

智能乒乓球拾取机器人的设计与实现

邱高磊, 李红双, 姜春英, 潘五九

(沈阳航空航天大学 机电工程学院, 辽宁 沈阳 110136)

摘要: 针对在乒乓球运动中人工频繁捡球的问题, 设计了一款智能乒乓球拾取机器人。机器人实现功能需要硬件与软件的结合, 在硬件方面主要包含收球系统、存球系统和发球系统这三个系统, 三个系统之间的配合可实现将接触到的球体收集进存球系统, 并在达到容量上限后清空存球系统的功能; 而软件方面实现的功能则主要是规划合适的路径并控制机器人靠近球体, 主要包括方向修正算法、数据融合算法和路径规划算法, 分别基于非线性 PID、卡尔曼滤波和遗传算法实现。为验证机器人的有效性, 设计了实时性测试、全局定位系统误差测试、非线性 PID 模拟实验和实物测试四个实验。结果表明, 在实时性、误差和适应性方面表现优异, 且实物能够高效地完成收球任务。证明了设计方案具有可行性。

关键词: 捡球机器人; 乒乓球; 非线性 PID 算法; 数据融合; 路径规划

中图分类号: TP242 **文献标识码:** **文章编号:**

Design and implementation of an intelligent table tennis ball picking robot

QIU Gaolei, LI Hongshuang, JIANG Chunying, PAN Wujia

(College of Electromechanical Engineering, Shenyang Aerospace University, Shenyang 110136)

Abstract: To address the issue of frequent manual ball picking in table tennis, an intelligent table tennis ball picking robot is designed in this paper. The robot's functionality requires a combination of hardware and software. The hardware mainly includes three systems: a ball collection system, a ball storage system, and a ball serving system. The coordination of these three systems implements the functions of collecting balls into the storage system upon contact and emptying the storage system when it reaches capacity. The software implements the functions of planning suitable paths and controlling the robot to approach the balls. This includes a direction correction algorithm, a data fusion algorithm, and a path planning algorithm, based on nonlinear PID, Kalman filter, and genetic algorithm respectively. To verify the robot's effectiveness, four experiments were designed: real-time test, global positioning system error test, nonlinear PID simulation experiment and physical test. The results show that it performs well in terms of real-time capability, error and adaptability, and the physical prototype can efficiently complete the task of collecting balls. It proves that the design scheme is feasible.

Key words: Ball picking robot; Table tennis; Nonlinear PID algorithm; Data fusion; Path planning.

随着机器人技术的发展, 机器人技术在体育领域的应用层出不穷, 国内外在捡球机器人方面颇有研究。捡球机器人的核心功能为捡拾散落到球场中的球体, 在捡取方式方面可分为两种形式, 即机械臂形式和非机械臂形式, 下面从该两类介绍目前国内外的研究工作:

(1) 机械臂形式: Yeon H S^[1]等设计了自主式

乒乓球收集机器人, 其采用了斑点检测和级联分类器的组合, 实现对球体的检测, 并使用 A* 图搜索算法生成最佳路径, 最后通过吸力马达和一个2自由度的主动喷嘴来收集检测到的球; Nicolaus^[2]等设计了一款搭载两个摄像头的机器人, 分别用于确定球体的位置和机械臂的反馈控制能够实现精准收球; Wei^[3]等设计了一种利用机械臂夹球和流体吸球

功能的乒乓球拾球装置,可实现两种不同拾球机构的切换,应用场景更加广泛,实用性更强;刁跃虎^[4]等设计了一种搭载机械臂和机械手的捡球机器人,通过机械臂和机械手的配合实现对球体的精确捡拾并放到回收箱的功能;周海波^[5]等设计的捡球机器人具有4个机械臂,前后分别配置两个,并设有储球机构,能够在任意方位上高效地拾取周围的球体;周旭等^[6]设计的捡球机器人搭载一个双目摄像头和一个5轴的机械臂,通过双目摄像头确定球体的精确位置,并通过机械臂进行捡取。

(2) 非机械臂形式: Sato^[7]等使用旋转弹性叶片的独特收球机制,并使用深度学习算法检测球体,使机器人能够自主导航到最近的球并将其收集起来;张勇^[8]等通过进球叶轮实现收球,并设计了存球系统,可以临时存储收取的球体,并结合了多种技术,实现建图定位和路径规划,高效完成收球任务; Shan^[9]等设计了机器人,其通过旋转的风扇叶轮将球扫到存储装置,同时机器人还配备了红外传感器,用于实现自动避障,机器人具有较高的效率和稳定性;刘秀杰^[10]等设计的捡球机器人通过单目摄像头指引机器人运动至球体附近,通过收球桶的上下运动将球体收入桶中。

尽管与捡球机器人相关的研究有很多,但大部分应用性并不高,原因主要在于捡球效率存在瓶颈,主要表现在以下两个方面:

(1) 在捡球方式方面,类似于机械臂形式的捡球方式在效率上存在挑战。这主要是因为机械臂的动作需要精确的控制来适应不同的球体位置和环境。机械臂捡球效率与人工捡球相比,优势并不明显。文献[10]中提出的收球桶的方案较为新颖、控制简单,但在桶运动时车体只能静止而不能前往下一个球体,降低了收球效率。文献[7-9]提出的方案在原理上是等效的,本文称之为“风扇叶轮”方案,该方案可理解为旋转的挡板将球体“推入”车体内,该收球方式是高效的,但通常使用该方式时收球机构与存球机构是直接相连的,旋转的挡板将占用较多存球机构的空间,降低了存储容量。

(2) 在运动控制方面,现有的捡球机器人设计方案通常集中于机器人身边的球体,而未充分考虑整个场地内球体的分布情况。这种局限可能导致机器人在执行任务时,未能采用较短的路径,从而增加了不必要的移动时间。同时,通常机器人需要通过摄像头捕捉到的画面信息来决定当前的运动状态,达到使机器人能够运动至目标球体附近的目的,

而机器人在世界中的运动状态与球体在摄像头画面中的运动状态之间是非线性关系,一般需要设计算法来达到上述目的,但现有的文献中却并没有对此处所需的算法进行讨论。

针对以上问题,设计了一款同时具备收球、存球、发球功能的机器人,其中收球功能通过旋转刷式捡球机制实现,原理为高速旋转的胶圈轮与球体接触的过程中,在摩擦力的作用下球体能够加速并运动到存球系统中,而发球系统的功能是在存球系统中球体数量达到容量上限后,将球体发射出去,达到清空存球系统的目的。同时搭载双目相机、IMU、北斗定位模块,用于计算球场中球体相对于机器人的坐标,并基于此,通过遗传算法确定收球顺序,最后使用非线性PID算法控制车体进行收球。文章的主要创新工作如下:

(1) 采用旋转刷式收球机制。该机制解决了机械臂形式的低效率和需要高精度的问题,因为其在收取球体时也可保持运动状态,且收球时球体与收球机构任意一处接触即可实现收球,不要求与特定位置接触。同时也避免了“风扇叶轮”方案的对存球机构占用空间大的问题,因为旋转刷式收球机制的原理是基于滑动摩擦力的,其可对收取到的球体进行加速,使球体沿特定轨道运动至存球系统,实现收球系统和存球系统在空间上的解耦。

(2) 设计了发球系统和与之搭配的“人”字形存球轨道。发球系统与存球轨道的配合能够实现在存球系统剩余容量不足时通过发球系统清空的功能。

(3) 通过非线性PID算法控制车体前往每一个球体附近。该算法通过神经网络实现非线性,使机器人能够适应各种球体和机器人之间的相对位置状态。

(4) 通过遗传算法确定收球顺序。相较于其他大部分文献中只考虑收取最近球体的设计方案,本文设计的机器人将考虑全局信息,在计算球场中所有球体位置信息后,通过遗传算法优化收球顺序,减少时间开销。

1 整体结构设计(Overall structural design)

机器人的任务是自动捡取散落在场地上的球体,并在存球系统达到容量上限时,到指定位置清空存球系统。为了提高工作效率,机器人在执行任务时应最大限度地减少时间消耗。在硬件方面,机器人主要包含三个关键的系统:收球系统、存球系

统和发球系统，下面先简述整机工作流程，再介绍各个系统的设计。

1.1 整机工作流程

机器人整体结构如图1所示，在收球的过程中，收球胶圈轮不断旋转，球体与之接触后会产生较大

的速度，以此通过上升通道，进入存球轨道，在重力的作用下，球体沿着轨道运动至最低端，当存球轨道中球体数量达到上限时，通过发球系统清空存球轨道，然后继续收球。从开机到待机的完整工作流程如图2所示。



