

---

# “智在飞行” 无人机线下赛汇报

---

November 7, 2021

Chong Gao  
Nanchang University  
Information Engineering School  
Generic Operational and Optimal Data Group

# Contents

1	完成情况 . . . . .	2
2	总体方案 . . . . .	2
3	比赛思考 . . . . .	4

# 1 完成情况

无人机线下赛比赛场地和任务说明如下：

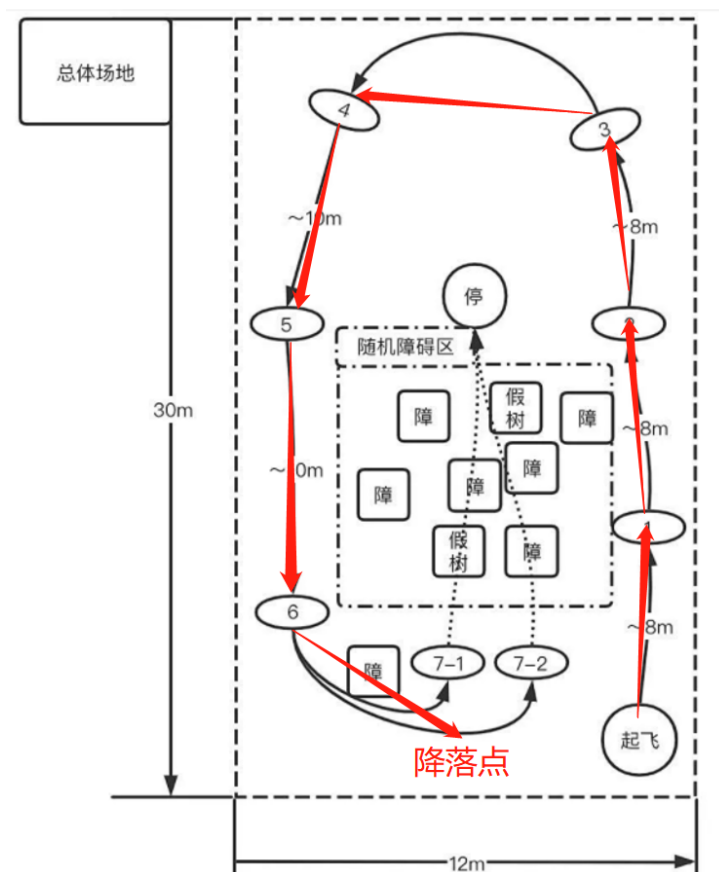


Figure 1: 比赛场地

总共分为 11 个任务，其中钻过 7 个圈记为 7 个任务，障碍物按照水平前进方向分为 3 个任务，停到停机坪视为 1 个任务，每完成一个任务 +10 分，发生明显碰撞-5 分，每个任务最多-5 分；选择 7-2 圆圈并钻过 +10 分（即无碰撞钻过 7-1 给 10 分，无碰撞钻过 7-2 给 20 分），最后按照分数排序；分数相同按照有效分数的最后记分时间排序。

我们最终的结果为：通过前 6 个障碍物圈，在第 5 个障碍物发生一次碰撞，用时时间为 2 分钟，总计分为 55 分。

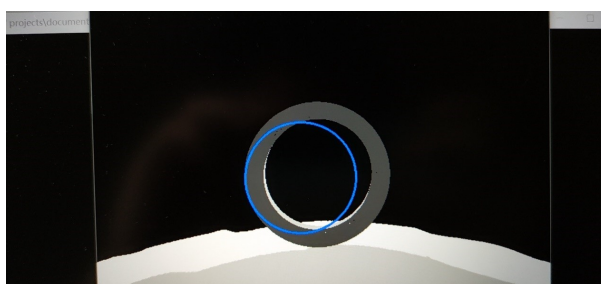
## 2 总体方案

同线上仿真相比，线下比赛存在无法获取完美的深度图，定位存在误差等问题，导致我们无法继续使用仿真时的方案。对此我们的解决方法是：

- 相较于线上获取的深度图（图 b），线下的深度图太过模糊（图 a），无法用传统霍夫圆检测的方式得到圆心点的世界坐标，因为在现场的时间比较紧张，我们没有找到合适的解决方案。因此我们采用固定航迹的方式，通过人为采集合理的航点世界坐标作为无人机飞行航迹。但是这种方式存在很多问题，比如十分依靠定位的精确度，然而在线下环境中，定位错误是十分常见的，所以这种方案的准确度比较低，具体的优化方式将在下面给出。



(a) 线下赛采集的深度图



(b) 线上赛采集的深度图

- t265 相机定位过程中，如果机身的加速度过大，比如急加速和急减速的时候，会出现特征点跟踪丢失，从而导致视觉漂移，定位出错的问题，而一旦定位错误，则定点飞行的方式就会失效。对此我们采用更加严格的飞行速度控制，来使无人机的飞行尽可能平稳，以减少因加速度过大导致的定位误差，定位准确之后，我们实现基本可以稳定地通过 5 到 6 个圈。这种方法的缺点是飞行过程非常缓慢，由于时间问题，我们没有调试满足定位准确时的最大速度来所短飞行时间。以下有部分飞控代码：

```

1
2 if (abs(local_pos.pose.position.x-pose.pose.position.x)<=0.15 && abs(
   local_pos.pose.position.y-pose.pose.position.y)<=0.10 && abs(
   local_pos.pose.position.z-pose.pose.position.z)<=0.15){
3
4     if(count<=2 && count > 1){//fly to circle1
5         pose=posearray[0]; //飞到预定圆圈前的航点坐标
6     } //一个count耗时0.1s, count用于等待无人机稳定飞行至目标点
7     else if(count<=3 && count>2){//circle1
8         pose=circleposearray[0]; //飞到通过圆圈的航点坐标
9     }
10 }

```

```

11 //通过位置误差发布速度
12 geometry_msgs::TwistStamped vel;
13 vel.twist.linear.x=pose.pose.position.x-local_pos.pose.
position.x;
14 vel.twist.linear.y=pose.pose.position.y-local_pos.pose.
position.y;
15 vel.twist.linear.z=pose.pose.position.z-local_pos.pose.
position.z;
16 vel.twist.linear.x=vel.twist.linear.x>=0.5? 0.5:vel.twist.
linear.x;
17 vel.twist.linear.y=vel.twist.linear.y>=0.5? 0.5:vel.twist.
linear.y;
18 vel.twist.linear.z=vel.twist.linear.z>=0.5? 0.5:vel.twist.
linear.z;
19 vel.twist.linear.x=vel.twist.linear.x<=-0.5? -0.5:vel.twist.
linear.x;
20 vel.twist.linear.y=vel.twist.linear.y<=-0.5? -0.5:vel.twist.
linear.y;
21 vel.twist.linear.z=vel.twist.linear.z<=-0.5? -0.5:vel.twist.
linear.z;
22 vel.twist.angular.z=0;
23 }

```

部分飞控代码

整体的思路如下：我们采用 px4 的飞控固件，用 t265 的深度追踪相机进行定位导航，通过 ROS 和 mavros 编程来对 px4 进行控制。我们通过现场测试来得到一系列目标航点的世界坐标，通过无人机当前的位置和目标航点位置的误差来控制无人机飞向目标点。

### 3 比赛思考

这次比赛中我们使用的方法较为保守，但可作为完成飞行任务的基础方案，下一次参加时可基于此方案进行优化。优化方向主要为用视觉获取目标点的世界坐标，px4 的实时调整飞行姿态。