目录

前言 5

1 Autobot机器人的结构设计 6

1.1 底盘结构设计 6

1.1.1 底盘结构综述 6

1.1.2 底盘结构分析 6

1.1.3 底盘结构总结 6

1.2 电机选型 6

1.2.1 电机综述 6

1.2.2 电机选型 6

1.2.3 电机驱动选型 6

1.3 配件设计与选型 6

1.3.1 万向轮 6

1.3.2 减震系统 6

1.3.3 滑轨 6

2 Autobot机器人的硬件设计 7

2.1 硬件结构概述 7

2.2 主要控制系统选型 7

2.2.1 上位机主控选型 7

2.2.2 下位机主控选型 9

2.3 传感器选型 10

2.3.1 惯性测量单元 11

2.3.2 激光雷达 12

2.3.3 深度相机 13

2.3.4 雷达与测距模块 14

2.3.5 电源电路设计 14

2.4 本章小结 14

3 Autobot机器人的软件设计 15

3.1 机器人下位机软件设计 15

3.1.1 基于FreeRTOS操作系统的程序框架设计 15

3.1.2 电机控制程序设计 16

3.1.3 基于DMA的ADC数据读取程序设计 16

3.1.4 基于VCP的模拟串口通讯设计 16

3.1.5 基于rosserial的下位机ROS通讯设计 16

3.2 机器人上位机软件设计 16

3.2.1 基于rosserial的上位机ROS通讯设计 16

3.2.2 机器人实体模型的TF分析 16

3.2.3 基于RIVZ的机器人传感器数据显示 16

3.2.4 基于话题通讯的机器控制程序设计 16

3.3 本章小结 16

4 人机交互系统设计 17

4.1 基于PyQt5的控制软件设计 17

4.1.1 控制界面功能总览 17

4.1.2 基于matplotlib的传感器数据图形化显示设计 17

4.2 基于stm32的网络遥控设计 17

4.3 遥控设计概述 17

4.3.1 OLED的GUI界面设计 17

4.3.2 局域网内stm32与ROS系统的通信系统设计 17

4.4 基于Web的网页控制界面设计 17

4.5 本章小结 17

5 SLAM算法分析与实践 18

5.1 SLAM算法概述 18

5.2 激光SLAM 18

5.2.1 Gmapping 18

5.2.2 Cartographer 18

5.3 视觉SLAM 18

6 人体姿态估计系统设计 19

6.1 人体姿态估计概述 19

6.2 OpenPose、AlphaPose、Adafuse算法对比 19

6.2.1 OpenPose原理与实验 19

6.2.2 AlphaPose原理与实验 19

6.2.3 Adafuse原理与实验 19

6.2.4 对比分析 19

6.3 Autobot机器人多人人体姿态估计实验 19

7 总结与展望 20

参考文献 21

# 前言

# Autobot机器人的结构设计

## 底盘结构设计

自主移动机器人的底盘结构决定了机器人的运动形式，机器人底盘作为移动指令的执行者，直接决定了移动机器人的运动轨迹与控制精度。不同的运动形式适用于不同场景，所以在对机器人的底盘结构设计时要了解不同底盘结构的特点，了解机器人的工作场景，才能保证机器人满足在某一场景下的工作要求。

### 底盘结构分析

1. 阿克曼模型

阿克曼转向模型是现代汽车转向的常用模型。四轮车在转向时，由于车宽的存在会导致左右轮的转向半径不一致。阿克曼原理是指汽车在行驶时，每个车轮的运动轨迹都必须完全符合它的自然运动轨迹。阿克曼原理的提出解决了四轮车转向时的内外轮转向半径差的问题，保证各轮转向半径集中于一点。任孝平等对基于阿克曼原理的轮式移动机器人运动学模型进行了详细的分析与仿真实验。[1]

|  |
| --- |
| 转向中心O |
| 图1-1 基于阿克曼原理的机器人运动模型 |
| Fig.1-1 Robot motion model based on Ackerman principle |

1. 四轮转向线控模型

在阿克曼模型的基础上，改善后轮的相关机构，使之可转动一定的角度，用以改善车辆的转向不足，也能进一步提高车辆的稳定性。其中后轮因后悬架而产生的转向也被称为后轮随动转向。徐晓美等对后轮随动转向技术的不足与对材料的要求做了详细说明[2]；陈栋对后轮随动系统的摆阵现象、半自助控制策略等做了仿真分析，设计了一套独立的后轮智能随动转向机构[3]。

