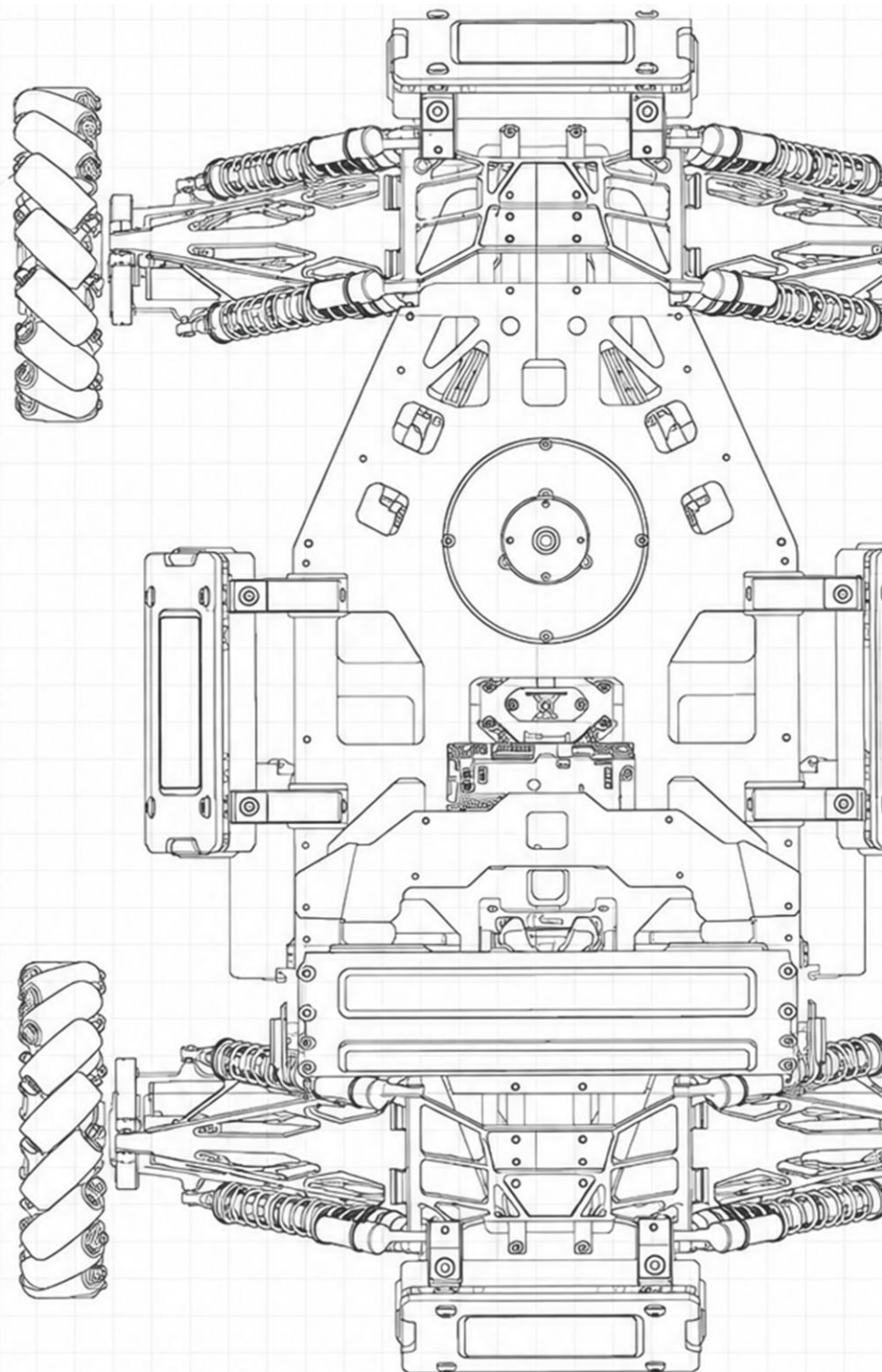




山东理工大学
SHANDONG UNIVERSITY OF TECHNOLOGY

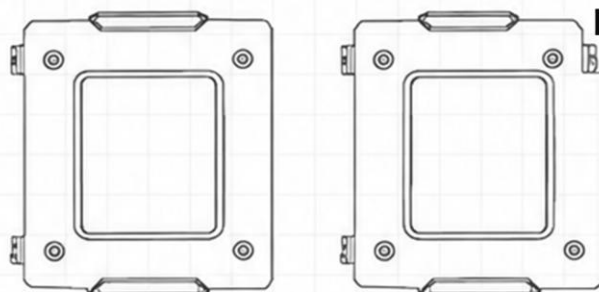


齐奇战队RoboMaster2026 开源报告

双目测距与飞镖系统前端制导 方法

山东理工大学齐奇战队

2026年8月



前言

机械镖一定要有机械！稳定的机械结构是本篇开源调试方法的前提，本赛季镖架可以保证镖体落点散布在一个小装甲板内，为后续调试打下了基础，具体设计思路可见战队本赛季飞镖系统结构开源，本篇不会涉及详细的机械设计。

我认为这篇文档最值得参考的是测距精度的优化方法部分和根据场地条件（roll 轴、pitch 轴偏角和基地水平距离、高度）动态调整拉力与 yaw 轴参数的方法。我认为这种方法不止适合用在飞镖调试场景，还可以应用在英雄吊射或自瞄等场景。

希望这篇文档不仅能帮助到设计镖架制导的同学，也能让大家可以把文中的思路应用在其他兵种的设计甚至是比赛以外的工程设计中。

1. 双目测距

1.1 双目测距原理

理论双目测距应该使用两台完全相同的左右摄像机，并尽量保持两个相机的光轴（通过镜头的光学中心并垂直于成像平面的假想直线）平行固定，此时双目测距原理可简化为以下模型（图 1）：

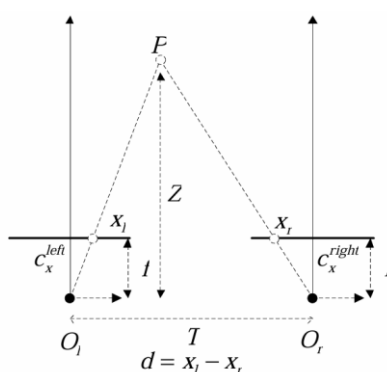


