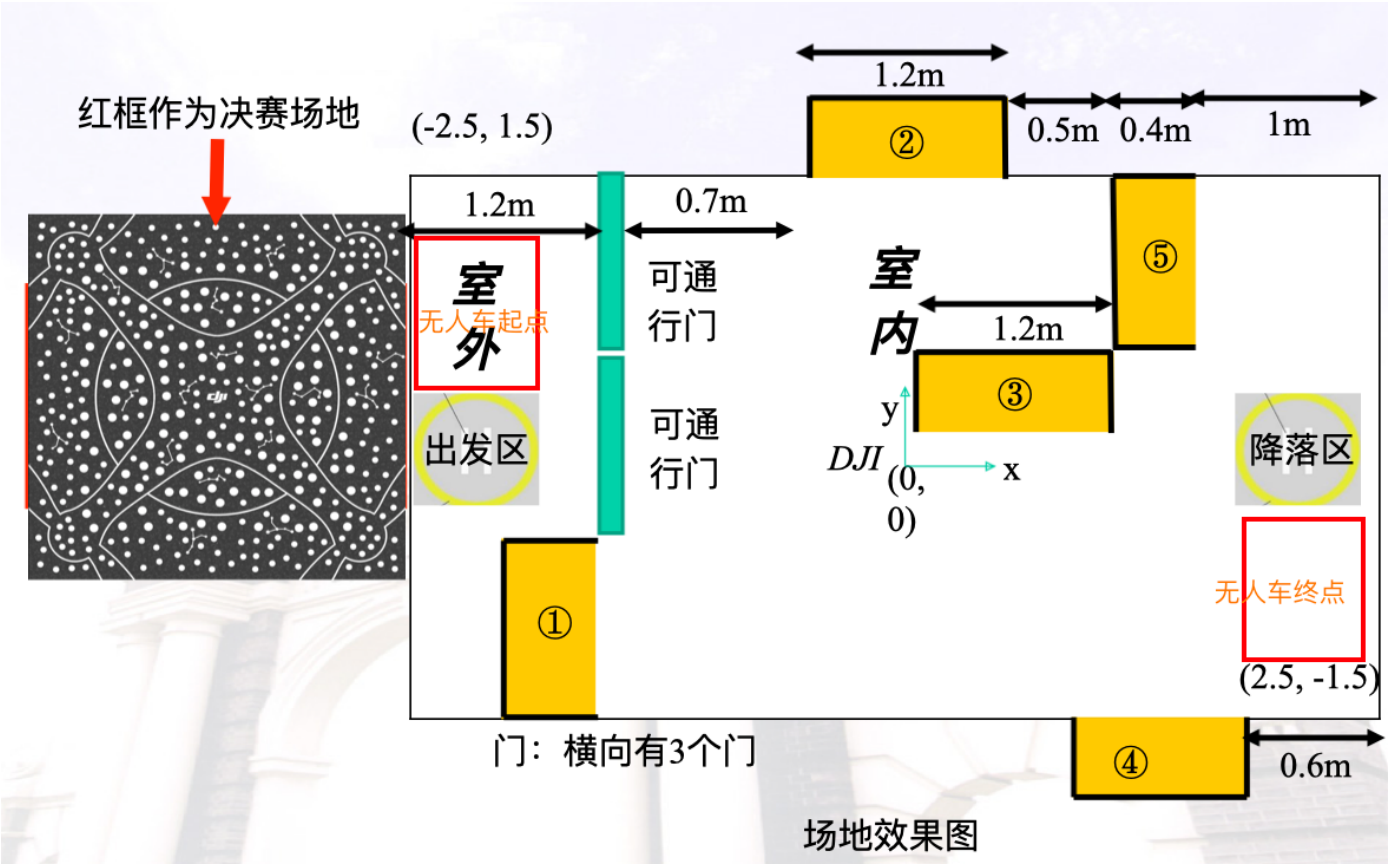


无人机课赛规则初版

比赛流程

- 1 无人机和无人车启动
- 2 无人机和无人车通过着火点
- 3 无人机和无人车于场内巡视，完成篮球、排球、足球的识别
- 4 无人机和无人车达到终点，向裁判机发送结束指令

比赛地图



红框为无人车起点/终点，灰框为无人机起点/终点

比赛细则

- 1 无人机和无人车启动受裁判机发送的“/cmd_start”话题（话题名为“/cmd_xtart”，类型为Bool)控制，裁判机发布“/cmd_start”值为“1”时比赛开始，无人机和无人车开始运行。若未于无人机和无人车中进行设置，将先开始计时再运行相关节点。
- 2 无人机和无人车在完成识别后，向裁判机发送名为“/target_result”（话题名为“/target_result”，类型为String)的话题来表达自己的识别结果，格式为长为5个字符串的String消息类型，其中‘b’‘f’‘v’分别表示篮球、足球、排球，‘e’表示无球,第n个字符表示第n号柜子中摆放什么球，示例：“befev”表示第1个柜子中是篮球，第3个柜子中是足球，第5个柜子中是排球，2、4号柜子中没有球。

3 三种球的摆放规则如下：在某号柜子处，既可以在柜子内摆放球，又可以在柜子下摆放球，一个柜子处最多摆放一个球，放置在**柜子下**的球最多为一个。

4 无人机和无人车到达终点后，分别向裁判机发送名为“finish_car”和“finish_drone”(话题类型为Bool)，值分别为“1”的话题。裁判机在接收到两个话题之后结束计时。

示例如下：

```
from std_msgs.msg import String, Bool
....

self.cmdfinishPub_ = rospy.Publisher('/finish_car', Bool, queue_size=100)
....
#小车完成比赛后
finish_car = Bool()
finish_car.data = 1
self.cmdfinishPub_.publish(finish_car)
```