

RadarCAS 使用说明

此项目用于实时接收以及显示雷达和 GPS 信息，并累计到一定数量以后计算特征，在第二次测量时，将特征和第一次测量时的特征做对比，并输出对应的 GPS 信息。在之后的详细说明里，会把第一次测量成为 prior 过程，第二次测量称为 unregistered 过程。

目录

RadarCAS 使用说明.....	1
项目结构	2
软件界面介绍	3
工具栏.....	3
雷达配置.....	3
GPS 配置.....	5
测量轮设置.....	5
频率设置.....	6
模拟文件设置	7
融合 GPR	7
载入 Prior 数据.....	7
画图板.....	8
状态栏.....	8
软件使用	9

项目结构

RadarCAS 项目

- **res**: 项目所用到的图标文件
- **connexions**: 连接 GPS 以及雷达的类文件
- **configurations**: 存放配置设置界面的类文件
- **value**: 语言本地化文件 strings, 以及缺省文件路径定义: respath
- **test**: 单元测试文件, 主要用于测试项目内不在类里的函数
- **tensorflow**: 神经网络 core 代码
- **nets**: 神经网络拓展部分
- **logs**: 暂时没用上, 如果把 appconfig 中 logging 设定为非 DEBUG 模式则 logs 会输出到此文件夹内
- **utils**: 文件解析, 格式转换等函数
- **scrachMesh**: 请忽略这个文件夹, 它是草稿纸
- **data**: 默认存放数据的文件夹
 - **mocks**: 默认存放模拟数据的文件夹
 - **performance**: 存放性能测试数据文件夹
- ***.py**: 此项目的主要图形界面以及数据控制的代码
 - **mainframe.py**: 这是代码的主体, 数据采集控制, 配置控制以及算法控制都在这个文件中, 可以说是 MVC 模型里的 controller。它包含了主界面窗体类以及 3 个线程类。
 - **findwaytohome.py**: 此文件是算法, 定义了如何对数据进行切片, 以及不同测量中的特征计算及后续处理。
 - **errorhandler.py**: 此文件定义了项目里可能出现的错误, 以及错误代码。供用户查阅。

- **appconfig.py**: 这是一个 python 代码写的配置文件，可以配置软件语言（中文，英文），数据保存地址，以及雷达，GPS，日志等等的缺省配置，其中大多数都可以在软件里修改。

