

VisualSFM+MeshLab 三维重建全过程记录

原创 Ladd7 2016-11-26 16:07:43 30064 收藏 118

版权

分类专栏: 计算机视觉 文章标签: 计算机视觉 三维重建



计算机视觉 专栏收录该内容

0 订阅

5 篇文章

订阅专栏

*VisualSFM+MeshLab*三维重建全过程记录，从下载到运行出结果，以及其中一些可能出现的小问题。通过这两个软件熟悉一下多目立体视觉重建的流程。

多目立体视觉重建关键步骤有以下两点：

从运动信息中恢复场景结构 (*Structure from motion, SFM*)和多视角立体 (*Multi-viewstereo, MVS*)

大致步骤分为如下：

特征提取与匹配→稀疏重建→稠密重建→表面重建

本文使用 *VisualSFM+MeshLab*工具来过一遍上述流程。

参考来自: <http://www.guudo.cn/tech/view/id/7129>

操作系统: 亲测 *win10 win8*可用 (*ubuntu*的弄好后再更新)

软件版本: *ViusalSFM V0. 5. 26* <http://ccwu.me/vsfm/>

CMVS+genOption+PMVS2<http://francemapping.free.fr/Portfolio/Prog3D/CMVS.html>

MeshLab 1. 3. 3 x64 (这里注意，如果重建场景比较大的话，32位软件会提示使用64位版本)

-1、环境准备

上述内容以打包上传: <http://download.csdn.net/detail/u012750702/9694352>

首先下载解压完成后，将 `.. \VisualSFM\CMVS-PMVS-master\binariesWin-Linux\Win64-VS2010` 文件夹下的内容，如图：

VisualSFM > CMVS-PMVS-master > binariesWin-Linux > Win64-VS2010			
名称	修改日期	类型	
cmvs.exe	2015/3/23 20:29	应用程序	
genOption.exe	2015/3/23 20:29	应用程序	
msvcp100.dll	2015/3/23 20:29	应用程序扩展	
msvcr100.dll	2015/3/23 20:29	应用程序扩展	
pmvs2.exe	2015/3/23 20:29	应用程序	
Readme.txt	2015/3/23 20:29	文本文档	
ws2_32.dll	2015/3/23 20:29	应用程序扩展	

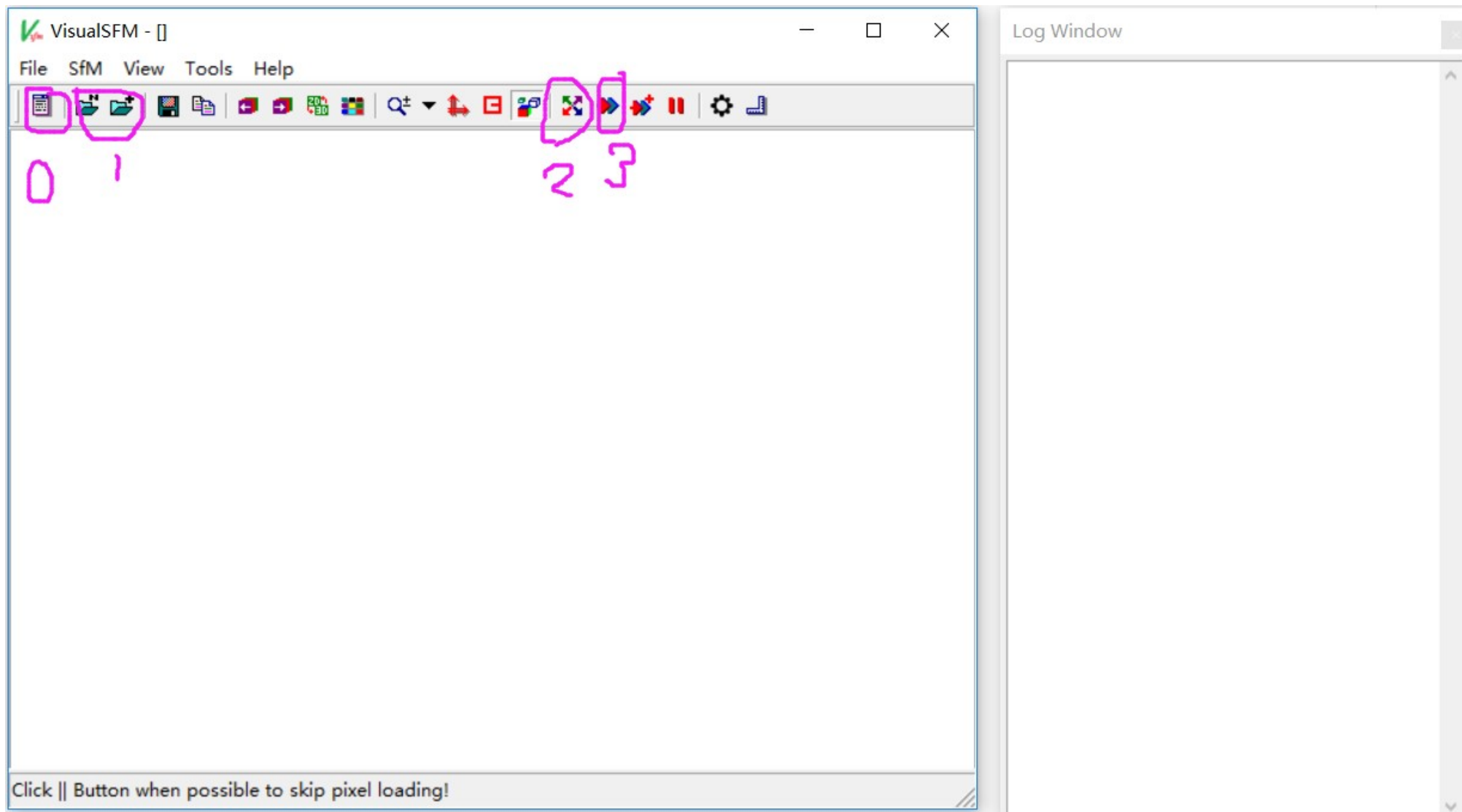
复制到 `..\VisualSFM\VisualSFM_windows_64bit\VisualSFM_windows_64bit` 文件夹下:

VisualSFM > VisualSFM_windows_64bit > VisualSFM_windows_64bit		
名称	修改日期	
log	2016/11/10 13:24	
glew64.dll	2010/6/3 1:19	
nv.ini	2016/11/10 13:24	
pba_x64.dll	2012/12/14 3:16	
README.txt	2015/3/11 11:56	
SiftGPU64.dll	2012/10/12 4:46	
VisualSFM.exe	2015/4/27 12:19	

