**【餐厅服务机器人】**

**软件设计说明书**

**【*SDD Team 104*】**

**【1.1】**

分工说明

|  |  |  |
| --- | --- | --- |
| 小组名称 | 机器人，你在干神魔？ | |
| 学号 | 姓名 | 本文档中主要承担的工作内容 |
| 17373175 | 郑耀彦 | 参与4、5小节编写 |
| 17373005 | 黄月姣 | 参与3小节编写 |
| 17373510 | 王蒙 | 参与1、2、4、6、7小节编写 |
| 17373515 | 李任宇 | 参与1、2、4、6、7小节编写 |
| 17373173 | 叶泽峰 | 参与8小节编写，审核 |

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1.0 | 2020.04.20 | 李任宇 | 叶泽峰 | 软件设计说明书初稿 |
| 1.1 | 2020.04.20 | 李任宇 | 叶泽峰 | 软件设计说明书初稿 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1.      范围

1.1    项目概述

本项目拟在ros启智机器人的基础上开发一个能够为顾客运送产品的餐厅服务机器人。支持机器人自动行驶运行与用户手动操控运行，并提供语音控制等辅助服务帮助用户使用，机器人具体的运行功能如下。

主要功能：

1、机器人的主动控制

2、静态或动态障碍物避障

3、机器人利用传感器实时建立环境地图

4、机器人根据地图和自身的位置信息实现动态路径规划及导航控制

5、检测、识别并定位环境中的特定目标，动态接近目标物

6、抓取目标物

应用场景：餐厅服务、医疗服务

1.2    文档概述

本文档是《机器人，你在干神魔？》小组的软件设计说明书，项目内容组织如下所示，其中包括：需求概述、数据库设计、体系结构设计、接口设计、详细设计运行与开发环境和需求可追踪性说明。

1.3    术语和缩略词

|  |  |
| --- | --- |
| 缩略词 | 全称 |
| ROS | Robot Operating System |
| L | Low |
| M | Moderate |
| H | High |

1.4    引用文档

启智ROS版\_开发手册\_20181109

启智ROS机器人

2.需求概述

2.1 业务需求描述

本项目旨在设计一款用于餐饮行业辅助运送菜品的机器人，主要功能为帮助将一定量的物品自动从一个地点运送到指定地点，或者通过用户手动操控或者语音控制完成这一过程。（注：目前仅支持运送单个货物，多个货物的扩展工作可能会在后续版本中逐步完成）

初次使用或者重置系统后需要对机器人进行相关的配置，用户可手推或手柄控制机器人遍历业务实地场景，使用激光雷达及相关SLAM算法对地图进行建模并保存。可以保存常用起点、与固定地点在内存中简化后续操作。

用户使用机器人时，有三种操控方式。用户可以发出语音指令指定机器人回到地点、检测抓取饮料操作。执行回到地点操作时，机器人从当前位置自动规划路径回到规定地点。也可以通过手机或电脑连接机器人通过我们提供的方向控制板控制机器人前进或者通过填写行程表让机器人按照此表自动规划路线。

机器人将对成功获取到的指令及时做出相应，在到达取物地点和抓取物品成功后发出提示，并对错误和异常情况做出判断、处理和提示，以提供良好的可调试性。

2.2 数据需求

用户通过交互窗口注册或者登录账号，验证身份后可以通过交互窗口或者语音输入对机器人下达任务指令。机器人在用户验证身份后启动，采集周围环境的数据，根据用户下达的任务指令进行相应的操作。

