

在 RoboMaster 比赛中，机器人需要精准的命中敌方机器人身上搭载的四块装甲板才能造成有效伤害，这就需要我们精准的获取目标装甲板的位置用于计算发射时的云台姿态。在比赛中，有两种不同的装甲模块，大型装甲板（230mm*127mm）和小型装甲板（135mm*125mm）。它们都贴有数字标签（从 1 到 5）以及一些特殊图案，并且在装甲模块的边缘有两个灯条，灯条的灯光可以是红色或蓝色。当己方为红队时，机器人上的灯光将是红色的，目标机器人上的灯光将是蓝色的。同时装甲模块上的数字标签用于区分不同的机器人。不同的机器人在装甲模块上有不同的数字标签。



图 2.1 不同类型的装甲板模块

同时由于比赛中发射的是塑胶弹丸，弹速一般为 10~15m/s，会在出膛后有一定的飞行时间，因此估计出目标的速度进行目标的预测是非常有必要的。

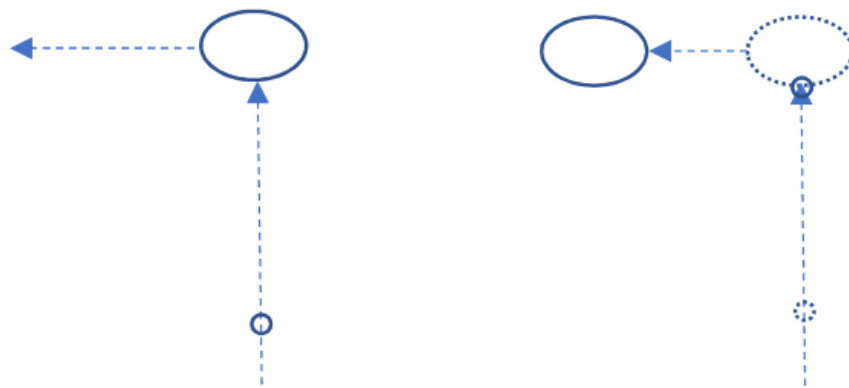


图 2.2 没有进行预测时的射击效果

如图 2.2 所示，在左图状态下发射一枚弹丸，当弹丸到达目标在发射时的位置时，目标已经离开了瞄准的位置。

总的来说，想要在比赛中实现良好的自动瞄准效果，我们需要：

- (1) 在图像中实时的识别目标并精准的获取目标相对己方机器人的三维位置；
- (2) 较为精确的估计目标的运动状态以获得实时的预测瞄准位置。

三维重建算法

由于我们需要获取目标相对己方机器人的三维位置才可计算出弹丸发射时的云台姿态，因此我们需要使用三维重建算法。

三维重建方法有很多种，包括但不限于：

- (1) 基于 RGB-D 深度相机^[8]或融合相机及激光雷达^[9]直接获取深度信息计算对应三维位置；
- (2) 基于深度学习的深度估计^[10]；
- (3) 通过单目 PnP 从单张图像中恢复物体的三维位置姿态^[6]。

方法（1）可以直接通过硬件实现获取深度信息，因此在合适的距离下的距

离精度应当是最高的，但受制于目前市面上的深度相机及激光雷达 FOV 较大，在远距离情况下得到的深度图较为稀疏，取装甲板大小的深度信息抖动较大，并且由于相机 FOV 较大，识别器的有效识别距离不足，以 Intel Realsense D455 深度相机为例，最远识别距离仅为 3m，无法满足比赛的需求。方法（2）分为绝对深度估计及相对深度估计，在该任务场景下我们需要绝对深度估计，而由于绝对深度估计泛化能力极差，并且由于网络推理延时较高，也不满足自动瞄准任务的需求。方法（3）的劣势主要为需要知晓目标三维角点的先验信息，但在 RoboMaster 比赛中目标都是固定且可知的，并且方法（3）只需要单目 RGB 图像即可获取目标的位置姿态，而一般工业相机都可以提供高帧率的 RGB 图像，还可根据不同机器人识别距离的需求更换不同焦距的镜头，故笔者选用方法（3）作为自动瞄准系统的三维重建算法。

目标识别算法

由 2.1 中可知，我们需要获得图像中的目标对应角点的信息，因此需要选用合适的目标识别算法。由 1.2 可知，目前在 RoboMaster 比赛中应用的识别算法主要分为传统方法识别灯条结合机器学习进行图像分类以及使用神经网络直接回归装甲板角点。传统方法一般为对图像二值化中找寻符合装甲板灯条特征的轮廓，再对候选灯条进行匹配。图像分类任务与常见的数字分类任务比较接近，数字分类方法一般有：