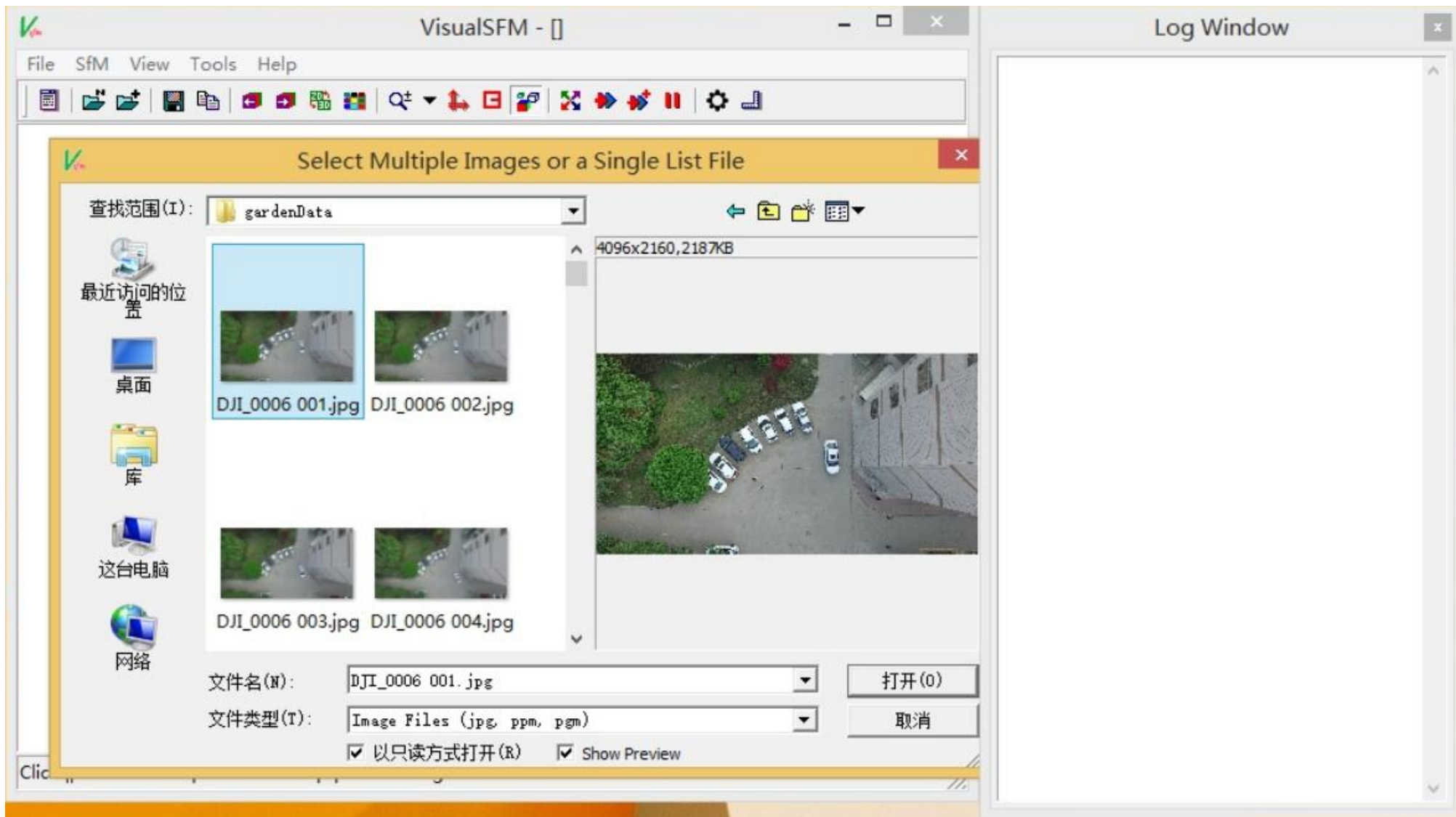
因为 *VisualSFM* 在进行稠密的时候会调用 *pmvs* 和 *cmvs* 的部分，否则运行会报错。

0、图片导入

双击 `.exe` 运行:



其中按钮0用来显示日志界面（即上图右侧那个界面），可以查看运行进度，如果运行时找不到在哪里显示了点击此按钮唤出。
按钮1导入图片或者运行文件，可以一次选择多张图片：



