

移动机器人自主导航系统及上位机软件设计与实现

崔奇, 夏浩, 滕游, 刘安东

(浙江工业大学 信息工程学院, 杭州 310023)

摘要: 针对移动机器人自主导航系统, 采用 C++ 语言设计了一款基于 Qt 的跨平台实时数据可视化上位机软件; 该软件执行 SLAM 技术和路径规划算法, 实现可视化移动机器人建图与导航过程以及实时读取数据参数等功能; 首先介绍移动机器人的硬件结构和功能; 其次给出了自主导航所运用到的改进 RRT* 算法和动态窗口法; 在详细叙述上位机软件工作流程的基础上, 开发和设计了实时话题显示、读取以及界面可视化等功能; 最后基于 ROS 系统完成移动机器人自主导航功能, 并通过实时地图与数据可视化来验证所设计上位机软件功能的有效性。

关键词: 移动机器人; 自主导航; Qt; 上位机; 数据可视化

Design and Implementation of Host Computer Software for Mobile Robot Autonomous Navigation Systems

CUI Qi, XIA Hao, TENG You, LIU Andong

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: In this paper, a cross-platform real-time data visualization software is designed, which based on Qt for autonomous navigation systems of mobile robot by using C++ language. The software implements SLAM technology and path planning algorithm to realize visualization of mapping navigation process of mobile robot and real-time data parameter reading. Firstly, we introduce the hardware structure and function of mobile robot. Secondly, give the improved RRT* algorithm and dynamic window method used in autonomous navigation. Based on the detailed description of the work flow of the host software, the functions of real-time topic display, reading and interface visualization are developed. Finally, the mobile robot autonomous navigation function is completed under ROS system, by the real-time map and data visualization, the function validity of the designed host software is verified.

Keywords: mobile robot; autonomous navigation; Qt; host software; data visualization

0 引言

移动机器人是一种在复杂环境下工作, 具有自行感知、自行规划、自我决策功能的智能机器人。随着机器人与传感器技术的高速发展, 移动机器人被广泛应用于各种各样的领域, 如智能仓储机器人^[1]、清扫服务机器人^[2]、自主水下机器人^[3]等。

上述各类机器人的功能实现需要机器人具有自主导航的能力, 而实现自主导航则需要 SLAM 与路径

规划技术。文献 [4] 利用改进 A* 的路径规划算法在机器人平台进行了实验仿真, 结果表明该算法可提高仓储系统的运行效率。文献 [5] 利用高效模板法与动态窗口法进行路径规划, 可以在动态环境中有效提高清扫服务机器人的安全性和工作效率。文献 [6] 利用电子海图的先验知识和改进粒子群优化算法对机器人进行全局路径规划, 在复杂海域航行时, 可以缩短航程, 提高安全性及工作效率。文献 [7] 利用改

收稿日期:2021-06-01; 修回日期:2021-07-20。

基金项目:国家自然科学基金项目(61973275);浙江省省属高校基本科研业务费(RF-A2020004)。

作者简介:崔奇(1997-),男,安徽亳州人,硕士生,主要从事移动机器人路径规划方向的研究。

引用格式:崔奇,夏浩,滕游,等.移动机器人自主导航系统及上位机软件设计与实现[J].计算机测量与控制,2022,30(1):141-146.

进 RRT-Connect 算法删除冗余节点来简化路径, 通过添加障碍物周围的模型和定义安全距离, 提高家庭服务机器人工作的安全性。

在自主导航系统中, SLAM 技术^[8-9]是通过激光雷达传感器, 获取周围环境的信息, 实现即时定位与地图构建的过程。相比于视觉传感器, 激光雷达传感器不受环境光照的影响, 对障碍物识别准确度与可靠性高。路径规划技术^[10-11]则是在 SLAM 建立完整地图基础上, 按照某种评价指标寻找一条从起始点到目标点的最优无碰撞安全路径。然而由于自主导航系统的定位、建图与导航等一系列操作较复杂, 且移动机器人地图、数据信息和运行轨迹无法直观化, 为此需要设计上位机将定位、建图与导航等功能集成到上位机操作界面, 同时实时显示移动机器人地图、数据和运行轨迹。文献 [12] 使用目标跟踪算法提取特征点完成视觉跟踪任务, 并设计了上位机软件实时绘制机器人运动轨迹。文献 [13] 设计了车辆勘查机器人上位机软件, 通过无线网络实现机器人远程操作和车辆图像信息实时显示。