轮式独立电驱动机器人底盘是通过轮毂电机驱动，以电池或混合动力作为能源的机器人结构。使用轮毂电机驱动能极大的减少部分机械传动系统，大大的简化车辆结构。

1. 双轮差速模型
2. 四轮差速模型
3. 全向轮运动模型
4. 轮足式运动模型

### 差速模型分析

### 底盘结构总结

## 电机选型

### 电机选型

### 电机驱动选型

## 配件设计与选型

### 万向轮

### 减震系统

在第1.1.3节确定了机器人的底盘结构，在第1.1.2节中也分析了这种结构的特点。本节主要分析减震系统的结构形式。

韩乐说明了AGV中减震系统的作用，分析并列举了AGV中常见的减震系统类型，如下表[4]：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 减震结构类型 | 空间占用 | 优点 | 缺点 | 适用范围 |
| 铰接摆动式 | 中等 | 小刚度的弹簧能提供更大的地面贴紧力、减震适应性较好 | 存在双向受力差异性 | 大载重、空间充足 |
| 垂直导柱式 | 小 | 占用空间小、结构简单 | 导柱容易因受扭而卡滞，应补充抗扭措施 | 轻中载、空间较小 |
| 剪叉式 | 较大 | 便于与差速转向系统结合 | 占用体积大、减震性差 | 具备减震以及相对车体旋转的差速驱动 |
| 摆动桥式 | 大 | 结构简单、适应性好 | 多轮系布局须多桥组合的结构才能适应不平路面 | 多轮系且高度空间充足 |
| 四边形式 | 中等 | 小刚度的弹簧能提供更大的地面贴紧力、适应性好 | 结构复杂、空间占用大 | 大载重、空间充足 |

本机器人的底盘结构布局空间较小，工作时机器人负载较小，所以机器人的减重系统选择垂直导柱式。垂直导柱式减震系统是驱动轮与安装座固定，连接到装有压力弹簧的导杆导套机构，压力弹簧在垂直方向上为驱动系统提供竖直的反力。在布置导柱位置时应将两导柱布置在车轮宽度中心线上，避免因车轮支撑力与弹簧的反力产生力矩，导致导套与导柱间的卡滞。

### 滑轨

# Autobot机器人的硬件设计

## 硬件结构概述

## 主要控制系统选型

### 上位机主控选型

上位机作为机器人的大脑，承担机器人最复杂的计算任务，如建图、导航、人体姿态估计等。所以对上位机型号的选择尤为重要，它决定能否自如的完成机器人的计算任务。另外，在机器人设计时不仅要考虑当前功能的需要，还要留有一定计算冗余，便于后期增强神经网络算法深度，用于实现更复杂的功能。常用上位机主控有树莓派、JetsonNano、JetsonTX2、miniPC等，下表选择了其中具有代表性的几款产品进行对比：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表2-1 上位机主控对比** | | | | |
| **Tab.2-1 Comparison of host computer controls** | | | | |
|  | Raspberry-Pi-4 | Jetson TX2 | Jetson Nano | Intel-i5-6300HQ miniPC |
| CPU | 1.5GHz四核64位ARM Cortex-A72 CPU | HMP Dual Denver 2/2 MB L2 +Quad ARM® A57/2 MB L2 | 四核 ARM Cortex A57 | 2.3GHz 四核Intel-i5 CPU 最大睿频频率：3.2GHz 缓存：6MB Intel Smart Cache 散热设计功耗（TDP）：45W |
| GPU | Broadcom VeoCore IV@500MHz | NVIA Pascal™，256 CUDA 核心 | 128 Maxwell cores | NviaN16P-GX(GTX960M), VRAM 4G GDDR5 |
| 内存 | 2GB/4GB/8GB 可选 | 8GB 128bit LPDDR4 | 4GB 64bit LPDDR4 | 8GB DDR4-2133 |
| USB | 2 x USB2.0  2 x USB3.0 | 1 x USB2.0  1 x USB3.0 | 1 x USB2.0  4 x USB3.0 | 2 x USB2.0 2 x USB3.0 |
| WiFi | 802.11b/g/n/ac 2.4GHz/5GHz双频 | 802.11 2x2 ac/BT Ready | M.2 Key E扩展WiFi | 802.11bgn |
| 功耗 | 15W | ≤7.5W | 5-10W | 65W(UMA model) 120W (Nvia GPU model) |