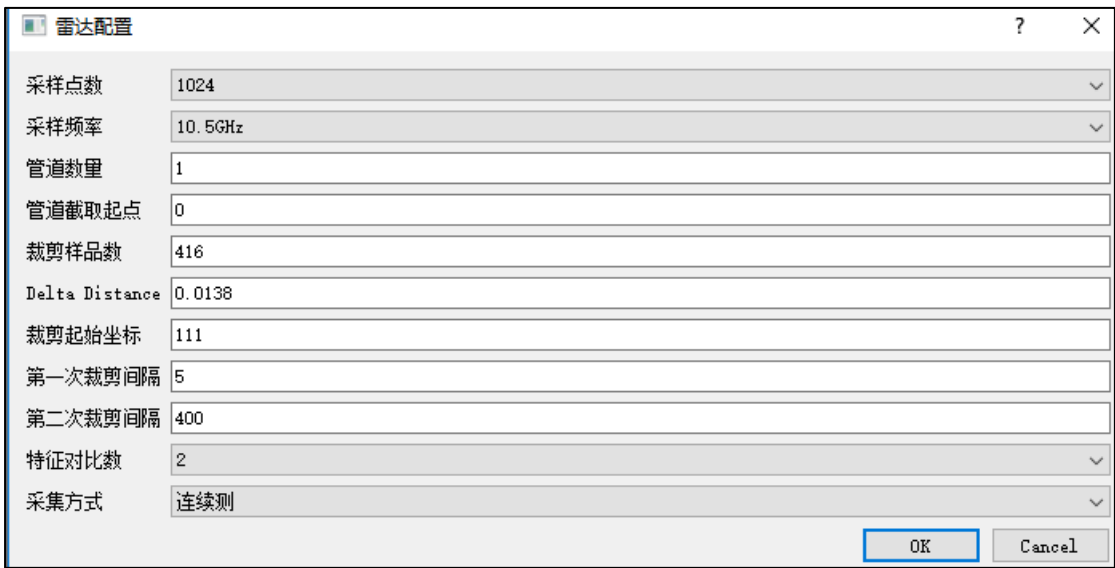
软件界面介绍

工具栏



当前保存路径显示的是，数据保存的文件夹路径。

雷达配置

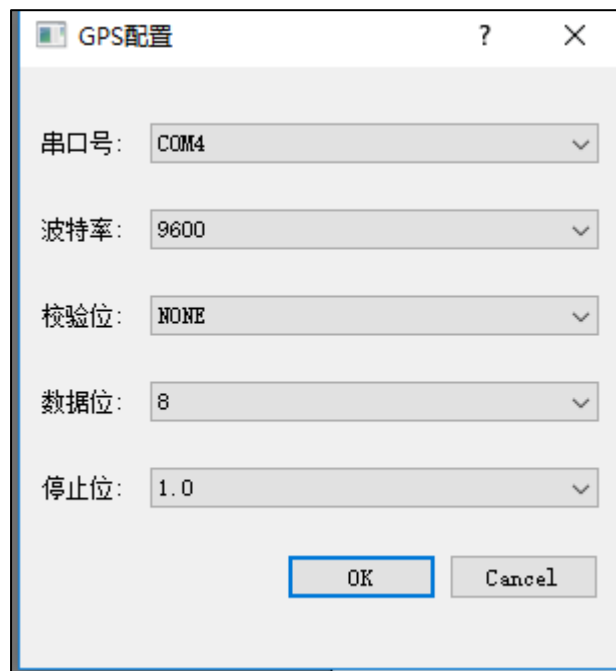


雷达配置包含了雷达的基础配置以及一部分算法配置。比如，从裁剪样品数开始到特征对比数都属于算法里的配置。

- **采样点数**: 设置雷达的采样点数
- **采样频率**: 设置雷达采样频率
- **管道数量**: 设置所接雷达的管道数量
- **管道截取起点**: 设置采样的管道起点，比如 8 管道的雷达，如果需要第二管道数据则写 1，（从 0 开始计算）。

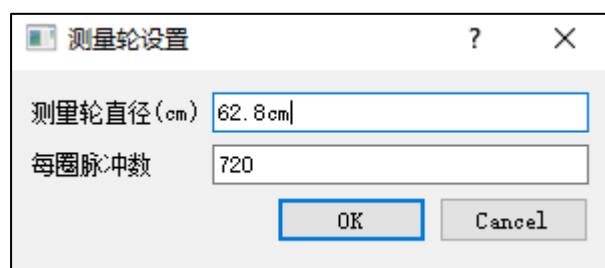
- **裁剪样品数**：算法中所需要的窗口大小，默认为 416，不可更改
- **Delta Distance**：prior 测量和 unregistered 测量的起点差距
- **裁剪起始坐标**：由于算法不需要所有点数，只截取中间 416 个点，裁剪起始点就是设置从哪一个点开始截取。
- **第一次裁剪窗口间隔**：prior 过程中，为了获得尽可能多的道路特征，默认设置为 5，意思就是每多 5 条雷达数据则计算一个特征。
- **第二次裁剪窗口间隔**：unregistered 过程是为了比对 prior 过程获得的特征并得出对应的 GPS 信息，间隔不需要密集，默认 400 条雷达数据计算一个特征并进行比对。
- **特征对比数**：在算法里的 unregistered 过程中，需要用当前最新的特征以及上一个特征做对比，对比的数量可以在这里设置。
- **采集方式**：配置雷达采集方式，可以是连续测，也可以是测距轮。

GPS 配置



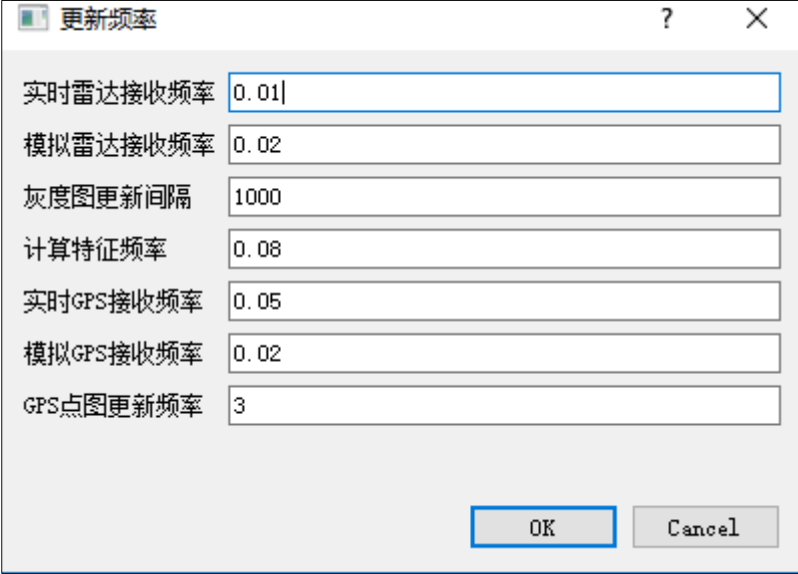
GPS 配置只包含了 GPS 的连接配置，具体需要用户自己查看计算机设备管理器中的串口号。

