
一组哨兵机器人 技术文档

目录

一、	机械部分.....	1
1.	整体设计说明.....	1
1.1.	设计方案.....	1
1.2.	方案的优缺点.....	1
1.2.1.	优点.....	1
1.2.2.	缺点.....	2
2.	易损结构及零件说明.....	2
二、	嵌入式部分.....	3
1.	控制方案.....	3
1.1.	元件清单.....	3
1.2.	硬件、软件资源配置.....	3
1.2.1.	硬件电路.....	3
1.2.2.	软件资源分配.....	4
1.3.	控制电路布线结构，拓扑图.....	5
1.4.	软件架构、控制框图.....	5
2.	细节.....	6
2.1.	方案的优化过程.....	6
2.2.	算法详细介绍.....	7
三、	算法部分.....	7
1.	整体.....	7
1.1.	硬件环境.....	7
1.1.1	软件环境.....	8
1.2.	总体系统框图.....	8
1.3.	算法方案.....	8
1.4.	枪管偏移.....	11
1.5.	平滑滤波平滑输出数据.....	11
1.6.	多摄像头处理图像.....	12
1.7.	串口通信.....	12
1.8.	开机自启动配置.....	12
1.9.	视觉方面总结.....	12
1.9.1.	预期效果与最终结果.....	12

1.9.2.	方案的创新点	12
1.9.3.	存在的坑	13
1.9.4.	可以改进的地方	13
1.10.	元件清单	13
四、	总结	14
1.	预期效果与最终效果	14
2.	方案的创新点	14
3.	存在的坑	14
4.	可以改进的地方	15
	附录	16
1.	测试过程和数据	16
1.1.	测试一 哨兵方案设计测试（中间转轴加导轮）	16
1.2.	测试二 拨弹轮拨弹测试	16
1.3.	测试三 哨兵丝滑测试及清弹测试	16
1.4.	整车测试	16
2.	操作手册	17

一、机械部分

1. 整体设计说明

哨兵机器人作为一个新加入战场的元素，要求其全自动运行，整个过程不能手动控制，这就需要它可以自主移动和自主防御射击。从战略战术上来看哨兵机器人也起着非常关键的作用，如果己方哨兵机器人一直存活，基地就可以增加 50%的防御。而且敌方近距离攻击基地的活动范围在哨兵的火力范围内，这就对敌方机器人造成了一定的威慑，在哨兵机器人存活的情况下，敌方不敢贸然攻击。所以，在其他机器人战斗能力相近的情况下，哨兵机器人成为拉开比赛差距的一个关键点。

综合分析，哨兵机器人需要实现在轨道上平滑快速地移动，搭载视觉辅助射击之后可以准确命中在火力范围内的敌方机器人，在场间三分钟的时候，可以快速从轨道上拆下并更换弹仓中的子弹，再将哨兵挂在轨道上。

针对此设计要求，做出了该哨兵机器人，在总决赛中，成功击杀一辆正在偷基地的敌方步兵机器人。

1.1. 设计方案

哨兵机器人的地盘运动部分主要采用两个驱动轮，两个从动轮的方式在轨道上表面移动，轮子支架与底盘固连，两侧装有两对导向轮，通过中间转轴连接在一起，靠导向轮来引导整个车身在弯道处平滑的转弯。关于导向轮的设计，第一版中采用两侧分别加导轮，靠弹簧将其压紧在轨道的两侧，后经过运动仿真和分析之后发现，哨兵机器人需要走过两个相反的弯道，导致导轮在垂直轨道的方向上位移较大，弹簧不能满足设计要求，后来改为中间转轴加两侧导轮，经过试验之后发现中间转轴加两侧导轮满足设计要求。且可以平稳的转弯。

1.2. 方案的优缺点

1.2.1. 优点

1) 快拆机构方便快捷。

利用简单的机构，使得整个哨兵机器人在从拆下到清弹再到安装到轨道上面整个过程耗时 27 秒左右，比另一组快了一倍多的时间。而且便于维护和安装。

2) 布置紧凑，整机重量轻。

-
- 3) 在整个设计过程中，尽量压缩哨兵的竖直方向的尺寸，使哨兵不超过尺寸限制，重量也尽量减轻，这样能够使哨兵机器人运动的更加迅速。
 - 4) 云台俯仰角足够大，配合视觉辅助，足够打到己方桥上，如果视觉允许的话，可以打到资源岛上的敌方机器人。
 - 5) 裁判系统下沉，减小哨兵在竖直方向上的尺寸。

1.2.2. 缺点

- 1) 供弹管路较长，且管路布置在结构内部，如果进入杂物或者异形子弹之后会卡在管内，维修拆卸费时间。
- 2) 没有做到真正的模块化设计，维护不方便。
- 3) 滑环部分的结构设计尺寸太紧凑，影响安装。
- 4) 钣金件存在加工误差，影响弹仓安装。