本文针对移动机器人自主导航问题, 采用 C++ 语言与图形界面设计了一款基于 Qt 的跨平台实时数据可视化上位机软件。该软件界面控件布局合理, 可以分页显示数据与地图信息, 方便高效地实现远程自主导航, 具备远程通信、实时显示机器人数据信息、发布初始位置与目标位置、可视化地图和机器人运动轨迹等功能, 从而达到数据与地图可视化和自主导航的效果。

1 移动机器人硬件系统与功能简介

如图 1 所示, 本文实验设备是 Turtlebot2 移动机器人。该移动机器人底盘左右两边各有一个差分电机, 驱动机器人前进。前后各有一个辅助轮, 对机器人起到平衡的作用, 保证机器人有较好的稳定性, 防滑性。该机器人的主要硬件组成如下所述:



图 1 TurtleBot2 移动机器人实物图

1) 有刷直流电机: 移动机器人使用型号为 RP385-ST-2060 的有刷直流电机, 额定电压为 12 V, 额定负载为 5 mN·m, 额定负载电流为 750 mA, 电枢电感为 1.51 mH, 有电压源和脉宽调制两种控制方法。有刷直流电机作为移动机器人的动力系统重要组成部分, 控制机器人移动。

2) 3 轴数字陀螺仪: 移动机器人使用的 3 轴数字陀螺仪测量范围 ±250 度/秒, 偏航轴默认设置为 ±20 度/秒至 ±100 度/秒的范围内校准。三轴陀螺仪可以测量 6 个方向数据信息, 最主要的作用是测量角速度, 判别物体当前运动状态。

3) 角位移编码器: 角位移编码器采用的是 52 线/转, 精度为 11.7 ticks/mm, 能够将角位移转换成周期性的电信号, 通过该电信号可以计算出机器人里程计信息。

4) 激光雷达传感器: 移动机器人使用由 HOKUYO 公司推出的 URG-04LX-UG01 激光雷达。测量范围 240°, DC5V 输入, 用于机器人避障和位置识别, 精度与分辨率较高, 低功耗且供电方便。

2 移动机器人自主导航

本文利用移动机器人执行 SLAM 技术和路径规划算法, 进行数据显示与地图可视化来验证上位机功能效果。首先通过 SLAM 技术建立真实环境地图, 再利用改进 RRT* 算法^[14-16]进行全局路径规划以及动态窗口法^[17-18]进行局部路径规划。最后通过上位机界面完成一系列远程操作, 实时读取机器人的线速度、角速度和当前位置信息等, 可以高效、便捷地实现移动机器人自主导航功能。

改进 RRT* 算法将椭圆状态子集采样方法与智能样本插入相结合, 可以减少采样生成的节点数和得到最优路径解所需要的迭代次数, 从而优化路径长度和提高算法收敛速度。改进 RRT* 算法如下所示。

