本电脑是苹果 M1 芯片，运行 windows 虚拟机时会存在卡顿和速度变慢的情况，因此后面给出的时间结果只能作为参考。
- 3) 本程序只能读 obj 格式的文件，loop 算法只能处理均为三角面片的模型。
- 4) 在\src\main.cpp 中修改宏定义开关以控制程序运行时通过哪种算法进行细分。

5) 本程序的可视化用的是之前计算机图形学作业中我顺便写出来的 Z-Buffer 消隐算法。

6) \src\vector.h 和 \src\vector.cpp 定义了三维矢量，是我所在的课题组所用的代码，不是我原创的。

## 6. 实验结果

### 1) Doo-Sabin 算法和 Doo-Sabin 算法的比较

以 \cases 中的几个四边形网格为例，Doo-Sabin 算法和 Catmull-Clark 算法时间上的对比如下表：

case	第几次细分	Doo-Sabin 算法	Doo-Sabin 算法
spherecylinder	1	0.004s	0.003s
	2	0.013s	0.013s
	3	0.048s	0.049s
spring	1	0.005s	0.005s
	2	0.021s	0.021s
	3	0.079s	0.083s
torus_filled	1	0.008s	0.007s
	2	0.028s	0.03s
	3	0.11s	0.123s

可以看出这两种算法的时间效率是差不多的。

对于细分结果，这里给出 spherecylinder 这一例子在两种算法下分别细分五次后的结果：



图 4 从左至右分别是：原始网格、Doo-Sabin 算法、Catmull-Clark 算法

从结果来看，相同次数的细分过后两种算法得到的网格规模相差并不多。而 Catmull-Clark 算法得到的网格在原网格的圆柱形顶面部分更加圆润一些，Doo-Sabin 算法得到的更加硬朗，特别是在距离圆柱形顶面这一平面更近的部分，二者的差别尤其显著。由此做出推断，如果想要得到更加圆润一些的表面细分网格，推荐使用 Catmull-Clark 算法。

## 2) Loop 算法

与上述两种算法不同，Loop 算法应用与三角网格细分。算例 bunny 的细分结果如图 3 所示。