2. 易损结构及零件说明

- 1) 两端的触碰开关在工作或者搬运的过程中容易损坏。
- 2) 由于快拆机构采用碳板作为材料，长时间使用之后会被磨损，快拆机构产生间隙。
- 3) 加弹时应注意子弹的形态和清洁程度，如果有杂物或者变形严重的子弹进入供弹管路中，有可能会卡在里面，需要将其拆下来清理管道。拨弹轮部分在使用过程中，若突然有子弹进入出弹口上方位置时，容易产生卡弹的现象。
- 4) NUC 外接线路插头处，在调试过程中容易产生接触不良的现象。
- 5) 集电滑环在拆装的过程中应该注意线路，电线容易破损。

二、 嵌入式部分

1. 控制方案

1.1. 元件清单

序号	名称型号	数量	作用
1	Robomaster 开发板 STM32F427IIH6	2	系统主控
2	RoboMaster M3508 减速直流电机	2	驱动底盘
3	RM C620 无刷电机调速器	2	驱动底盘电机
4	微动开关	3	限位
5	正交编码器	1	对机器人进行定位
6	红外反射型传感器	1	对编码器清零
7	RM 6025 电机	2	驱动双轴云台
8	RM 2006 电机	1	拨弹电机
9	RM 820R 电调	1	驱动拨弹电机
10	RM 2312 电机	2	摩擦轮电机
11	SNAIL 电调	2	驱动摩擦轮电机
12	DR16 接收机	1	遥控器接收机
13	DT7 遥控器		给机器人保护作用
14	TB47 电池	1	电源
15	工业摄像头		
16	NUC	1	