图 1 机器人整体结构图

Fig.1 Robot overall structure diagram

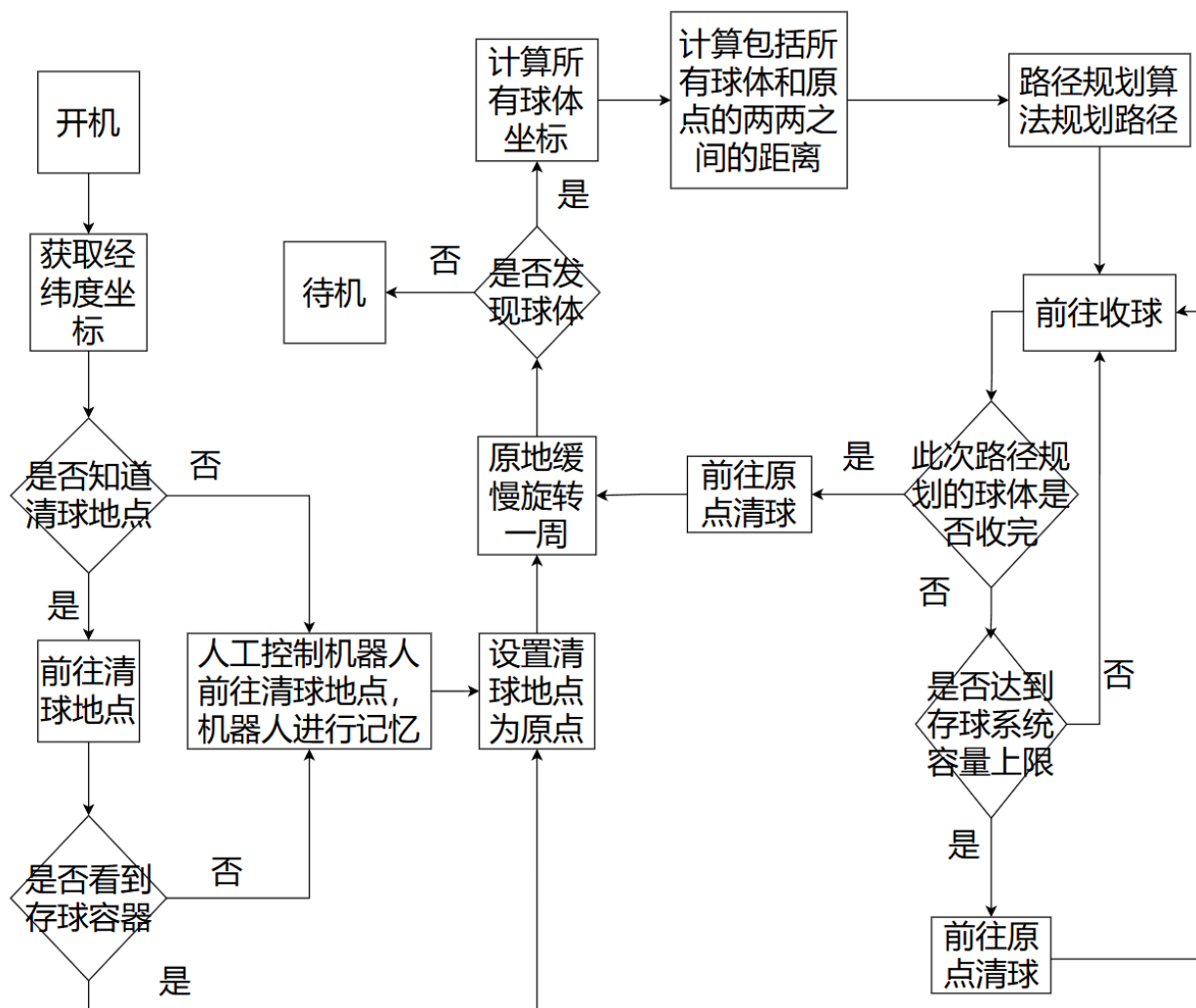
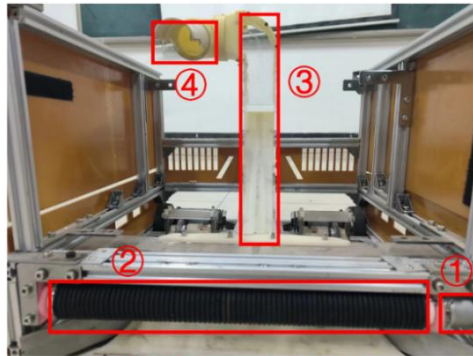


图 2 控制系统工作流程图

Fig.2 Control system workflow diagram

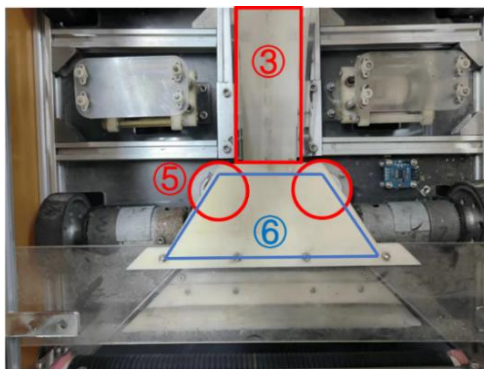
1.2 收球系统设计

收球系统的核心任务是通过机械装置将场地上的球体收集到机器人内部。该系统由收球胶圈轮、上升通道、收球机构出口、升球摩擦轮和挡球板组成，如图3所示。



①收球电机 ②收球胶圈轮
③上升通道 ④收球机构出口

(a) 收球系统正视图



⑤升球摩擦轮 ⑥挡球板

(b) 收球系统俯视图

图3 收球系统结构图

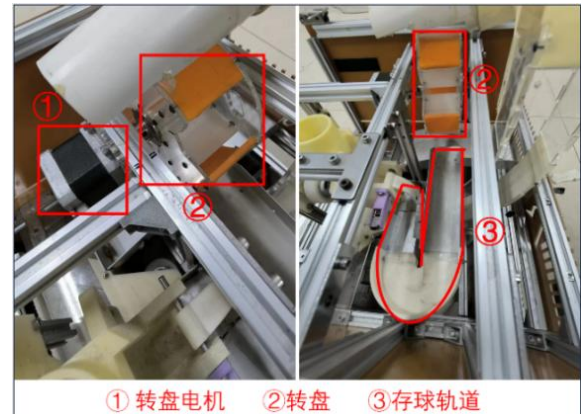
Fig.3 Ball receiving system structure diagram

收球胶圈轮在功能和原理上与摩擦轮相似，即通过旋转使其与物体之间存在相对运动，从而在接触时产生滑动摩擦力，物体受该摩擦力的作用而产生加速度，进而加速运动。收球胶圈轮的作用是通过上述原理将场地上的球体送进通道中，在通道中由升球摩擦轮进行加速。由此可知，收球胶圈轮的速度并不需要很快，选用直流减速电机驱动其转动。考虑到升球摩擦轮与地面有一定距离，故在收球胶圈轮下方安装一个倾斜向上的收球铲。在收球铲的作用下，球体能够和升球摩擦轮在同一高度处，使其能够获得良好的加速效果。为了避免球体在收球胶圈轮的作用下速度过快而直接飞到上升通道中，在收球铲末端和升球摩擦轮之间水平放置了挡球板，限制球体的高度。通过收球胶圈轮、收球铲和

挡球板，球体能够顺利地与升球摩擦轮接触，并获得足够的速度通过上升通道和收球机构出口，最后进入存球轨道。

1.3 存球系统设计

存球系统的任务是临时存储收球系统收取的球体。该系统由转盘、转盘电机和存球轨道组成，如图4所示。



①转盘电机 ②转盘 ③存球轨道

图4 存球系统结构图

Fig.4 Ball storage system structure diagram

转盘设计有若干凹槽，用于接收从收球机构出口滚出的球体。当球体到达转盘上方，如果遇到挡板，球体将暂时停留，起到了缓冲的作用。在机器人工作时，转盘电机始终驱动转盘旋转，在凹槽对准球体时，球体便落入凹槽中。当转盘继续旋转至一定角度后，球体将滚入“人”字形的存球轨道。轨道整体向下倾斜，使得球体在重力作用下沿轨道下滑，直至到达发球系统入口或被前方的球体阻挡。

