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年秋学期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。

1. 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

1. 编译

请使用cmake进行编译。

1. 程序使用

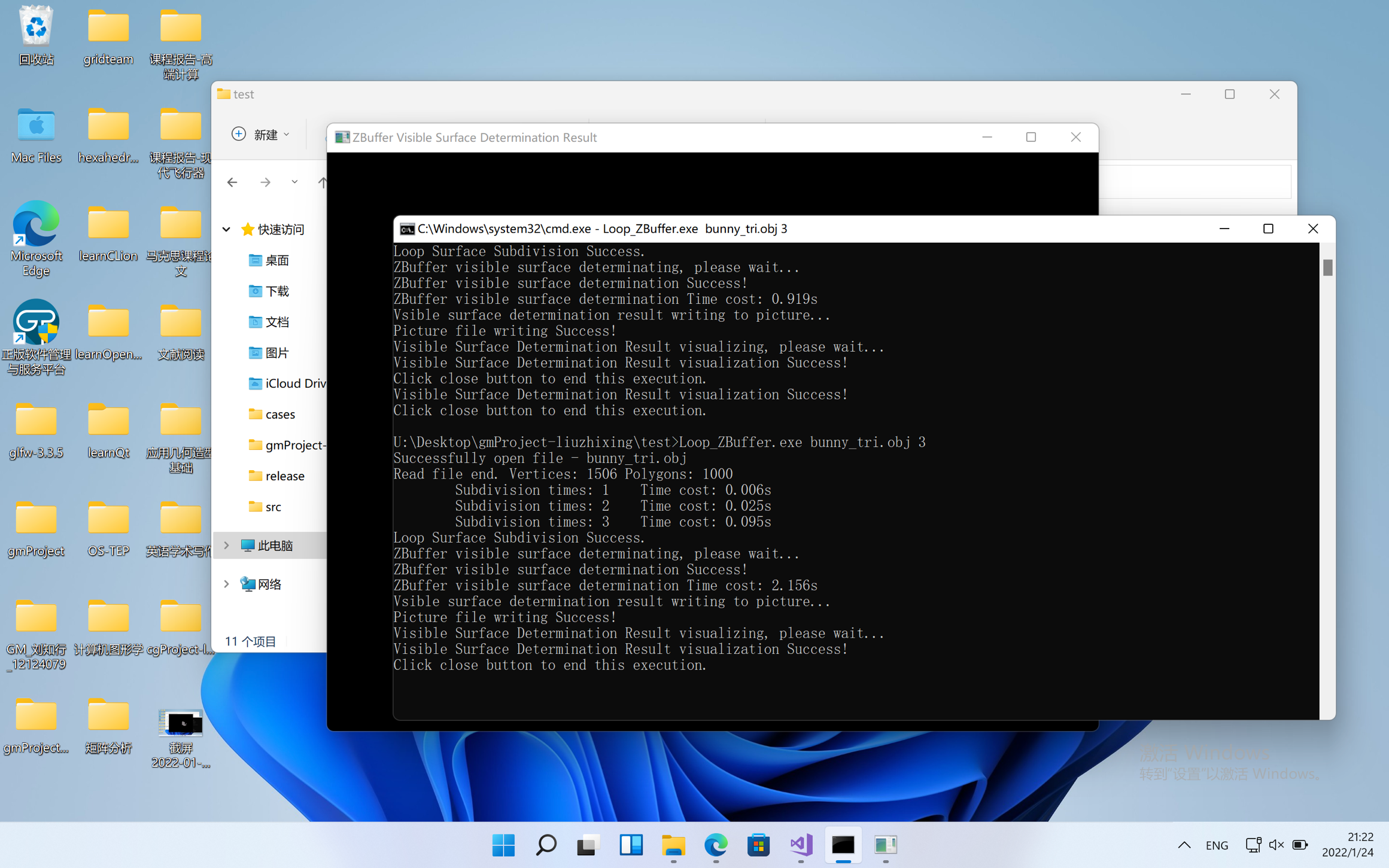
在windows powershell或cmd中使用方法如图1，命令行需输入需要进行细分的模型文件名和细分次数。读文件成功后将开始曲面细分，细分成功后将展示细分结果（用的是计算机图形学作业涉及到的Z-Buffer消隐算法），如图2所示，同时生成这一细分结果的obj文件，用三维模型查看器可以打开该模型进行查看，如图3所示。每次细分消耗的时间会在命令行中打印出来。