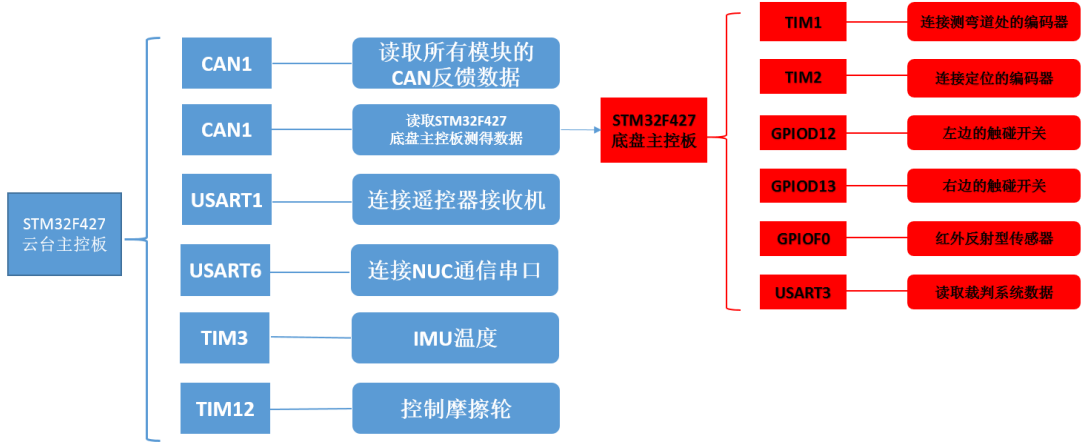
1.2. 硬件、软件资源配置

1.2.1. 硬件电路

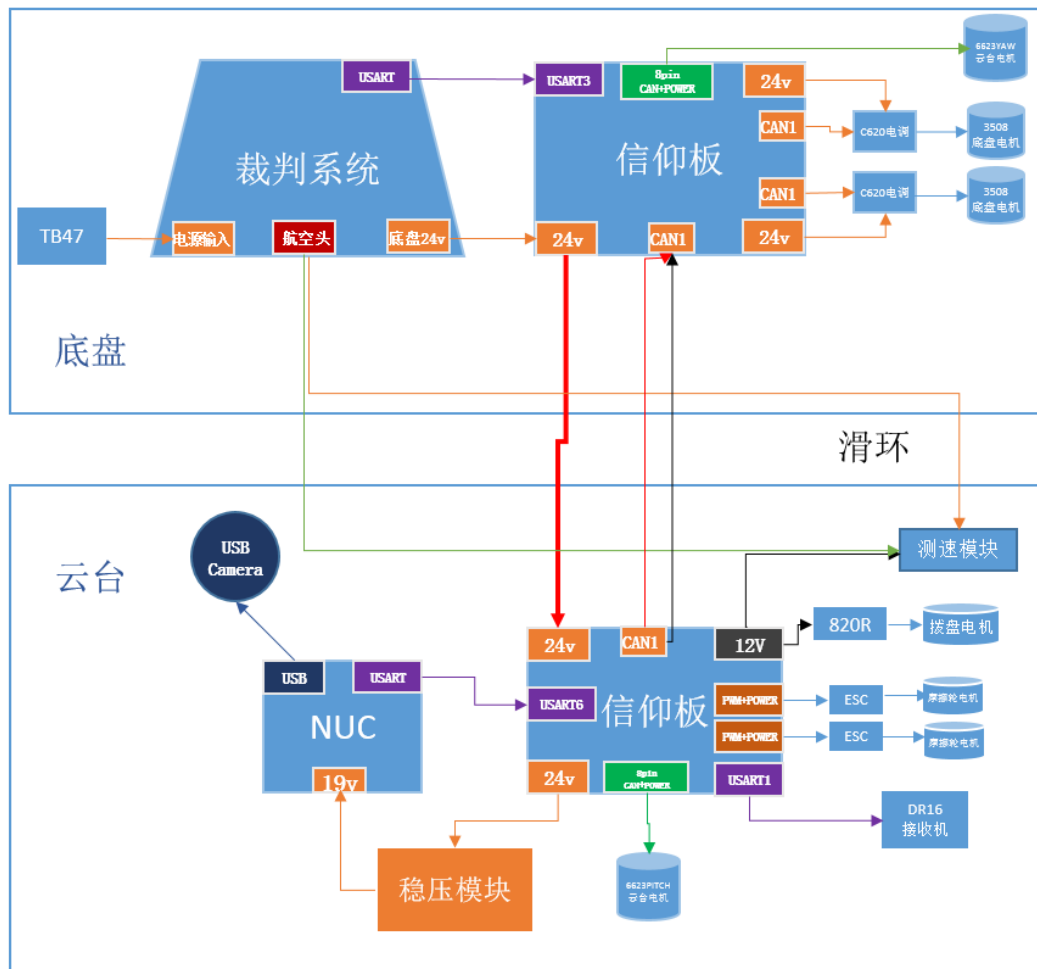
使用了两块 Robomaster 开发板，一块放在底盘上把编码器、触碰开关和枪口热量数据通过 CAN1 发送给云台主控板。云台主控板还通过 CAN1 控制云台电机、拨弹电机。发射机构采用步兵的发射机构，用 PWM 控制 2312 电机。