数据范围描述：用户个人用户名、密码、id等基本信息；机器人收集的环境信息；用户指定的任务信息；用户的语音信息。

2.3 非功能需求

**2.3.1性能指标**

响应时间：

主要组成部分为指令识别时间、系统处理时间和指令执行时间。指令识别与系统处理规划应在10s内完成；执行时间根据具体路径和机器人速度而定。

功耗：

根据硬件实际功耗而定。

处理能力：

可以处理简单的语音指令。

可以在地形条件下完成较优的路径规划。

可以自主判断、躲避障碍并重新规划路线。

**2.3.2质量属性**

系统可用性：

可以实现物品抓取、运送等功能。

可以实现人工设置目标与简单的语音控制。

可移植性：

该机器人针对ROS系统。

完整性：

应实现的完整功能包括物品抓取与放下、自主路径规划、简单的语音识别、人工指令设置和避障功能。

效率：

根据实际算法效率和硬件效率而定。

2.4 功能需求

2.4.1 用户手动操控

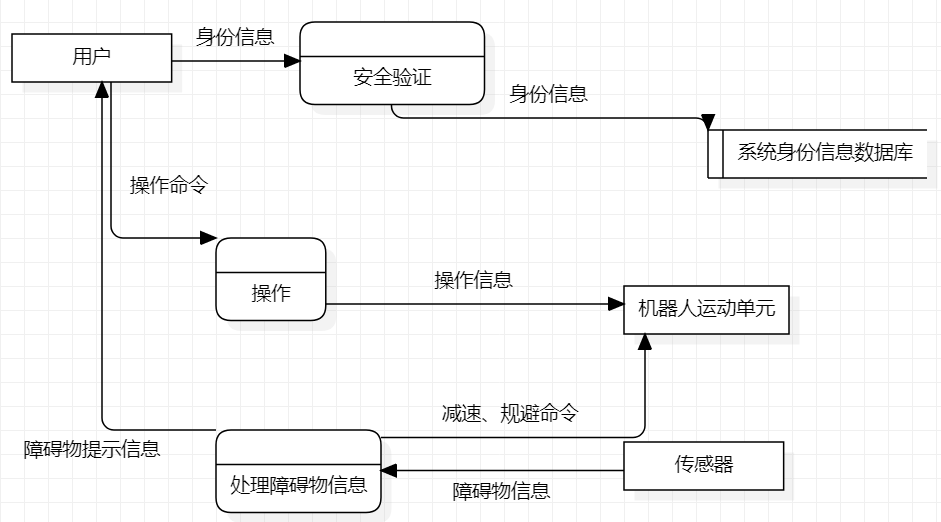


图1 手动操控数据流图

2.4.2 机器人自动运输

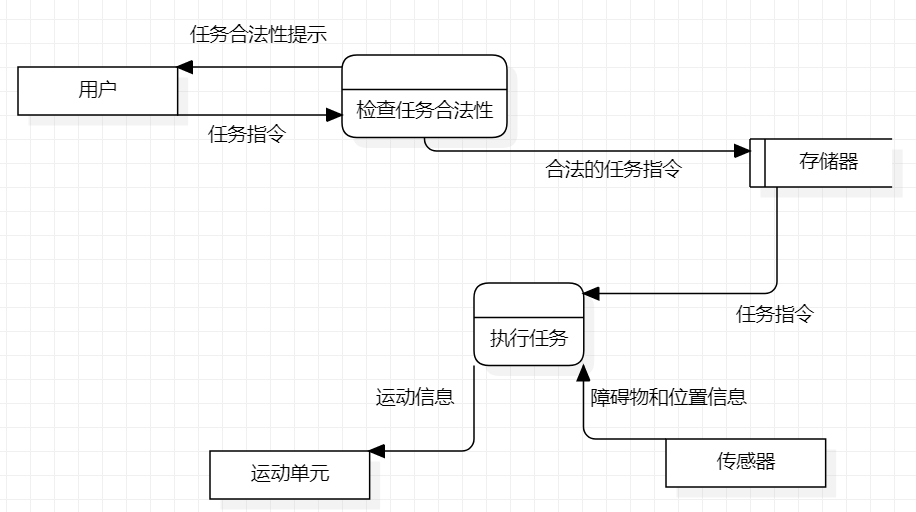
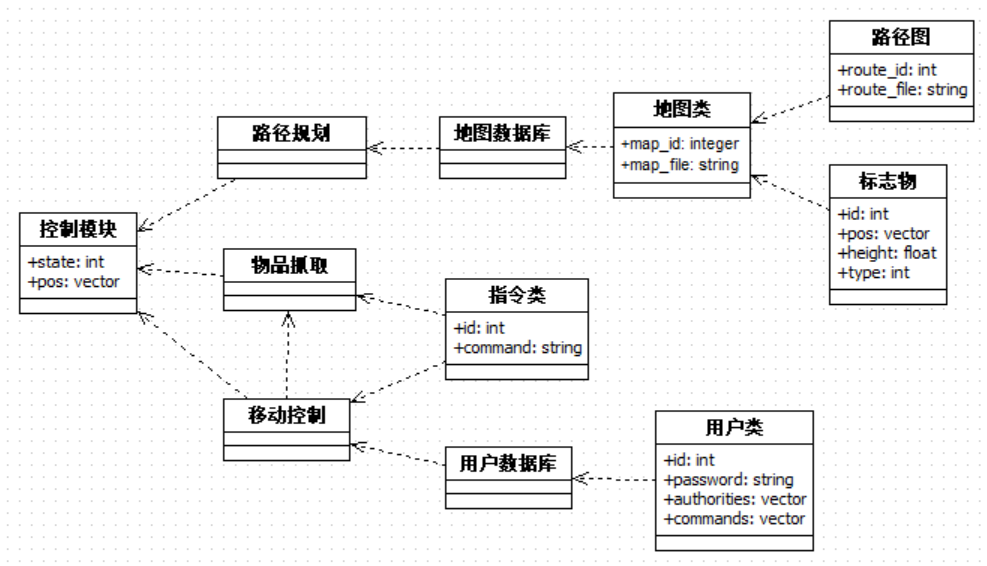