在存球轨道始端和末端分别放置有激光测距模块，其作用分别是检测轨道中球体数量是否达到容量上限和检测轨道中是否有剩余球体——当轨道无剩余容量后，则执行清空存球系统的功能；当执行完清空存球系统的功能后，则继续收球或待机。

1.4 发球系统设计

当存球系统的容量达到上限后，发球系统需要清空存球系统，使存球系统为后续工作提供足够的容量，结构如图5所示，主要由送球托杆、托杆电机、发球摩擦轮和发球通道组成。当存球系统中的球体数量达到容量上限后，托杆电机将驱动送球托杆不断旋转，带动存球轨道最下端的球体运动，球体沿着发球轨道上升，当接触到高速旋转的升球摩擦轮后，将从发射口发射出去，如图6所示。当存球系统中的球体被清空后，发球系统将停止工作，机器人继续执行捡球任务或待机。

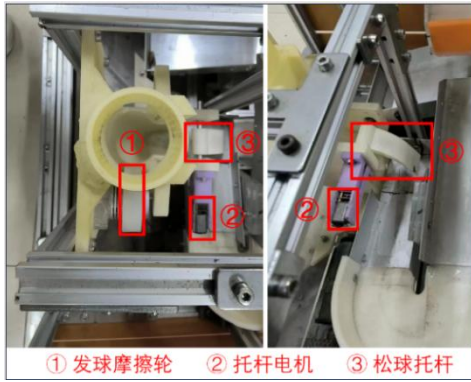


图 5 发球系统结构示意图 (1)

Fig.5 Schematic diagram of the serving system structure (1)

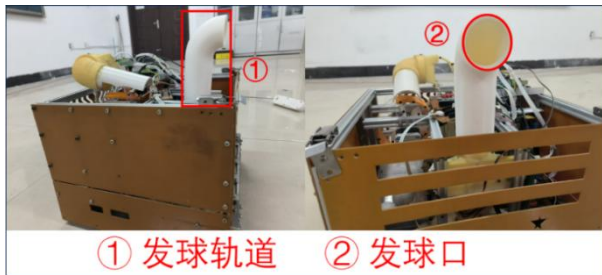


图 6 发球系统结构示意图 (2)

Fig.6 Schematic diagram of the serving system structure (2)

2 算法设计 (Algorithm Design)

本文设计的机器人搭载全局定位系统,其本质就是固结于地面的参考系,以清球地点为原点,以沿地磁场方向和垂直于地磁场方向为坐标轴的正方向,通过该参考系来描述机器人的位姿和球体的绝对坐标,进而计算球体相对于机器人的相对坐标。

在机器人的运动中,方向修正算法发挥着关键作用,它根据摄像头捕获的实时画面信息,或根据全局定位系统中机器人与球体的绝对坐标计算出的球体相对于机器人坐标系的相对坐标,动态调整左右车轮的转速,以确保机器人能够快速修正速度方向,从而提高捡取球体的精度和效率。

此外,为了有效进行全局定位并优化路径规划,机器人需要准确知晓自身的位置和速度。因此,使用数据融合算法,将来自多个传感器的数据集成,以获得更精确的位置和速度信息。同时,机器人需要能够进行合适的路径规划,以减少不必要的时间开销。

2.1 方向修正算法

在机器人收球的过程中,速度方向并不总是指向目标球体的,即并不总是向目标球体方向运动的,但成功收球的前提是机器人能够运动至目标球体

附近。这就要求设计一个方向修正算法,不断调整两侧车轮转速差以修正机器人运动方向,使得机器人能够逐渐地靠近目标球体,提高收球准确率和效率。

修正方向需要知道方位信息,获取方位最直接的方法是通过摄像头捕获的画面计算得到。但在机器人运动过程中,摄像头捕获的画面中并不总是存在球体,因此需要建立全局定位系统间接计算方位。下面建立摄像头画面信息和全局定位系统的数学模型。

2.1.1 摄像头画面信息模型

如图7所示,坐标系固结于机器人,设机器人位于点 O ,其形心相对于地面的速度大小为 v ,角速度大小为 ω_a ,目标地点为点 S ,距离为 $|OS| = L$,与速度方向的夹角为 θ ,假设相机能够捕捉的最大范围为 θ_{max} 。相机画面模型如图8所示,矩形框为画面边界,以画面中心为原点,以一个像素作为单位长度建立坐标系,设点 S 与中轴线(即 Y 轴)的距离为 x ,画面最大距离为 x_{max} 。

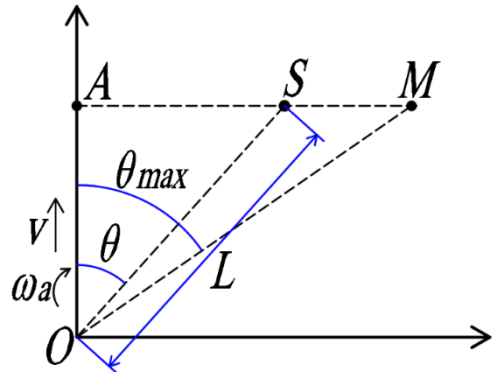


图 7 机器人运动模型

Fig.7 Robot motion model

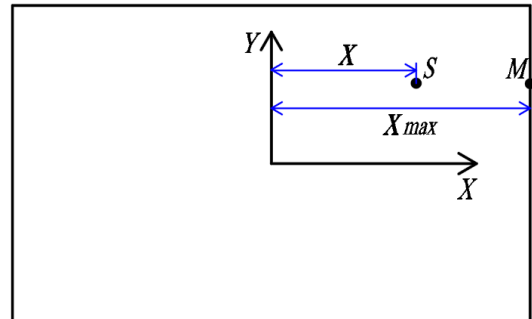


图 8 相机画面模型

Fig.8 Camera screen model

根据图 7 的模型易知

$$\begin{cases} |AS| = |OA|\tan\theta \\ |AM| = |OA|\tan\theta_{max} \end{cases} \quad (1)$$

运动模型和画面模型通常有以下关系:

$$\frac{|AS|}{|AM|} = \frac{x}{x_{\max}} \quad (2)$$

根据式(1)(2), 易知

$$x = \frac{x_{\max}}{\tan \theta_{\max}} \tan \theta \quad (3)$$

等式两边同时对时间求导得

$$\dot{x} = \frac{x_{\max}}{\tan \theta_{\max}} \sec^2 \theta \cdot \dot{\theta} \quad (4)$$

其中 $\dot{\theta}$ 表示点 S 相对于机器人的角速度。设机器人前进时平动速度为正, 顺时针旋转时旋转速度为正, 显然

$$\dot{\theta} = \frac{v \sin \theta}{L} - \omega_a \quad (5)$$

2.1.2 全局定位系统模型

本文所指全局定位系统可理解为只定位而不建图的SLAM(同步定位与建图)。SLAM是一类强大而实用的算法, 在多个领域都是研究和应用^[11-13]。但完整的SLAM对算力要求较高, 且该机器人通常在单一而简单的环境中工作, 建图的意义并非很大, 故该机器人只有定位功能。

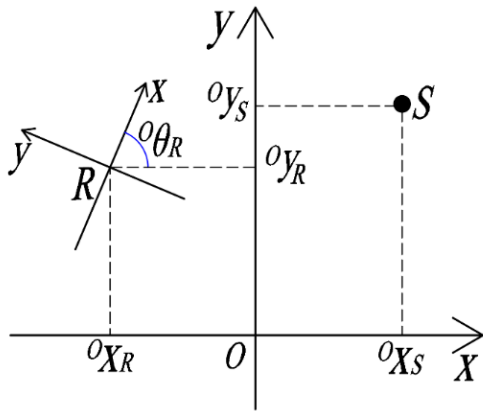


图9 全局定位系统世界坐标系示意图

Fig.9 Schematic diagram of the world coordinate system of the global positioning system

如图9所示, 坐标系 O 为世界坐标系, 坐标系 R 为机器人坐标系, 点 S 为目标点, 各个变量如图9所示, 设从坐标系 O 到坐标系 R 的齐次变换矩阵为 ${}^O_R T$, 目标点 S 在坐标系 O 中的坐标为 ${}^O P_S$, 则点 S 在机器人坐标系中的坐标为

$${}^R P_S = ({}^O_R T)^{-1} \cdot {}^O P_S$$

整理得

$$\begin{bmatrix} {}^R x_S \\ {}^R y_S \end{bmatrix} = A \begin{bmatrix} {}^O x_S & {}^O y_S & {}^O x_R & {}^O y_R \end{bmatrix}^T \quad (6)$$