算法 1: 改进 RRT* 算法

- 1 $V_a \leftarrow \{x_s\}; E_a \leftarrow \emptyset; T_a \leftarrow (V_a, E_a)$
- 2 $V_b \leftarrow \{x_g\}; E_b \leftarrow \emptyset; T_b \leftarrow (V_b, E_b)$
- 3 $c_{best} \leftarrow \infty$
- 4 for $i \leftarrow 0$ to N do
- 5 ConnFlag \leftarrow True
- 6 $x_{rand} \leftarrow \text{Sample}(x_s, x_g, c_{best})$
- 7 $\{X_{near}^a, X_{near}^b\} \leftarrow \text{NearNodes}(x_{rand}, T_a, T_b)$

```

8  if  $X_{near}^a = \emptyset$  &&  $X_{near}^b = \emptyset$  then
9   $\{X_{near}^a, X_{near}^b\} \leftarrow \text{NearestNode}(x_{rand}, T_a, T_b)$ 
10  $\text{ConnFlag} \leftarrow \text{False}$ 
11  $\{L_s^a, L_s^b\} \leftarrow \text{SortedList}(x_{rand}, X_{near}^a, X_{near}^b)$ 
12  $\{x_{min}, \text{Flag}, c_{best}\} \leftarrow \text{BestParent}(L_s^a, L_s^b, \text{Conn})$ 
13 if  $\text{Flag}$  then
14  $T_a \leftarrow \text{InsertNodes}(x_{rand}, x_{min}, T_a)$ 
15  $T_a \leftarrow \text{RewireNodes}(x_{rand}, L_s^a, E_a)$ 
16 else
17  $T_b \leftarrow \text{InsertNodes}(x_{rand}, x_{min}, T_b)$ 
18  $T_b \leftarrow \text{RewireNodes}(x_{rand}, L_s^b, E_b)$ 
19 return  $\{T_a, T_b\} = (V_a \cup V_b, E_a \cup E_b)$ 

```

改进 RRT* 算法主要包括以下几个关键步骤:

设置一个连接标志 ConnFlag, 使得两棵树能够尝试互相连接 (算法 1, 第 5 行)。将随机采样点 x_{rand} 限制在椭圆子集采样空间内 (算法 1, 第 6 行)。以 x_{rand} 为圆心, r 为半径的圆形区域内计算两个邻近节点集合 X_{near}^a, X_{near}^b (算法 1, 第 7 行)。分别对两个集合中的元素按照总路径代价升序排序, 将排序结果分别放入 L_s^a, L_s^b 列表中 (算法 1, 第 11 行)。选择距离 x_{rand} 最近的节点 x_{min} 作为 x_{rand} 的父节点 (算法 1, 第 12 行)。将节点 x_{rand} 插入到搜索树中, 需要对最佳搜索树上的节点重新布线 (算法 1, 第 15&18 行)。依次重复上述过程直到搜到一条从起始点到目标点的无碰撞路径。

上述改进 RRT* 算法仅适用于基于环境已知的情况, 若机器人在移动的过程中出现未知障碍物, 则有可能发生碰撞。因此, 需要与动态窗口法相结合, 实现局部动态避障。动态窗口法^[19]主要是在速度空间内采样多组角速度和线速度, 在这些速度下模拟机器人下一时间间隔内的轨迹。在获取多组轨迹后, 根据相应的评价指标, 分别对这些轨迹进行评分, 选取评分最高轨迹, 再利用该轨迹所对应的速度来控制机器人运动。

在局部路径规划中, 动态窗口法通过设计合适的评估函数选取最优的轨迹。确保最优轨迹规避未知障碍物后, 能够回到改进 RRT* 算法规划的全局参考路径上, 朝着目标位置快速运动, 从而完成移动机器人自主导航避障任务。评估函数定义如下:

$$G(v, w) =$$

$$k[\alpha \text{head}(v, w) + \beta \text{dist}(v, w) + \lambda \text{vel}(v, w)] \quad (1)$$

其中: k 表示平滑系数; α, β 与 λ 表示加权系数, $\text{head}(v, w)$ 表示机器人的方位角评价函数, $\text{dist}(v, w)$ 表示当前速度所对应轨迹与障碍物之间的最近距离, $\text{vel}(v, w)$ 表示当前轨迹速度大小的评价函数。

通过上述算法, 移动机器人可实现自主导航功能, 能够安全、高效地到达指定目标点。同时上位机可以获取机器人数据信息与当前话题, 并在地图中实时可视化机器人的运动与轨迹状态。

