
1、实验名称及目的

目标跟踪算法实验：通过平台接口进行对图像的获取，然后通过目标跟踪算法控制无人机对目标物体的跟踪。

2、实验原理

首先进行 Config.json 文件进行相机传感器的配置，配置参数如下解释：

“SeqID”代表第几个传感器。此处表示第 1 个传感器（免费版只支持 2 个图）。

“TypeID”代表传感器类型 ID，1:RGB 图（免费版只支持 RGB 图），2:深度图，3:灰度图。“TargetCopter”传感器装载的目标飞机的 ID，可改变。

“TargetMountType”代表坐标类型，0：固定飞机上（相对几何中心），1：固定飞机上（相对底部中心），2：固定地面上（监控）也可变。

“DataWidth”为数据或图像宽度此处为 640，“DataHeight”为数据或图像高度此处为 480。“DataCheckFreq”检查数据更新频率此处为 30HZ。

“SendProtocol[8]”为传输方式与地址，SendProtocol[0]取值 0：共享内存（免费版只支持共享内存），1：UDP 直传 png 压缩，2：UDP 直传图片不压缩，3：UDP 直传 jpg 压缩；SendProtocol[1-4]：IP 地址；SendProtocol[5]端口号。

“CameraFOV”为相机视场角（仅限视觉类传感器），单位度也可改变。

“SensorPosXYZ[3]”为传感器安装位置，单位米也可改变。

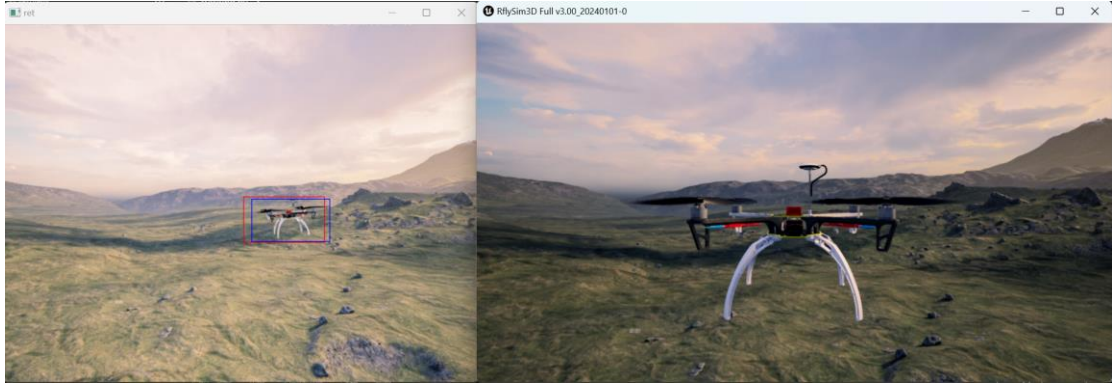
“SensorAngEular[3]”为传感器安装角度，单位度°也可改变。

深度相机输出的数据是以 uint16 存储和传输的，它的数据范围是 0~65535。默认情况下，一个单位表示 1mm（由 otherParams[2]控制），也就是说最大范围是 0 到 65.535 米。但是，数据范围并不代表相机的实际探测距离，还需要 otherParams[0]设置最小探测距离 otherParams[1]设置最大探测距离。otherParams[0]：深度相机的最小识别距离（单位米），如果深度距离小于本值，那么输出 NaN 对应 65535。otherParams[1]：深度相机的最大识别距离（单位米），如果深度距离大于本值，那么输出 NaN 对应 65535。OtherParams[2]：深度相机 uint16 输出值的刻度单位（单位米），默认情况下深度值以毫秒为单位，因此需要填 0.001。注，默认值填 0 的话，会被替换为 otherParams[2] = 0.001。实际深度值（单位米）= 深度图片值（uint16 范围）* otherParams[2]。

通过平台在 windows 下客户端进行图像数据的发送，然后在服务端进行图像数据的接收，然后对图像进行图像算法的运算处理以及目标跟踪算法运算，控制跟随无人机跟踪另一架无人机运动。

3、实验效果

会看到在飞机右侧又生成了一架飞机，两架飞机起飞后右侧飞机会飞行到前方并进行运动，同时生成一个窗口，显示目标飞机识别检测框与跟踪检测框重合效果。



4、文件目录

文件夹/文件名称	说明
Tracking.bat	启动仿真配置文件
Tracking.py	Python 实验脚本
Config.json	视觉传感器配置文件

5、运行环境

序号	软件要求	硬件要求	
		名称	数量(个)
1	Windows 10 及以上版本	笔记本/台式电脑 ^①	1
2	RflySim 平台免费版及以上		
3	Visual Studio Code		

①：推荐配置请见：<https://doc.rflysim.com/1.1InstallMethod.html>

②：须保证平台安装时的编译命令为：droneeye_zyfc-h7_default，固件版本为：1.12.1。其他配套飞控请见：<http://doc.rflysim.com/hardware.html>

6、实验步骤

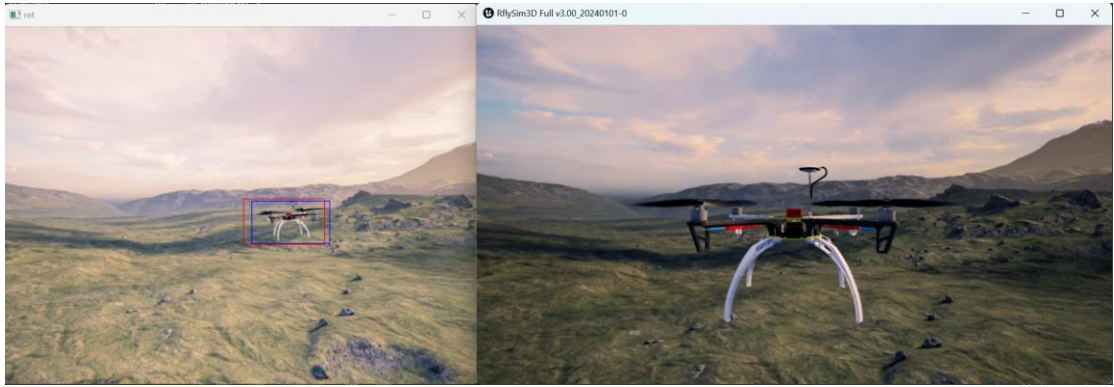
Step 1:

运行 PX4PSPRfySimAPIs\RflySimSDK 目录下的 ReLabPath.py 文件。

Step 3:

以管理员身份打开 Tracking.bat 脚本；

然后运行 Tracking.py，可以 1 看到在飞机右侧又生成了一架飞机，两架飞机起飞后右侧飞机会飞行到前方并进行运动，同时生成一个窗口，显示目标飞机识别检测框与跟踪检测框重合效果。



7、参考文献

[1]. 无

8、常见问题

Q1: 无

A1: 无