1.2.2. 软件资源分配

根据系统控制需求，调用了 CAN 总线、串口、定时器和 GPIO 等资源，具体功能分配如下图：



1.3. 控制电路布线结构，拓扑图

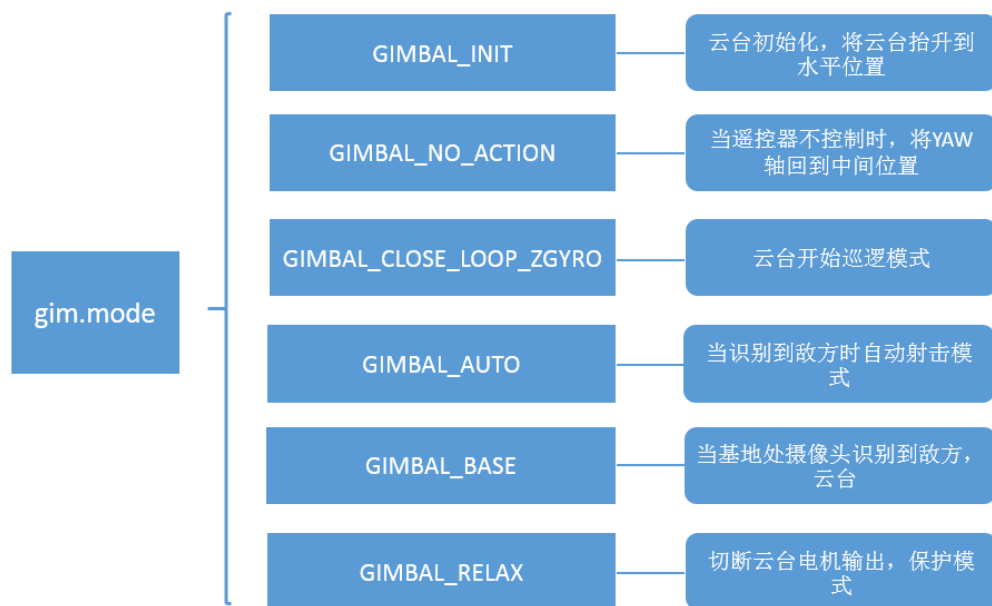


1.4. 软件架构、控制框图

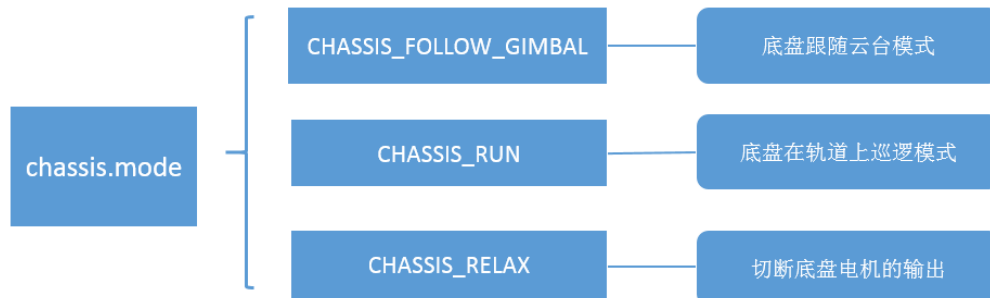
用到了 FREERTOS，软件架构采用步兵车的架构

Task 相关

Gimbal Task 云台的几个模式说明：



Chassis Task 底盘的几个模式说明：



2. 细节

2.1. 方案的优化过程

1.1 版本：用一个主控板控制所有传感器与电机，安装时发现过滑环线的口开的特别小，强行插进去，好多线都搞坏了。

1.2 版本：用两个主控板，把底盘电机、YAW 电机、触碰开关、编码器连接在底盘主控板上，云台主控板负责其他的外设。

1.3 版本：之前对编码器清零是通过两边的触碰开关撞击，容易损坏触碰开关和底盘结构，后来采用光电传感器在中间清零，在底盘上加入了 USART3 读取裁判系统的数据，发送给云台主控板，控制枪口热量。

1.4 版本：云台是一直 360 度旋转，后来考虑我们组整体情况，加上视觉有很多误识

别、快速运动的车和倾斜着对着摄像头的车都识别不到，最后总决赛采取“守家”策略，在面向基地处 180 度左右旋转。

2.2. 算法详细介绍

```
/* pit target */
pnp_dis = rece_data.vision_dis / 100.0; // rece_data.vision_dis 接收到的距离(cm)
dynamic_bias_pit = pnp_dis*dia_p + offset;
pit_vision_bias = rece_data.vision_y*VisionPithP + dynamic_bias_pit;
gim.pit_angle_ref = gim.pit_relative_angle + PitTestP*pit_vision_bias;

/* yaw target */
yaw_vision_bias = rece_data.vision_x + dynamic_bias_yaw;
gim.yaw_angle_ref = yaw_zgyro_angle - gim.yaw_offset_angle + YawTestP*yaw_vision_bias;
Last_Pit_Angle_Ref = gim.pit_angle_ref; // 记录识别到的 PIT 轴角度
Last_Yaw_Angle_Ref = gim.yaw_angle_ref;
```

三、算法部分

1. 整体

功能需求：

多摄像头装甲板检测：

功能：

- 1) 用多摄像头检测哨兵周围地区敌人
- 2) 优先锁定英雄车
- 3) 清楚知道敌人在哨兵的前方，桥头或者基地附近
- 4) 多摄像头轮流读图，节省 CPU 负载

1.1. 硬件环境

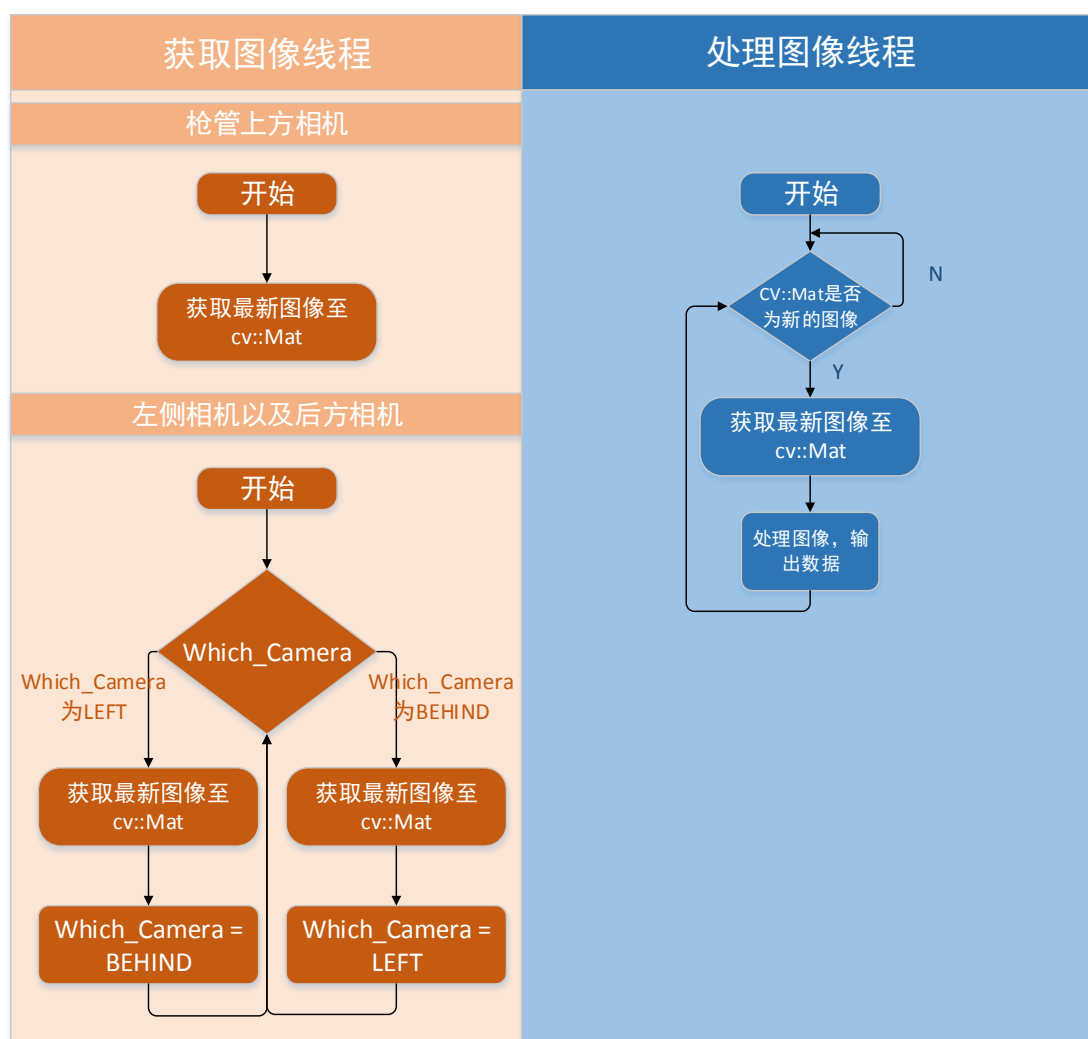
- 一组：NUC + 工业相机 + 2 个普通 USB 相机（总决赛和半决赛一组决定只用一个摄像头）
- 二组：TX1 + 2 个普通 USB 相机