图 1 理想视差测距原理

其中， T 为两台摄像机投影中心 O_l 与 O_r 的距离， f 为摄像机焦距，物理世界中的点 P 在左右图像上的成像点横坐标分别为 x_l 与 x_r ， Z 表示该物理点 P 与左右摄像机连线的垂直距离，视差定义为 $d = x_l - x_r$ 。通过三角形相似原理可推导出 Z 值如下：

$$Z = \frac{fT}{x_l - x_r} = \frac{fT}{d}$$

1.2 双目测距精度校准

我们假设两台相机为理想相机，且固定后光轴完全平行，投影中心距离已知，将引导灯中心在两个相机的成像坐标代入上面方程即可得到测试数据，将相机放至水平，引导灯高度与相机平齐，每隔一米记录一组实际距离值及其对应测量值，下面是一次安装后测量得到的数据：

实际值 (m)	15	16	17	18	19	20	21	22	23	24	25	26	27	28
测量值 (m)	17.874	18.89	19.871	20.853	21.785	22.725	23.622	24.512	25.402	26.081	27.067	27.922	28.732	29.514



图 2 测距所用相机

数据显示，原始测量值与实际值存在显著偏差，在 25 米处误差可达 2 米以上。此现象源于现实中难以满足理想安装条件——包括内参标定误差、机械装配偏差（如相机光轴非完全平行）等因素叠加所致。