如上表所述，树莓派虽功耗较低，但其整体性能有限，对于同时部署SLAM算法与人体姿态估计的神经网络算法来说，略显不足。Jetson系列是NVIA公司推出的嵌入式人工智能超级计算平台，在思科的电视电话会议系统、法拉赫的工厂自动化等都采用了Jetson系列计算平台。JetsonNano的四核A57CPU性能较为一般，与JetsonTX2的Denver核心要差不少。JetsonTX2的CPU相较于Intel Core系列的I5-3XXXU计算性能大致相当，所以相较于i5-6300HQ的miniPC还要差一些。

本人在使用JetsonTX2时，ARM架构下的部分功能包与工具稍有不足。在计算平台使用神经网络进行实验时，会用到较多不同深度学习框架，如Tensorflow、Pytorch等，其对系统环境的依赖有严格的要求，所以我们在实验时会使用Anaconda创建多个不同虚拟环境，用于支撑不同算法的运行。但Anaconda并不支持ARM架构，有着类似功能的miniconda、Archiconda等对程序功能包的支持却不如Anaconda。所以在实际使用中，Intel i5的miniPC更为便利。

所以考虑主控性能与实际使用，miniPC应为首选。但其额外突出的功耗却是十分值得考虑的问题。对于普通的实验平台，整体框架较小，驱动能力与承载能力有限，所以更加青睐高性能低功耗的计算平台。但本机器人的设计为实用平台，而非实验平台，其驱动能力与承载能力足以支撑miniPC的功耗。

综上所述，本机器人的主控型号选择，确定为搭载GTX960M的Intel-i5-6300HQ miniPC。

### 下位机主控选型

下位机主要用于控制电机、读取传感器数据、与上位机通讯等功能。由于SLAM算法中的导航算法大多基于概率模型，加上计算速率与传输速率的影响，仅仅依靠上位机很难实现实时性特别高的功能。尽管机器人设计的目标为低速移动机器人，出于安全考虑，机器人也要有多重安全防护措施。所以上位机SLAM算法的避障处理作为第一重防护，下位机直接读取的测距模块或雷达的距离信息作为第二重防护，为避免电信号受磁场、短路、断路等影响，在机器人机身布置防撞条作为机器人的第三重防护。所以下位机要尽量准确、快速的读取传感器信息，灵活、精准的实现通讯功能。

较为常用的下位机主控有STC51、MSP430、Arduino、STM32、飞思卡尔等。MSP430系列资料较少、飞思卡尔价格较高、STC51与Arduino的性能较STM32相差较多，ST公司推出的STM32CubeMX、CubeMonitor、X-CUBE-AI等为STM32的开发提供了极大便利。所以下位机主控确定在STM32单片机系列。

综合考虑前文说明的下位机主控要求，将主控型号的锁定在主流MCU和高性能MCU之间。下面选择两类中具有代表性的两款同配置单片机进行对比。

|  |  |  |
| --- | --- | --- |
| 表2-1 单片机参数对比 | | |
| Tab.2-1 Comparison of host computer controls | | |
| 单片机型号 | STM32F103VGT6 | STM32F407VGT6 |
| Frequency(MHz) | 72 | 168 |
| Core | Cortex-M3 | Cortex-M4 |
| I2C | 2 | 3 |
| CAN | 1 | 2 |
| USB OTG\_HS/FS | 0 | 1 + 1 |
| Ethernet | 0 | 1 |
| U(S)ART | 3 + 2 | 4 + 2 |

如上表所述，STM32F4系列主控频率更高，处理性能更强。我们常用的OLED、IMU等大多要用到I2C或USART通信，所以从通信接口看，STM32F4系列更胜一筹。STM32与ROS间使用串口通信时，需要借用USB-TTL电路实现，本人在魔方机器人的设计中使用USB-TTL模块通信时，常遇到信号不稳定与信号乱码情况，所以在本机器人的设计中使用VCP虚拟串口实现，这依赖于单片机的USB OTG\_HS/USB OTG\_FS。为使机器人配置更加灵活，机器人上位机与下位机还可使用Ethernet通信，所以选定STM32F4系列单片机。

在下位机的程序设计中，传感器数据读取、OLED显示、通讯、APP等互有交流，但运行互不影响，所以使用嵌入式操作系统将这几个功能分别置于不同线程运行，能更大效率利用单片机性能。使用实时操作系统依赖于较大FLASH。