3 移动机器人上位机软件设计

3.1 Qt 开发环境

Qt 是一款基于 C++ 的跨平台图形用户界面应用程序开发类软件, Qt 不仅支持 PC 和服务器平台, 包括 Windows、Linux、macOS 等, 还支持移动和嵌入式操作系统, 包括 iOS、Embedded Linux、Android 等。跨平台意味着只需要编写一次程序, 在无需改动源码情况下进行编译, 就可以形成在不同平台上运行的版本。这种跨平台功能为程序开发者提供了极大的便利^[20]。Qt 使用独特的信号与槽机制进行对象间通信替代传统的回调函数机制进行对象间信息传递, 降低了 Qt 对象之间的耦合度, 确保参数传递的准确性与唯一性。Qt 有着丰富而又全面的 C++ 图形库, 基础的封装模块主要有 Core 模块、OpenGL 模块、SQL 模块、XML 模块、Network 模块与 Test 模块等, 这些封装模块都支持多线程操作, 极大地提高了 Qt 软件开发大规模复杂工程项目的能力^[21]。

Qt Creator 是由 Nokia 公司推出的一款新的轻量级集成开发环境, 功能强大, 操作界面简洁, 配置环境简单, 利用 Qt 应用程序框架可以快速方便的完成开发任务。Qt Creator 包括项目生成向导、高级 C++ 程序语言代码编辑器、浏览文件以及图形化的 GDB 调试前端等功能。本文上位机软件的设计是基于 Qt5.9.9 版本下开发的, 由于该版本较高, 能够满足项目开发的需要。其操作界面如图 2 所示。

3.2 上位机软件设计过程

本文基于 QMainWindow 作为主窗口类设计的主窗口。主窗口包括主菜单栏、工具栏和状态栏, 操作方便、可扩展性好。在主窗口中加入 QWidget 类, QWidget 是所有用户界面对象的基类, 可以在其中加入各种界面组件, 如 QLabel、QPushButton、QLineEdit 等, 这些界面组件的具体功能都是通过

QWidget 类得以实现。同时, 使用了 QTabWidget 组件进行分页处理, 实现不同功能界面的切换, 上位机软件的主界面如图 3 所示。



图 2 Qt Creator 操作界面示意图



图 3 上位机软件主界面

该上位机软件主要由 2 个主界面组成: 主从机的登录通信模块以及机器人当前话题列表和运动信息显示界面、可视化界面以及机器人启动设置和控制模块显示界面。上位机工作流程如图 5 所示, 利用全局规划与局部规划算法完成自主导航功能。上位机软件所实现的主要功能如下:

1) 建立主从机之间的通信: 将两台电脑连接在相同的局域网下, 通过上位机分别获取主从机的 IP 地址以及 URL 地址, 再通过 SSH 协议让从机登录主机并且控制主机, 从而完成登录通信过程。

2) 获取机器人当前话题列表及运动信息: 当运行 Launch 文件时, 会产生许多 ROS 话题。为了方便查阅相关话题, 在上位机界面中进行实时显示。此外, 订阅速度与里程计话题, 可以读取当前机器人的线速度、角速度以及当前位置信息。

3) 界面可视化及图层信息获取与发布位置信息: 创建一个 librviz 界面类, 利用 RenderPanel 对象向 ui 界面中添加显示界面, 并且可以实现界面放大、缩小和平移操作。通过 createDisplay 函数在界面中添加图层, 如栅格图层、地图图层、雷达图层等。通过可视化图层信息, 可以实时观察机器人自主导航过程中的运动状态与轨迹。此外, 可以利用上位机发布机器人的初始位置与目标位置, 通过 rviz 控制对象获取 ToolManager 工具对象, 通过工具对象选取导航初始位置与目标位置。可视化机器人导航界面如图 5 所示。

4) 启动设置与实现机器人控制: 为了让机器人自主导航操作更加方便、简洁, 在 QGroupBox 组件中加入启动机器人、激光雷达等按钮。此外, 在启动激光雷达建图后, 通过控制按钮对周围环境进行建图, 不需要利用键盘对机器人控制。保存好地图后, 就可以利用路径规划算法对机器人进行定点导航。

