

# RoboMaster 冬令营算法技术报告

## 一、RoboVision 装甲板检测系统：

1. 功能需求：检测场景中地方机器人装甲板的位置，将旋转信息发送给云台，实现对目标的跟踪，起到辅助瞄准的作用

2. 其他要求：①在目标旋转或快速移动时保证检测的稳定性

②尽量减少系统延迟（通过优化算法）

3. 整体架构：

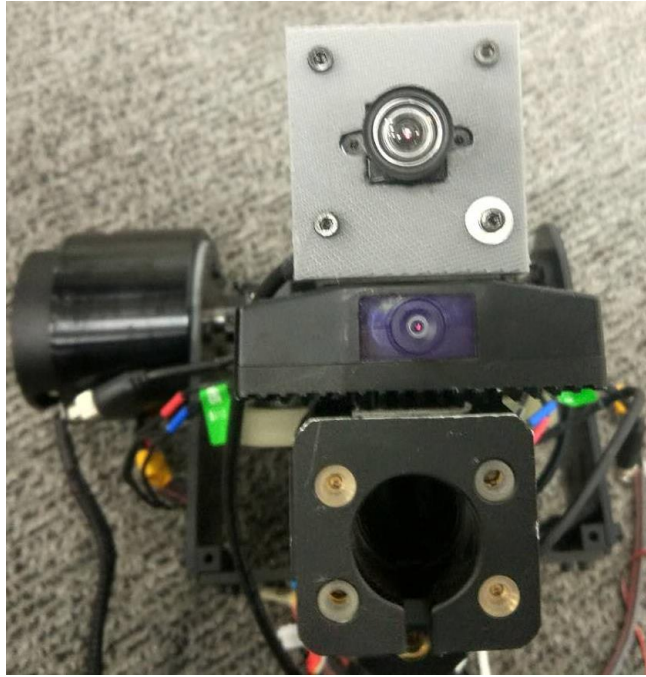
1) 文件描述：

文件名	用途
Camera.cpp	独立文件，用于标定摄像头，计算相机内参
uart.cpp	Linux 串口通信，用于在上下位机间发送数据
RoboVision.cpp	对装甲板进行检测，通过算法确定最终云台需要转动的角度，并通过串口通信发送给下位机
main.cpp	程序入口

2) 程序逻辑：系统启动后，摄像头开启，以 120 fps，一帧一帧对装甲板进行检测，通过算法确定最终云台需要转动的角度，并通过串口通信发送给下位机

4. 实现方案：

1) 摄像头：购买的摄像头模组位 K2SA17( 90 度无畸变 )，支持以 120fps MJPG 640\*480 摄像。用 3D 打印件对摄像头模组进行保护，并用觉将摄像头安装至图传上方，如下图。



- 2) 运算平台 :搭载了 Tegera TK1 的 DJI Manifold 妙算平台 ,运行 ubuntu 系统 ,  
固定在车体上 ,通过 4Pin 线与开发板连接( 需要将 4Pin 线的 Tx 和 Rx 对调 )



- 3) 第三方库支持 : OpenCV > 3.0
- 4) 运行效果 : 绿色方框位最终确定要击打的装甲板 ( 注 : 此车辆并非我组车辆 )



5. 检测部分具体算法：

- 1) 由 R 通道与 B 通道作差，以凸显蓝色或红色（蒸馏图像）
- 2) 对蒸馏后的图像进行二值化处理，得到场景中特定蓝色物体
- 3) 对原图像的灰度图进行二值化处理，得到场景中特定亮度的物体
- 4) 对两张二值化后的图像进行按位与操作，得到场景中蓝色与高亮的公共物体，  
即灯条蓝到发白的部分和蓝色光云部分的轮廓
- 5) 对找到轮廓进行矩形拟合，得到单条灯柱的部分
- 6) 运用约束条件排除形态差异过大的灯条以及显示血量的横着的灯条
- 7) 由灯条之间的关系判断两两灯条是否能够构成一块装甲板
- 8) 由灯条位置得到装甲板位置，并确定装甲板中心点位置
- 9) 对装甲板进行排序，选择较大的装甲板作为最终装甲板
- 10) PnP 解算，计算摄像头相对于目标位置的姿态（需要知道相机内参，提前对相机进行标定）
- 11) 由于云台只有 yaw 轴和 pitch 轴，故可以在一个平面上考虑 yaw 轴的旋转角