其中

$$A = \begin{bmatrix} \cos {}^O \theta_R & \sin {}^O \theta_R & -\cos {}^O \theta_R & -\sin {}^O \theta_R \\ -\sin {}^O \theta_R & \cos {}^O \theta_R & \sin {}^O \theta_R & -\cos {}^O \theta_R \end{bmatrix}$$

式(6)能够将点 S 在世界坐标系中的坐标映射到机器人坐标系中, 而能够根据点 S 的坐标计算方位 θ 的公式如式(7)所示

$$\theta = \begin{cases} \theta_1({}^R x_S, {}^R y_S), & {}^R y_S > 0 \\ \theta_2({}^R x_S, {}^R y_S), & {}^R y_S < 0 \text{ and } {}^R x_S \neq 0 \\ \theta_3({}^R x_S, {}^R y_S), & {}^R y_S < 0 \text{ and } {}^R x_S = 0 \\ \theta_4({}^R x_S, {}^R y_S), & {}^R y_S = 0 \end{cases} \quad (7)$$

其中

$$\theta_1(x, y) = \arctan x/y$$

$$\theta_2(x, y) = (\pi + \arctan(x/y)) \text{sign}(x)$$

$$\theta_3(x, y) = \pi$$

$$\theta_4(x, y) = \pi \text{sign}(x)/2$$

在全局定位系统的模型中, 机器人的运动状态对 θ 的作用效果与式(5)同理。

2.1.3 算法设计

由于算法的目的是修正方向, 因此算法的输入理应为球体在机器人坐标系中的方位, 而输出为机器人目标角速度, 最后根据目标角速度计算两侧车轮的目标转速, 值得注意的是, 计算方位和控制电机的实际转速稳定在目标转速并非该算法的目的。根据模型, 方位由式(3)和式(7)计算得到, 而机器人的运动状态对方位和摄像头画面的影响可从式(4)和式(5)中体现出, 显然与硬件相关的参数仅有摄像头的 x_{\max} 和 θ_{\max} , 而这两个参数的误差通常极小。因此, 当计算得到的方位足够精准且算法不控制现实的电机转速而仅计算目标角速度时, 能够避免算法设计与硬件特性的高度耦合, 换言之, 不论实际的电机具有怎样的特性、两侧车轮的间距如何, 等等, 只要电机的实际转速能够使得机器人的实际角速度在目标角速度附近, 就能够实现修正方向使机器人向目标球体运动的功能。因此可以先在仿真环境中确定好算法, 再移植到实体机器人中而无需修改参数。

显然系统是非线性的, 可以考虑非线性PID算法。相较于传统的PID算法, 非线性PID算法可以根据误差, 通过非线性函数来计算比例、积分和微分三者的参数^[14-16], 如式(8)所示

$$u(k) = K_p(e(k)) \cdot e(k) + K_i(e(k)) \cdot T \sum_{i=0}^k e(i) + K_d(e(k)) \cdot \frac{(e(k) - e(k-1))}{T} \quad (8)$$

其中 $K_p(\cdot)$, $K_i(\cdot)$ 和 $K_d(\cdot)$ 均为非线性函数。

神经网络具有强大的拟合能力，而遗传算法具有强大的搜索能力，本文将简单的全连接神经网络作为上述的非线性函数来计算PID参数，并通过遗传算法来优化神经网络参数。定义适应度函数如下

$$F_{\text{fitness}} = \begin{cases} \frac{S}{v_0 t}, & \text{successful} \\ \frac{S_{\text{flag}}}{L} - 1, & \text{unsuccessful} \end{cases} \quad (9)$$

其中 S 为初始距离， t 为机器人实际运行时间， v_0 为机器人基础速度， L 为结束时机器人与目标点的距离，当在一定时间内机器人与目标点的距离小于一定范围时，即 S_{flag} ，则定义为“成功”，否则为“失败”。

全连接神经网络和遗传算法的参数设置如表 1 所示，其中考虑到运动的平滑性，神经网络输入层的其中一个输入为上一次 PID 控制器的输出，同时训练时需要最大的角速度和角加速度进行限制。经过 100 次迭代，最优个体适应度达到0.9904，优化过程如图 10 所示。对该个体进行100次的随机测试（对于单次测试，当误差小于0.5 cm时停止本次测试），平均误差为4.31e-3m，最大的10次误差如表 2 所示。

表 1 非线性 PID 的神经网络和遗传算法的参数设置

Tab.1 Parameter Setting of Neural Network and Genetic Algorithm for Nonlinear PID

项目	内容
每层神经元数量	(2, 4, 5, 3)
种群中个体数	200
迭代次数	100
选择算子	规模为 3 的锦标赛算子
交叉算子	模拟二进制算子
突变算子	高斯突变

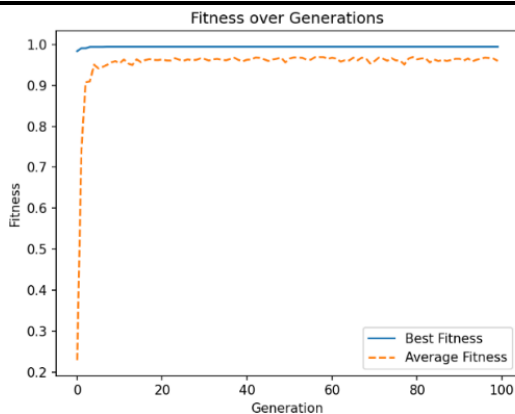


图 10 神经网络优化过程

Fig.10 Neural network optimization process

表 2 最优个体在随机测试中最劣 10 次表现

Tab.2 The worst 10 performances of the best individual in random tests

序号	误差/m	序号	误差/m
1	4.987e-3	6	4.994e-3
2	4.991e-3	7	4.995e-3
3	4.992e-3	8	4.995e-3
4	4.992e-3	9	4.998e-3
5	4.994e-3	10	4.999e-3

2.2 数据融合算法

在全局定位中和方向修正时，需要知道机器人的位置信息和速度信息。如果只依靠单一的传感器，由于误差的存在，得到的数据可信度较低，所以可以考虑融合来自不同传感器的数据，并依据理论公式，计算得到可信度更高的数据。根据文献[17]，若已知地球上两点 A 和 B 的经纬度分别为 (x_1, y_1) 和 (x_2, y_2) ，其中 x 为纬度， y 为经度， R 为地球半径，则两点距离 Δ 满足

$$\Delta = 2R \cdot \arcsin\left(\frac{\sqrt{2m}}{2}\right) \quad (10)$$

其中

$$f(x, y) = \cos(x - y)$$

$$g(x, y) = \cos x \cdot \cos y$$

$$m = 1 - f(x_1, x_2) + g(x_1, x_2) \cdot (1 - f(y_2, y_1))$$

通常机器人的工作范围相对于地球而言非常微小，尽管当机器人在不同纬线上沿纬线运动时，其经度变化量相同时位移不同，但该差别可以忽略不计，因此，如果物体 O 从点 A 运动到点 B ，其在纬线和经线方向上的位移分别近似为

$$\begin{cases} \Delta_{\text{warp}} = 111.199|x_1 - x_2| \\ \Delta_{\text{weft}} = \sqrt{\Delta^2 - \Delta_{\text{warp}}^2} \end{cases} \quad (11)$$

然后可以根据当地的磁偏角计算沿地磁场方向和垂直于地磁场方向的位移分量。速度大小可以通过编码器计算得到，结合IMU的数据，可以知道速度沿地磁场方向和垂直于地磁场方向的分量，同理，也可以将IMU中的加速度投影到沿地磁场方向和垂直于地磁场方向上。

卡尔曼滤波是一种广泛使用的数据融合算法[18-20]，本文使用该算法来融合北斗导航模块和编码器的数据，获取更加可信的位置和速度信息。经典的卡尔曼滤波有如下5个公式：

$$\hat{\mathbf{X}}_k^- = \mathbf{A}\hat{\mathbf{X}}_{k-1} + \mathbf{B}\hat{\mathbf{U}}_{k-1} \quad (12)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \quad (13)$$