4 实验验证与分析

本文所采用的实验设备为第 1 节中所提及的 Turtlebot2 移动机器人, 该机器人搭载了一台主机和 Hokuyo 激光雷达。另外一台从机运行上位机软件, 远程操作移动机器人。具体实验步骤如下:

1) 机器人硬件系统与主机通过 USB 接口进行串口通信, 从机的上位机软件系统获取主机与本机的 IP 地址及 URL 地址, 进一步与主机建立通信并且获得主机的控制权。同时, 不需要在终端进行输入指

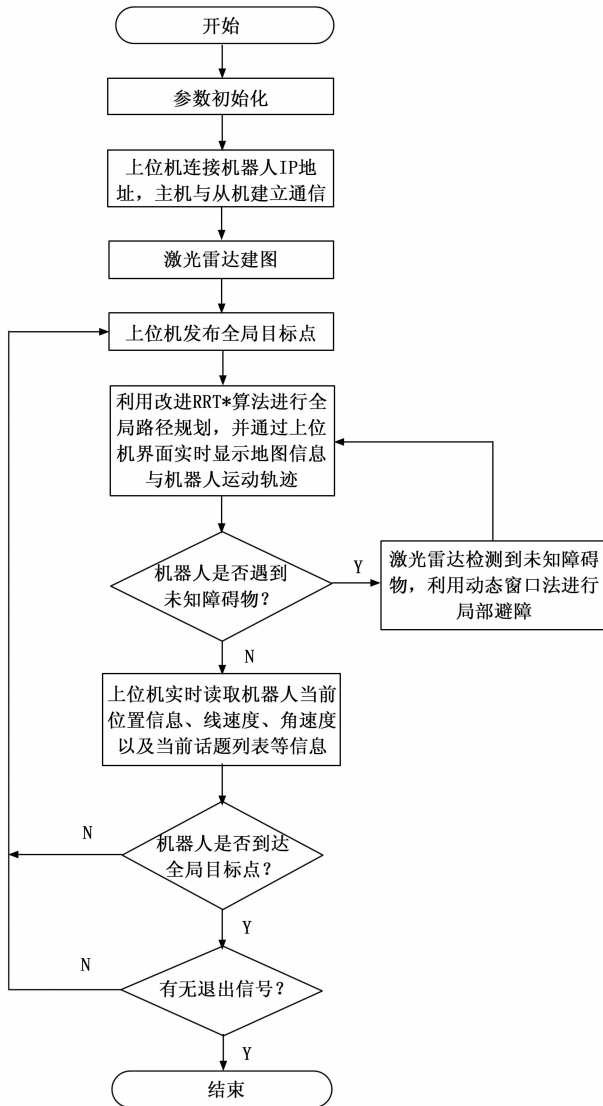


图 4 上位机工作流程图

令, 就可以查看当前机器人的话题列表信息, 并且可以实时获取当前机器人运行的线速度、角速度以及当前位置信息, 如图 6 所示。



图 5 可视化机器人导航界面



图 6 登录通信界面以及机器人信息

2) 根据如图 7 所示的实验环境, 通过点击上位机界面中激光雷达建图按钮, 利用激光雷达和 gmapping 包对真实环境建图。通过界面中的控制按钮移动机器人, 完成建图任务。建立完整的地图如图 8 所示。

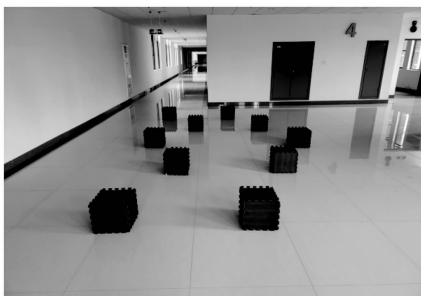


图 7 真实环境

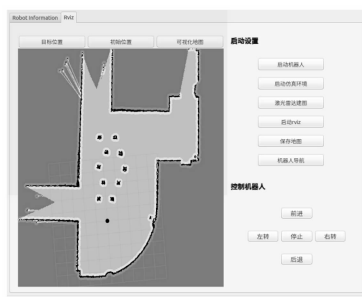


图 8 建立地图

3) 首先通过上位机初始位置按钮在地图中为移动机器人选择初始位置, 再通过目标位置按钮选择导航的全局目标点。此时, 利用改进 RRT* 算法规划出一条全局最优路径, 并在地图中实时记录机器人的运行轨迹, 如图 9 所示。