这里注意，运行过程中会在与图片相同目录下生成中间文件（建议图片文件夹先做备份），如图：



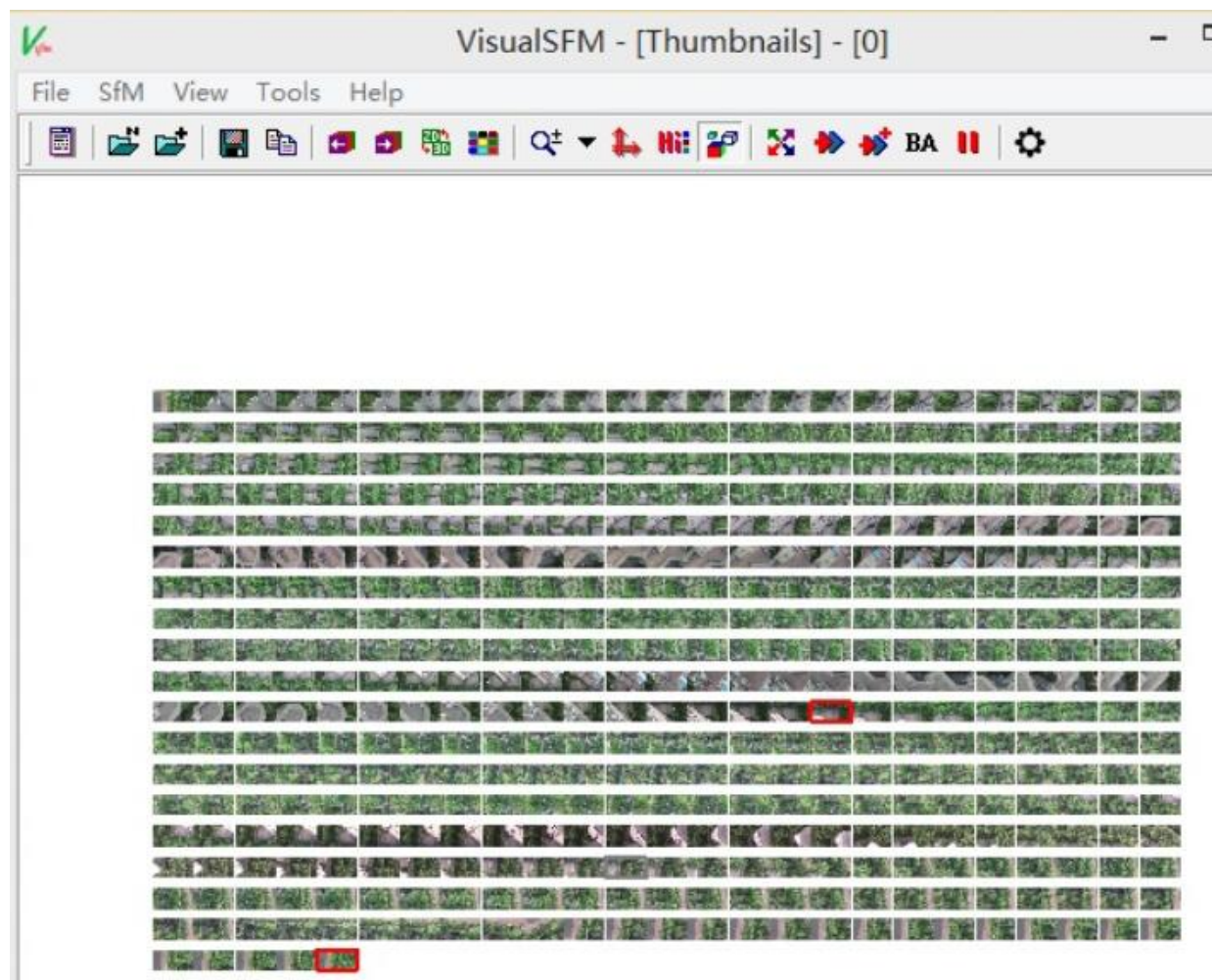
其中包括 *sift* 求解结果和 *match* 的结果。

提示：导入的图片文件不全在同一文件夹下也可以，之后生成的中间文件会跟对应的各自图片文件在一起。

导入并加载结束时会输出：*Loading image pixel data ...done in XXXs*

注意：导入过程中如果出现 *done in 0s* 的情况是电脑内存不足造成的，也就是数据量太大了。

所有照片加载无误后：



可进行下一步，

有时候个别照片的缩略图会不显示，不要紧。

一、特征提取与匹配

这一步旨在寻找多个视角下同一空间点之间的位置关系，使用描述子提取图片中的特征点后，在其余图像中进行特征点匹配，匹配过程中还需确定特征的尺度、方向等特性。通俗来讲就是找到不同二维图片中拍摄到的同一个三维点。

*VisualSFM*用的是changchang wu写的siftGPU算法，提取每幅图像中的特征点，并利用GPU加速计算。

点击按钮2, 开始进行 *sift* 和 *match* 步骤。

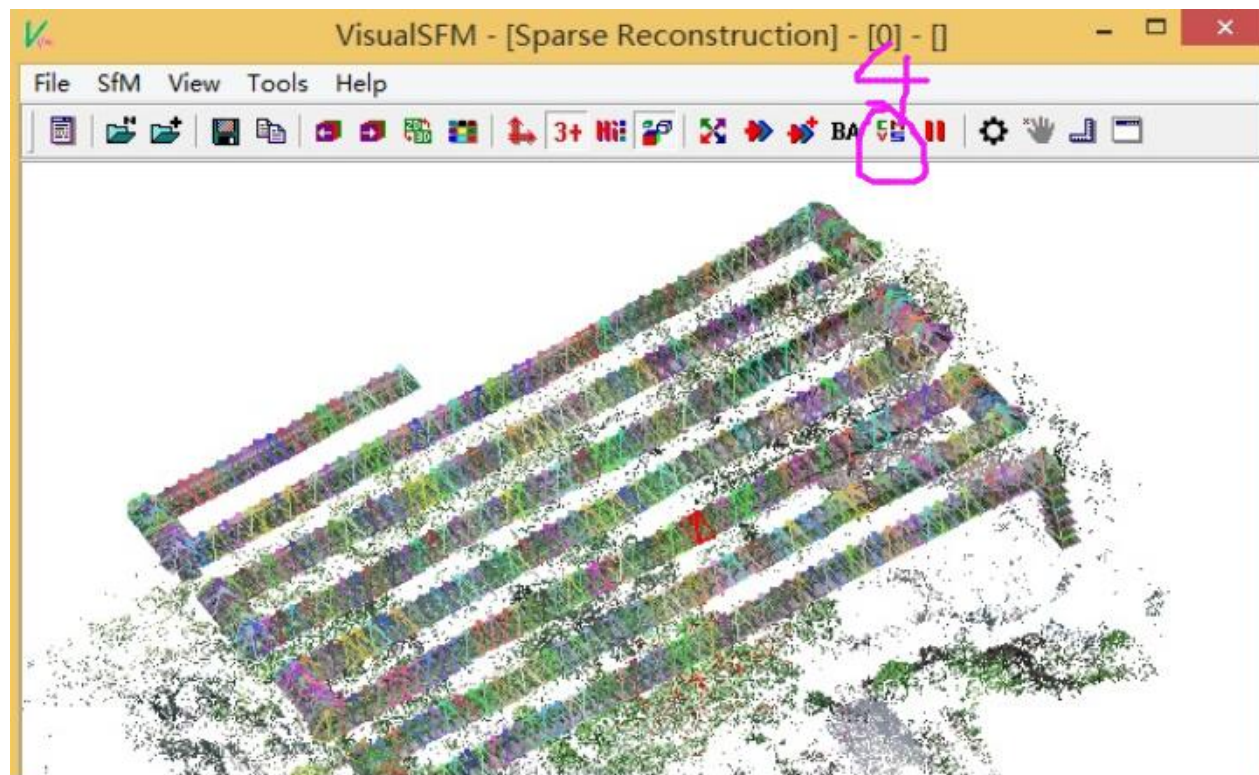
运行过程会占满 *cpu*。

Log 界面可查看进度, 此步骤需要等待一段时间。

二、稀疏重建

利用上一步特征提取和匹配后的结果, 求解每张图对应相机的相机姿态和稀疏的三维点, 这里稀疏的三维点来自于上一步匹配后的特征点。

点击按钮3, 开始稀疏重建, 稀疏重建结束后界面会变成这样, 可以看到相机位置及对应图像已经被求解出来了。

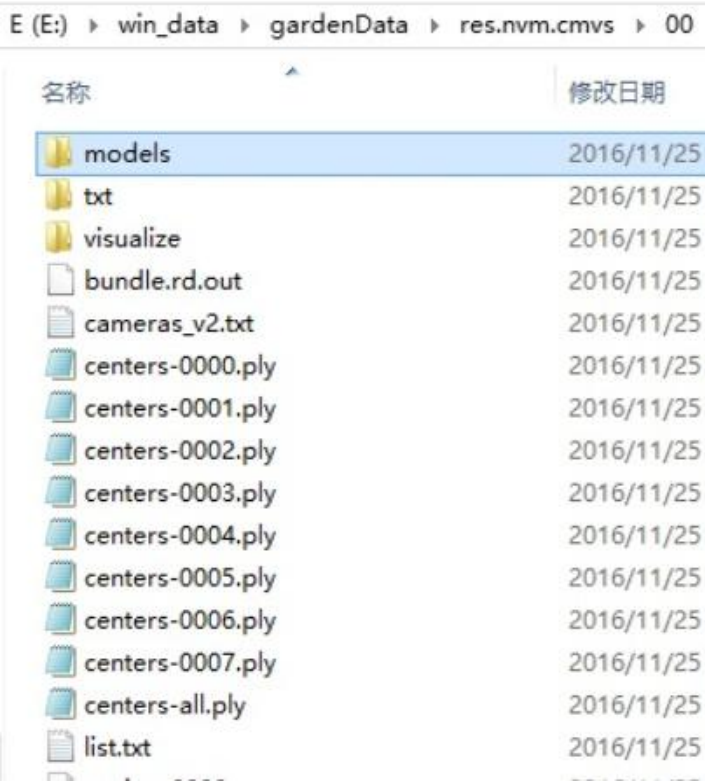


三、稠密重建

稠密重建使用了 *CMVS/PMVS* 算法, 其中 *CMVS* 对图像序列进行聚类, *PMVS* 利用稀疏重建结果将其向空间周围扩散得到有向点云, 同时利用局部光度一致性和全局可见性约束完成稠密重建过程。