- （1）逻辑回归^[11] (Logistic Regression)，用于预测二元结果，如是/否，通过/失败等。
- （2）朴素贝叶斯^[12] (Naive Bayes)，用于计算一个数据点属于某个类别的概率，基于特征之间的独立性假设。

- (3) K-最近邻^[13] (K-Nearest Neighbors), 用于根据训练数据集中最接近的 k 个数据点来对新数据点进行分类。
- (4) 决策树^[14] (Decision Tree), 用于通过不断将数据集分割为更小的子集来构建分类或回归模型。
- (5) 支持向量机^[15] (Support Vector Machines), 用于通过寻找能够最大化两个类别之间边界距离的超平面来进行分类。
- (6) 多层感知机^[16] (Multilayer Perceptron), 用于分类或回归问题, 通过反向传播算法来学习网络的权重和偏置, 以最小化损失函数。
- (7) 卷积神经网络^[17] (Convolutional Neural Network), 它的人工神经元可以响应一部分覆盖范围内的周围单元, 对于大型图像处理有出色表现。

而通过神经网络回归装甲板角点主要分为 1-stage 方法及 2-stage 方法, 代表方法为:

- (1) 1-stage 的单发多框检测器^[18] (Single Shot MultiBox Detector) 及 YOLO^[19] (You Only Look Once), 可以在一次前向传播中完成目标的定位和分类, 而不需要借助候选区域生成的步骤。
- (2) 2-stage 的 Fast RCNN^[20], 先使用选择性搜索 (selective search) 算法在图像中提取候选区域 (region proposals), 再将原始图像输入卷积神经网络 (CNN) 中, 得到整个图像的卷积特征图 (feature map), 再通过神经网络进行分类及回归。

考虑到自动瞄准系统对于算法实时性以及角点定位精确度的高要求, 笔者选择采用传统方法识别灯条结合机器学习进行图像分类的方法。在分类器的选择上,

考虑到目标图案为印刷体数字，并且不会有位移及形变，而且需要进行多分类，故笔者选择使用多层感知机作为装甲板中图案的分类器。

跟踪算法

由于图像中会同时出现多个目标，在自动瞄准时云台应该保持持续锁定同一目标，所以跟踪算法是非常有必要的。跟踪算法主要有基于图像信息的 KCF 跟踪算法^[21]，是一种基于核相关滤波（Kernel Correlation Filter）的目标跟踪算法，它是在 CSK 算法的基础上进行了改进，提高了跟踪的准确度和鲁棒性，以及使用卡尔曼滤波器（Kalman Filter）对每个目标的状态进行预测和更新同时仅使用 IoU 作为数据关联的指标的多目标跟踪算法 SORT^[6]（Simple online and realtime tracking）等。考虑到对于实时性的要求以及在该自动瞄准任务中，目标的图像特征都十分相似，KCF 不适合用于该任务的跟踪，故笔者选用 SORT 算法中对于目标滤波及关联的处理思路作为单目标跟踪器的算法基底。

状态估计算法

由 2.1 可知我们需要对于运动目标有一定的预测，这需要我们对于目标的运动状态（例如目标的运动速度）有一定的估计，而除位置姿态外其他的状态是无法直接被我们的系统所观测的，这就需要状态估计器来从观测数据来推断系统状态。

状态估计算法有很多种，其中比较常见的有：

- （1）卡尔曼滤波（Kalman Filter）：一种基于线性系统模型和高斯噪声假设的最优状态估计算法，它利用系统的状态转移方程和观测方程，通过预测和更新两个步骤，来递归地求解最小均方误差的状态估计。
- （2）扩展卡尔曼滤波（Extended Kalman Filter）：一种基于非线性系统模

型的状态估计算法,它通过对系统的状态和观测方程进行线性化近似,然后应用卡尔曼滤波的框架,来处理非线性系统的状态估计问题。

- (3) 无迹卡尔曼滤波 (Unscented Kalman Filter): 一种基于非线性系统模型的状态估计算法,它通过对系统的状态分布进行采样,然后根据采样点的变换和加权,来近似系统的非线性变换和预测分布,从而避免了线性化近似带来的误差。
- (4) 粒子滤波 (Particle Filter): 一种基于蒙特卡洛方法的状态估计算法,它通过对系统的状态空间进行随机采样,然后根据观测数据对采样点进行重要性加权和重采样,来近似系统的后验分布和最优状态估计。
- (5) 多变量状态估计技术 (Multivariate State Estimation Technique, MSET): 一种基于非线性多元预测诊断技术的状态估计算法,它通过分析对比实际监测参数与设备正常运行时的健康数据为基础,对正常运行时的各个参数进行运算并做出估计,以这种正常的状态估计作为标准。

