

NX PX4控制操作手册

作者 贾振&唐金亮

欢迎不断补充

日期：2021.6.3

(ego-planner 为例)

1. 下载

从 github 中下载一份最新的 ego_swarm_ws (或者向大鑫要下), 把飞机里面已有文件 run_in_exp.launch 和 advanced_param_exp.yaml 拷贝出来放到最新从 github 下在的对应目录里面, 后开始编译。

可能有遇到<quadrotor_msgs/Takeoffland.h>的报错情况, 有两种解决方式, 第一是文件里面出错, 这中情况是要用实验室飞机的文件夹 quadrotor_msgs, 把这个文件夹拷贝到 ego_swarm_ws/src/utils 中, 第二种情况是编译的时候多核起冲突。

输入 catkin_make -DCATKIN_WHITELIST_PACKAGES='quadrotor_msgs'
等编译完后在编译其他的文件夹, 此命令用途是专门编译等号后面这个包。

2. 相机调内参

在终端中输入 realsense_viewer (要先安装 realsense SDK) 在 More 菜单中点 Install recommended firmware, 等待时间安装, 在安装过程中看下串口是不是 3.2 的 (realsense 的数据量太大, 如果不是 3.2 串口, 传输速度不够), 装好后点开 stereo module 打开 infrared1 和 infrared2, 正常情况下应该是有 2 个黑白图和一个深度图在画面中。

打开一个终端, 在里面输入 roslaunch realsense2_camera rs_camera.launch, 开第二个终端在里面输入 rqt_image_view, 开第三个终端在里面输入 rostopic hz /camera/depth/image_rect_raw 确定是 15hz。用 rostopic echo /camera/depth/camera_info 查看参数把 fx 和 fy 上的参数填写到 ego_swarm_ws/src/realflight_modules/Vins.../config/Px4/camera_pinholes.yaml 里面 (其中第一第二个参数是 320 左右, 第三第四个参数是 240 左右)。

相机 0 和相机 1 里面都有 4 个参数，同样要改，文件在 `plan_manage/launch/run_in_exp` 中，填写格式参考 `max_yel`。

文件中参数含义：在 `advanced_param_exp.xml` 中 `grid_map/resolution` 表示地图分辨率，这个参数一般是飞机尺寸的一半，`grid_map/obstacles_inflation` 表示障碍物膨胀的数量。

把飞机放在离地面一定高度的位置，避免让相机能够直接拍到地面，在 `realsense` 应用中，把 `More` 菜单中的 `Tare Calibration` 的参数对话框中的数值换成飞机摄像头离墙面的距离（单位 `mm`），在放飞机的过程中，微调飞机的角度，让画面中的红白点尽量均匀分布，点 `calibration` 相机开始计算离墙面的距离，一般只有 10-20mm 的误差。（相机内部的固件，`realsense` 版本和 `ros` 版本要匹配上）。

3. 调试 VINS

在输入所有 `rosXXX` 的情况下都要另外开一个窗口输入 `roscore`，防止关一个 `rosXXX` 整个系统都崩坏。开 `vins_run.sh` 的时尽量不要开 `rviz` 这样网速压力比较大。

提示:`vins` 代码在 `ego_swarm_v2/src/realflight_modules/VINS-Fusion-gpu` 中。

把 `ego_swarm_V2` 拷贝到小电脑中，放在自己设定的文件夹下面（下面设定的文件夹名称为 `ABC`），编译好之后开始，进入 `ABC/ego_swarm_V2` 文件夹下之后打开三个终端，每个终端下都输入命令 `source devel/setup.bash`，在第一个终端里面输入 `ABC/ego_swarm_V2/src/planner/plan_manage/launch/roslaunch rviz.roslaunch` 起 `rviz`，在第二个终端里面输入 `cd src` 进入 `src` 文件夹下面，输入 `vins_run.sh` 查看能不能接上相机和 `IMU`。接上相机之后打开 `realsence` 可视化界面，查看串口是不是在 3.2 上，不是的话可能需要将 `TYPE-C` 接口换个方向插进去，小电脑的 `TYPE-C` 串口是分上下的，确定串口是 3.2 的之后，在第三个终端里面输入 `rostopic list` 查看有 `imu_propagate` 的那一个，然后输入 `rostopic hz /vins_estimator/imu_propagate`，要是通信线没连接好会反馈 `no new messages`，如果反馈不是 `no new message` 就是正常的，没连接上的话就重新插拔一下就行（一般是这样，或者是接触不良了）。

设置相机外参 `RT` 矩阵：在 `ABC/ego_swarm_V2/src/realflight_modules/VINS-Fusion-gpu/config/px4/stereo_imu_config.yaml` 文件中设置，自定义 `output_path`: 输