点击按钮 4 进行稠密重建, 这个时候需要选择存盘文件名和保存目录, 选择目录 `../gardenData`, 文件为 `res`。

稠密重建结束后，到刚刚选定的目录下，会有 *res. 0. ply* 和 *res. nvm* 文件和 *res. nvm. cmvs* 文件夹：



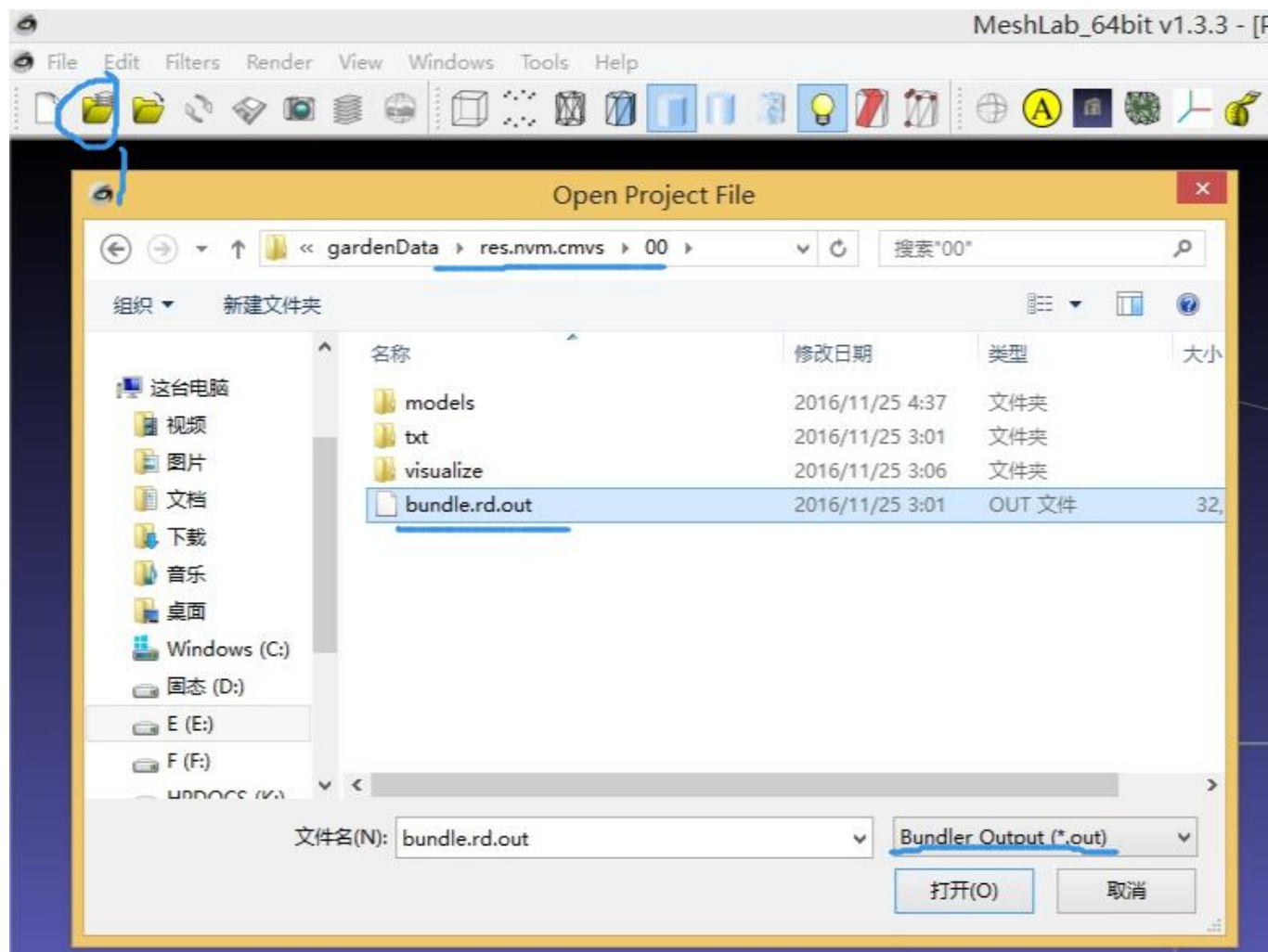
其中，*res. nvm. cmvs\00*（这里有可能不是00而是其他数字）文件夹下的 *centers-0000. ply* 等文件保存着重建后相机坐标文件，*res. nvm. cmvs\00\models* 文件夹下保存着 *option-0000. ply* 文件，也就是稠密点云。当场景比较大的时候会保存不止一个文件，*centers* 文件与 *option* 文件的编号是对应的。此步骤也会占满 *CPU* 运行。

在日志界面可以看到进度，*option* 文件会逐一生成，剩余时间可通过 *centers* 文件个数估计。

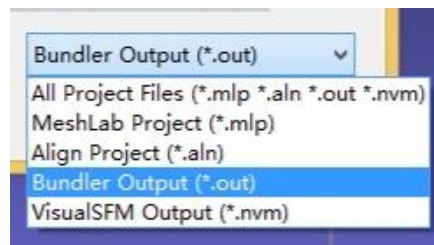
四、表面重建

表面重建用的是 *MeshLab* 提供的泊松表面重建 (*poisson surface reconstruction*) 算法。

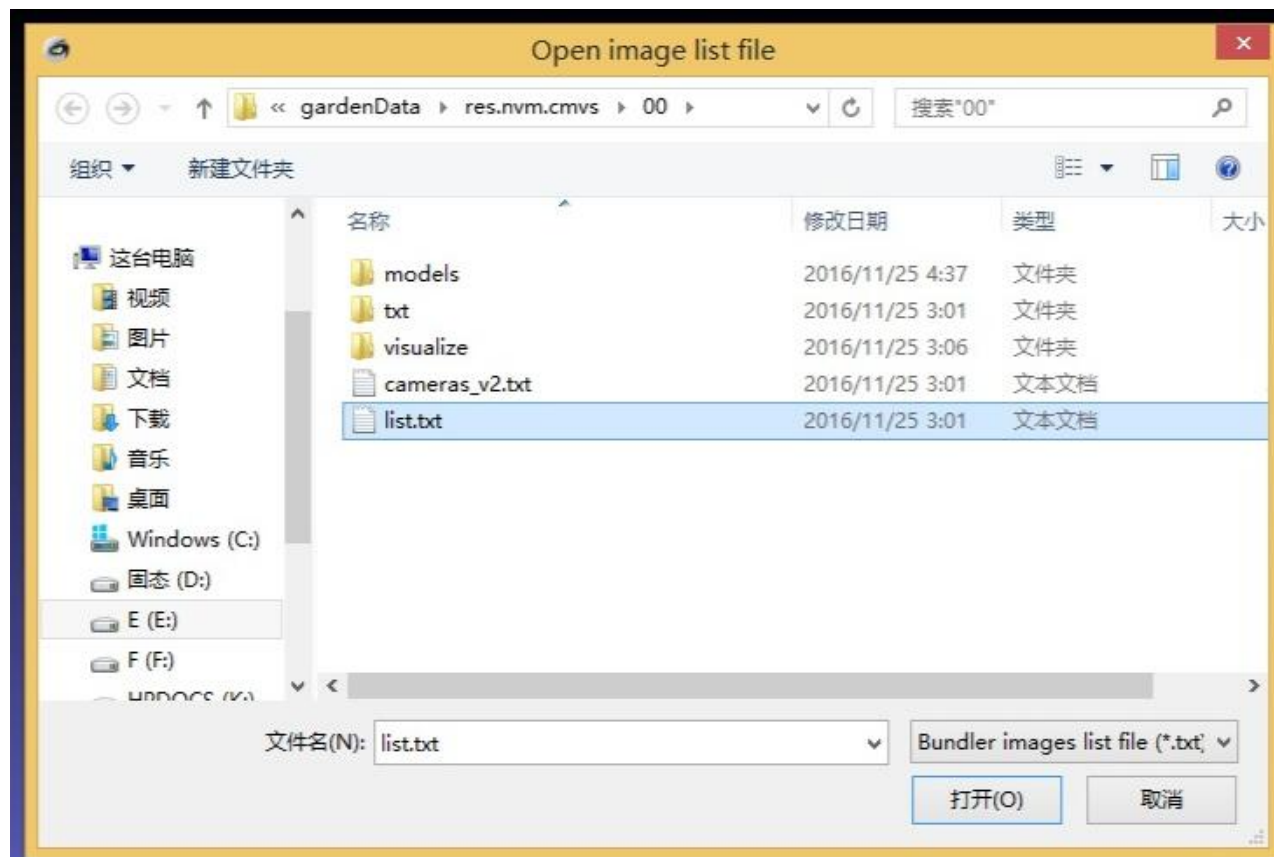
关闭 *VisualSFM* 打开 *MeshLab*，大致过程如下，点击按钮 *I* 打开工程文件：