将上述数据绘制为散点图（图 3）。

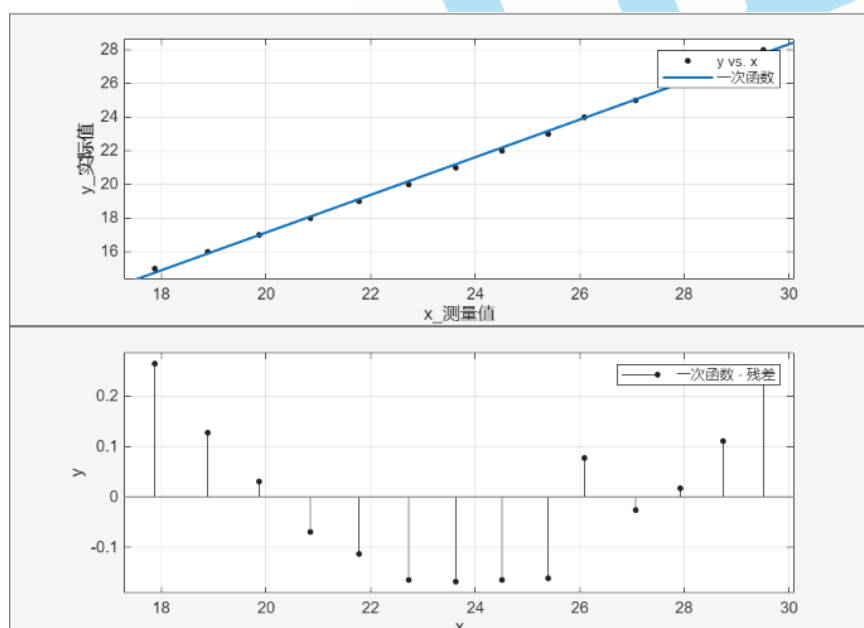


图 3 测量值与实际值

分析测量值 x 与实际值 y 的关系散点图发现，二者近似呈线性关系。初步尝试一次函数拟合后观察残差分布，发现其趋势接近二次函数形态。因此采用二次多项式进行拟合，并剔除明显异常的两个数据点得到测量值与实际值的关系图（图 4）：

