**项目简介**

**四旋翼无人机室内自主飞行系统**

属于校级的科研项目，于2018年12月开始实施。该项目的最终目标是实现无人机在室外的自主航点飞行和室内环境下的避障飞行，包括无人机飞行模式的切换、加解锁、自主起飞、悬停、航点飞行、offboard模式飞行、障碍物的识别及避障策略，自主降落等动作。

*本人工作* ：

*1*、负责并搭建整个系统的框架。 市场上主流的飞行控制器分为PIXHAWK和APM两种，其中PIXHAWK以stm32为控制核心，APM以ardiuno单片机为控制核心，对比之下，PIXHAWK处理速度更快，提供的接口更加丰富，扩展性更强，同时二者分别对应PX4和APM两种开源软件体系，PX4以模块化编程，代码结构清晰，而APM以支持的对象编程，针对多旋翼、固定翼、无人车、无人船都有所涉及，软件结构混乱且复杂，研究使用难度较大，故选用PIXHAWK和PX4作为本系统的软硬件核心之一。主处理器32 bit STM32F427，32 bit STM32F103，它的特点就是安全稳定。所以就算主处理器死机了，还有一个协处理器来保障安全。

*2*、本系统中有图像处理部分和激光雷达数据处理部分，将数据处理后与飞控信息融合。图像处理需要占用cpu资源，激光slam定位、建图、导航同样需要占用大量cpu资源，PIXHAWK在运行PX4之后，剩余资源完全无法满足图像和建图的需要，也无法兼顾飞控的实时性，会造成很大的安全隐患。故引入板载计算机，来完成图像处理和slam定位等工作。对比市面常见的几款嵌入式计算机，英伟达旗下的TX2、TX1、nano，价格都比较昂贵且重量较大，在一千到三千左右的价位。在项目初期，树莓派的处理能力能够满足前期的需要，且接口支持丰富，将树莓派与pixhawk通过串口进行连接。接下来拟更换机载计算机，进行室内飞行工作。

*3* 、在确定好系统处理器及所需要的传感器之后，需对四旋翼的机架、电机、电调和桨叶尺寸进行筛选，进行筛选的条件主要有三个：1保证各个器件的重量和小于飞机所能提供最大升力；2保证四旋翼具有15-20分钟的飞行时间；3提供必要的冗余，用来系统未来存在的可能扩展。通过计算，整个系统的总质量在600g左右，故设计系统选用的机架轴距尺寸在450及以上，考虑到室内飞行的安全性，机架尺寸应越小越好，故选用轴距450的机架，四旋翼通常选用5200mA 4S电池，为续航时间及升力，选用1255桨叶及kv980电机。

*4*、实现室外有GPS条件下的航点避障飞行

在飞控体系架构中，GPS主要提供的是XY方向的坐标信息，融合飞控自身所带的加速度计、磁力计和陀螺仪，获得飞机的当前位置信息，与期望位置进行比较，进行PID运算，得到下一时刻的飞行位置。

其次，对飞控的飞行控制部分进行分析，飞控采用的控制方式是双闭环的PID控制方式，外环为位置环，由期望的位置信息和实际的位置信息得到期望的速度，内环为速度环，由期望的速度和实际的速度得到加速度，转换为四旋翼四个电机的转速，控制飞机的姿态变化，最终得到变换成为飞机的位置变化。

所以，关键就是避障算法利用激光雷达数据处理得到一个期望的位置信息，进入到飞控的位置控制部分进行运算即可，也就是说，我们需要在树莓派上通过计算得到一个期望的位置信息，传输到飞控中。考虑到激光雷达信息，图像信息与飞控信息的融合和系统未来的可扩展性，我们这里引入机器人操作系统ROS。

**Ros特点**

*Ros* 是一个开源工具,是为了兼容不同机器人平台软件开发所设计，并且得到广泛使用的机器人系统的软件架构。*Ros*基本思想是无需改动就能够在不同是机器人上复用代码。

*Ros* 基于一个集中式拓扑的图结构。节点是任意进程，从传感器读取数据、控制执行器，或者运行用于在环境中自主映射或导航的高级复杂机器人导航算法或视觉算法。

*Ros*的运行可以看做是有多个耦合度较低的进程组成。每个进程称为一个节点，所有的节点可以运行在一个处理器上，也可以分布式的运行在多个处理器上。实际使用时，这种松耦合结构的设计，可以让开发者根据机器人所需功能灵活添加各个功能模块，并且能分散处理器的计算压力。