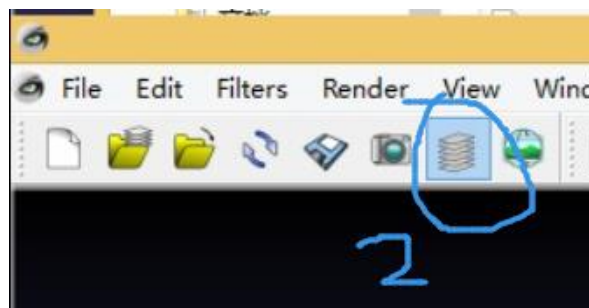
此文件在上一步保存目录的00文件夹下，*bundle.rd.out*，没找到的话记得修改一下显示文件的拓展名：

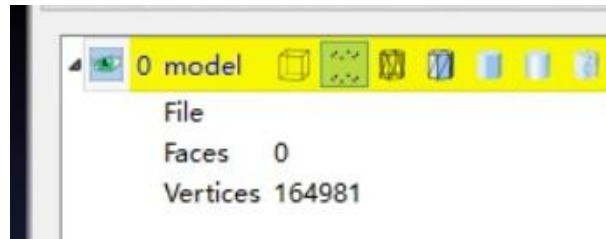


之后会又弹出一个窗口选择同一目录下的 *list.txt* 文件：



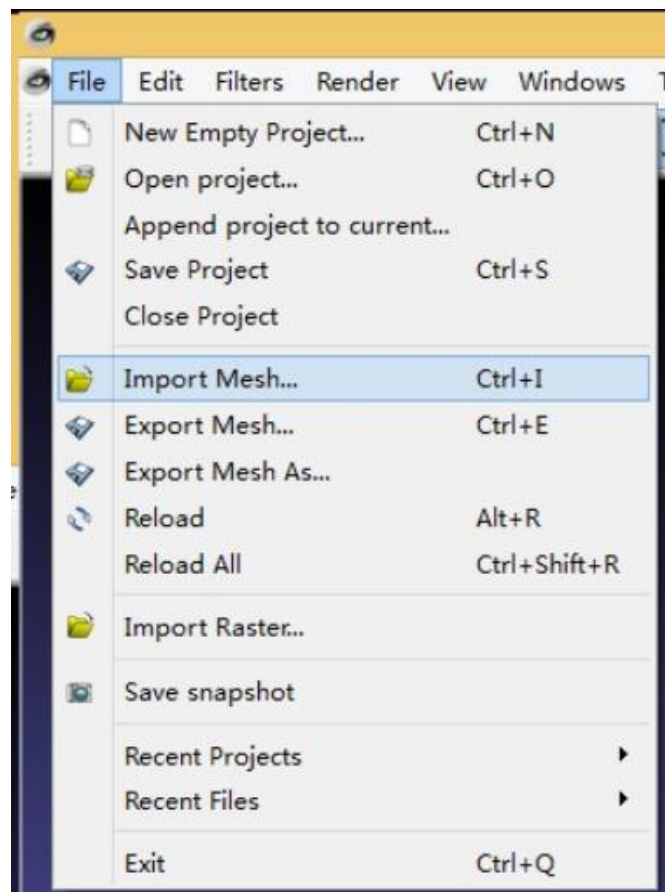
导入图片和系数点云时会耗费一些时间，软件可能处于卡死状态，不要强制关闭。结束后点击按钮2，界面右侧会出现文件列表，如图：