图1

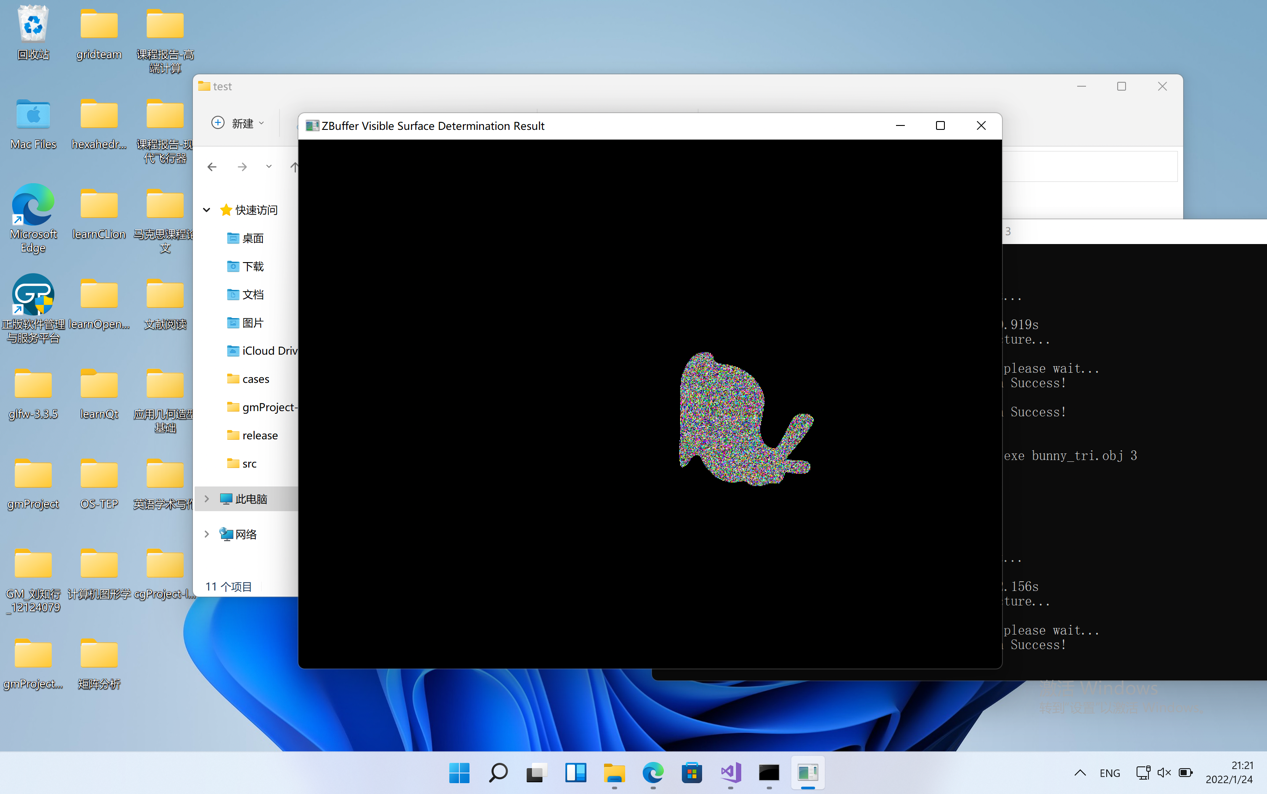