*Ros*中提供*gazebo*仿真软件平台，可以直接将代码在软件中仿真。

首先需要完成局域网络的配置，板载计算机和个人pc，飞控工作在同一个局域网内。

ROS操作系统将激光雷达，飞控，避障算法处理看做是三个独立的节点，他们之间的数据通信是通过（ROS下的协议包）MAVROS相连接。激光雷达节点主要是获取无人机周围环境场的信息，障碍物的分布情况；飞控则是获取一些自身所带的传感器数据信息，解算出无人机当前的姿态以及位置信息。我们使用地面站上规划航点，由数传通过MAVROS上传到PIXHAWK中，再通过MAVROS传递到ROS操作系统中。通过远程连接方式，进入板载计算机内的ROS操作系统，启动避障节点。在避障算法处理里，通过订阅发发布的方式，输入为无人机的状态信息，目标航点信息，以及激光雷达跟障碍物的距离和角度信息。

接下来就是避障算法的处理过程：VFH避障算法

VFH避障算法是是一种由人工势场法改进而来的机器人导航算法，是属于局部路径规划范畴之中。主要分为两个阶段，第一个阶段是扫描机器人障碍物距离信息，根据障碍物的分布情况，构建直方图。第二个阶段结合无人机的下一个目标点，以及直方图的分布，寻找合适的机器人驱动方向。

vfh避障算法最终解算出的每个时刻的期望位置，同时对速度信息进行了限制，通过MAVROS发布到飞控，作为期望位置发送至PX4中用于位置控制，无人机在飞抵目标点的过程中实现无人机的避障飞行。

5、实现室内激光雷达避障飞行

在GPS信息缺失之后，单一的依靠飞控自身所带的传感器提供的数据，误差累计会越来越大，最终会造成炸机的危险，需要借助其他传感器（摄像头，激光雷达等）进行信息的弥补工作。首先确定定位方式。

1. 在最新版飞控中使用ekf2模块（位置和姿态估计），将外部定位方式启用。针对室内缺乏gps信号的情况，通过外部传感器获得无人机位置信息。
2. 定位方式按照是否机载分为两类。外部和机载定位。外部定位mocap和UWB，使用多个摄像头记录飞机的姿态，刷新率很高，精度达到毫米级别，价格几万到几百万不等。机载定位，采用视觉slam或者激光slam。项目中使用的是intel的T265摄像头，集成视觉slam算法，直接输出本地位置信息。
3. 将外部定位信息通过mavros发送到飞控中。传感器T265发布的位置信息和高度信息。通过MAVROS发给飞控。飞控就获得了基于无人机机体坐标系的本地位置信息。定位完成。其次是避障处理：
4. 使用MATLAB 中的提供的ros toolbox工具箱，ROS 工具箱为 MATLAB和 Simulink 与机器人操作系统（ROS 和 ROS 2）相互连接提供了一个接口，以便能够创建一个 ROS 节点网络。通过设置主从通信方式，将工作在同一个ROS网络下的不同ros节点，分布式的处理在不同的处理器上，他们之间也是能够完成互相通信。

整个避障系统由四个关键节点组成，mavros，激光雷达节点，matlab避障节点。该系统采用分布式通信系统，mavros节点，激光雷达节点，运行在板载计算机上，matlab 避障节点运行在个人PC上。这种做法的主要好处是缓解板载计算机的处理压力。针对一些处理性能较差的板载计算机，或者在板载计算机要运行一些计算量较大的算法时，将处理器的压力转移到个人PC上。

避障算法的计算过程是在个人PC上完成，它的输入主要是无人机的本地坐标系的位置信息，以及激光雷达的局部环境信息，使用的避障算法也是VFH算法。

最终matlab发布的速度控制指令通过mavros发布到飞控控制无人机避障飞行

7、无人机室内定位与建图。

catrtographer 谷歌2018年开源的新算法，加入回环检测，减少了误差。对雷达频率要求不是很高，并且在很多扫地机器人上得到了应用。

将激光雷达和机载电脑通过usb口连接，在机载计算机上将雷达的节点运行起来，接着运行激光slam算法，雷达的数据经过slam算法运算后，解算出飞机的X、Y坐标信息和偏航角。通过mavros发布到飞控中。

由于我们采用的是分布式多机通信，PC和机载计算机处在同一个ros系统中，在PC端启动rviz图形窗口，可以实时观看到雷达的建图与定位情况。