1.1.1 软件环境

- Ubuntu16.04 系统+ROS Kinetic

1.2. 总体系统框图

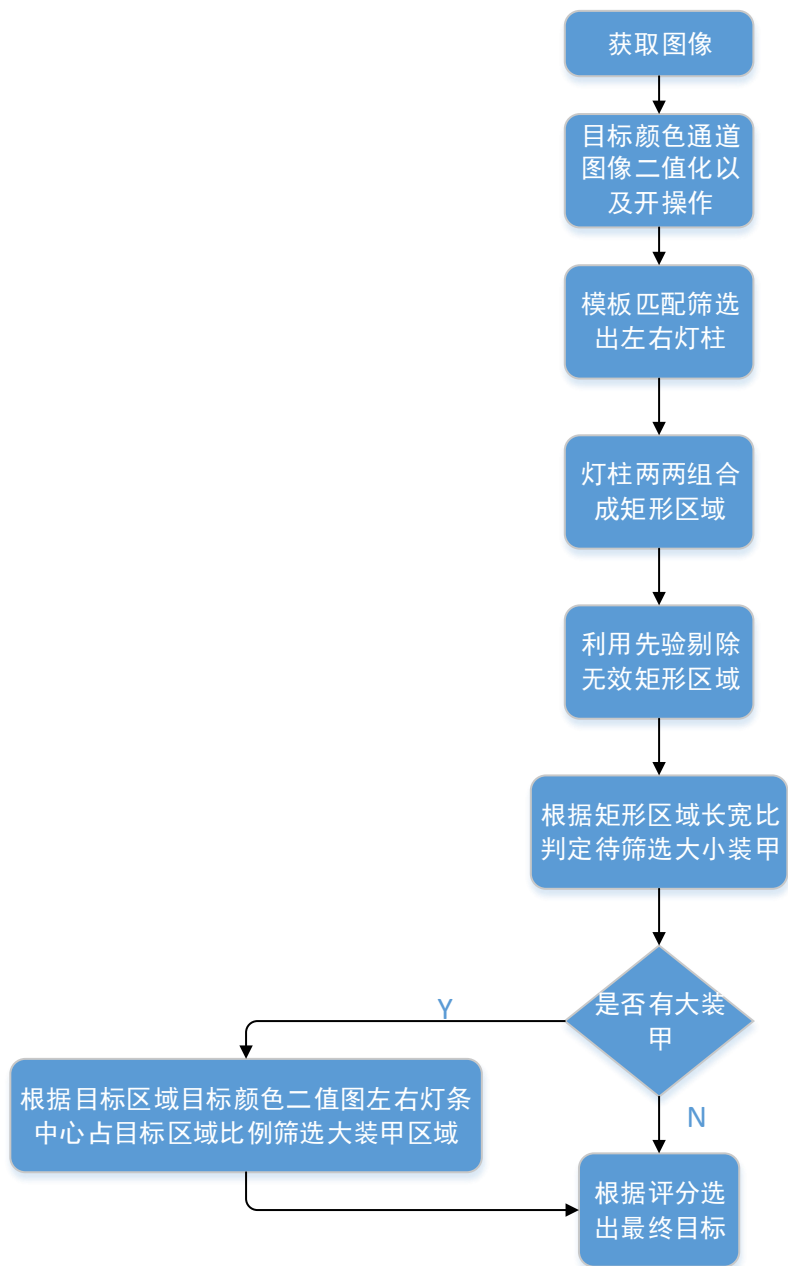
因为需要用到多摄像头进行对装甲板进行检测识别，一个摄像头放在枪管上面，用于检测哨兵前方荒地区域有无敌方，还有用于枪管的自动跟踪瞄准；左方的摄像头专门用于检测桥头方向有无敌人；后方的摄像头专门用于扫描基地区域有无敌人。