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (14)$$

$$\hat{X}_k = \hat{X}_k^- + K_k (Z_k - H \hat{X}_k^-) \quad (15)$$

$$P_k = (I - K_k H) P_k^- \quad (16)$$

显然该系统的状态方程和观测方程分别为

$$\begin{bmatrix} s_k \\ v_k \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_{k-1} \\ v_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ \Delta t \end{bmatrix} a_{k-1} \quad (17)$$

$$\begin{bmatrix} s_k \\ v_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_k \\ v_k \end{bmatrix} \quad (18)$$

则 A, B, H 的值分别为 $\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ \Delta t \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, 式(12)到式(18)中所有参数和变量的意义和取值表3所示

表3 卡尔曼滤波参数表

Tab.3 Kalman filter parameter table

参数或变量	意义	取值
Δt	时间步长	0.005s
a_k	加速度	IMU 读取
s_k	位移	经纬度计算得到
v_k	速度	编码器数据计算得到
A	位置和速度的状态转移矩阵	$\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$
B	控制矩阵	$\begin{bmatrix} \frac{1}{2}(\Delta t)^2 \\ \Delta t \end{bmatrix}$
P_k	先验估计协方差矩阵	式(16)计算得到, 初始值取 $\begin{bmatrix} 1E-3 & 0 \\ 0 & 1E-3 \end{bmatrix}$
P_k^-	后验估计协方差矩阵	式(13)计算得到
Q	过程噪声协方差	$\begin{bmatrix} 1E-3 & 0 \\ 0 & 1E-3 \end{bmatrix}$
\hat{U}_k	控制输入	a_{k-1}
H	观测矩阵	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
Z_k	当前观测值	$\begin{bmatrix} s_k \\ v_k \end{bmatrix}$
\hat{X}_k	当前状态估计	式(15)计算得到
\hat{X}_k^-	预测的状态估计	式(12)计算得到
K_k	卡尔曼增益	式(14)计算得到
R	观测噪声协方差	$\begin{bmatrix} 3 & 0 \\ 0 & 1E-2 \end{bmatrix}$

除了卡尔曼滤波, 诸如无迹卡尔曼滤波和扩展卡尔曼滤波等算法也可以实现数据融合的功能, 但

该系统是线性的, 而其他算法大多用于解决非线性的数据融合问题, 使用卡尔曼滤波足够满足该机器人的需求。

2.3 路径规划

路径规划的目的在于规划一条合适的收球顺序以降低机器人运动路程, 进而降低工作的时间开销, 提高效率。

假设球场上有30个球待捡取, 而机器人每次最多一次性收取10个球, 且每次均从原点出发, 当存球系统达到容量上限后, 回到原点清空系统, 而后继续工作。对于贪心算法, 每次均捡取距离机器人最近的球, 对于其他3个算法, 先将30个球随机分为3组, 每组10个球, 将原点考虑在内, 分别对3组进行路径规划, 最后得到整体的收球顺序。在第 k 轮测试中, 将算法 i 所得的路径规划对应的路程 $s_{k,i}$ 与该轮测试中4个算法中最短路程 $s_{k,\min}$ 的差值 $e_{k,i}$ 定义为“误差”, 将 $e_{k,i}$ 与 $s_{k,i}$ 的比值 $r_{k,i}$ 定义为“误差率”, 如式(19)所示。

$$r_{k,i} = (s_{k,i} - s_{k,\min}) / s_{k,i} \quad (19)$$

经过80轮随机测试, 得到数据如表4所示。

表4 不同路径规划算法性能比较

Tab.4 Performance comparison of different path planning algorithms

算法	平均时间/s	平均路程/m	平均误差率
贪心算法	4.82e-4	191.12	1.22e-1
动态规划	4.89e-2	176.21	5.05e-2
分支限界	387.18	175.12	4.40e-2
回溯算法	18.33	176.24	5.06e-2

理论上, 动态规划、分支限界、回溯算法都能够得到TSP问题的最优解, 所得的路程是一致的, 但由于引入了随机性, 故测试结果略有差别, 但随机性对时间开销几乎没有影响, 根据实验结果, 动态规划是这4个算法中最优秀的。

表5 路径规划遗传算法参数设置

Tab.5 Path planning genetic algorithm parameter setting

项目	内容
种群中个体数	43
迭代次数	500
交叉率	0.75
变异率	0.25
精英保留规模	1
交叉算子	部分交叉匹配
变异算子	逆转变异
选择算子	规模为4的锦标赛算子

下面比较动态规划算法与遗传算法^[21]的性能。测试方法与上述同理，动态规划算法不做变化，而遗传算法的分组通过优化得到（每组依然10个球），参数设置如表5所示，动态规划与遗传算法性能比较如表6所示。

表 6 动态规划和遗传算法性能比较

Tab.6 Performance comparison between dynamic programming and genetic algorithm

算法	平均时间/s	平均路程/m	平均误差率
动态规划	2.28e-2	177.87	0.31
遗传算法	1.06	122.83	0

由表6的实验结果可知，在所有情况中，遗传算法所得到的解均优于动态规划的，且在时间上，仅仅比动态规划慢约1s。在机器人实际工作中，即使以最大速度 v_0 运动1s，其路程也只有0.5m，远小于两个算法平均路程之差55.05m，显然遗传算法是最优选择。

3 实验 (Experiment)

3.1 实时性测试

机器人的控制器之一是STM32F407ZGT6，其主频最高可达168MHz，卡尔曼滤波和非线性PID的神经网络的运算均通过该芯片实现，该实验的目的在于验证该芯片能否胜任这两个算法的计算任务。

通过计算可知，一次神经网络的计算包含42次乘法和12次加法，一次卡尔曼滤波的计算包含76次乘法和47次加法，由于单次计算耗时极短，故定义一百万次运算为一个周期，统计100个周期的数据，图11和图12分别以折线图的形式展示了卡尔曼滤波和神经网络的实验数据，单位为ms，表7统计了最长、平均和最短用时。神经网络和卡尔曼滤波单个周期的平均运行时间分别为1310.18ms和3131.29ms，计算可知单次运算时间平均为1.31μs和3.13μs。神经网络用于间接控制电机，卡尔曼滤波用于计算机器人运动状态，如果每5ms采样一次并计算，显然STM32F407ZGT6是满足对神经网络和卡尔曼滤波的计算需求的。

表 7 神经网络和卡尔曼滤波运行周期的最长时间、平均时间和最短时间

Tab.7 The longest, average and shortest operation cycles of neural networks and Kalman filters