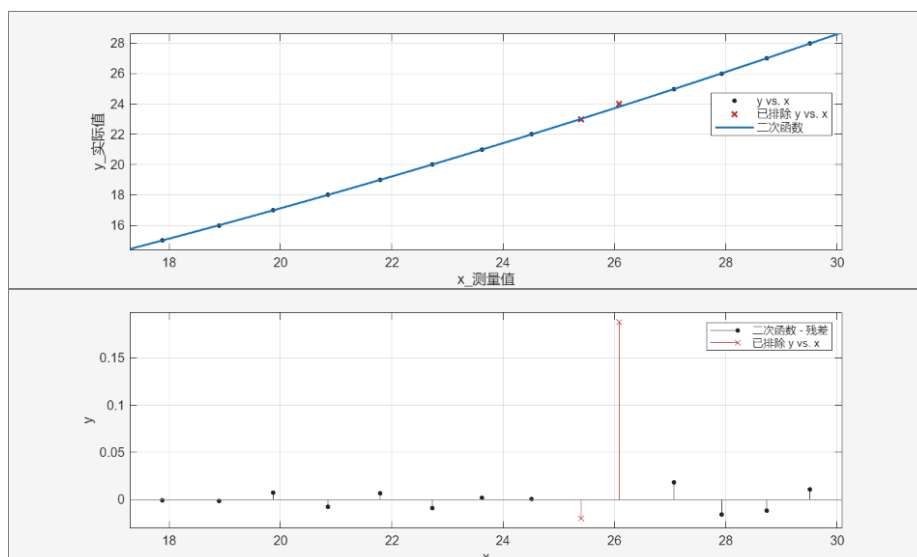


图 4 实际值与测量值关系拟合图

通过上面方式拟合可得方程（此方程仅适用于这一次装配，在这里只起演示作用，实际拟合结果受两相机光轴夹角和焦距等因素影响）：

$$y = 0.0123x^2 + 0.5331x + 1.5432$$

其中 x 为测量值， y 为实际值，由上面方程进行简单换算即可得到精准的测距参数，函数内插区间为 15m 到 28m，完全满足赛场使用需求。

对于更大范围或更高精度的应用场景，可通过引入含具体误差项（如相机夹角、内参畸变）的复杂模型进行拟合，快速反演参数。但在当前应用范围内，二次函数已提供足够精度，无需额外复杂建模。

至此，已实现对相机至引导灯直线距离的高精度估计。结合陀螺仪读取当前 Pitch 轴倾角 θ ，通过三角函数运算即可解算出引导灯相对于发射机构的水平距离 D 与相对高度 H ：

$$D = Z \cdot \cos(\theta), H = Z \cdot \sin(\theta)$$

2. 前端瞄准和拉力计算

2.1 前端瞄准

本赛季飞镖发射架采用 yaw 轴和 pitch 轴都可动的结构，但是并不通过调节 pitch 轴角度改变飞镖落点，而是通过可动扳机调整拉力，从而改变落点。pitch 轴始终通过相机返回的 y 坐标调整角度，保持引导灯在 y 轴坐标中间值，即使引导灯在相机光轴平面的状态。

视觉识别方面，我们使用 OpenCV 做传统视觉的色块识别，将画面中引导灯色块中的中心点像素坐标作为相机的成像点，并以右相机的坐标值作为返回值与设定的目标值作差进行瞄准，当 x 和 y 方向差值都小于 3 个像素点时视为瞄准完成。

下图（图 5）为上场时使用的每一发镖体的基础参数，用来区分不同镖体由于打印质量和装配误差产生的落点偏移，不直接用于瞄准，须结合镖架 roll 轴倾角、pitch 轴倾角和与引导灯直线距离计算得到实时补偿量之后，与基础参数相加才可用于瞄准，补偿量的具体解算方法见后续章节。

```
// 第一发参数
//535 360 -3
auto_ctrl.dart_number[0].target_coord_x = 535;//540//640//505//541
auto_ctrl.dart_number[0].target_coord_y = 360;
auto_ctrl.dart_number[0].trigger_angle = -6 + TRIGGER_OUTPOST_BASE_ANGLE;

// 第二发参数
//530 360 -6
auto_ctrl.dart_number[1].target_coord_x = 560+30;
auto_ctrl.dart_number[1].target_coord_y = 360;
auto_ctrl.dart_number[1].trigger_angle = - 8 + TRIGGER_OUTPOST_BASE_ANGLE;

// 第三发参数
//548 360 -9
auto_ctrl.dart_number[2].target_coord_x = 548;
auto_ctrl.dart_number[2].target_coord_y = 360;
auto_ctrl.dart_number[2].trigger_angle = -9 + TRIGGER_OUTPOST_BASE_ANGLE;

// 第四发参数
//548 360
auto_ctrl.dart_number[3].target_coord_x = 548;
auto_ctrl.dart_number[3].target_coord_y = 360;
auto_ctrl.dart_number[3].trigger_angle = -9 + TRIGGER_OUTPOST_BASE_ANGLE;
```