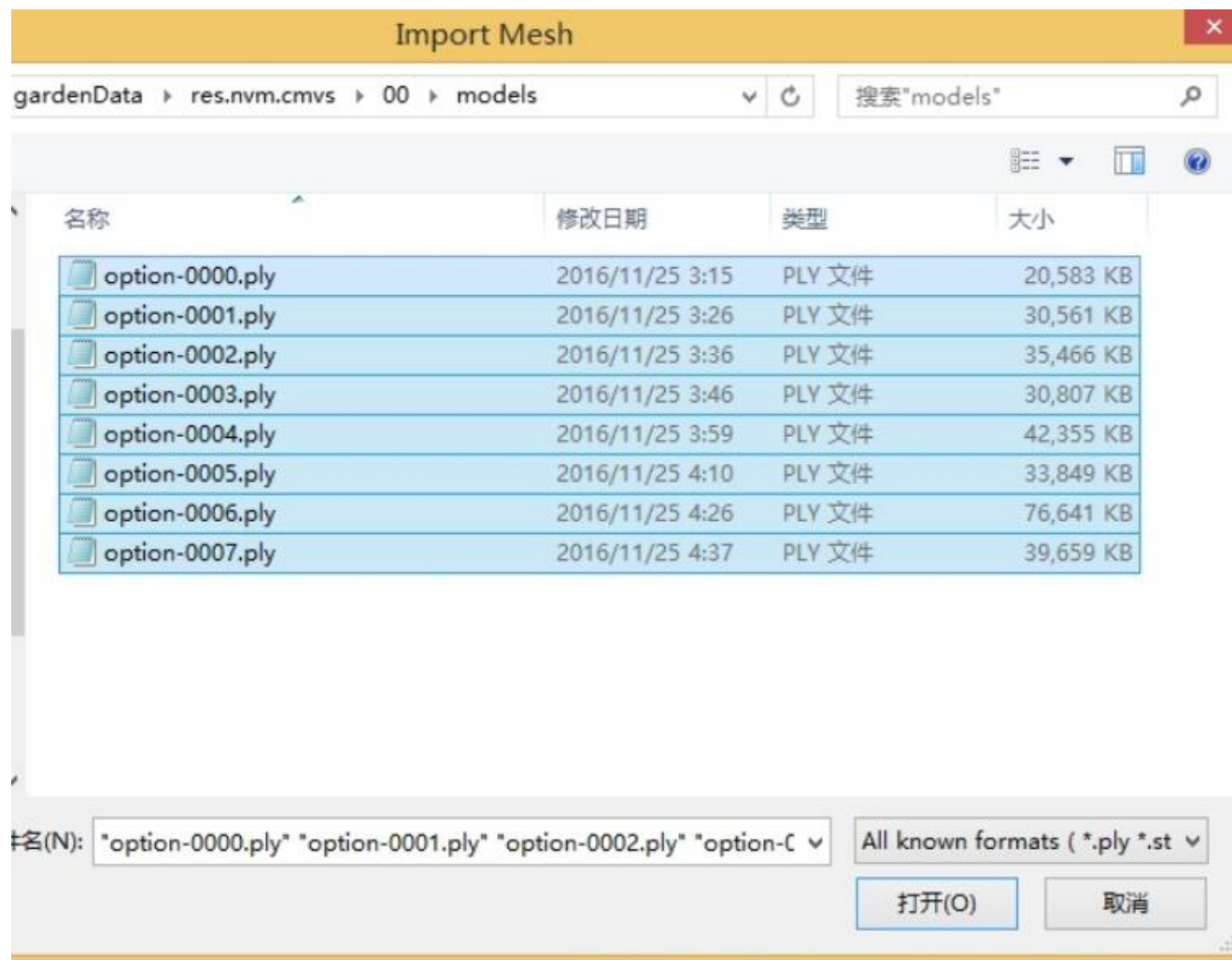


点击“眼睛”图标可操作本模型是否显示，图标前的小三角可查看模型顶点/面片个数。

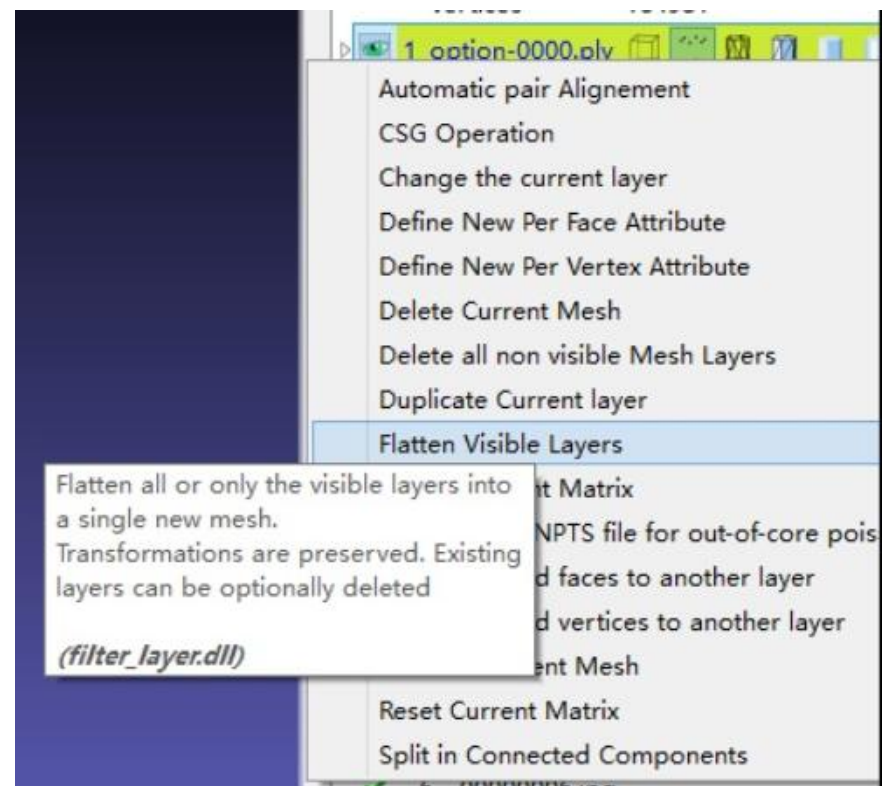
下一步选择菜单栏->File->Import Mesh:



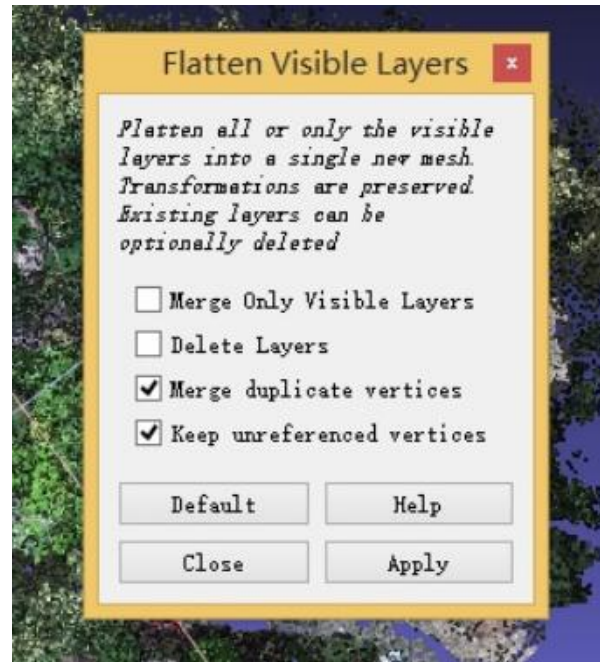
导入稠密点云文件，文件在../gardenData/res.nvm.cmvs/00/models路径下：



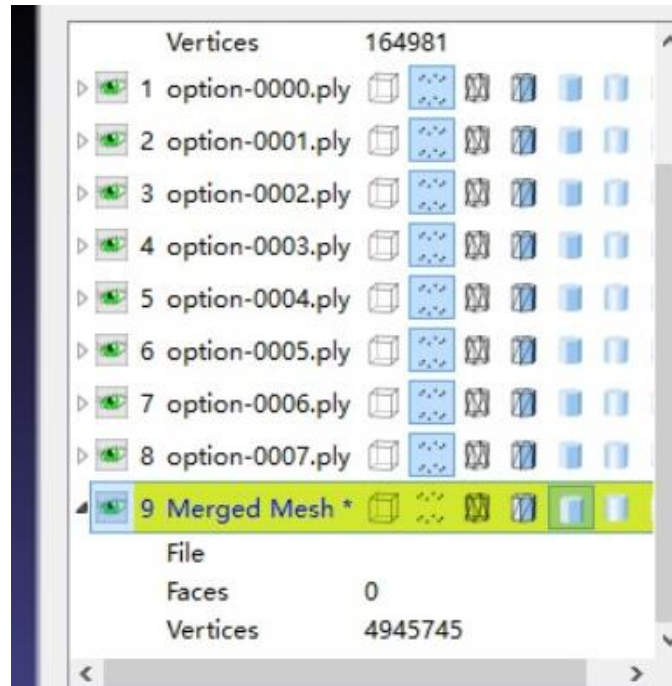
接下来合并点云（若只有一个文件可跳过此步），在随便某个点云文件上右键->*Flatten Visible Layers*->修改选项->*Apply*;



