

# 1. 实验名称及目的

## 1.1 实验名称

无人机集群跟随圆形案板移动实验

## 1.2 实验目的

本实验旨在通过控制多架无人机协同跟踪一个或多个移动的红色圆形目标，深入理解和实践基于计算机视觉的无人机集群自主跟随技术。通过该实验，学习者可以掌握以下核心技能：

- 理解视觉伺服控制原理：**掌握从图像信息（目标位置、大小）到无人机运动指令的闭环控制流程；
- 掌握HSV色彩空间分割：**学习如何在不同光照条件下稳定地识别特定颜色（红色）目标；
- 实现编队飞行控制：**理解并实践无人机在集群中的相对位置保持与协同运动策略；
- 设计与调试PID控制器：**通过调整比例系数（ $k_x, k_y, k_z$ ），优化无人机的响应速度与稳定性；
- 集成多模块系统：**熟悉RflySim平台中视觉采集、图像处理、运动规划与飞控执行等模块的协同工作方式；
- 分析与解决工程问题：**通过常见问题排查（如目标丢失、控制震荡），提升系统性思维与故障诊断能力。

此外，通过对比双机与四机跟随模式，学习者能够直观感受到系统扩展性、多目标管理以及复杂轨迹适应能力的重要性，为后续研究更高级的集群智能算法奠定坚实基础。

## 1.3 关键知识点

本实验基于计算机视觉的无人机编队跟随技术，主要涉及以下关键技术点：

### 整体思路和框架

实验采用双目标、四无人机编队跟随的架构，其中两个红色圆形目标分别被两组无人机跟随。整个系统分为以下几个核心模块：

1. **视觉采集模块**：通过RflySim的虚拟相机获取无人机第一视角图像数据。
2. **图像处理模块**：将RGB图像转换至HSV色彩空间，利用 `cv2.inRange()` 函数提取红色区域，生成二值掩膜（mask）。
3. **目标检测模块**：通过对掩膜进行非零像素分析，计算出目标的边界框（Bounding Box）及其几何中心点。
4. **控制算法模块**：基于目标中心点相对于期望位置的偏差，使用比例（P）控制器计算无人机的速度指令（ $v_x, v_y, v_z$ ）。
5. **运动控制模块**：通过PX4MavCtrl接口向无人机发送速度控制指令，实现精确的位置跟踪。
6. **目标运动模块**：利用UE4CtrlAPI控制圆形案板按照预设轨迹（正弦波/余弦波）运动，模拟动态目标。

## 核心程序解析

### VisFollowingUAVSwarm2.py 程序分析

该程序实现了双无人机对单一移动目标的协同跟随。其核心逻辑如下：

- **初始化**：创建两个无人机控制实例（`mav1`，`mav2`）和一个视觉采集实例（`vis`），并设置各自的期望成像位置（`target_x_1`，`target_x_2`），形成左右编队。
- **`detect_circle()` 函数**：是图像处理的核心。它接收原始图像，通过HSV阈值分割提取红色区域，并返回目标的水平、垂直范围及中心坐标。此函数还绘制十字线用于可视化调试。
- **`calculate_velocity()` 函数**：是控制算法的核心。它根据当前检测到的中心点与期望位置的偏差，结合预设的PID参数（`k_x`，`k_y`，`k_z`），线性计算出前后、左右、上下的速度指令，实现简单的视觉伺服。

### VisFollowingUAVSwarm4.py 程序分析

此程序是双机版本的扩展，支持同时控制四个无人机跟随两个独立目标。

- **多目标支持**：引入了第二套目标位姿变量 `circle_pose2`，并通过 `auto_move_target2()` 函数独立控制第二个案板沿不同的余弦波轨迹运动。
- **扩展性体现**：代码结构清晰，易于扩展至更多无人机和目标，体现了模块化编程的优势。