4) 若移动机器人在运动过程中出现未知障碍物, 如图 10 所示, 灰色箱子是地图中添加的未知障碍物。

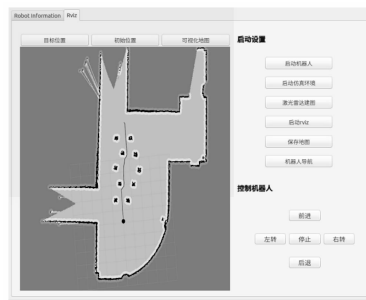


图 9 机器人自主导航

移动机器人可以根据当前位置信息, 结合激光雷达检测到未知障碍物, 利用动态窗口法进行局部路径规划, 在规避未知障碍物后重新回到全局参考路径上, 如图 11 所示。

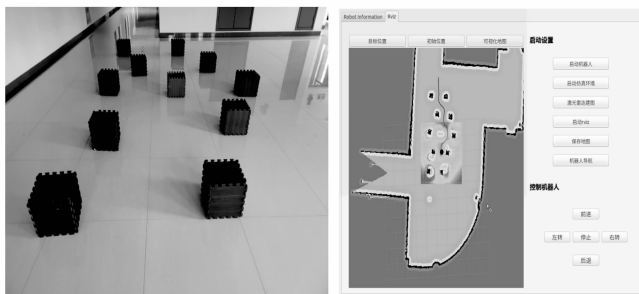


图 10 添加未知障碍物 图 11 动态窗口法局部避障

5) 判断机器人是否到达全局目标点。如果机器人到达指定目标点, 则完成自主导航任务。否则, 如果未到达目标点, 则返回步骤 4) 继续运动。

实验过程中, 移动机器人的实验参数设置如下: 最大线速度为 0.5 m/s, 最大角速度为 1 rad/s, 障碍物的膨胀半径 0.3 m, 机器人半径为 0.2 m, 传感器上传信息频率为 5.0 Hz, 地图更新频率设置为 0.5 Hz。图 9 中栅格地图的分辨率为 20 cm×20 cm, 设坐标原点为左下角弧形顶点, 水平方向为 x 轴, 垂直方向为 y 轴, 起始位置设为 (1.6 m, 3.2 m), 目标位置设为 (1.5 m, 12 m), 机器人实际所走的路线如图中的线所示, 最终到达目标点, 与期望目标点误差 0.12 m。由于地图定位不准确, 里程计计算出的机器人当前位置存在偏差和机器人与地面会存在打滑的可能性, 导致了机器人实际到达目标点与期望的目标点存在误差, 但误差不超过 0.2 m, 可以满足实际需求与应用。

5 结束语

本文在跨平台的 Qt 开发框架下, 以 C++ 面向

对象编程语言,设计和开发了一款界面简洁且功能齐全的机器人自主导航系统的上位机软件。首先介绍了上位机软件涉及的改进 RRT* 算法与动态窗口法,并且详细阐述了上位机软件工作流程、实现功能以及使用的组件。通过上位机与机器人之间建立通信,远程控制机器人实现建图与导航功能,并且上位机界面实时可视化机器人地图、速度、里程计以及轨迹等信息。最后,通过真实环境进行实验,在线实时显示机器人各状态信息和可视化地图等,验证了改进 RRT* 算法与动态窗口法的可行性与实用性、上位机软件功能的实时性与有效性。

参考文献:

- [1] PENG Y H, ZHANG H, HU T Y, et al. Design and realization of car-like mobile robot based on automated warehousing [C] // 2020 Chinese Automation Congress (CAC). Shanghai, China; IEEE, 2020: 2659 - 2664.
- [2] DUBEY S, CHINNAIAH M C, KIRAN C S, et. al. An FPGA based service robot for floor cleaning with autonomous navigation [C] // 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS), Bangalore, India, IEEE, 2016: 1 - 6.
- [3] CAI W Y, WU Y, ZHANG M Y. Three-Dimensional obstacle avoidance for autonomous underwater robot [J]. IEEE Sensors Letters, 2020, 4 (11): 1 - 4.
- [4] 李金钊, 张世宣, 周小铃, 等. 智能仓储机器人动态路径规划研究 [J]. 软件工程, 2020, 23 (7): 30 - 32.
- [5] 陈超勇, 熊禾根, 陶永, 等. 基于高效模板法与动态窗口法的服务机器人全覆盖路径规划方法 [J]. 高技术通讯, 2020, 30 (9): 949 - 958.
- [6] 张岳星, 王铁群, 李硕, 等. 基于海图和改进粒子群优化算法的 AUV 全局路径规划 [J]. 机器人, 2020, 42 (1): 120 - 128.
- [7] LI S Y, ZHAO D H, SUN Y Z, et. al. Path planning algorithm based on the improved RRT - Connect for home service robot arms [C] // 2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR). Tokoname, Japan; IEEE, 2021: 403 - 407.
- [8] 危双丰, 庞帆, 刘振彬, 等. 基于激光雷达的同时定位与地图构建方法综述 [J]. 计算机应用研究, 2020, 37 (2): 327 - 332.
- [9] CHAN S H, WU P T, FU L C. Robust 2D indoor localization through laser SLAM and visual SLAM fusion [C] // 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Miyazaki, Japan; IEEE, 2018: 1263 - 1268.
- [10] 宋晓茹, 任怡悦, 高嵩, 等. 移动机器人路径规划综述 [J]. 计算机测量与控制, 2019, 27 (4): 1 - 5, 17.
- [11] EELE A, RICHARDS A. Path-planning with avoidance using nonlinear branch-and-bound optimization [J]. Journal of Guidance Control & Dynamics, 2015, 32 (2): 384 - 394.
- [12] 仇翔, 陈旭, 刘安东, 等. 移动机器人视觉伺服系统上位机软件的设计与实现 [J]. 计算机测量与控制, 2019, 27 (4): 106 - 110.
- [13] 张丰丰, 高文斌. 车辆底盘勘察机器人上位机软件设计与实现 [J]. 现代制造技术与装备, 2020, 56 (7): 56 - 58.
- [14] GAMMELL J D, SRINIVASA S S, BARFOOT T D. Informed RRT*: optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic [C] // 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. Chicago, IL, USA; IEEE, 2014: 2997 - 3004.
- [15] 杨也, 倪建军, 陈一楠, 等. 改进 RRT* 的室内机器人路径规划算法 [J]. 计算机测量与控制, 2020, 28 (1): 241 - 245.
- [16] QURESHI A H, AYAZ Y. Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments [J]. Robotics & Autonomous Systems, 2015, 68: 1 - 11.
- [17] FOX D, BURGAAR W, THRUN S. The dynamic window approach to collision avoidance [J]. IEEE Robotics & Automation Magazine, 2002, 4 (1): 23 - 33.
- [18] 王永雄, 田永永, 李璇, 等. 穿越稠密障碍物的自适应动态窗口法 [J]. 控制与决策, 2019, 34 (5): 34 - 43.
- [19] FOX D, BURGARD W, THRUN S. The dynamic window approach to collision avoidance [J]. IEEE Robotics & Automation Magazine, 2002, 4 (1): 23 - 33.
- [20] 杨晓星. 基于 Qt 的工业机器人示教器交互界面的研究与设计 [D]. 广州: 广东工业大学, 2020.
- [21] 文生平, 陈端平, 刘其信. 基于 Qt 的激光导引移动机器人监控系统设计 [J]. 自动化与仪表, 2016, 31 (8): 6 - 9.