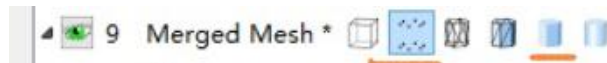
选项如图:



*Apply*后会多出一个名为*Merged Mesh*的文件（注意，此文件未存盘，如果要保存的话需要进行导出操作），看一下顶点个数，若为0则可能是上一步设置有问题：



若想查看合并后的结果需要关闭其余文件的显示（点击“眼睛”图标），并修改*Merged Mesh*文件的显示方式（默认*Flat*改成*Points*）：



提示：

重建的过程中是会有新文件产生的，如果对效果不满意，可以重复进行某些步骤，这样做不会覆盖产生的文件，直至结果满意为止，选择*export*导出文件，将结果存盘。

如果过程中出现某些失误影响到了，只要不导出存盘就不会影响打开的原始点云数据。

如果像我这样重建场景很大的话中间需要耐心等待（程序可能会显示未响应），*MeshLab*跑的时候不会满*CPU*的跑（*VisualSFM*会），我的*CPU*占用率一直在20%多（仅供参考）。

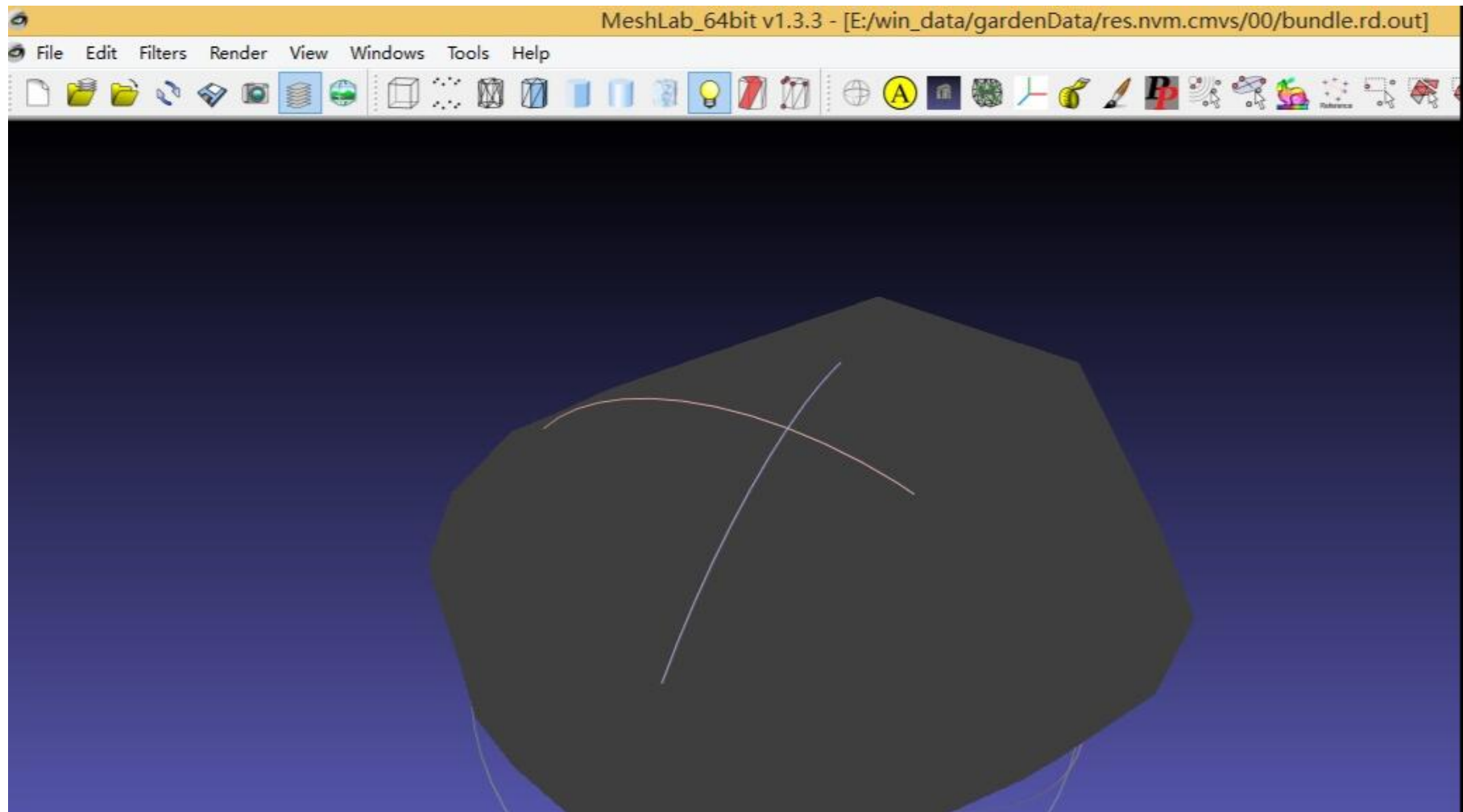
接下来：

1、

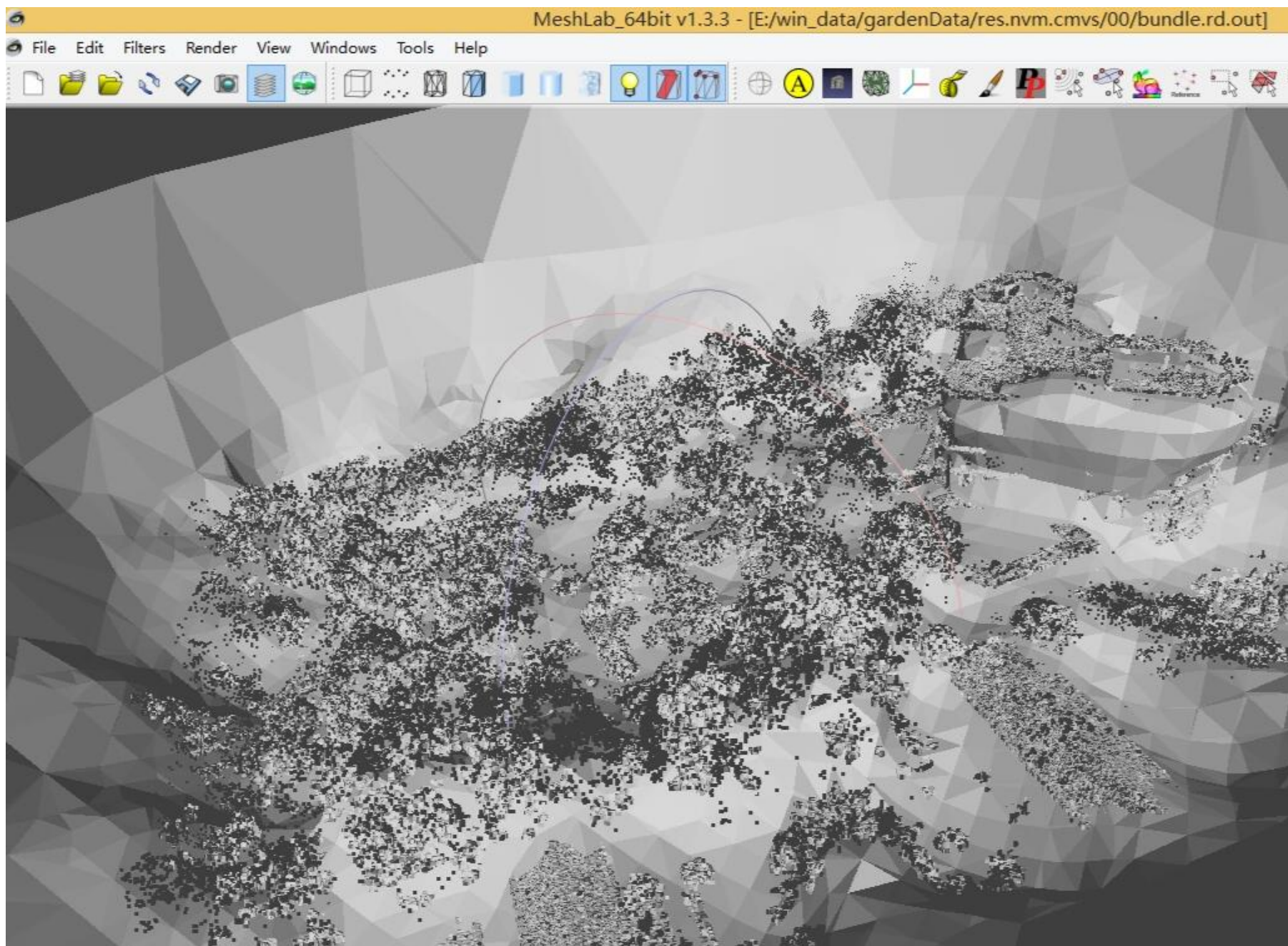
选中*Merged Mesh*文件，菜单栏中*Filters*-> *Point Set*-> *Surface Reconstruction: Poisson*，八叉树深度控制着细节精细程度，可根据需要自行调整。