**注**：本实验的前序实验为：[[安装目录]\RflySimAPIs\8.RflySimVision\1.BasicExps\1-VisionCtrlDemos\6\_Circle-follow\Readmemd\Readme.md]。

## 2. 实验效果

本实验成功实现了基于视觉的无人机集群稳定跟随，具体效果如下：

- 高鲁棒性目标识别：**系统能有效抵抗环境光干扰，准确分割并定位红色圆形目标，即使在部分遮挡情况下也能维持跟踪。
  - 平滑稳定的跟随性能：**通过合理整定PID参数，无人机能够平稳地跟随目标移动，无明显超调或振荡，响应及时。
  - 灵活的编队能力：**双机模式下，两架无人机能保持预设的横向间距，形成稳定编队；四机模式下，可实现两组独立编队的协同作业。
  - 多轨迹适应能力：**无论是直线后退还是复杂的正弦/余弦波运动，无人机均能有效跟踪，展示了系统的泛化能力。
  - 实时可视化监控：**程序运行时弹出的OpenCV窗口实时显示图像处理过程，便于观察算法效果和快速调试。
- 这些效果充分验证了所采用的视觉伺服方案的有效性和实用性。

## 3. 文件目录

例程目录：

[安装目录]\RflySimAPIs\10.RflySimSwarm\1.BasicExps\2.PyCtrlUAVsSim\1.VisFollowingUAVSwarm

## 4. 运行环境

### 4.1 软件要求

Windows 10及以上版本；RflySim工具链。

①：推荐配置请见：<https://rflysim.com/>

### 4.2 硬件要求

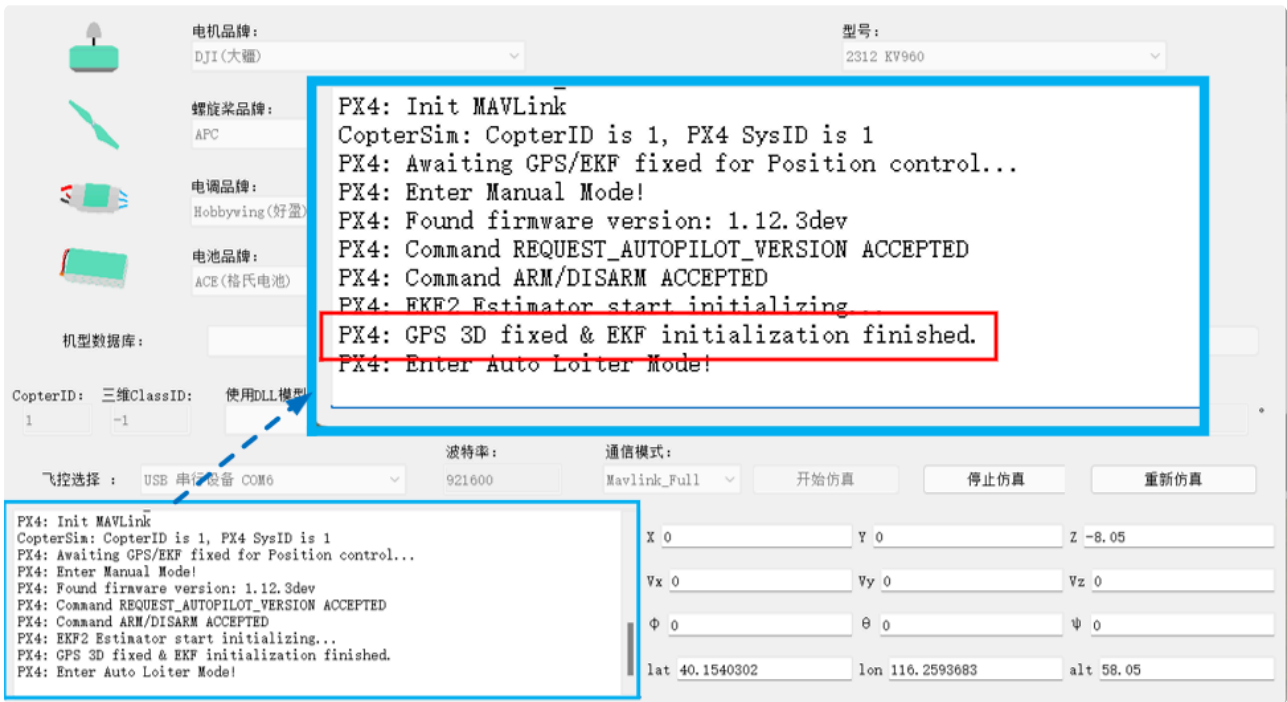
笔记本/台式电脑1台

①：推荐配置请见：<https://rflysim.com/doc/zh/HowToInstall.pdf>

## 5. 实验步骤

### 5.1 启动仿真

双击运行[VisFollowingUAVSwarm2.bat]文件，等待仿真环境初始化完成。脚本将会启动 1 个 QGC 地面站，2 个 CopterSim、1 个 RflySim3D 软件，等待2个CopterSim软件下侧日志栏必须打印出 `GPS 3D fixed & EKF initialization finished` 字样代表初始化完成。如下图所示：