图 5 每发镖体参数

2.2 拉力与 yaw 轴补偿计算

2.2.1 roll 轴倾角对 yaw 轴偏角的影响

在测试中不难发现，roll 轴倾角对镖体落点的左右方位有影响，具体为左倾落点偏左，右倾反之。

此时就需要根据 roll 轴的倾角调整 yaw 轴瞄准的参数，于是我们可以像上面校准测距一样如法炮制得到 yaw 轴的校准方程。

下面是测量镖体 yaw 轴基础参数和补偿量的过程：

- ① 确定基础参数和零点：部署发射架并瞄准后进行试射，根据落点调整 yaw 轴基本参数，调整至落点正对目标，没有左右偏移后，得到当前镖体 yaw 轴基本参数；记录当前 roll 轴角度，并将补偿量记为零。
- ② 测量补偿量：改变镖架 roll 轴偏角，再次进行试射，并根据落点调整 yaw 轴补偿量，调整至落点正对目标，没有左右偏移后，记录当前 roll 轴角度和 yaw 轴补偿量。
- ③ 重复第二步得到不同 roll 轴偏角情况下的 yaw 轴补偿量，将 roll 轴偏角和 yaw 轴补偿量数据导入 MATLAB 曲线拟合器中拟合得到补偿量的计算方程。

将 roll 轴倾角作为输入量，代入补偿量计算方程，得到 yaw 轴瞄准的补偿量，将补偿量与每一发镖体原本的目标值相加得到最终的 yaw 轴目标参数。

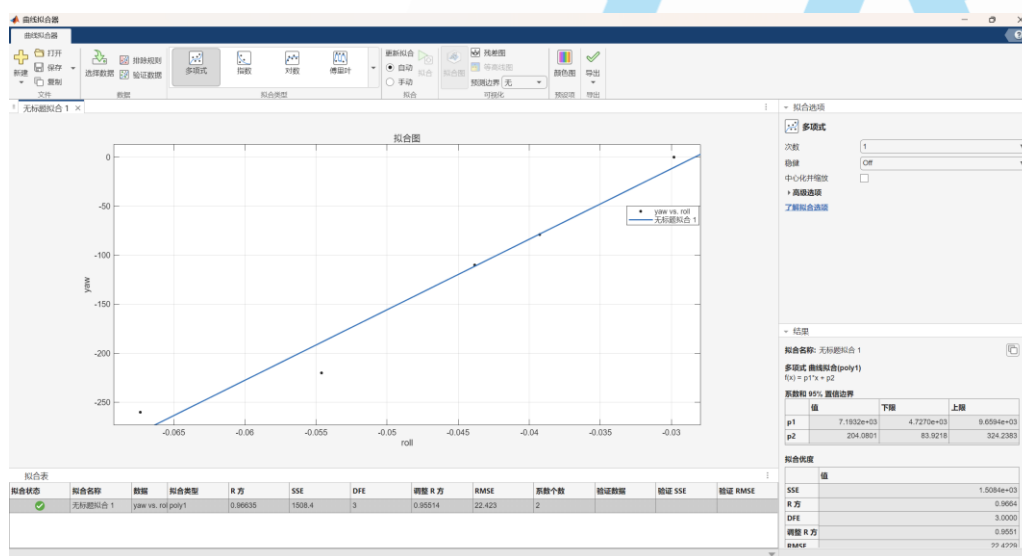


图 6 yaw 轴补偿值方程拟合结果

2.2.2 拉力计算方法

2.2.2.1 拉力影响因素

这里的拉力计算方法和前面双目测距精度校准方法一样，都是由实测的数据通过拟合的方法得到对应的函数，只是输入变量由一个变为多个。

拉力大小受多重因素影响，其中引导灯水平距离与引导灯高度为主要影响因素。经实验验证，roll 轴倾角也会对拉力产生显著影响。因此在计算拉力时应将上面三个因素作为自变量纳入考量，而水平距离与高度是由引导灯直线距离和相机角度计算而来，所以在这里直接用直线距离、相机角度和 roll 轴倾角作为自变量。

2.2.2.2 得到拉力计算方程的步骤

- ① 试射：将飞镖靶摆在目标距离附近进行试射，根据落点远近调整拉力补偿值，调整至命中装甲板后，记录当前读取到的直线距离、pitch 轴倾角、roll 轴倾角。
- ② 获取更多样本点：调整飞镖靶位置，改变发射架 roll 轴倾角，重复上一步，得到在不同距离、高度，不同发射架 roll 轴倾角时需要的拉力补偿值。
- ③ 将上面得到的样本点导入 MATLAB 拟合得到拉力补偿值计算方程。