图2 自动运输数据流图

1. 数据库设计

本系统数据库包含地图数据库和用户数据库。各数据库/单元关系如下：



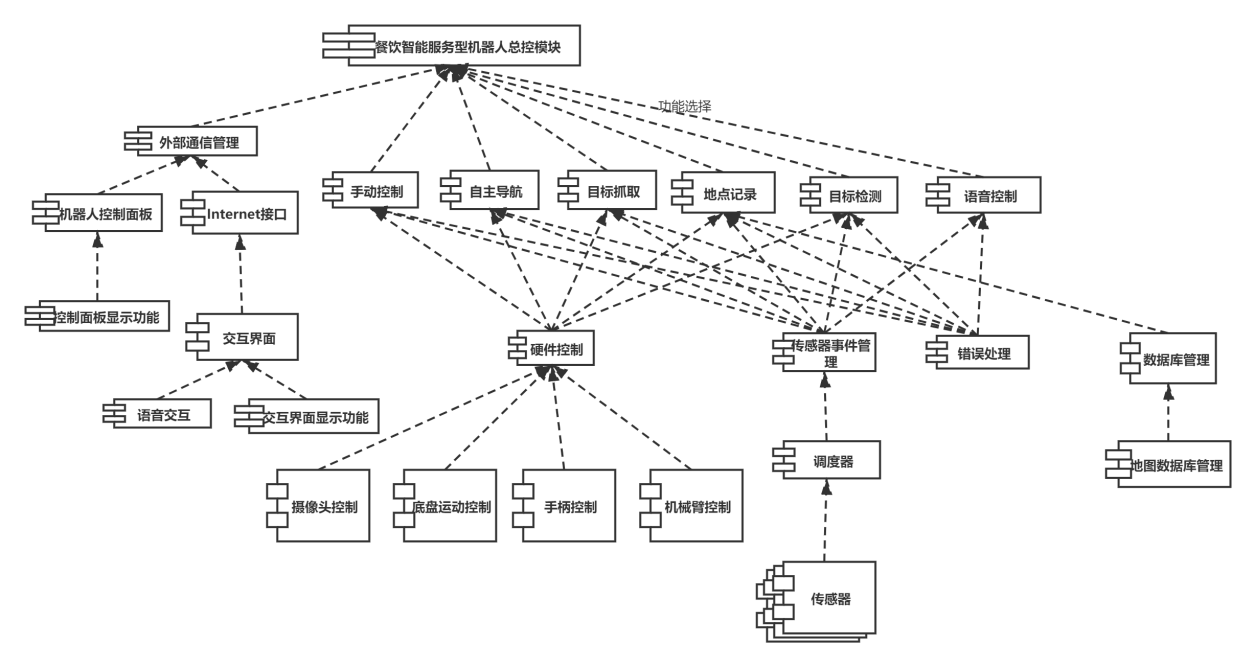
3.1 地图数据库

地图数据库包含了地图信息，受依赖于路径规划模块。地图类的信息包括地图标识以及地图文件索引。地图类内包含的信息有路径和标志物信息。路径包含地图路径信息；标志物为建立地图信息时记录的特殊位置信息，例如餐桌等。

3.2 用户数据库

用户数据库管理用户信息，受依赖于控制模块。用户在通过验证后方可使用机器人执行命令。用户信息包括用户标识、验证密码、用户权限以及用户命令。用户通过标识与密码验证身份通过后，在自身权限内对机器人发出命令。

