Robot Detection and Tracking System for RoboMaster Competition

Qiwei Wu, Lihao Xu

July 29, 2024

1 Introduction

RoboMaster University Championship(RMUC) was initiated by **Dajiang Innovation**. As one of the competitions under the National University Robot Competition, it is a robot competition and academic exchange platform specially built for global science and technology enthusiasts. As a member of the computer vision group of our team, we participated in the competition on behalf of Sichuan University(SCU). During the competition, we were defeated by Harbin Engineering University(Figure 1). After the game, we studied our plans and those of our opponents in detail, and sorted out and optimized them.

Figure 1: Game Scene.

As a computer vision group member, my job is to identify the robot of the enemy through the camera and locate the robot of the enemy so that our robot can accurately hit them. In this paper, we will briefly introduce how we identify the position of the enemy robot relative to our robot and how we use the Extended Kalman Filter(EKF) to estimate the position of the enemy robot(Figure 2).

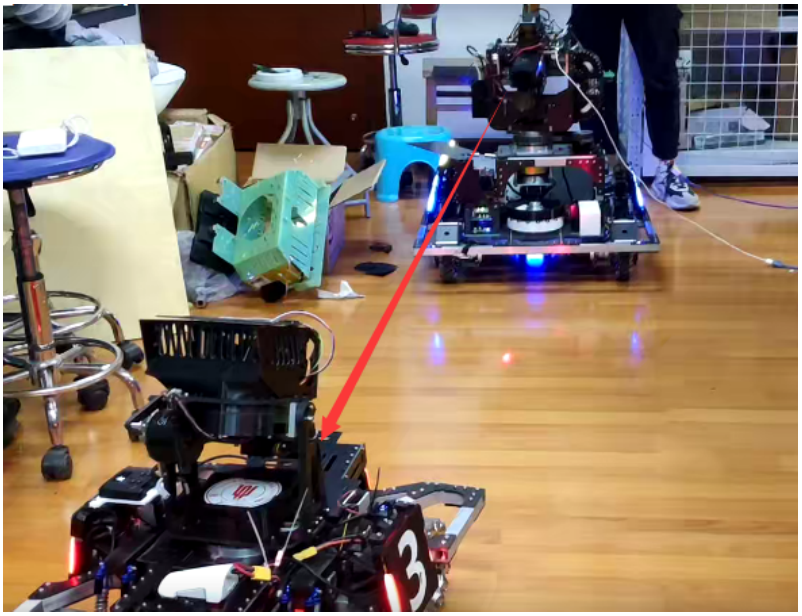

Figure 2: Robots aim at enemy armor.

2 Priori estimation of robot position

2.1 Armor Detection

For the position of the robot, we do not measure it directly, but measure the position of the robot's armor plate. Firstly, we need to identify the enemy's armor. Using object detection algorithms, we can get four feature points of the armor, which represent the four vertices of the armor (Figure 3). The object detection algorithms can be divided into two categories, the deep neural network methods and the feature point extraction methods. The neural network method is often combined with a depth camera to directly measure the position and distance of objects. In this article, we will focus on the deep neural network method, which can accurately identify the four feature points. For the feature point extraction methods, we will introduce the coordinate system transformation algorithm.

Based on the feature point extraction methods, we can get four feature points. However, we can not directly use these points to estimate the position of enemy robots, because these points only represent the relative position of the enemy in the pixel plane under the pixel coordinate system. Therefore, we need to convert the position of the enemy robot in the pixel coordinate system to the position in the world coordinate system through algorithms.

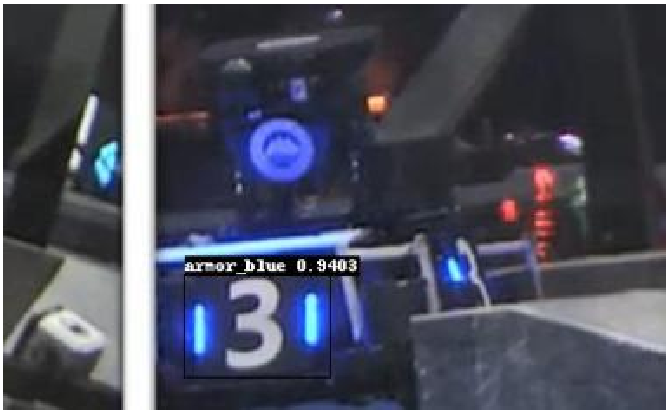

Figure 3: Armor detection result.