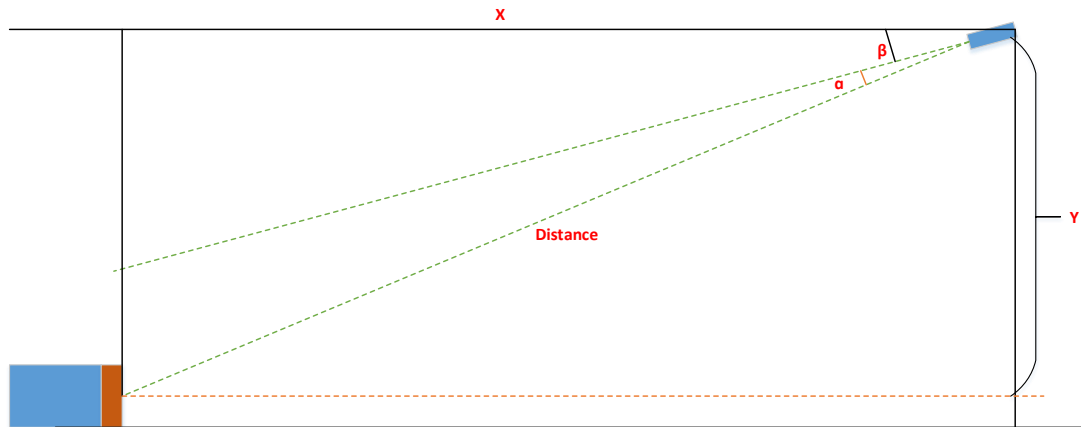
1.3. 算法方案

- 对目标颜色通道进行阈值二值化，同时加上滤掉白色灯光的部分，然后进行一个开操作

-
- 提取所有的灯柱，通过对所有灯柱进行模板匹配进行筛选出左右灯柱
 - 根据每个灯柱偏离垂直方向的角度以及水平方向角度等阈值排除灯柱，并左右灯柱包围成矩形组合成待挑选装甲板，每个矩形区域都有根据以下先验知识进行筛选和评分：
 - 左右灯柱角度差绝对值
 - 左右灯柱偏离垂直方向的角度绝对值
 - 左右灯柱中心点的连线与水平线之间的角度值
 - 对目标区域进行透视变换，变为正视图，利用以下先验知识对目标区域进行筛选：
 - 根据大小装甲长宽比筛选目标区域
 - 若存在上一次的检测目标，则当前目标与上一次目标的宽度和高度差不能超过一定阈值
 - 灯柱周围区域的目标通道灰度均值需大于其他通道的灰度均值
 - 黑色装甲中间区域绿色通道的梯度值较大的比例小于一定阈值
 - 若剩下的目标区域根据长宽比阈值判定是大装甲板，则对目标区域目标颜色二值图左右灯条中心矩形区域占目标区域的比例一定阈值进行筛选，若筛选出多于或等于一个大装甲板目标区域则选取评分最高的为目标；若都没有大装甲板目标区域，则根据选取评分最高的小装甲板区域为目标；
 - 若剩下的目标区域根据长宽比阈值判定是小装甲板，则根据选取评分最高的小装甲板区域为目标；
 - 每次识别到相应的装甲板后，只在其所在的一小部分进行识别运算，从而减小运算量，当目标丢失超过 10 帧时时，扩大搜索范围，不断累积，直到遍历全图；
 - 当连续识别到目标连续超过 5 帧才判定为识别到目标，才进行位姿的输出。
 - 实现的总体方案如下面流程图所示：