算法	最长时间	平均时间	最短时间
神经网络	1311.34	1310.18	1309.09
卡尔曼滤波	3132.60	3131.29	3130.16

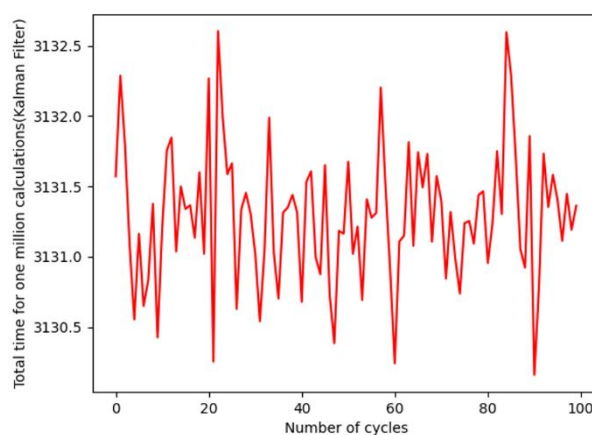


图 11 卡尔曼滤波 100 个周期的计算耗时

Fig.11 The calculation time of 100 cycles of Kalman filter

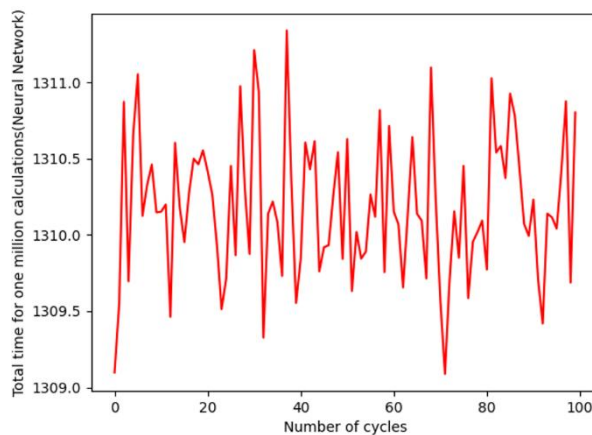


图 12 神经网络 100 个周期的计算耗时

Fig.12 The calculation time of 100 cycles of neural network

3.2 全局定位系统误差测试

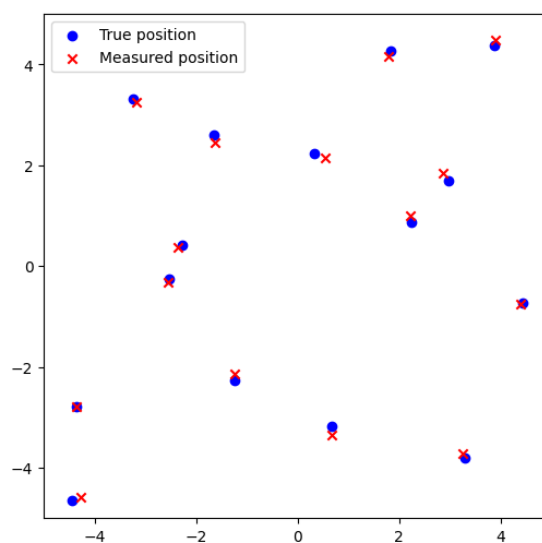


图 13 全局定位系统中球体位置

Fig.13 The positions of the balls in the global positioning system

全局定位系统主要包含机器人和所有球体的位置信息，在机器人工作过程中，只在路径规划以及摄像头无法直接获取球体位置信息时发挥作用，故该实验旨在测试全局定位系统中机器人和球体的位置误差。本实验中所有的“真实位置”均为人工测量，而“测量位置”为全局定位系统计算的位置。

表 8 球体的真实位置和测量位置

Tab.8 The true and measured positions of the balls

序号	真实位置	测量位置	误差
1	(-4.36, -2.78)	(-4.36, -2.78)	4.13e-3
2	(4.43, -0.74)	(4.40, -0.76)	3.88e-2
3	(-2.53, -0.25)	(-2.55, -0.31)	6.76e-2
4	(-3.23, 3.32)	(-3.18, 3.25)	8.31e-2
5	(3.30, -3.80)	(3.26, -3.71)	9.56e-2
6	(-2.26, 0.41)	(-2.35, 0.37)	9.86e-2
7	(3.88, 4.38)	(3.90, 4.48)	1.00e-1
8	(1.83, 4.26)	(1.80, 4.15)	1.10e-1
9	(-1.24, -2.26)	(-1.23, -2.14)	1.23e-1
10	(2.25, 0.86)	(2.23, 1.00)	1.38e-1
11	(-1.65, 2.61)	(-1.63, 2.45)	1.63e-1
12	(0.68, -3.18)	(0.68, -3.34)	1.67e-1
13	(-4.44, -4.65)	(-4.26, -4.57)	1.89e-1
14	(2.98, 1.69)	(2.87, 1.85)	2.04e-1
15	(0.33, 2.23)	(0.54, 2.15)	2.25e-1

球体的“真实位置”和“测量位置”如图13所示，具体数据如表8所示，其中单位均为 m ，表8中的数据按误差大小升序排序。经过计算，平均误差大约为 $12.05cm$ ，最大误差大约为 $22.46cm$ 。球体的位置信息仅在路径规划，以及机器人运动时摄像头无法直接捕获球体画面时发挥作用。在路径规划中， $12.05cm$ 的平均误差相对于球体本身而言较大，但相对于球体与球体之间的距离而言很小，对规划的结果几乎不会产生影响；在方向修正中，通过球体相对于机器人的方位来修改机器人速度方向，在利用全局定位系统计算方位时，方位的最大误差可由式(20)计算得到，

$$\Delta\theta_{\max} = \arcsin \frac{e}{L} \quad (20)$$

方位的最大误差即为在误差一定但“测量位置”相对于“真实位置”的具体位置未知时，所有可能的方位误差中的最大值，其中 e 为误差， L 为机器人和球体的距离。当 L 为 e 的10倍时， $\Delta\theta_{\max}$ 仅为 5.74° 。且显然在拾取到球体之前的某一段时间内，摄像头

一定可以直接获取球体的方位信息，以修正使用全局定位系统时可能产生的较大误差。

在全局定位系统中，除了球体的位置信息外，还有机器人的位置信息。机器人的位置信息通过上文的数据融合算法计算得到，所以通过测试机器人的位置误差，也可以反映数据融合算法的效果。在该实验中，使机器人以固定角速度运动一段圆弧，期间人工测量10个点的“真实位置”，包括起点和终点。“真实位置”和卡尔曼滤波得到的轨迹如图14所示，“真实位置”及其对应的“测量位置”和误差如表9所示。

表 9 机器人的真实位置和测量位置

Tab.9 The true and measured positions of the robot

序号	真实位置	测量位置	误差
1	(0, 0)	(2.93e-4, 8.97e-4)	1.75e-3
2	(7.20e-2, 8.31e-1)	(1.13e-1, 8.13e-1)	4.46e-2
3	(2.72e-1, 1.55)	(3.03e-1, 1.51)	5.56e-2
4	(6.14e-1, 2.20)	(6.21e-1, 2.14)	5.85e-2
5	(1.03, 2.73)	(1.01, 2.66)	7.24e-2
6	(1.51, 3.15)	(1.50, 3.10)	4.95e-2
7	(2.06, 3.60)	(2.05, 3.50)	9.98e-2
8	(2.71, 3.91)	(2.72, 3.86)	5.22e-2
9	(3.42, 4.08)	(3.45, 4.02)	6.92e-2
10	(4.20, 4.14)	(4.24, 4.08)	7.57e-2

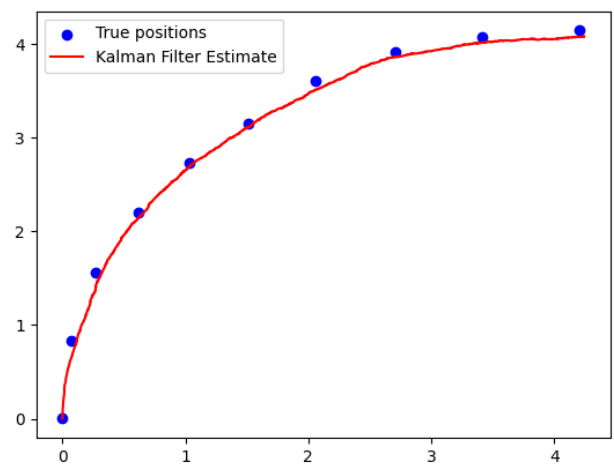


图 14 全局定位系统中机器人位置

Fig.14 The position of the robot in the global positioning system

该测试的平均误差大约为 $5.79cm$ ，最大误差大

约为9.98cm。全局定位系统中的机器人位置只有在通过该系统计算球体相对于机器人的方位时才发挥作用，而该位置误差对方位误差的影响与球体的位置误差对方位误差的影响同理，且同样地，在摄像头能够捕获球体位置信息时，由此产生的误差能够被修正。同时说明了数据融合算法具有较好的效果。

3.3 非线性 PID 模拟实验

该实验的目的在于表现理性情况下机器人的运动状态，并验证机器人运动的平滑性。由于方向修正算法不需要考虑底层的硬件特性，只需考虑模

拟中的平滑性即可。该实验使机器人从原点出发运动至(-2.8, 4.6)处，初始时速度大小为0.5m/s，方向沿Y轴正方向。

如图15所示，从上至下每个子图的含义分别为机器人轨迹、机器人与目标点直线距离 L 随时间的变化、机器人与目标点的连线与机器人速度方向的夹角 θ 随时间的变化、机器人速度大小 v 随时间的变化、机器人角速度 ω 的大小随时间的变化。可知，当 θ 较大时， ω 会不断增大以减小 θ ；而当 θ 较小时， ω 将逐渐减小以尽量避免超调。表明具有不错的适应性。

