# 说明

## 解决了什么问题？

老版本里核心算法是聚类，树和电力线离得太近聚成一类，无法进行后续的识别。

新版本核心算法是基于多种特征的多类分类器。

## 2，开发了什么功能？

### 1，QColorTool.exe

功能：三维浏览、三维选取、编辑点云、修改颜色、简化点云

### 2，PointCloudTool.exe

功能：批量训练、点云分类

调用示范：

* PointCloudTool.exe --help
* PointCloudTool.exe --inputfile testdata\powerline.las --classdir classdir --retrain false

## 3，最终提供什么接口？

目前做到的是输入一个las文件，输出一个las文件（以颜色区分类别）。

最终做到的是输入一个las文件，输出一个点云obj和一堆模型obj（待确定，取决于场景需要什么数据，pnts还是obj，或者是两者交叉？）。

## 4，面临哪些问题？

1. cgal的训练算法是基于随机森林分类器，它的特点是适合小数量样本训练，与最初希望大量训练有些不符。
2. 分类器返回的结果是每个点属于什么分类，并没有把每个实体区分开。比如说场景有两棵树，程序认为这些点是树，并不能把两棵树区分开来。但是幸运的是，老版本已经做过相关的工作，我把老版本的一些功能合并进来就好。
3. 对铁塔样本的识别效果很差，去掉铁塔样本就没问题了。
4. QColorTool.exe辅助工具里选取内容为空时容易崩溃，也特意加了锁防重入之类的还崩，它是事件驱动的，但是事件还没有接收就崩了，要解决的话估计得调试pcl源码了。
5. 程序目录里还有qt的动态库依赖，是因为cgal的源码里使用到很多qt的东西，为了节省时间我没有管它。如果不想看到这些dll，剩余工作做完后我把qt静态编译一下，或者把qt的东西都去掉。

## 5，接下来做什么？

1. 样本文件已经训练了，试了两份数据，质量可以支撑后续的步骤，但是为了保险起见，我把数据试完。
2. 把分类结果再分割为单个实体（见4-2）。
3. 把老版本的危险距离检测和电力线建模加上。