### 5.2 运行2架次无人机跟踪圆形案板程序

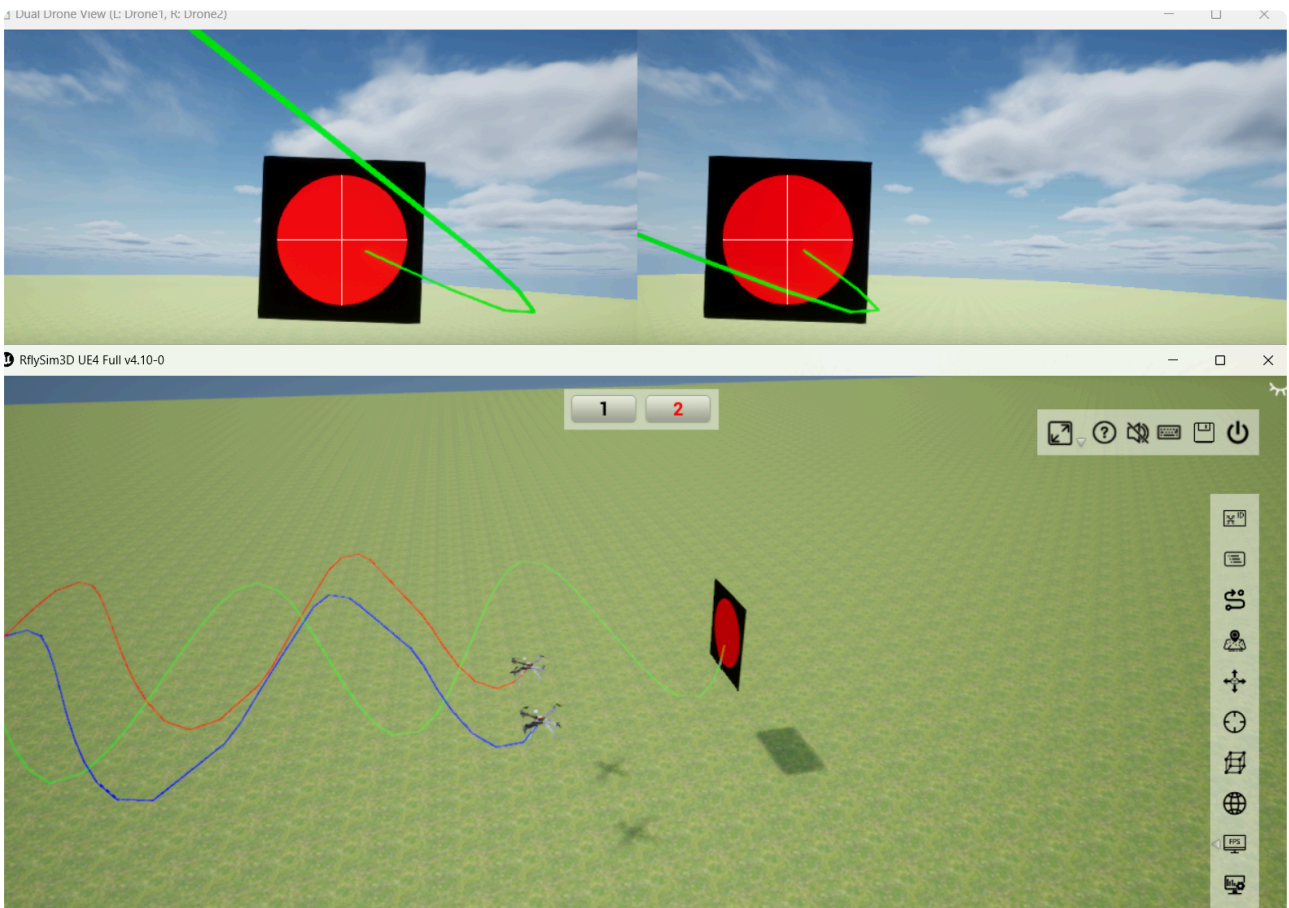
双击运行[Python38Run.bat]脚本，在弹出的窗口中输入：

```
1 | python VisFollowingUAVSwarm2.py
```

```
C:\WINDOWS\system32\cmd. x + v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python

D:\RflySim\RflySimAPIs\10_RflySimSwarm\1.BasicExps\e2.PyCtrlUAVsSim\1.VisFollowingUAVSwarm>python VisFollowingUAVSwarm2.py
正在初始化环境...
Json use relative path mode
jsonPath= D:\RflySim\RflySimAPIs\10.RflySimSwarm\1.BasicExps\e2.PyCtrlUAVsSim\1.VisFollowingUAVSwarm\Config.json
Got 4 vision sensors from json
No CopterSim time, use image time to calculate.
视觉捕获已启动。
No CopterSim time, use image time to calculate.
No CopterSim time, use image time to calculate.
正在连接无人机...
切换到 Offboard 模式...
```