测量轮设置



设置轮子的参数，当用户在使用测量轮的情况下能正确计算雷达的移动距离。

频率设置



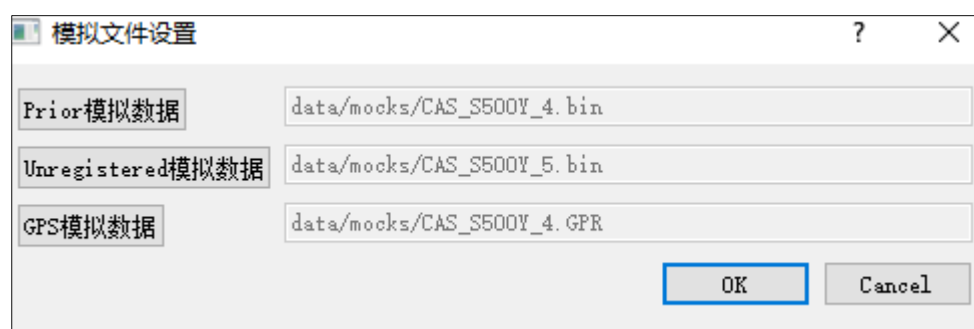
实时雷达接收频率	0.01
模拟雷达接收频率	0.02
灰度图更新间隔	1000
计算特征频率	0.08
实时GPS接收频率	0.05
模拟GPS接收频率	0.02
GPS点图更新频率	3

OK Cancel

频率设置通常使用默认即可，但是为了防止用户数据量过大引起程序崩溃，可以通过减小频率避免崩溃。每一项频率设置是有最大值和最小值，超过范围配置不成功。

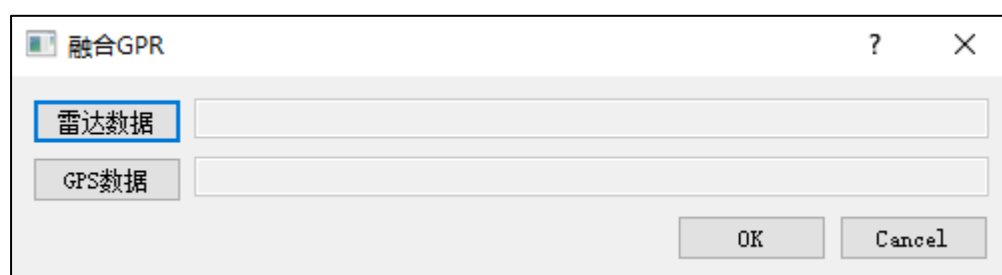
- **实时雷达接收频率**：在实时测量情况下的雷达接收频率
- **模拟雷达接收频率**：在使用本地雷达数据模拟实时测量情况下的读取数据的频率
- **灰度图更新间隔**：由于画灰度图相当耗时，更新方式为，每收到 N 条数据，更新最新的 N 条数据的灰度图。
- **计算特征频率**：特征计算的步骤相对比较多， 频率偏慢一些。
- **实时 GPS 接收频率**：实时测量情况下接收 GPS 信号的频率，这个设置即使再调小，收到一条 GPS 数据的频率还是和 GPS 本身的发送频率相关。
- **模拟 GPS 接收频率**：使用本地 GPS 数据模拟实时测量情况下的读取数据的频率，最好和模拟雷达接收频率保持一致
- **GPS 点图更新频率**：画点图相当耗时间，并且点越多所消耗时间越多，尽量往 3S 以上走。