出路径，确保这个路径是存在的（如果不存在输出文件就不会生成），然后在终端中起 `vins_run.sh`，手拿着飞机转一圈，后打开输出路径的文件夹，会有一个 `extrinsic_parameter.vsc` 文件，把数据复制到 `stereo_imu_config.yaml` 中的 `dody_T_cam0`和 `dody_T_cam1`的两个相机参数里面，里面参数的意思是：每一个摄像头有一个4*4的矩阵，左上角3*3的矩阵是旋转矩阵，里面的数据一般是靠近0或者1的值，第四列的前三行分别是摄像头相对飞控的 XYZ 方向的位移偏差，若偏差不正常按上面的方法再来一遍，直到偏差正常就行（XYZ 偏差可以用卷尺量）。

还有一种办法是让相机内部估计 RT 矩阵：修改 `stereo_imu_config.yaml` 里面的 `print_extrinsic` 的0换成1，在终端中就会打印 RT 矩阵，开 `rostopic echo /vins_estimation/odometry`(`roslaunch list`看下有没有类似的)，这是看 RT 的 position 的偏差，一般在0.5左右，把 `rostopic echo /vins_estimation/odometry` 里面的参数填到 `stereo_imu_config.yaml` 中。

如果对想提高 vins 精度可对其相机和 imu 的时间戳进行补偿 `stereo_imu_config.yaml` 中 `estimate_td: 1` 开启终端 vins 后会输出 dt 时间差,然后将其填写至 `stereo_imu_config.yaml` 中 `td:` 填写时间保存重启即可。

都设置完后,可将在 rviz 订阅 ins 的 path 话题，然后将飞机拿在手上在场地上走动，一般移动30m 有1m 的偏差是正常的。

如果 vins 会飘可以打开 `stereo_imu_config.yaml` 中 `show_track` 设置为1,从 RVIZ 中添加话题 `img_track` 可以看到双目的特征点,方便新手确认环境中的特征是否丰富。

相机 IMU 的标定结束，在终端中起 `vins_run.sh` 测试飞机的重复定位精度，可根据上述进行迭代至可接受误差。

4. 调试控制器

调控制器的时候要一直开着 `rostopic echo /vins-estimator/odometry` 然后在开 `run.sh` 调控制器。

建议开启 ROS 多机通讯以免 NX 会炸机

nx 电脑:

执行 `sudo gedit /etc/hosts`

在127.0.0.1 localhost

127.0.1.1 nv

下方添加自己的 IP 和用户名(自己的电脑建议固定 IP,改完记得重启)

*****分割线

自己电脑:

执行 `sudo gedit /etc/hosts`

在127.0.0.1 localhost

下方注释自己的127.0.0.1 用户名

添加自己的 ip 好人用户名 和 NX 的 ip 和用户名

例如:

192.168.100.167 jz

192.168.100.114 nv

打开自己的 bash

加入(nx 为远程机用户名)

`export ROS_MASTER_URI=http://nx:11311`

`export ROS_HOSTNAME=jz`

多机配置参考 <https://zhuanlan.zhihu.com/p/101331694>

与控制器相应的文件在PX4Ctrl文件夹中,参数设定都在 `ctrl_param_fpv.yaml` 文件中,(下面坑比较多,可能不是顺序来调整)。在 `ABC/ego_swarm_V2` 中运行 `run.sh` 可能有 `run.id` 的报错,出现这个问题就调整 `run.sh` 中每个 `sleep` 后面的值,一般设定为第一个设定是1,第二个设定是9,后面的设定成5,后面当 `ros` 通讯出现问题的时候也有可能是这个的原因,设定 `sleep` 是调整每个节点起的速度,确保每个节点有充足的时间起。调试遥控器 `channel5` 和 `channel7`,运行 `run.sh` 后,用遥控器控制电机旋转,后拨 `channel7` 电机立即停转, `channel7` 是急停开关。控制飞机起飞,拨 `channel5`,在 `run.sh` 那个终端中显示是 `MANUAL_CRTL->AUTO-HOVER` (绿色) 表示有效果,在把 `channel5` 拨回去应该显示 `AUTO-HOVER->MANUAL_CRTL` (黄色) 表示正常, `channel` 是悬停开关,到这里遥控器控制测试完成,下一步就是调参数。

5. 调参数

参数在 `ctrl_param_fpv.yaml` 中

- 5.1. 把 `print_value` 换成 `True` 表示油门量在终端中打印。
- 5.2. `hover_percentage`:是根据飞机重量来定, Q250机架一般是0.25附近的值。
- 5.3. KP0-KP2和 KV0-KV2是上层 `poistion` 和 `PID` 参数的值需要自己调整下。
- 5.4. 在起了 `run.sh` 后手控飞机, 在 `channel1-4`中反馈若是与操作相反, 在 `rc_reverse` 中的4个值, `roll` 对应的是 `y`, `pitch` 对应的是 `x`, `yaw` 对因的是 `yaw`, `throttle` 对应的是 `z`。
- 5.5. `run_in_exp.launch` 是设置位置和速度的文件