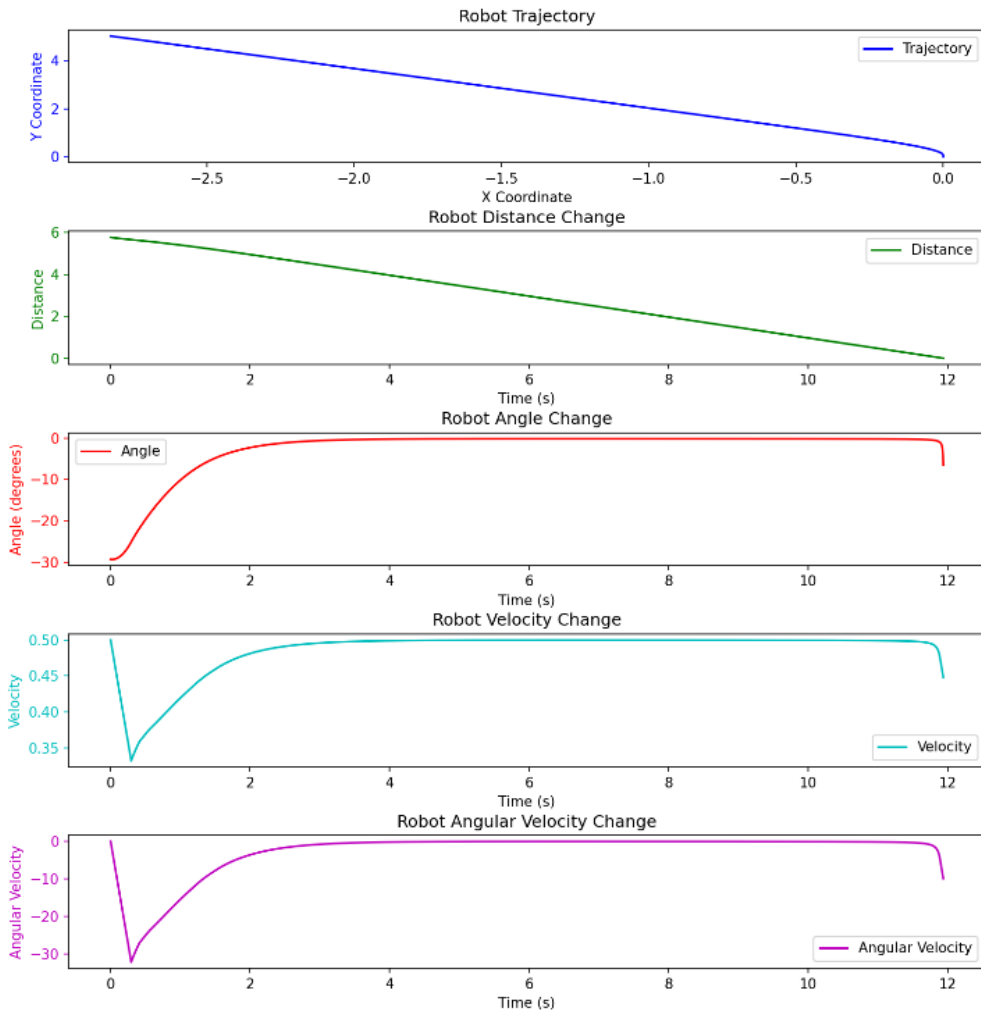


图 15 模拟结果示意图

Fig.15 Schematic diagram of simulation results

3.4 实物测试

作为一个在球场中工作的机器人，首先要能够顺利地完捡球任务，其次再考虑工作效率。定义如下两个评价指标以量化性能：

(1)若某次实验中，计算得到的路径为 p ，其理论路程（即收球顺序不变，而任意两点之间仅考虑

直线距离）为 s_p ，则规定工作时间上限为

$$T_{\text{limit}} = \frac{2s_p}{v_o} \quad (21)$$

若 t 为单次实验中实际运行时间（不考虑清空存球系统所用时间，下文同理）， N 为实验次数，定义 $I_{\text{isSuccess}}$ 为单次实验中是否成功收球的标志，

$S_{\text{SuccessTimes}}$ 为成功收球的次数, $S_{\text{SuccessRate}}$ 为整个实验中, 能够成功收球的次数与总实验次数的比值, 则

$$I_{\text{isSuccess}} = \begin{cases} 1, & t < T_{\text{limit}} \\ 0, & t \geq T_{\text{limit}} \end{cases} \quad (22)$$

$$S_{\text{SuccessTimes}} = \sum_{k=1}^N I_{\text{isSuccess},k} \quad (23)$$

$$S_{\text{SuccessRate}} = \frac{1}{N} S_{\text{SuccessTimes}} \quad (24)$$

(2) 工作效率仅考虑成功收球的情况。若某次实验中, 机器人的速度与时间的关系为 $v = v_k(t)$, 则该次实验的平均速度为

$$\bar{v}_k = \frac{1}{t} \int v_k(t) dt \quad (25)$$

由于仅考虑成功收球的情况, 且理论上在多次独立同分布实验中, 每次实验的权重是一样的, 因此定义整个实验中的平均速度为

$$\bar{v} = \frac{1}{S_{\text{SuccessTimes}}} \sum_{k=1}^N \bar{v}_k \cdot I_{\text{isSuccess},k} \quad (26)$$

定义 $S_{\text{SpeedRate}}$ 为整个实验的平均速度与基础速度的比值, 即

$$S_{\text{SpeedRate}} = \bar{v} / v_0 \quad (27)$$



图 16 实验照片

Fig.16 Experimental photos



图 17 发球系统清空存球系统

Fig.17 Ball serving system clears the ball storage system

如图16所示, 使机器人做10随机实验, 在每次实验中, 由人工抛掷16个球体, 随后让机器人以8个球为一组进行收球。

在每收完8个球之后, 机器人返回原点, 通过发球系统清空存球系统, 如图17所示。

每次实验结束后记录机器人的 $I_{\text{isSuccess}}$ 和 \bar{v}_k , 实验结果如表10所示

表 10 机器人实验结果

Tab.10 Robot Experiment Results

号	$I_{\text{isSuccess}}$	\bar{v}_k	号	$I_{\text{isSuccess}}$	\bar{v}_k
1	1	3.94e-1	6	1	3.90e-1
2	1	4.10e-1	7	1	4.08e-1
3	1	4.04e-1	8	1	4.12e-1
4	1	4.41e-1	9	1	3.02e-1
5	1	4.27e-1	10	1	4.25e-1

根据实验数据, 计算 $S_{\text{SuccessRate}}$ 和 $S_{\text{SpeedRate}}$ 如下所示

$$S_{\text{SuccessRate}} = 1.0$$

$$S_{\text{SpeedRate}} = 0.825$$

在这10次实验中随机选择一次实验的过程数据, 绘制其路径图, 如图18所示。

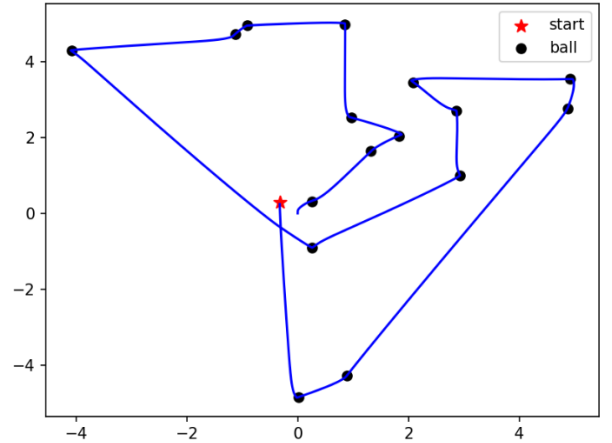


图 18 实验的路径图

Fig.18 Experimental Path Map

以该路径图为例分析机器人的工作效果和算法的有效性。首先, 从路径图中可以看出, 在每一个工作周期内 (即相邻两次位于原点附近的时间间隔), 机器人都是准确地收取8个球体, 并接着运动到原点附近清空存球系统, 直接说明了收球系统、存球系统和发球系统这些主要由机械结构组成的系统都能够正常工作; 其次, 机器人能够准确地收取球体并返回原点附近, 说明了基于卡尔曼滤波的

数据融合算法、基于神经网络的非线性PID算法以及方向修正的数学模型在实际工作中是有效的。

4 结论 (Conclusion)

针对现在人工捡拾乒乓球效率较低的问题,设计了一款智能乒乓球拾取机器人。在系统整体设计方面,设计了收球系统、存球系统和发球系统提出了旋转刷式收球机制,并设计了存球装置,且当容量达到上限后通过发球系统清空存球装置。这种工作方式不需要对机器人的位置进行非常精确地控制,相较于机械臂式捡球方式,提高了捡球效率,且具备存球和清空存球系统的功能,为机器人持续工作提供条件。为了机器人能够顺利工作,设计了方向修正算法、数据融合算法以及路径规划算法。首先建立了机器人方向修正的模型,通过基于全连接神经网络的非线性PID算法实现运动控制;接着,通过各种传感器测得的数据,利用卡尔曼滤波计算机器人位置和速度;最后对比5个路径规划算法的性能,选择效果最好的遗传算法作为实际的路径规划算法。