1.4. 枪管偏移



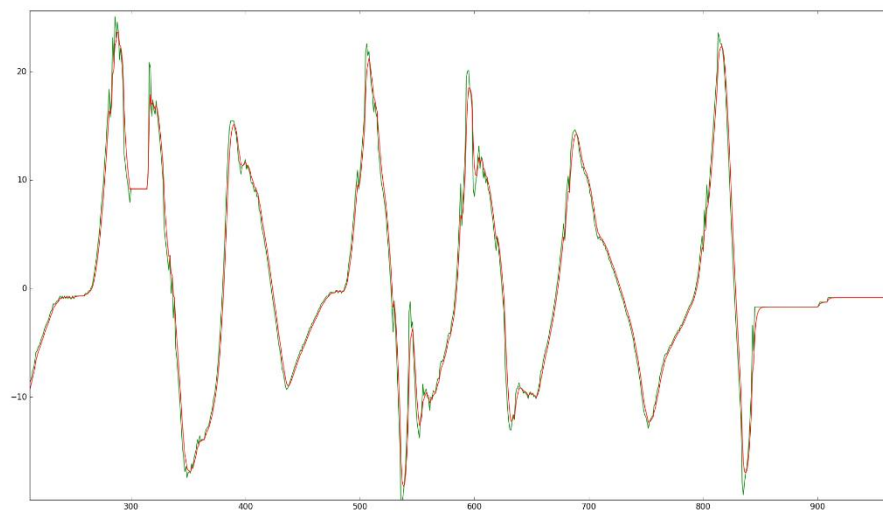
$$\begin{cases} X = V \cos \theta * t \\ Y = V \sin \theta * t - \frac{1}{2} g t^2 \end{cases}$$

$$X = \text{Distance} * \cos(\alpha + \beta)$$

在近似情况下， α 为通过 PnP 计算出来的角度， β 为云台当前 Pitch 轴角度， θ 为需要计算出来云台的 pitch 角度

根据以上公式就可以计算出云台 pitch 角度进行有效打击。

1.5. 平滑滤波平滑输出数据



拟合前五帧的数据进行平滑滤波。

1.6. 多摄像头处理图像

因为云台摄像头不仅用于搜索敌人，还用于瞄准跟踪射击敌人，所以云台的摄像头用一个节点读图处理，不进行轮流读图；

左方和后方的摄像头用另外一个节点，进行隔帧轮流读图处理图像，当轮到某一个摄像头发现敌方时会发送对应该摄像头的标志位给 `stm32` 进行转向瞄准射击。

1.7. 串口通信

byte0	byte1	byte2-byte5	byte6-byte9	byte10-byte13	byte14-byte17	byte18-byte21	byte22-byte25	byte26
8bit Int8_t	8bit Int8_t	32bit float	32bit float	32bit float	32bit float	32bit float	32bit float	8bit Int8_t
帧头	云台相机是否看到敌方的标志位	装甲板与目标点水平方向的偏差	云台需要抬高的角度	装甲板与相机的距离	后方相机是否看到敌方的标志位	左方相机是否看到敌方的标志位	输出的帧数	帧尾

1.8. 开机自启动配置

运用了 ROS 包中的 `robot_upstart` 包

1.9. 视觉方面总结

1.9.1. 预期效果与最终结果

- 预期效果：前方云台 100 度缓慢旋转扫描能迅速发现过往敌方 4m 以内车辆，其他相机能灵敏发现敌方；当发现敌方时云台会转到相应的位置迅速锁定敌方进行打击。
- 最终效果：在比赛中前方摄像头基本能迅速发现过往敌方 4m 以内车辆，其他摄像头在近距离可以迅速发现敌方，但距离变远时而且哨兵运动过快时就不能很好的发现敌方。

1.9.2. 方案的创新点

- 在一个处理器里连接了三个摄像头，采用了隔帧轮流读图并处理图像，这样子处理使得 `cpu` 的负载不会太大，减少程序崩溃的可能。
- 多摄像头互相配合可以使哨兵几乎无死角的发现敌方，做到迅速响应并且打击敌

方。

1.9.3. 存在的坑

- 视觉算法有时候会误识别到场地上方的灯柱
- 解决方法：
 - 对目标颜色通道进行阈值二值化时加入对白色的阈值筛选，然后再进行一个开操作，这样子误识别就会减少；
 - 当得知云台的 **pitch** 轴的时候，当误识别到上方白灯时，结合 **pitch** 轴角度与 PnP 计算出来目标区域相对于镜头的角度即可剔除误识别区域。

1.9.4. 可以改进的地方

- 当左方摄像头或者后方摄像头识别到敌方时，可以发出敌方相对于左方或者后方摄像头的相对位置，使云台可以根据当前 **yaw** 轴的角度快速知道敌方相对于当前的位置并快速转到该位置进行锁定攻击，由于后期的大部分时间都花在调试上，这个细节就没有去做；
- 预测敌方的运动，因为相机只有 30 帧，如果帧与帧之间物体运动速度特别快，整个控制系统的延时不能满足跟踪需求。因此需要结合陀螺仪返回的 **pitch** 角和 **yaw** 角信息做一个预测。如果出现丢帧情况，该帧的目标数据能通过历史帧的图像像素坐标进行二次拟合得到修正。在之前的手自一体测试中，我尝试结合前 5 帧对运动敌方进行预测，当敌方小车在运动时效果还行，但是当速度有急剧突变时，云台就会抖，呈一个发散的状态，越来越抖，后来如果放在嵌入式上做预测控制上会好很多；
- 这一次我只预测 1 帧的数据，该方法会减少掉帧对控制系统的影响。