模拟文件设置



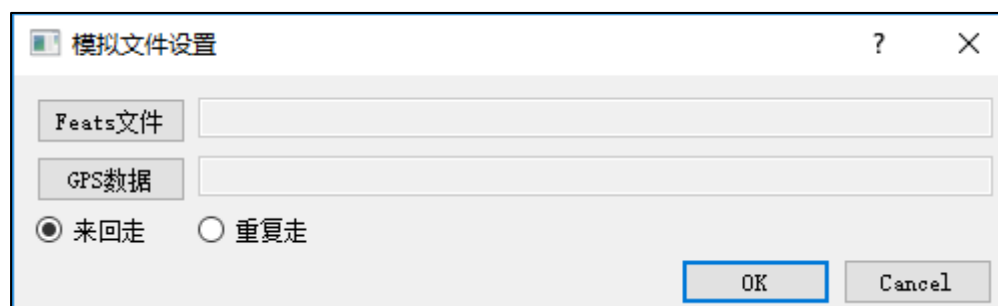
此功能是为了配置模拟情况下的数据路径，配置好以后，如果用户选择使用模拟数据，程序则会从此路径中获得对应的文件。

融合 GPR



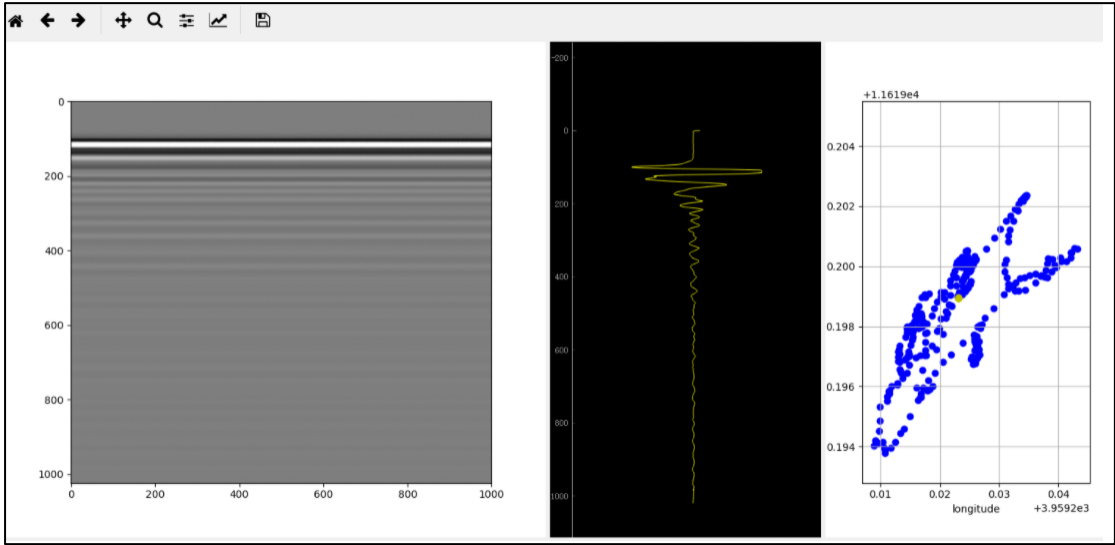
由于在结束采集后，如果数据量过大再融合为 GPR 有可能会造成程序长时间无响应的状态，所以现在改为数据是以 pickle 的形式保存，如果用户需要 GPR 格式可以使用这个小工具获得所需要的 GPR 文件并保存在设定的文件夹中。

载入 Prior 数据



由于考虑到雷达状态可能不稳定，电量低时可能会出现不可预知的错误，这里加了一个功能，允许用户可以载入 prior 过程后保存下来的数据，并直接开始 unregistered 过程的测量和计算。

画图板



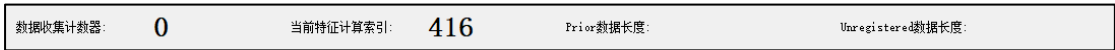
画图板位于软件界面的中部，用于实时显示雷达和 GPS 的信息。

左边为**灰度图**：显示最新部分的雷达数据

中间为**波形图**：显示最新收到的雷达波形

右边为 **GPS 图**：显示最新 GPS 坐标信息

状态栏



状态栏显示了数据收集和计算的情况。

- 数据收集计数器：显示当前收集到的雷达回波数量
- 当前特征计算索引：显示计算下一个所需要的雷达回波数量

- Prior 数据长度：在完成 prior 过程的测量后，这里会显示此过程收集到的总回波数量
- Unregistered 数据长度：在完成 unregistered 过程后，这里会显示此过程收集到的回波总数