2、

完成后会形成一个“气泡”将模型包起来（也有可能没有，只是延展出几片面）：



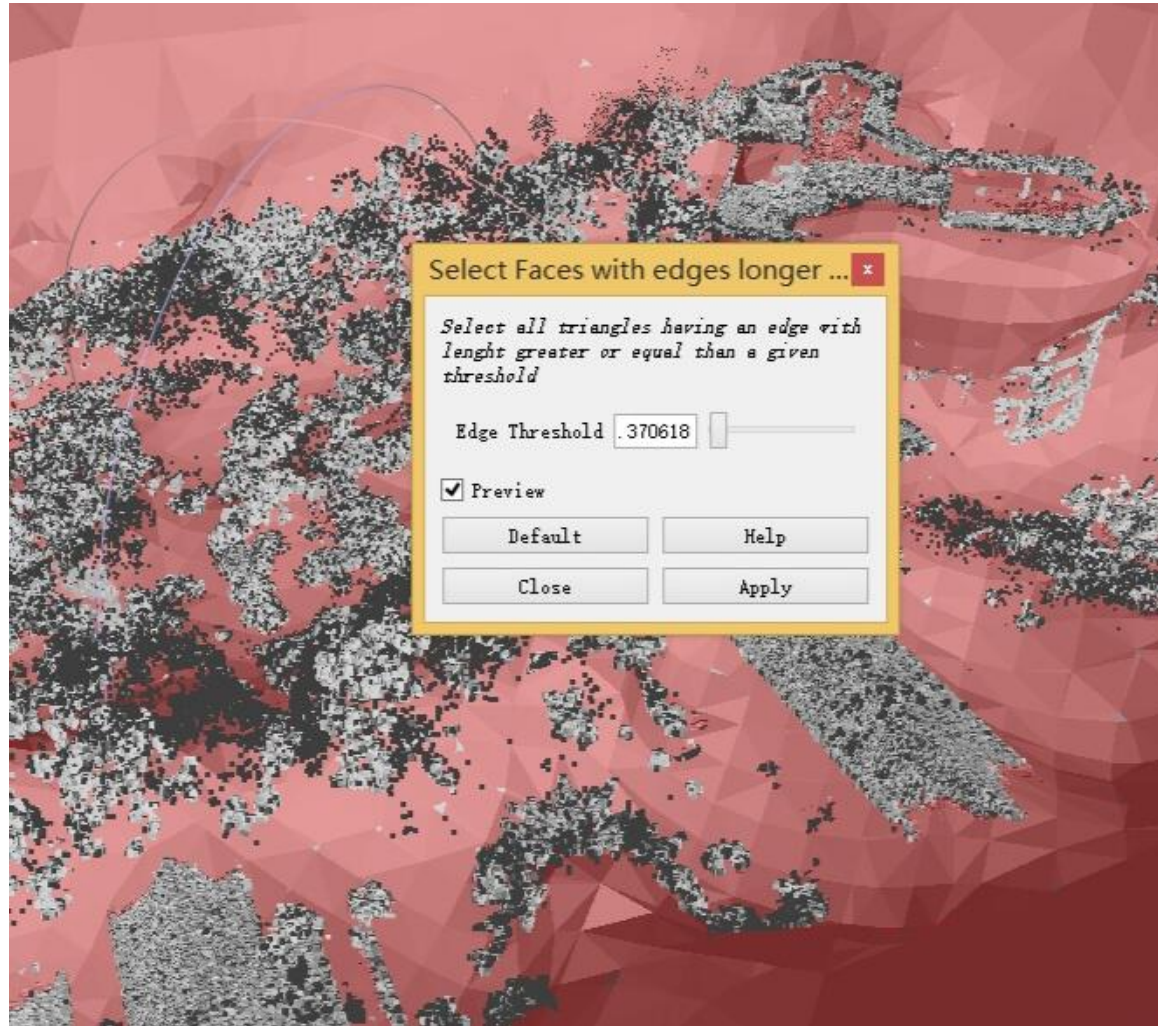
这时不用怀疑哪里出问题了，滚动鼠标滚轮，视角钻进气泡里，可以看到模型好好的在里面：



如果没有形成多余的面，检查一下进行此步骤时选中的文件是否正确。

3、

然后, *Filters -> Selection-> Select faces with edges longer than* 删除多余的面, 这个时候记得把*preview/预览*打开, 同时其余文件取消显示, 红色的面片就是选中需要删除的:



调整门限到自己认为合适的地方, 软件会根据模型给出个默认门限, 调整乱了后可以点击*Default*回到默认值。

apply选中相应面片后，点击工具栏中的删除面片的图标将选中的面删除：



4、

修复流型边缘，*Filters->Selection-> Select Non-Manifold edges*，同上，选中然后删除。

5、

参数化和纹理投影，菜单栏->*Filter->Texture-> Parameterization + texturing from registered rasters*，要设置贴图大小，一般是2的幂次：1024\2048等等：



我重建的场景比较大，这一步耗费了很长时间，程序会处于未响应状态，并且进度条没有提示。

当看见模型显示出颜色后，即可：



FOV: 60
FPS: 8.5

Current Mesh: Poisson mesh(2)
Vertices: 846233 (10781881)
Faces: 1688505 (1695445)
Selection: v148 f0