考虑到在 RoboMaster 比赛中,目标机器人为了规避弹丸的击打,运动状态一般为旋转运动叠加平移运动,而我们观测的对象为目标机器人上所搭载的装甲板,为了获取最佳的击打效果,我们应该估计出目标机器人的整体运动状态而非单一装甲板的运动状态,这意味着状态空间到观测空间的转换是一非线性过程,所以卡尔曼滤波是不能直接使用的。同时考虑到这一过程的非线性程度不大,再加上系统对于实时性的要求,故笔者采用扩展卡尔曼滤波作为自动瞄准系统的状态估计算法。

综上所述,由于我们需要在图像中实时的识别目标并精准的获取目标相对己方机器人的三维位置以及较为精确的估计目标的运动状态以获得实时的预测瞄

准位置，我们需要使用三维重建、识别、跟踪及状态估计算法来实现这一过程。考虑到比赛中击打距离的需求、待识别目标的特征、整体运动状态以及系统对于实时性的要求，笔者选用单目 PnP 算法进行三维重建，传统方法识别灯条结合机器学习进行图像分类进行目标识别，采用 SORT 思想的单目标跟踪器进行目标跟踪，以及扩展卡尔曼滤波进行状态估计。

算法系统框架

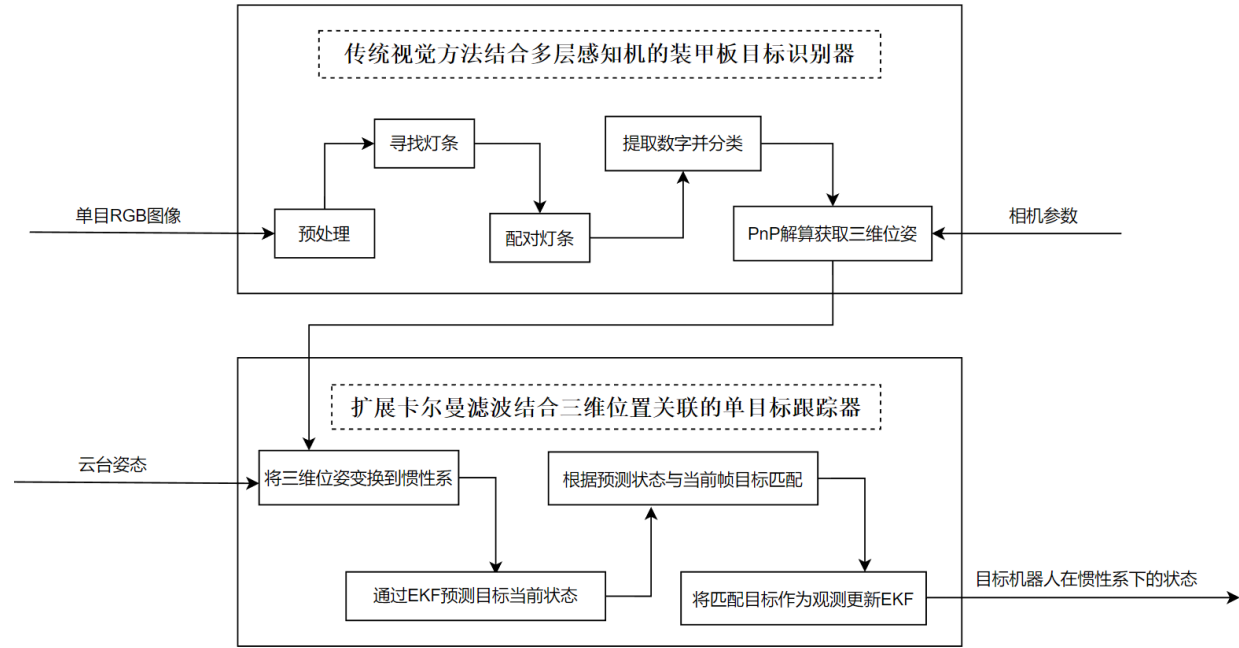


图 3.1 自动瞄准系统整体框架示意图

该算法系统的输入为单目 RGB 图像以及相机的内参矩阵及镜头畸变系数，以及云台姿态信息，最终输出在惯性系下跟踪着的机器人状态，其中包含机器人中心位置及机器人的运动线速度和角速度，以及两对装甲板的相对位置关系。