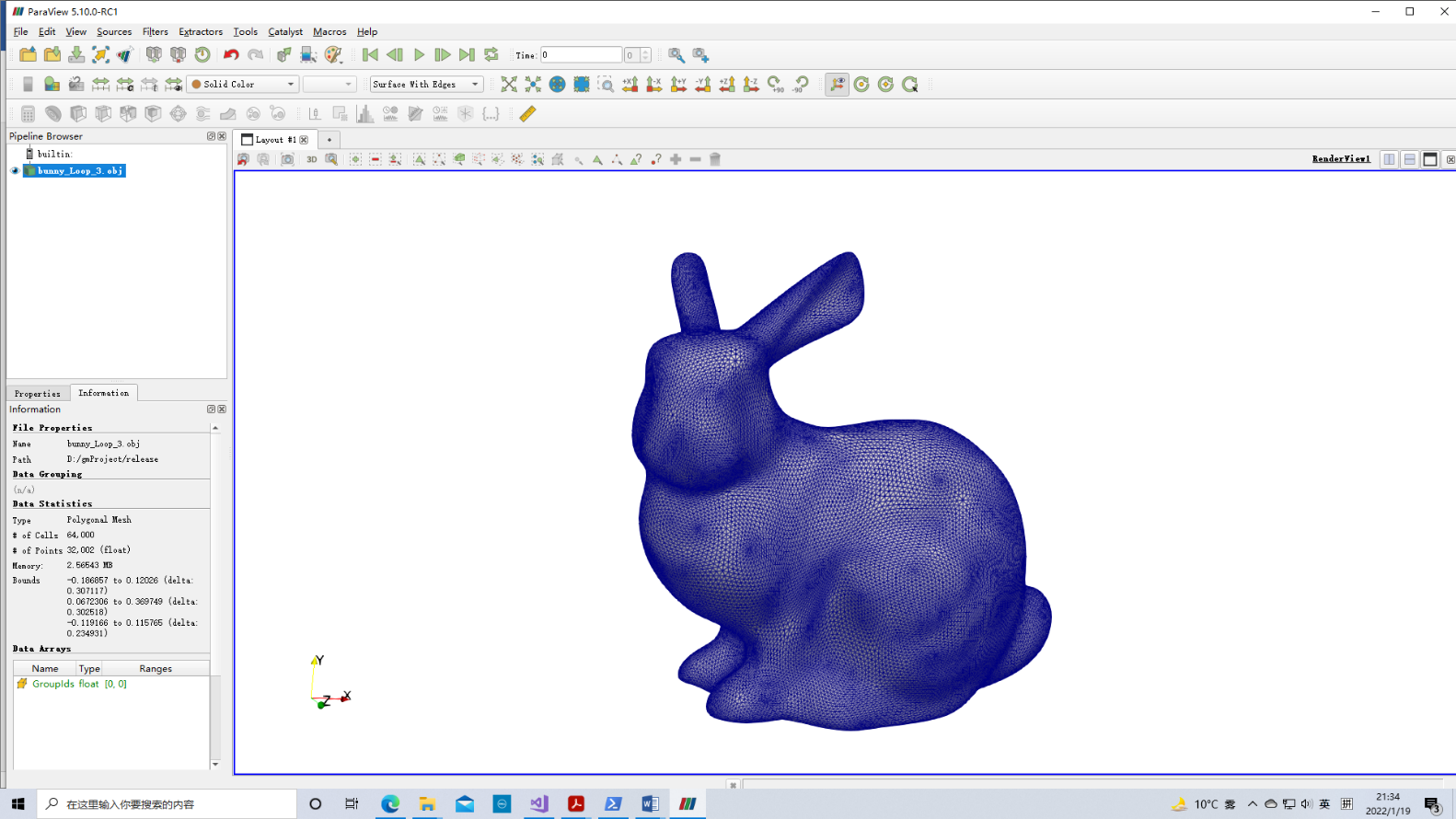
图2