2.2 YOLO series algorithm

Target detection algorithms have been widely used in Robomaster competitions, especially YOLO series algorithms have become one of the mainstream algorithms for detecting robot armor. Currently, YOLOX is the most widely used algorithm in YOLO series algorithms. The light version of YOLOX-S has also achieved good results in practical applications. The main improvement of YOLOX-S is to modify the detection head of the head part, introduce the structure of the decoupled head, and use the anchor-free free anchor frame method. In addition, it also adds multiple positive multi-normal and SimOTA optimal transmission methods to improve the screening of pre-selected frames. In the training method, EMA weight update, Cosine learning rate mechanism, and other training techniques are also used. In the part of the loss function, YOLOX uses the IOU loss function to train the Reg branch, and the BCE loss function to train the Cls branch respectively.

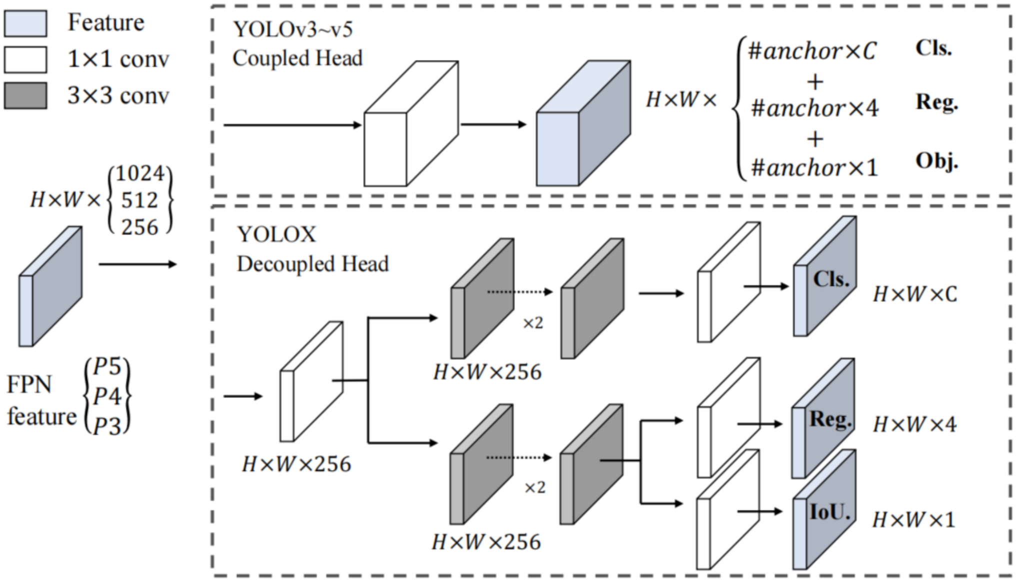

Figure 4: The comparison diagram of YOLOX and YOLO series algorithms

2.3 Coordinate Transformation

Secondly, by solving the pnp problem, we can easily realize the transformation from 2D coordinate system to 3D coordinate system. Assume that the principal point is $(C_x, C_y)$and the feature points in the pixel coordinate system are $(u_i, v_i), i = 1, 2, 3, 4$(Figure 4), if we know the actual size of the target object and the focal length f of the camera, we can get the coordinates $P_i = (X_i, Y_i, Z_i), i = 1, 2, 3, 4$ of the target object in the world coordinate systemX_c, Y_c, Z_c. Indeed, the robot's armor is made according to the competition rules, so the size of the target object can be known. Furthermore, we use a pnp solving method "P3P", which can directly estimate the center of the robot armor by solving the pnp problem of the four feature points.
The coordinates of the target which represents the center of the robot armor in the camera frame are:

$$\boldsymbol{r}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \tag{1}$$

And it also represents the priori estimation of robot position.

3 Position Estimation

The system captures images from the camera fixed to the rotational station, and the on-board miniPC identifies the target armor plate based on the target appearance features using OpenCV, and solves the position of the target in the camera frame by PnP. The position of the target in the rotational station frame can be obtained according