综上，考虑运行性能、FLASH大小、读取传感器占用引脚等因素，选定LQFP100封装的STM32F407VGT6作为下位机主控。

## 传感器选型

### 惯性测量单元

惯性测量单元，即IMU，是机器人获取当前位姿和运动状态的传感器。惯性测量数据与双目视觉图像进行紧耦合优化，能显著提高系统的鲁棒性[5]；与激光雷达点云数据的融合能有效提高对物体的检测能力[6]。

本人在武术擂台机器人的设计中，发现短时间内使用六轴传感器能保持一定精度，当机器人长时间运行时，其Z轴误差较大。这是因为在六轴传感器中，Z轴角度是由角速度积分计算得到的，存在累计误差，当机器人运行时间过长时，累计误差会产生一定影响。九轴传感器相较于六轴传感器多了三轴的磁场数据，其角度信息与磁强计的磁场数据进行融合，能保持一定的稳定性。但九轴传感器容易受到外部磁场影响，除地理位置的特殊磁场，机器人的电机、电磁手指、扬声器等都会对九轴传感器产生影响，所以在机器人的结构设计时要注意减小机器人其他系统的磁场干扰。

|  |
| --- |
|  |
| 图2-1 惯性测量单-JY901 |
| Fig.2-1 Inertial measurement unit-JY901 |

本机器人的惯性测量单元选取维特智能JY901。JY901内置MPU9250芯片，能读取三轴加速度、三轴角速度、三轴磁场等原始数据，在JY901模块内还内置一个Cortex-M0内核处理器，使用高动态卡尔曼滤波融合算法对MPU9250的原始数据进行预处理，可通过串口或IIC输出更加平滑稳定的数据。JY901还可链接符合NMEA-0183的GPS模块，组成惯性组合导航单元。由于GPS卫星信号穿透能力差，在室内环境中不适用，所以本机器人的设计过程不包含GPS数据的解算，但在电路设计时为GPS数据的读取留出了接口，便于机器人功能的扩展。

### 激光雷达

激光雷达是机器人建图与导航实现的核心传感器，2D的激光雷达能够获取统一平面内的景深信息，得到二维点云图。本人在大学四年的学习过程中共使用过三款激光雷达，分别为：思岚A2、思岚S1和杉川Delta 3A，现将其参数列举如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 表2-1 激光雷达参数对比 | | | |
| Tab.2-1 Comparison of lidar parameters | | | |
|  | 思岚A2 | 思岚S1 | 杉川Delta 3A |
| 测量半径 | 0.2m~16m | 白色物体：40m  黑色物体：10m | 0.25m~16m（反射率80%） |
| 采样频率 | 8000Hz | 9200Hz | 8000Hz |
| 扫描频率 | 5Hz~15Hz可调 | 8~15Hz可调 | 5~15Hz可调 |
| 角度分辨率 | 0.9° | 0.313°~0.587°（取决于扫描频率） | 0.25°~0.7° |
| 测距分辨率 | ≤实际距离的 1%（测距≤12m)  ≤实际距离的 2%（测距12m～16m) | 3cm | 1mm |
| 测量精度 | 实际距离的 1%（≤3 m）  实际距离的 2%（3-5 m）  实际距离的 2.5%（5-16m） | ±5cm | <1% (16m) |

以上激光雷达参数上思岚S1要更胜一筹，在实际使用时思岚S1的综合表现也要比其余两个好一点，但本机器人的设计目的为室内环境下的巡检与协作机器人，在本机器人的实际使用时无法发挥出该款激光雷达的突出特点，如40m测量半径，较高测量精度等。所以综合考虑价格与实用性能选择杉川雷达的Delta-3A。

### 深度相机

视觉SLAM是通过单目摄像头、双目摄像头、鱼眼摄像头、RGBD摄像头等作为传感器，将传感器数据处理后传入SLAM算法。我们常用的单目摄像头可以通过几帧图像间的信息对比得到画面内的深度信息；双目摄像头的原理与我们人眼的测距原理一致，但是不能看到黑暗处的深度信息，属于被动测量；RGBD相机能主动发射红外光，根据反射回的反射光得到深度信息，属于主动测量。所以本机器人选用RGBD相机。

本人共使用过三款不同的RGBD相机，分别为：Kinect V1、Kinect V2和Real Sense D435。

Kinect相机由微软公司生产，其中KinectV1是基于结构光原理，KinectV2是基于TOF原理。后者相较于前者有更宽阔的视场角，生成的深度图质量较高。Kinect的软件工具包里的部分工具性能较强，如人体骨架追踪，能同时进行六个人的骨架追踪。但是在本人的实际使用中，Kinect相机对Ubuntu系统兼容性较差，特别是KinectV2，在Ubuntu操作系统中需要借助OpenNI2和libfreenect2功能包，其功能有一定限制。