如果用户使用测距轮模式，还会显示 prior 过程移动的距离，以及 unregistered 过程的移动距离。

数据收集计数器	0	当前特征计算索引	416	Prior数据长度	0	Unregistered数据长度	0	prior移动距离	0.0m	Unreg移动距离	
---------	---	----------	-----	-----------	---	------------------	---	-----------	------	-----------	--

软件使用

软件提供 4 种采集模式：

1. 使用模拟 GPS 数据， 模拟雷达数据， 当数据读取完后自动停止：

☐ 使用GPS

☒ 模拟雷达数据

2. 使用模拟 GPS 数据， 真实雷达数据， 当用户按了停止或者模式数据读取完了以后自动停止：

☐ 使用GPS

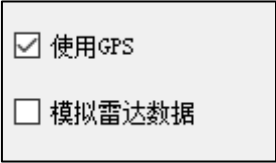
☐ 模拟雷达数据


3. 使用真实 GPS 数据， 模拟雷达数据， 当用户按了停止或者模式数据读取完了以后自动停止：

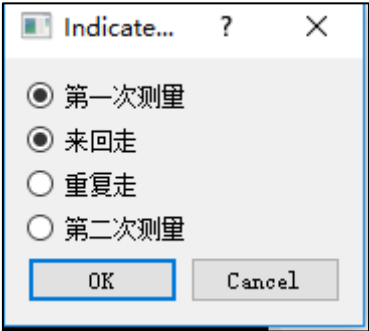
☒ 使用GPS

☒ 模拟雷达数据

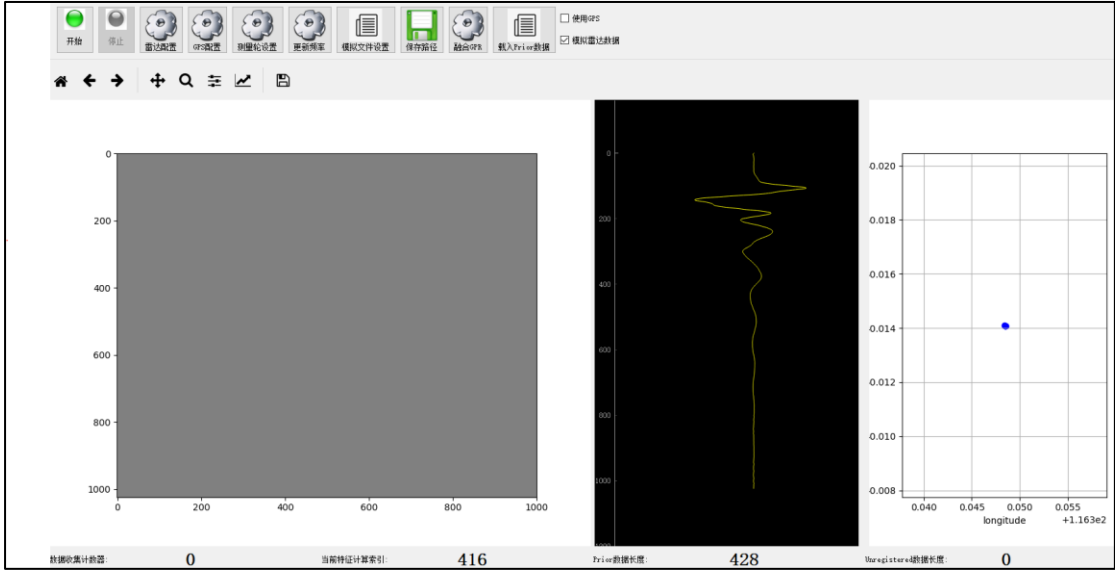
4. 使用真实 GPS 数据， 真实雷达数据， 当用户按下停止或者出现不可预知的错误自动停止采集：



在用户选定了一种模式并通过连接检测后， 点击开始按钮 ， 会出现一个询问框如下图：



在用户选择对应的测量策略之后软件开始收集数据并计算， 可从画图板中观察到数据变化。



如果用户收集的雷达回波数小于 416 则不会保存任何数据。

在完成两次完整的测量和计算后，软件会显示三张数据对比图，效果如下图：