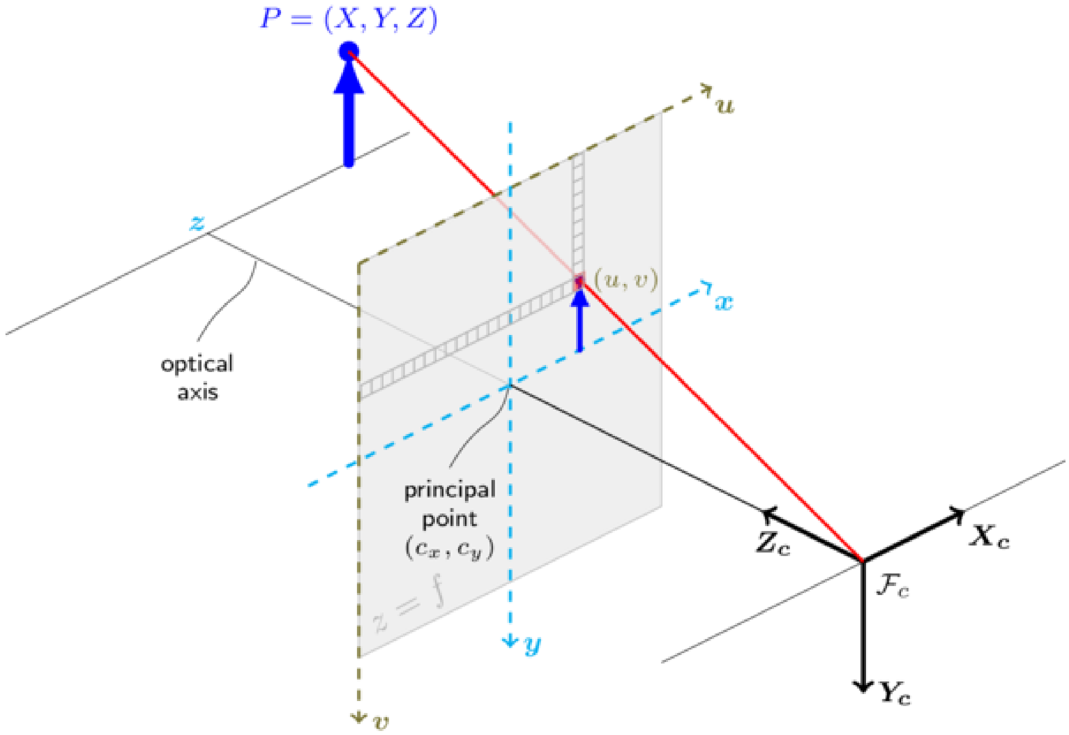

Figure 5: Coordinate system conversion.

to the relative position and pose of the camera and rotational station IMU, and then the position of the target in the inertial frame can be obtained through the coordinate transformation matrix determined by rotational station pose estimation. After the position of the target in the inertial frame is obtained, the moving state of the target in the inertial frame, that is, the position and velocity, is estimated by using the uniform model Kalman filter.(**?**)

So we need to utilize Kalman filter to estimate the pose of the rotational station and the motion state of target.

3.1 rotational station pose estimation

In this article, quaternions are used to describe the carrier pose, and IMU data are fused through Extended Kalman Filter (EKF), that is, accelerometer is used to correct pose and estimate gyro axis bias. In order to reduce the influence of motion acceleration on filtering accuracy, chi-square test is used to eliminate the acceleration measurement when the motion acceleration is too large.

3.1.1 Frame definition

(1) rotational station frame(b-frame):

The three axes of the rotational station frame are fixed to the carrier, and the three axes are respectively parallel to the three axes of the rotational station IMU, which is marked as $OX_bY_bZ_b$.

(2) Camera frame(c-frame):

The camera frame and the rotational station frame are relatively static, and the transformation relationship is determined by the camera position and pose, which is marked as $OX_cY_cZ_c$.

(3) Inertial frame(n-frame):

The direction of each axis of the inertial frame relative to the inertial space remains unchanged, and the coordinate origin moves with the robot, which is marked as $OX_nY_nZ_n$.

3.1.2 Pose description

The pose describes the rotation relationship between the rotation station frame (b-frame) and the inertial frame(n-frame). There are three common descriptive methods, each of which has its advantages and disadvantages. In this chapter, Euler angle and quaternion pose descriptions are given.

(1) Euler angle:

Euler angle is a common and intuitive pose description method with clear geometric meaning, so it is widely used in pose control. Robot pose is that the b-frame is obtained by rotating the n-frame in z-x-y order and rotating angles are ψ, θ, γ , in which the ψ, θ, γ,is yaw, Pitch and Roll respectively.