Real SenseD435是由英特尔公司生产的深度相机，这款相机在2018年推出，支持Windows、Linux、Mac OS等平台。与Kinect相机相比，Real Sense D435相机性能更强，配置更新。在实际使用中，Real Sense实现的功能也比较丰富，廖萱等人[7]使用RealSense相机实现了基于RGB-D传感器的机器人视觉里程计；周雄才[8]实现了一种基于Real Sense深度信息的障碍物检测和避障方法；W. Ma等人[9]提出了一种基于Real Sense D435的三维物体点云快速重建方法。

综上，本机器人的深度相机使用Real Sense D435深度相机。

### 雷达与测距模块

### 电源电路设计

## 本章小结

# Autobot机器人的软件设计

## 机器人下位机软件设计

### 基于FreeRTOS操作系统的程序框架设计

在第2章内，经过对比与分析，选定了单片机STM32F407VGT6作为下位机主控。stm32单片机的常规程序结构为通过定时器中断或外部中断等修改标志位，主循环内通过标志位的修改执行不同的计算任务。这种程序结构实时性较差，中断的回调函数内容易阻塞，通过人工指定标志位实现的轮询调度系统能在一定程度上缓解程序阻塞，但整体效率有限。

嵌入式实时操作系统是能运行在单片机中的轻量操作系统，一般的嵌入式实时操作系统都具有低功耗、实时性强、低成本等优势。FreeRTOS系统是一款开源免费的实时操作系统，具有较强的实时特性与可靠性，对比自定义的轮询调度系统在效率上有极大的提高。通过FreeRTOS的封装，对任务调度、内存管理、中断管理等程序的执行效率有明显的提高。通过ST公司开发的STM32CubeMX能更便捷的配置基于HAL库的FreeRTOS系统。

在上一节说明了机器人下位机需要实现的功能，根据以上功能，在FreeRTOS系统中定义程序框架图如下：

### 电机控制程序设计

### 基于DMA的ADC数据读取程序设计

### 基于VCP的模拟串口通讯设计

### 基于rosserial的下位机ROS通讯设计

Rosserial是用于非ROS设备与ROS设备进行通讯的一种协议，它为非ROS设备提供了ROS节点和服务的发布/订阅功能，使无法运行ROS系统的嵌入式设备或终端能通过串口或网络与ROS网络中的其他节点进行数据交互。Rosserial分为两部分，客户端和服务端。客户端用于在非ROS环境的终端设备，接收ROS网络中控制节点的信息，上传相关传感器、电机、电池等信息；服务端运行在ROS环境下，负责接收终端设备的数据并发布相关指令。本节主要进行rosserial的客户端实现。

1. 通讯内容设计
2. 消息格式转换

在确定了下位机与上位机间需要通讯的内容并定义了msg文件后，需要将ROS环境中的msg文件转换为适用于stm32中的头文件。

1. Stm32中C/C++混合编译实现通讯

一般的stm32单片机使用C语言编程，转换为hex文件烧录到单片机内，要在stm32单片机内实现ROS节点的创建需要使用C++语言实现。对于编写stm32程序常用的Keil5、Stm32CubeIDE等开发软件进行C/C++的混合编译需要比较复杂的设置。Jetbrains系列软件中的Clion使用cmake构建项目并运行，将其作为stm32的开发软件能较大简化混合编译的流程，软件的详细使用在本文中不再赘述。

## 机器人上位机软件设计

### 基于rosserial的上位机ROS通讯设计

在3.1.5节中说明了rosserial的原理并设计完成了其客户端实现，本节进行rosserial的服务端实现。

### 机器人实体模型的TF分析

### 基于RIVZ的机器人传感器数据显示

### 基于话题通讯的机器控制程序设计

## 本章小结

# 人机交互系统设计

## 基于PyQt5的控制软件设计

### 控制界面功能总览

### 基于matplotlib的传感器数据图形化显示设计

## 基于stm32的网络遥控设计

## 遥控设计概述

### OLED的GUI界面设计

### 局域网内stm32与ROS系统的通信系统设计

## 基于Web的网页控制界面设计

## 本章小结

# SLAM算法分析与实践

## SLAM算法概述

## 激光SLAM

### Gmapping

### Cartographer

## 视觉SLAM

# 人体姿态估计

## 人体姿态估计概述

人体姿态估计是指在一张图片或一段视频中识别出人的关键骨骼信息，判断人体的姿势。识别出的人体姿态信息对机器学习模型进行训练后，可以通过人工智能算法学习出某个人的行走习惯、运动习惯、生活习惯等，进而进行人体动作的预测、辅助人脸识别进行实名认证、更加便捷的人机交互、运动员动作分析等。如###对通过对人体行走动作模型训练、运动员动作分析、人体动作预测的论文。