1.10. 元件清单

组别	一组	二组
处理器	NUC	TX1
云台相机	CMOS迷你工业相机90°镜头	1920x1080的90°USB摄像头
辅佐相机	罗技C270摄像头 KS2A17	KS2A17

四、总结

1. 预期效果与最终效果

预期效果：哨兵机器人可以实现在轨道上平滑快速移动，搭载视觉辅助射击之后，可以准确命中在火力范围内的敌方机器人，正常运行时在轨道上进行来回巡逻，且云台做 360° 持续扫描。当识别到敌方机器人装甲板时炮管指向敌方机器人装甲板，靠近机器人进行打击。当基地启动区中发现敌方机器人时，立刻将炮管转向基地一侧，进行防御打击。在场间三分钟的时候，可以快速从轨道上拆下并清除弹仓中的子弹，再将哨兵挂在轨道上。

最终效果：哨兵机器人在靠近基地的一半轨道上做往复运动，进行防御。云台 180° 扫描基地启动区区域，当识别到敌方机器人装甲板时，立即开始射击。在总决赛的比赛中，击杀了一辆敌方步兵机器人。

2. 方案的创新点

- 1) 快拆机构方便快捷，在拆装的过程中节省时间和人力。（比另外一种方式快一倍多）
- 2) 子弹从滑环中心流过，云台可持续旋转。
- 3) 采用中间转轴加两侧导轮的形式辅助哨兵机器人过弯。
- 4) 利用红外反射型传感器，辅助编码器清零，避免了到两端碰柱子对机器人的损伤。
- 5) 在底盘和云台上分别使用一块主控板，然后两个主控板相互通信，这样处理之后大大减少了滑环的走线，可以减小滑环的尺寸。
- 6) 把所有设备都挂载在 CAN1 上，走线更加方便。
- 7) 每次打击到装甲板时装甲板都会闪灭，此时视觉就识别不到，通过记录上一帧的云台的数据，可以做到精确打击。

3. 存在的坑

- 1) 该哨兵机器人进行了模块化设计，但是在设计过程中将供弹结构中的其中一部分包在了其他结构中，导致管路中卡弹时不容易清除，拆卸比较困难。
- 2) 视觉方面会误识别到很多地方，比如场地的灯光、场地围栏反射到的蓝色灯光、当步兵车不是正对着时也会检测不到。

4. 可以改进的地方

- 1) 通过使用两块主控板，减少了滑环线路的数量，可以更换一个 6 路的滑环，减小滑环的尺寸，这样就更加有利于整个哨兵机器人的结构布局。
- 2) 供弹管路应该便于维修拆卸。或者供弹机构可以考虑安装在云台发射机构后方，每次定量从弹仓中落下子弹到云台弹仓。这样就避免了复杂的管路，减少哨兵机器人在运行过程中卡弹的现象。
- 3) 加装三个或者三个以上摄像头，实时监控各个关键区域。在轨道中间建立二维坐标系，结合视觉反馈的距离，做到最近距离打击。

附录

1. 测试过程和数据

1.1. 测试一 哨兵方案设计测试（中间转轴加导轮）

A. 实验目的

测试中间转轴加导轮是否可以在轨道上平滑的移动。

B. 实验工具

加工三块亚克力板，及四个摩擦轮

C. 实验结论

实验发现采用中间转轴的方式可以使哨兵底盘在轨道上平滑的移动，轴距对前后两轮的转向角有影响，轴距越大，转向角越大，应选取合适的轴距。

1.2. 测试二 拨弹轮拨弹测试

测试拨弹轮拨弹速度以及在拨弹过程中会不会产生卡弹的情况。

1.3. 测试三 哨兵丝滑测试及清弹测试

	第一次	第二次	第三次
丝滑测试（完成一个来回所用的时间）	12.81s	11.56s	11.69s
清弹测试（从轨道拆下加弹，然后安装到轨道的时间）	42.85s	38.70s	28.93s

表一 哨兵丝滑测试及清弹测试

1.4. 整车测试

完成整车功能之后，在场地上操作步兵机器人当做靶子，通过射击子弹来调节摩擦轮的占空比，设定过两个不同的射速。并且通过打击装甲板来调节云台参数

2. 操作手册

A. 跑整个轨道

把哨兵机器人放在右边，遥控器右开关上，机器人可以工作，触碰右边的触碰开关，机器人即可以运行，触碰左边的触碰开关，机器人底盘即停止运行，把遥控器右边开关拨到中下，机器人都不工作。

B. 跑半个轨道：

- a) 动：把哨兵机器人放在中间，遥控器右开关拨到上，即可运动；
- b) 不动：把哨兵放在中间，不开遥控器，用手推动机器人向左运动，挪到你想要的位置，打开遥控器，机器人不动，适合定点守家。