(2) Quaternion:

Quaternions can be defined as Eq. (2).

$$q = q_0 + q_1 i + q_2 j + q_3 k \quad (q_0, q_1, q_2, q_3 \in \mathbb{R}) \tag{2}$$

where $i^2 = j^2 = k^2 = i$. Quaternion can be regarded as linear combination of basis $\{1, i, j, k\}$.Therefore, quaternions can also be written in vector form:

$$\mathbf{q} = \left[\begin{array}{c} q_0 \\ q_1 \\ q_2 \\ q_3 \end{array} \right] \tag{3}$$

Unit quaternion can be used to describe pose. Coordinate transformation matrix $\boldsymbol{C}_b^n$ from b-frame to c-frame expressed by quaternion:

$$\boldsymbol{C}_b^n = \left[\begin{array}{ccc} 1 - 2\left(q_2^2 + q_3^2\right) & 2\left(q_1 q_2 - q_0 q_3\right) & 2\left(q_1 q_3 + q_0 q_2\right) \\ 2\left(q_1 q_2 + q_0 q_3\right) & 1 - 2\left(q_1^2 + q_3^2\right) & 2\left(q_2 q_3 - q_0 q_1\right) \\ 2\left(q_1 q_3 - q_0 q_2\right) & 2\left(q_2 q_3 + q_0 q_1\right) & 1 - 2\left(q_1^2 + q_2^2\right) \end{array} \right] \tag{4}$$

The differential equation of quaternion with respect to time is:

$$\dot{\mathbf{q}} = \frac{1}{2} \boldsymbol{\Omega} \mathbf{q} \tag{5}$$

where:

$$\boldsymbol{\Omega} = \left[\begin{array}{cccc} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{array} \right] \tag{6}$$

where $\omega_x, \omega_y, \omega_z$ is the angular velocity of the b-frame relative to the c-frame.

3.2 Target motion state estimation

In this section, we will estimate the target motion state.The inertial frame is selected to estimate and predict the target motion state. The Kalman filter is designed by using the uniform linear model to estimate the position and velocity of the target in the inertial frame, and the chi-square test is used to judge whether the target switches.

3.2.1 Coordinate transformation

Suppose that the coordinates of the target in the camera frame(c-frame) calculated by MiniPC are:

$$\boldsymbol{r}_c = \left[\begin{array}{c} x_c \\ y_c \\ z_c \end{array} \right] \tag{7}$$

According to the position of the camera, the coordinates of the target in the rotational station frame(b-frame) can be obtained after $\boldsymbol{r}_c$ rotation and translation:

$$\boldsymbol{r}_b = \left[\begin{array}{c} x_b \\ y_b \\ z_b \end{array} \right] = \boldsymbol{C}_c^b \boldsymbol{r}_c + \boldsymbol{t}_c^b \tag{8}$$

Where $\boldsymbol{C}_c^b$ is the rotation matrix (order ZXY) and $\boldsymbol{t}_c^b$ is the translation vector.

Considering that the time of target coordinates and pose information is not synchronized, the time offset relationship between them should be determined according to their time stamps. As the image acquisition takes 3ms and the target recognition and calculation takes 1−3 ms, it is necessary to find the quaternion corresponding to the target coordinate time from the historical pose information according to the time offset relationship, and use the coordinate transformation matrix $\boldsymbol{C}_b^n$ formed by the quaternion to convert the target coordinate from the rotational station frame (b-frame) to the inertial frame (n-frame):

$$\boldsymbol{r}_n = \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} = \boldsymbol{C}_b^n \boldsymbol{r}_b \tag{9}$$

3.2.2 Target EKF estimation

(1) process model

Kalman filter is designed by using uniform velocity model to estimate the position and velocity of the target in the inertial frame. Set the state:

$$\boldsymbol{x} = \begin{bmatrix} x_n \\ \dot{x}_n \\ y_n \\ \dot{y}_n \\ z_n \\ \dot{z}_n \end{bmatrix} \tag{10}$$

The process model is:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{\Gamma}_k \boldsymbol{w}_k, \quad \boldsymbol{w}_k \sim N\left(\boldsymbol{0}_{3\times1}, \boldsymbol{Q}_k\right) \tag{11}$$

where:

$$\boldsymbol{F}_k = \begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

The process noise variance matrix Q_k is:

$$\boldsymbol{Q}_k = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \tag{13}$$

Wherein, $\sigma_x^2, \sigma_y^2, \sigma_z^2$ is respectively x, y, z three-axis process noise variance.