当下常用的人体姿态估计算法分为自顶向下（top-down）和自底向上（bottom-up）两类。自顶向下的算法是先在图像中检测出所有人，然后使用单人姿态估计的方法对图像中的所有人进行姿态估计。自底向上的算法是先检测出所有人的骨骼点，再将骨骼点连接成图，最后通过图优化的方式剔除错误的骨骼点连接，实现多人姿态估计。通过这两种方式的原理可以看出，自顶向下算法的准确度依赖于图像中目标区域框检测的质量，而在自底向上的算法中，如果两个人距离过近，两个人体信息的区分度就会降低，失去对全局信息的利用。

AlphaPose采用自顶向下的方式。

OpenPose使用自底向上的方式。

Adafuse使用多视图特征融合方式解决遮挡环境对人体姿态估计的影响。

## AlphaPose、OpenPose、Adafuse算法对比

### AlphaPose原理与实验

### OpenPose原理与实验

### Adafuse原理与实验

### 对比分析

## Autobot机器人多人人体姿态估计实验

# 总结与展望

在本文的第六章人体姿态估计中用到的Adafuse算法借助多个相机间视图的补充实现了遮挡环境下对人体姿态估计的优化。在服务型机器人中对遮挡环境中人体姿态的估计格外重要，但是在家庭中部署多个相机配合服务机器人使用不够便利，网络中暴露出多个视频接口也会引发安全问题。但是在工厂车间、商厦、机器人巡检道路等场景中会有较为丰富的监控设施，也会有全面的网络系统，对于图像信息的协同调用也比较方便。所以在工厂车间或商厦内，某一机器人调取附近的监控画面作为辅助画面，机器人中的相机用于关键信息处理，能使人体姿态估计准确率与巡检效率更高。但机器人调用处理多图像时高度依赖于网络带宽与计算能力，较高的计算能力会使巡检机器人有较大的功耗。

# 参考文献

[1] 任孝平和蔡自兴, 《基于阿克曼原理的车式移动机器人运动学建模》, 智能系统学报, 卷 4, 期 06, 页 534–537, 2009.

[2] 徐晓美, 陈宁和.P.Lee H, 《后轮随动转向技术应用研究综述》, 汽车技术, 期 07, 页 1-6+16, 2016.

[3] 陈栋, 《后轮随动转向车辆的半主动控制及智能随动转向系统研究》, 硕士, 南京林业大学, 2019. [在线]. 载于: https://kns.cnki.net/KCMS/detail/detail.aspx?dbcode=CMFD&dbname=CMFD202001&filename=1020802051.nh&v=

[4] 韩乐, 《AGV常见减震浮动结构对比分析》, 中国设备工程, 期 10, 页 130–134, 2020.

[5] 任金伟, 郑鑫, 李昱辰和朱建科, 《基于新型多传感器融合策略的移动端双目视觉惯性SLAM闭环算法研究》, 高技术通讯, 卷 31, 期 07, 页 681–691, 2021.

[6] 张艳国和李擎, 《基于惯性测量单元的激光雷达点云融合方法》, 系统仿真学报, 卷 30, 期 11, 页 4334–4339, 2018, doi: 10.16182/j.issn1004731x.joss.201811034.

[7] 廖萱, 陈锐志和李明, 《基于Realsense传感器的机器人视觉里程计研究》, 地理空间信息, 卷 18, 期 02, 页 1-4+6, 2020.

[8] 周雄才, 《基于RealSense的深度信息的障碍物检测与避障方法》, 硕士, 重庆邮电大学, 2019. [在线]. 载于: https://kns.cnki.net/KCMS/detail/detail.aspx?dbcode=CMFD&dbname=CMFD202001&filename=1019643556.nh&v=

[9] W. Ma和K. Yang, 《A fast Reconstruction Method of 3D Object Point Cloud Based on Realsense D435》, 收入 *2020 39th Chinese Control Conference (CCC)*, Shenyang, China, 7月 2020, 页 6650–6656. doi: 10.23919/CCC50068.2020.9188665.