在RflySim3D软件中，按下“T”键即可打开载具运动轨迹显示，如下图所示，圆形案板将沿着X轴做正弦运动基于屏幕向后移动，2架无人机将自动跟随。



## 5.3 运行4架次无人机跟踪圆形案板程序（选做）

双击运行[VisFollowingUAVSwarm4.bat]文件，等待仿真环境初始化完成。脚本将会启动 1 个 QGC 地面站，4 个 CopterSim、1 个 RflySim3D 软件，等待4个CopterSim软件下侧日志栏必须打印出 `GPS 3D fixed & EKF initialization finished` 字样代表初始化完成。如下图所示：



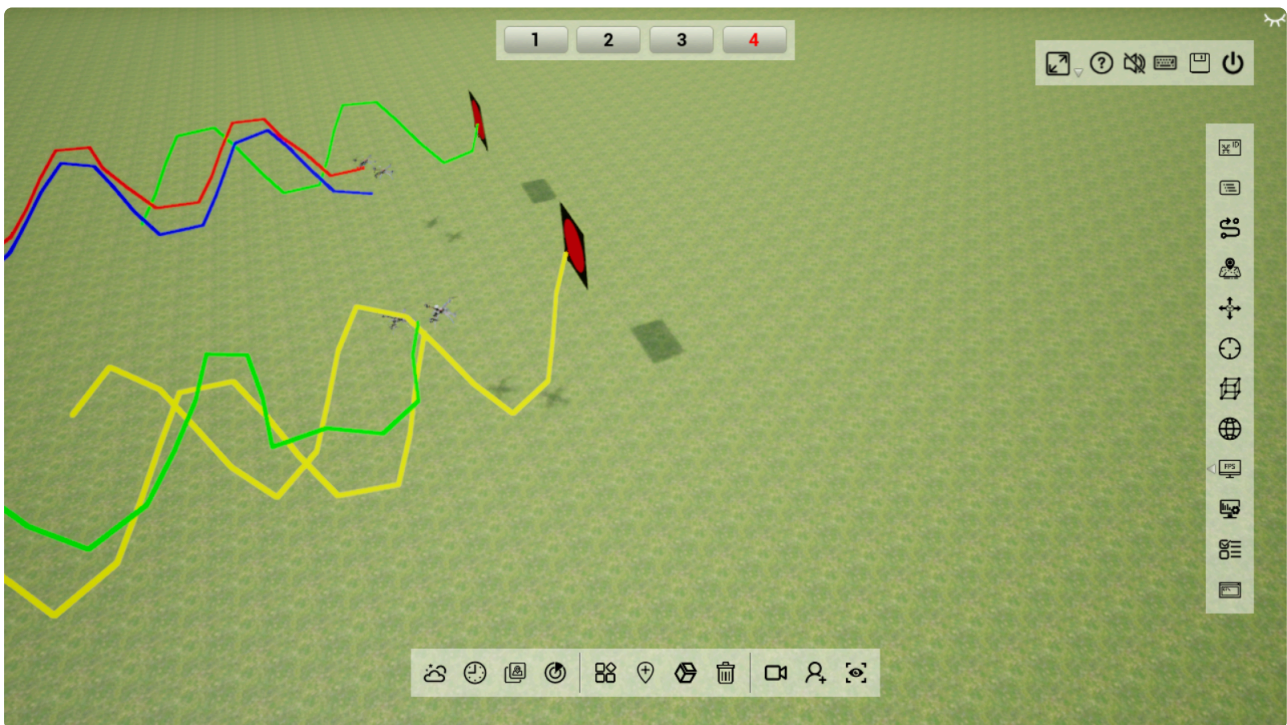
双击运行[Python38Run.bat]脚本，在弹出的窗口中输入：

```
1 | python VisFollowingUAVSwarm4.py
```

```
C:\WINDOWS\system32\cmd.exe
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python

D:\RflySim\RflySimAPIs\10.RflySimSwarm\1.BasicExps\e2.PyCtrlUAVsSim\1.VisFollowingUAVSwarm>python VisFollowingUAVSwarm4.py
正在初始化环境...
Json use relative path mode
jsonPath= D:\RflySim\RflySimAPIs\10.RflySimSwarm\1.BasicExps\e2.PyCtrlUAVsSim\1.VisFollowingUAVSwarm\Config.json
Got 4 vision sensors from json
No CopterSim time, use image time to calculate.
视觉捕获已启动。
No CopterSim time, use image time to calculate.
No CopterSim time, use image time to calculate.
No CopterSim time, use image time to calculate.
正在连接无人机...
```