(2) Measurement model

The measurement model is as follows:

$$\boldsymbol{z}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k \tag{14}$$

If the target inertial coordinate $\boldsymbol{r}_n = [x_n, y_n, z_n]^T$is used as the measurement vector, then:

$$\boldsymbol{H}_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \boldsymbol{v}_k \sim N\left(\boldsymbol{0}_{3\times1}, \boldsymbol{R}_k\right) \tag{15}$$

According to the camera model in Fig. 6:

Define angle $\alpha = \tan^{-1}(x_c/z_c)$,$\beta = \tan^{-1}(y_c/z_c)$, with:

$$\boldsymbol{r}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = g(z_c, \alpha, \beta) = \begin{bmatrix} z_c \tan\alpha \\ z_c \tan\beta \\ z_c \end{bmatrix} \tag{16}$$

Keep that the size of the image in the original measurement $\boldsymbol{z}^r = [z_c, \alpha, \beta]^T$ image plane is determined by the Euclidean distance, but considering the narrow angle of view of the camera, to simplify the model, it can be assumed that z_c is determined by the size of the target image plane, α, β is determined by the coordinates of the image plane of the target center point, the three are independent, the variance is $\sigma_z^2, \sigma_\alpha^2, \sigma_\beta^2$ respectively, and the noise variance matrix $\boldsymbol{R}^r$ is:

$$\boldsymbol{R}^r = \begin{bmatrix} \sigma_z^2 & 0 & 0 \\ 0 & \sigma_\alpha^2 & 0 \\ 0 & 0 & \sigma_\beta^2 \end{bmatrix} \tag{17}$$

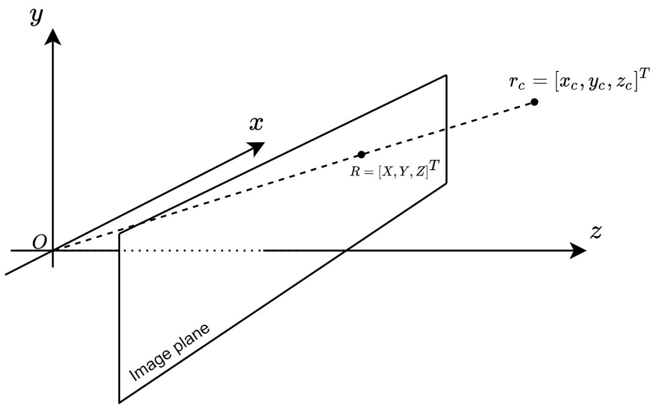

Figure 6: camera model

Linearize $\boldsymbol{r}_c = g(z_c, \alpha, \beta)$ at the working point, and obtain that the variance matrix $\boldsymbol{R}_k^c$ of camera coordinate noise at time k is:

$$\boldsymbol{R}_k^c = \boldsymbol{G}_k \boldsymbol{R}^r \boldsymbol{G}_k^T \tag{18}$$

where:

$$\begin{aligned} \boldsymbol{G}_k &= \frac{\partial g(\boldsymbol{z}_k^r)}{\partial \boldsymbol{z}_k^r} \\ &= \begin{bmatrix} \tan\alpha_k & \frac{z_{c|k}}{\cos^2\alpha_k} & 0 \\ \tan\beta_k & 0 & \frac{z_{c|k}}{\cos^2\beta_k} \\ 1 & 0 & 0 \end{bmatrix} \end{aligned}$$

According to $\boldsymbol{r}_b = \boldsymbol{C}_c^b \boldsymbol{r}_c + \boldsymbol{t}_c^b$, the rotation matrix $\boldsymbol{C}_c^b$ is close to the unit matrix, and the translation vector $\boldsymbol{t}_c^b$ is a definite constant, so the coordinate noise variance matrix $\boldsymbol{R}_k^b$ of the rotational station frame at time k is equal to the coordinate noise variance matrix $\boldsymbol{R}_k^c$ of the camera frame:

$$\boldsymbol{R}_k^b = \boldsymbol{R}_k^c \tag{19}$$

According to $\boldsymbol{r}_n = \boldsymbol{C}_b^n \boldsymbol{r}_b$, the coordinate noise variance matrix $\boldsymbol{R}_k^n$ of inertial frame at time k is:

$$\boldsymbol{R}_k^n = \boldsymbol{C}_{b|k}^n \boldsymbol{R}_k^b {\boldsymbol{C}_{b|k}^n}^T \tag{20}$$

Then the filter measurement noise variance matrix at time k is:

$$\boldsymbol{R}_k = \boldsymbol{C}_{b|k}^n \boldsymbol{G}_k \boldsymbol{R}^r \boldsymbol{G}_k^T {\boldsymbol{C}_{b|k}^n}^T \tag{21}$$