进行了4个实验,分别为实时性测试、全局定位系统误差测试、非线性PID模拟实验和实物测试,分别表明机器人具有很高的实时性、全局定位系统误差较低且由此产生的方位误差很小、理论上机器人的运动是平滑的且能够根据方位调整角速度、实物能够以较高的效率完成收球任务。

尽管如此,机器人有很大改进空间:

(1) 使用更高效的路径规划算法,使其能够以更大概率求解出更短的路径;

(2) 优化存球系统和发球系统的设计方案,扩大系统容量,并使其能够以更简便的方式清空存球系统。

(3) 设计更合适的算法以降低全局定位系统的误差。

参考文献 (References)

- [1] S. H. Yeon, D. Kim, G. Ryou, and Y. Sim. [System design for autonomous table tennis ball collecting robot](#)[C]//2017 17th International Conference on Control, Automation and Systems (ICCAS). Jeju, Korea (South): IEEE, 2017: 909-914. doi: 10.23919/ICCAS.2017.8204354.
- [2] Nicolaus K, Hooper J, Wood R, Ham C. [Development of an Autonomous Ball-Picking Robot](#)[C]//2016 International Conference on Collaboration Technologies and Systems (CTS). Orlando, FL, USA: IEEE, 2016: 373-378.
- [3] Wei M, Zhao C, Wang C, Chen Z. [Control Structure Design and Simulation Analysis of Table Tennis Picking Robot](#)[C]//2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT). Jilin, China: IEEE, 2023: 1558-1562.
- [4] 刁跃虎,吴涛,杨刚,等. [一种机械臂结构及其构建的捡球机器人](#)[P].云南省:CN201811073534.1,2024-03-29.
Diao Yuehu, Wu Tao, Yang Gang, et al. A Mechanical Arm Structure and Its Constructed Ball Collecting Robot [P]. Yunnan Province: CN201811073534.1, 2024-03-29.
- [5] 周海波,杨璐,王收军,等. [一种轮式多机械手智能捡球机器人](#)[P].天津市:CN201710244284.2,2024-02-20.
ZHOU Haibo, YANG Lu, WANG Shoujun, et al. A wheeled multi-manipulator intelligent ball picking robot[P].Tianjin City: CN201710244284.2,2024-02-20.
- [6] 周旭,周海波,陈睿,等. [基于双目视觉的机械手捡球机器人设计](#)[J].机床与液压,2020,48(03):34-40+79.
ZHOU Xu, ZHOU Haibo, CHEN Rui, ZHANG Zhaoqiang, ZHANG Zhongdang. Design of Pick-up Ball Robot Based on Binocular Vision[J].MACHINE TOOL & HYDRAULICS,2020,48(03):34-40+79.
- [7] T. Sato, K. Irie, S. Takaishi and H. Kawamoto. [Design and System Integration of Table Tennis Ball-Collecting Robot](#)[J]. 2024 IEEE/SICE International Symposium on System Integration (SII), Ha Long, Vietnam, 2024, pp. 1357-1362.
- [8] 张勇,周飞翔,王炜,等. [网球捡球机器人](#)[P].江苏省:CN202410614100.7,2024-07-26.
Zhang Yong, Zhou Feixiang, Wang Wei, et al. Tennis Ball Collecting Robot [P]. Jiangsu Province: CN202410614100.7, 2024-07-26.
- [9] Shan Fan, Zeping Chen, Changhe An, and Chaoying Meng. [Structure design of table tennis intelligent picking robot](#) [C]//Fourth International Conference on Mechanical Engineering, Intelligent Manufacturing, and Automation Technology (MEMAT 2023). Proc. SPIE 13082, 2024: 130821V. doi: 10.1117/12.3026612.
- [10] 刘秀杰,祝长生,王安航,等. [乒乓球捡取机器人的设计与实现](#)[J].机械设计与制造,2022,(10):251-256.
LIU Xiujie, ZHU Changsheng, WANG Anhang, LI Shiyu. Design and Implementation of Table Tennis Picking Robot[J].Machinery Design & Manufacture, 2022, (10): 251-256
- [11] 黄泽霞,邵春莉. [深度学习下的视觉 SLAM 综述](#)[J].机器

- 人,2023,45(06):756-768.
- YAO Erliang, SONG Haitao, ZHAO Jing, et al. LiDAR-inertial SLAM Algorithm Based on Point Cloud Structure and Appearance[J]. Robot, 2023, 45(06):756-768.
- [12] 路春晓, 钟焕, 刘威, 等. [复杂地形环境下的多传感器融合 SLAM 技术](#)[J]. 机器人, 2024, 46(04): 425-435.
- LU Chunxiao, ZHONG Huan, LIU Wei, et al. Multi-sensor Fusion SLAM in Complex Terrain Environments[J]. Robot, 2024, 46(04): 425-435.
- [13] 姚二亮, 宋海涛, 赵婧, 等. [考虑点云结构和表观信息的激光雷达-惯性 SLAM 算法](#)[J]. 机器人, 2024, 46(04): 436-449.
- YAO Erliang, SONG Haitao, ZHAO Jing, et al. LiDAR-inertial SLAM Algorithm Based on Point Cloud Structure and Appearance[J]. Robot, 2024, 46(04): 436-449.
- [14] 程苗苗, 翟朋辉, 张英杰, 等. [基于自学习非线性 PID 的音圈电机精密定位系统](#)[J]. 电工技术学报, 2023, 38(06):1519-1530.
- CHENG Miaomiao, ZHAI Penghui, ZHANG Yingjie, et al. A Voice Coil Motor-Driven Precision Positioning System Based on Self-Learning Nonlinear PID[J]. Transactions of China Electrotechnical Society, 2023, 38(06):1519-1530.
- [15] 苏杰, 曾喆昭. [非线性时变系统的自耦 PID 控制方法](#)[J]. 控制理论与应用, 2022, 39(02):299-306.
- SU Jie, ZENG Zhe-zhao. Auto-coupling PID control method for nonlinear time varying systems[J]. Control Theory & Applications, 2022, 39(02):299-306.
- [16] 曾喆昭, 刘文珏. [自耦 PID 控制器](#)[J]. 自动化学报, 2021, 47(02):404-422.
- ZENG ZheZhao, LIU WenJue. Self-coupling PID Controllers[J]. Acta Automatica Sinica, 2021, 47(02):404-422.
- [17] 卫宇. [考虑地球曲率情况下两点距离问题的求解](#)[J]. 航空兵器, 2008, (03):7-12.
- WEI Yu. Solution of Distance Between Two Points Considering Curvature of the Earth[J]. Aero Weaponry, 2008, (03):7-12.
- [18] 蒋帅, 陈铖, 段砚州, 等. [温度和老化影响下的锂离子电池荷电状态和健康状态联合估计方法研究](#)[J/OL]. 机械工程学报, 1-10[2024-09-06].
- JIANG Shuai, CHEN Cheng, DUAN Yanzhou, et al. Joint Estimation of State-of-charge and State-of-health for Lithium-ion Batteries under the Influence of Temperature and Aging[J]. Journal of Mechanical Engineering, 1-10[2024-09-06].
- [19] 南子寒, 刘大禹, 董明, 等. [GNSS 拒止下多源自主导航鲁棒滤波方法研究](#)[J/OL]. 航空学报, 1-13[2024-09-06].
- NAN Zihan, LIU Dayu, DONG Ming, et al. Research on Improved Robust Filtering Method for GNSS Denied Multisource Autonomous Navigation[J]. Acta Aeronautica et Astronautica Sinica, 1-13[2024-09-06].
- [20] 石饶桥, 李健, 张彪, 等. [基于卡尔曼滤波的烟气 NO_x 浓度融合测量方法](#)[J]. 机械工程学报, 2017, 53(24):63-69.
- SHI Raoqiao, LI Jian, ZHANG Biao, et al. Fusion Measurement Method for NO_x Concentration of Flue Gas Based on Kalman Filter[J]. Journal of Mechanical Engineering, 2017, 53(24):63-69.
- [21] Bladimir Toaza, Domokos Esztergár-Kiss. [A review of metaheuristic algorithms for solving TSP-based scheduling optimization problems](#)[J]. Applied Soft Computing, 2023, 148: 110908.

作者简介:

邱高磊 (2003--), 男, 本科生。研究领域: 机械创新设计。

李红双 (1980--), 男, 博士, 副教授。研究领域: 机械创新设计与控制。