在RflySim3D软件中，按下“T”键即可打开载具运动轨迹显示，如下图所示，圆形案板1将沿着X轴做正弦运动基于屏幕向后移动，无人机1、2将自动跟随。圆形案板2将沿着X轴做正弦运动基于屏幕向后移动，无人机3、4将自动跟随。



## 6. 参考资料

1. 《RflySim用户手册》
2. Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools.
3. Corke, P. (2017). Robotics, Vision and Control: Fundamental Algorithms in MATLAB. Springer.
4. Chen, C. H., & Wang, L. (2019). Visual servoing of quadrotor UAVs: A survey. Journal of Intelligent & Robotic Systems, 93(3-4), 507-527.
5. Chaumette, F., & Hutchinson, S. (2006). Visual servo control. IEEE Robotics & Automation Magazine, 13(4), 82-90.

## 7. 常见问题

### Q1: 仿真启动后无人机无法起飞或者飞行不稳定

A1: 检查是否正确完成了初始化过程，确保CopterSim软件下侧日志栏打印出 `GPS 3D fixed & EKF initialization finished` 字样后再进行后续操作。如果仍然存在问题，尝试重启仿真环境并检查网络连接。

### Q2: 无法检测到红色圆形目标或者检测不准确

A2: 这个问题通常由光照条件或颜色阈值设置引起。确保环境中红色目标足够明显且没有其他强红光干扰。可以适当调整HSV阈值参数：

```
1 | # 红色阈值 (HSV)
2 | self.low_hsv1 = np.array([0, 43, 46])
3 | self.high_hsv1 = np.array([10, 255, 255])
4 | self.low_hsv2 = np.array([156, 43, 46])
5 | self.high_hsv2 = np.array([180, 255, 255])
```

根据实际环境光照情况微调这些参数。

### Q3: 无人机跟随过程中出现震荡或者响应迟缓

A3: 这是控制参数不合适导致的问题。可以调整以下PID控制参数来优化跟随效果：

```
1 | # PID 控制参数 (比例系数)
2 | self.k_x = 0.015 # 前后速度系数 (基于目标大小变化)
3 | self.k_y = 0.009 # 左右速度系数 (基于水平位置偏差)
4 | self.k_z = 0.005 # 上下速度系数 (基于垂直位置偏差)
```

增大参数会使响应更快但可能导致震荡，减小参数会使响应更平滑但可能产生滞后。

## Q4: 程序运行过程中突然停止或崩溃

A10: 可能是由于资源不足或内存泄漏导致。尝试关闭不必要的程序以释放系统资源。检查图像处理部分是否有内存泄漏问题，确保及时释放不需要的图像资源。同时检查是否有异常处理机制防止程序意外终止。