下图是一次测试得到的数据，每一个样本点包含命中不同位置的目标时单片机读取到的 pitch 轴、roll 轴偏角、引导灯距离，以及对应的拉力补偿值。

拉力	Pitch (rad)	Roll (rad)	距离 (mm)
386	-0.56041962	-0.03613765	25481
383	-0.55938643	-0.03614958	25531
382	-0.55664694	-0.03620445	25515
382.5	-0.55397689	-0.03575422	25613
380	-0.55459923	-0.03632821	25432
381	-0.55575144	-0.03587065	25408
382	-0.55725616	-0.03610776	25465
385	-0.56002772	-0.03634723	25515
388	-0.56200165	-0.03625968	25533
350	-0.5592066	-0.03653842	25164
350.5	-0.56005704	-0.03680508	25114

346	-0.5548948	-0.03650634	25090
342	-0.55253547	-0.03620944	25128
347	-0.55221832	-0.03587084	25429
358.7000122	-0.55842727	-0.0365673	25364
400	-0.56039369	-0.03933974	25126
400	-0.56052649	-0.04166222	25141
395	-0.56088793	-0.02903787	25185

现设拉力为 y ，pitch 轴倾角、roll 倾角、距离分别为 x_1 、 x_2 、 x_3 拉力补偿值计算方程设为

$$y = a_0 + a_1x_1 + a_2x_2 + a_3x_3$$

将上面自变量导入 MATLAB 并合为一个矩阵后，用 regress 函数求出参数 a_0 、 a_1 、 a_2 、 a_3 ，得到方程：

$$y = -2770.4356 - 3654.5973 * x_1 - 680.9221 * x_2 + 0.04268 * x_3$$

```
X_data = [x1, x2, x3];
%加入截距列
n=18;%样本量
X_design = [ones(n, 1), X_data];

[params_builtin, bint, r, rint, stats] = regress(y, X_design);

disp('=== 内置 regress 函数求解的参数 ===');
disp(['截距 a0 = ', num2str(params_builtin(1))]);
disp(['系数 a1 = ', num2str(params_builtin(2))]);
disp(['系数 a2 = ', num2str(params_builtin(3))]);
disp(['系数 a3 = ', num2str(params_builtin(4))]);
disp(['R^2 = ', num2str(stats(1))]); % 决定系数，越接近1越好
```

```
>> fitting(x1,x2,x3,y)
=== 内置 regress 函数求解的参数 ===
截距 a0 = -2770.4356
系数 a1 = -3654.5973
系数 a2 = -680.9221
系数 a3 = 0.04268
R^2 = 0.44953
>>
```

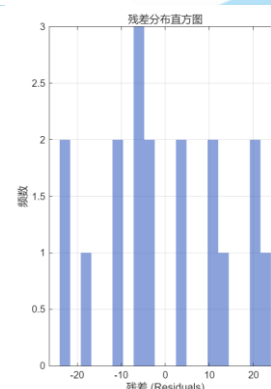


图 7 拉力解算方程拟合结果

接下来只要在瞄准后将 pitch 轴倾角、roll 倾角、距离代入方程即可得到当前所需拉力了。上面演示方程各自变量最高阶次均为一次，若需进一步提升方程精度，可适当引入高阶项，具体的阶次选择和样本容量应根据拟合后的验证结果调整。

3. 功能展示



图 8 获得的 MVP

视频链接:

https://www.bilibili.com/video/BV1JnuD6PEkC/?share_source=copy_web&vd_source=f4315bd7dc06ee5b0f616bc7c7494af2

参考资料

王代东. 基于双目视觉的测距方法研究与应用 [D]. 重庆: 重庆理工大学, 2016.

崔岩. 基于双目立体视觉的距离测量 [D]. 长春: 长春理工大学, 2007.