度，计算出旋转角度，通过串口通信发送给下位机

12) 由于每发子弹出膛速度不一致，需要操作手手动调节 pitch 轴

6. 调试与配置：算法中有众多参数需要调节，我一开始使用随便调节的方法，根据反馈情况调大调小，在意识到这种调试方法的低效后转变为对图像进行分析，精确到 mm 级别进行测量、观测与计算，得出参数的精确范围

7. 总结与展望：

1) 由于算法经过了多次优化，装甲检测操作耗时极短，同时精确度较高，但是当目标物体运动速度过快时将无法统计，预计是摄像头的问题，需要使用更高级的摄像头

2) 本次冬令营的任务是搭建一个步兵机器人，需要硬件、软件以及多个系统的配合，即使编写出了较好的算法，没有好的硬件，也是无用的，只有软硬件相配合，才能构建出更优秀的系统

3) 由于移动目标运动速度较快，而系统中有计算、通信、拨弹、出弹和飞弹的延迟，所以需要根据历史数据预测目标的下一位置，否则运动中的准确度将很低

8. 参考与感谢：特别感谢亚楠哥耐心地解答我遇到的问题

<https://github.com/RoboMaster/RoboRTS>

<http://bbs.robomaster.com/thread-3058-1-1.html>

二、TelloVision 无人机起飞校准系统：

1. 需求分析：在试验 Tello 飞行器的过程中，我们发现 Tello 在起飞时会有一定的误差，导致之后整体路径有了很大的偏差，为了消除起飞时的误差，需要在起飞后对 Tello 飞行器进行位置修正，而 API 中并没有开放图传数据，所以考虑外部架设摄像头，将摄像头检测的结果反馈给 Tello 飞行器

2. 实现方案：由于 Tello 飞行器特征并不明显，所以考虑在电池后部贴上一块蓝色贴纸，通过外部架设的摄像头对蓝色标记进行识别，以确定 Tello 飞行器的位置

1) 蓝色贴纸效果



2) 摄像头没有特定型号的要求，为了节省材料，使用上一阶段运用在步兵机器人上的摄像头，由于只是一个测试，用胶带临时固定



3) 红点为摄像头中心，需要调整到第一个环的中心，绿色方框为检测到的蓝色标记物轮廓，黄点为标记物中心



- 4) 误差减小的过程,当偏差值在一定范围内时,准许飞行器执行后续动作,如前进穿过第一个圆环等



### 3. 整体架构 :

- 1) 文件架构 : 由于工期较赶,完成该项目的时间为 2.10 晚上 23 : 00 到 2.11 早上 4 : 30 ,所以 udp、视觉识别部分和 Tello 操作部分写在同一文件内
- 2) 程序逻辑 :
  - ①操纵 Tello 飞行器进入控制模式并起飞
  - ②开启摄像头,检测蓝色标记物
  - ③减小标记物中心点与目标点之间的差距
  - ④先后在三个方向上进行微调
  - ⑤当差距在允许范围内时,退出姿态调整模式
  - ⑥操纵飞行器执行后续命令

### 4. 总结与期望 :

- 1) 在实际过程中，飞行器起飞时，yaw 轴会有不定偏转，即使将对齐中心点，也仍有可能偏离预定轨道，在与工程师交流过程中我们了解到，在 Tello 起飞至 30 公分前，下视将无法正常工作，故这个偏转将无法避免，可以考虑前进一定距离后分析轨迹，得出偏转角度，并修正这个角度，但由于单摄像头难以准确地三维重建，所以可能需要考虑双摄像头方案

三、总结：这次冬令营中，我挑战自我，实现了视觉识别，对自身编程能力和自主学习能力有很大的提升，同时对机器人有了较为系统的了解，也吸收了许多团队合作方面的经验，期待有机会的话，在夏令营再次挑战自我，做出优秀的项目