图3

1. 相关说明
2. 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。
3. \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。
4. 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。
5. 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。
6. 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。
7. \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。
8. 实验结果
9. 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：

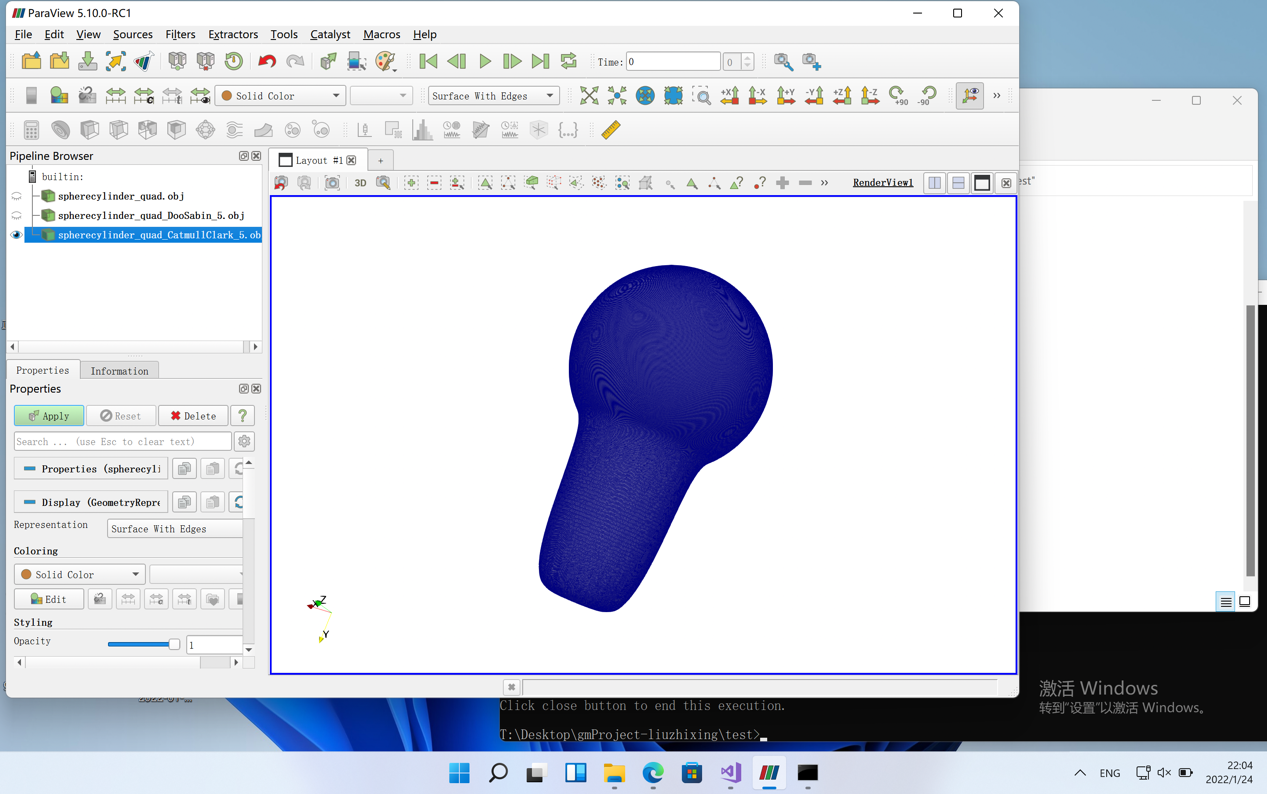
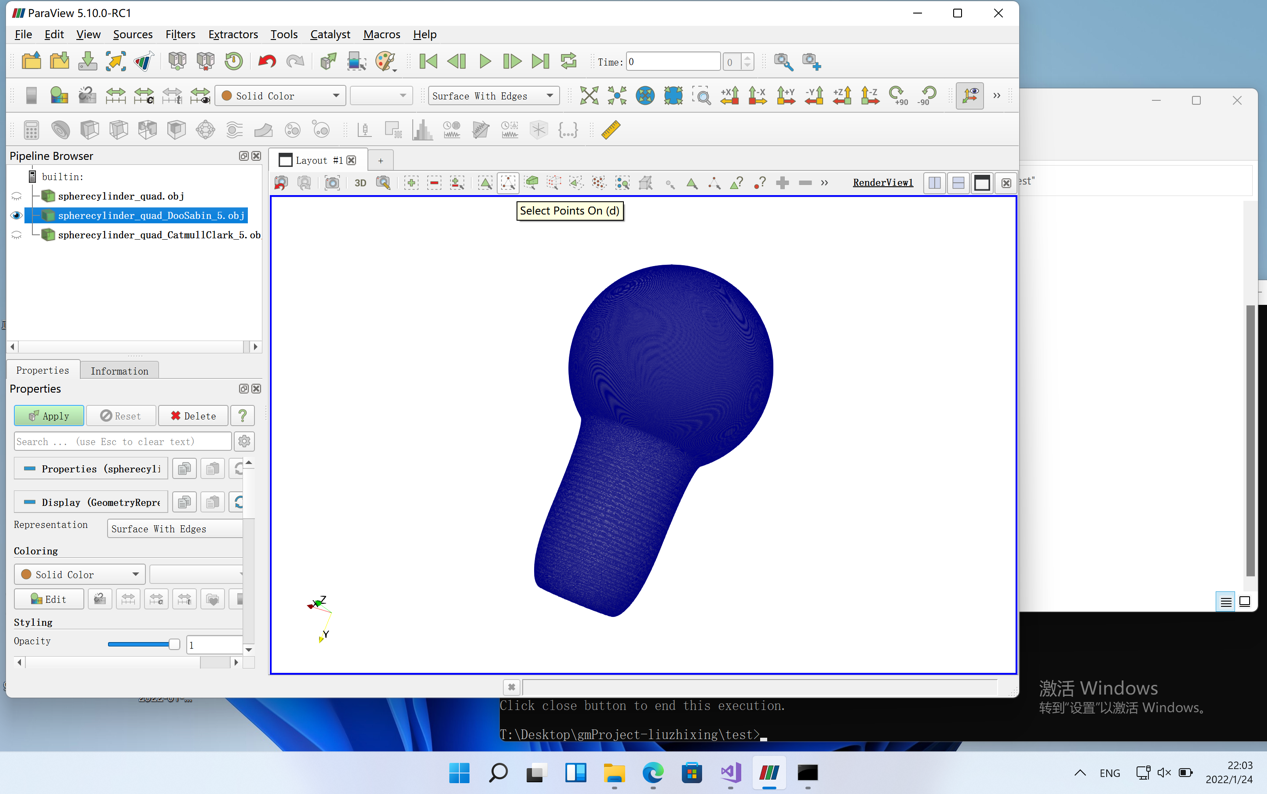
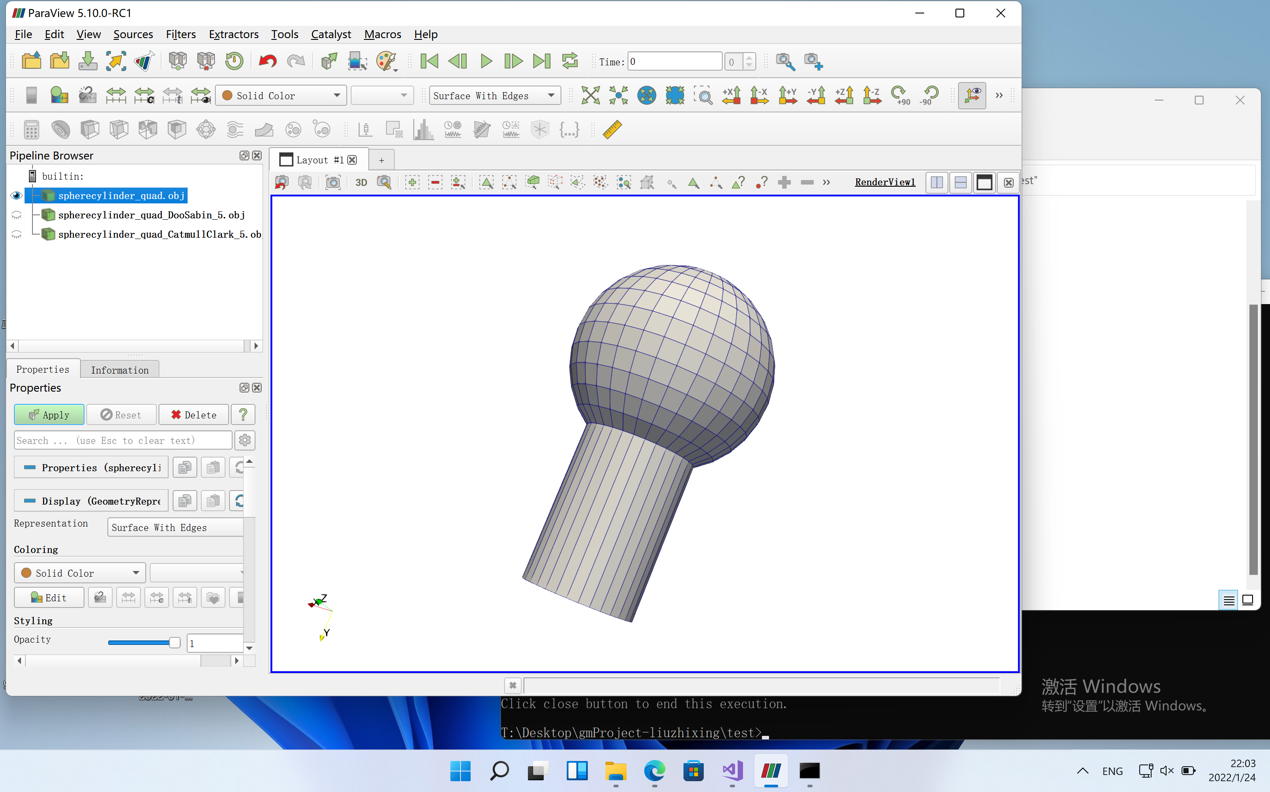


图4 从左至右分别是：原始网格、Doo-Sabin算法、Catmull-Clark算法

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。

1. 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。