(3) Chi-square test

If the target identified by MiniPC is switched during tracking, the position measurement obtained by Kalman filter will be significantly different from the current position estimation, and a maximum speed estimation will be obtained by directly updating the measurement. To solve this problem, the system judges whether the tracking target has changed by chi square test. Define residual:

$$\boldsymbol{e}_k = \boldsymbol{z}_k - \boldsymbol{H}_k \boldsymbol{x}_k^- \tag{22}$$

If $\boldsymbol{r}_k$ is larger than the threshold value, it indicates that there is a big difference between the location measurement and the location prior estimation, that is, target switching occurs. After target switching, reset the state and its covariance matrix to quickly re converge:

$$\begin{aligned} \boldsymbol{P}_k &= \boldsymbol{P}_0 \\ \hat{\boldsymbol{x}}_k &= \boldsymbol{G} \boldsymbol{z}_k \end{aligned} \tag{23}$$

where:

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

To sum up, the updating process of EKF in motion state estimation is shown in Fig. 7:

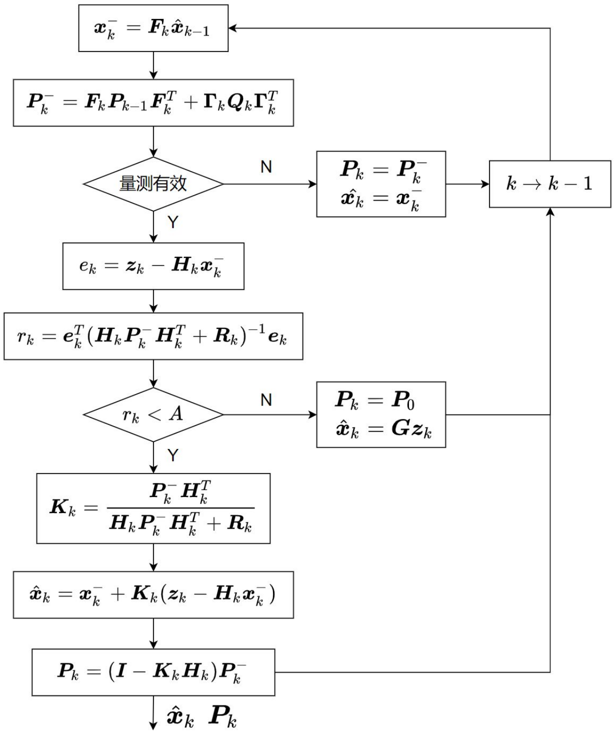
$\boldsymbol{x}_k^- = \boldsymbol{F}_k \hat{\boldsymbol{x}}_{k-1}$
$\boldsymbol{P}_k^- = \boldsymbol{F}_k \boldsymbol{P}_{k-1} \boldsymbol{F}_k^T + \boldsymbol{\Gamma}_k \boldsymbol{Q}_k \boldsymbol{\Gamma}_k^T$
量测有效
N
$\boldsymbol{P}_k = \boldsymbol{P}_k^-$
$\hat{\boldsymbol{x}}_k = \boldsymbol{x}_k^-$
$k \to k-1$
Y
$e_k = \boldsymbol{z}_k - \boldsymbol{H}_k \boldsymbol{x}_k^-$
$r_k = \boldsymbol{e}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k)^{-1} \boldsymbol{e}_k$
$r_k < A$
N
$\boldsymbol{P}_k = \boldsymbol{P}_0$
$\hat{\boldsymbol{x}}_k = \boldsymbol{G} \boldsymbol{z}_k$
Y
$\boldsymbol{K}_k = \dfrac{\boldsymbol{P}_k^- \boldsymbol{H}_k^T}{\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k}$
$\hat{\boldsymbol{x}}_k = \boldsymbol{x}_k^- + \boldsymbol{K}_k (\boldsymbol{z}_k - \boldsymbol{H}_k \boldsymbol{x}_k^-)$
$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_k^-$
$\hat{\boldsymbol{x}}_k \;\; \boldsymbol{P}_k$

Figure 7: flow chart of target estimation

References

[1] Simon D.Kalman filtering with state constraints: a survey of linear and nonlinear algorithms[J].IET Control Theory Applications, 2010, 4(8): 1303-1318.

[2] Kang Guohua, Liu Jianye, Xiong Zhi. Measurement lag asynchronous multi-sensor centralized filtering algorithm in navigation system [J]. Journal of Southeast University (Natural Science Edition), 2005 (05): 60-64

[3] RoboMaster Dream Wing Self-aiming System Design, 2022, https://zhuanlan.zhihu.com/p/416449365