1. 体系结构设计
   1. 总体结构



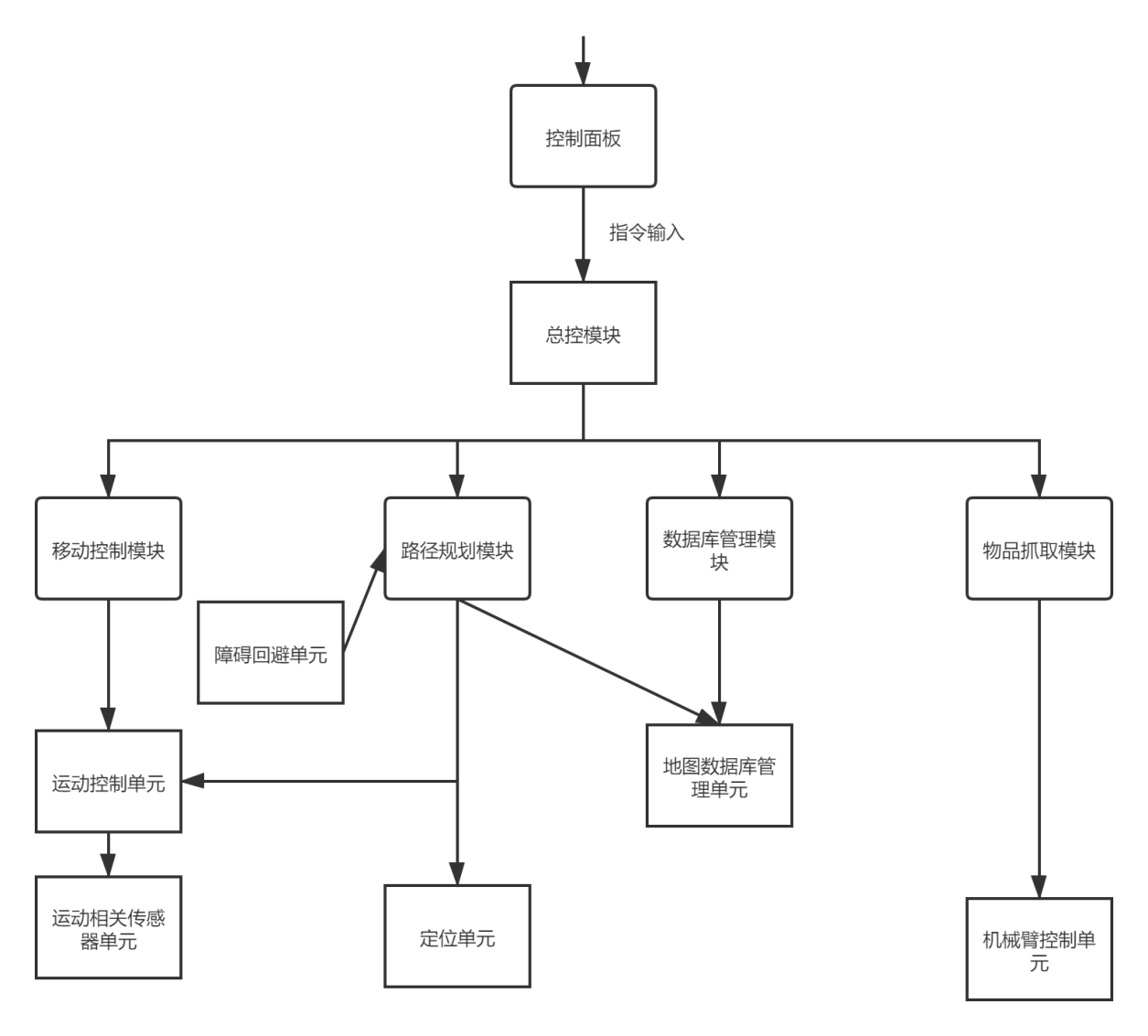
系统的核心单元为外部通信管理模块，总控模块，以及手动移动、自主导航、语音控制、目标检测、目标抓取五种功能的控制模块。附带地点记录的数据库建立模块。

外部通信管理模块获取来自机器人控制面板和Internet端的指令，向机器人总控模块发送消息。机器人控制面板需具备显示功能；Internet'端需要设置交互界面并具备处理语音和文本输入的功能。

机器人总控模块监听外部通信管理模块，解析指令，按需激活功能控制模块。

手动控制移动、自主导航、语音控制、目标检测、目标抓取五种功能的控制模块调用硬件控制模块和传感器事件管理模块完成任务。硬件控制模块需具备控制机械臂和底盘的功能，同时可以接受手柄的操控；传感器事件管理模块通过调度器来接收各传感器的状态及信息、与事件相关联。同时这些功能需要具备一定的错误处理能力。

* 1. 软件体系结构

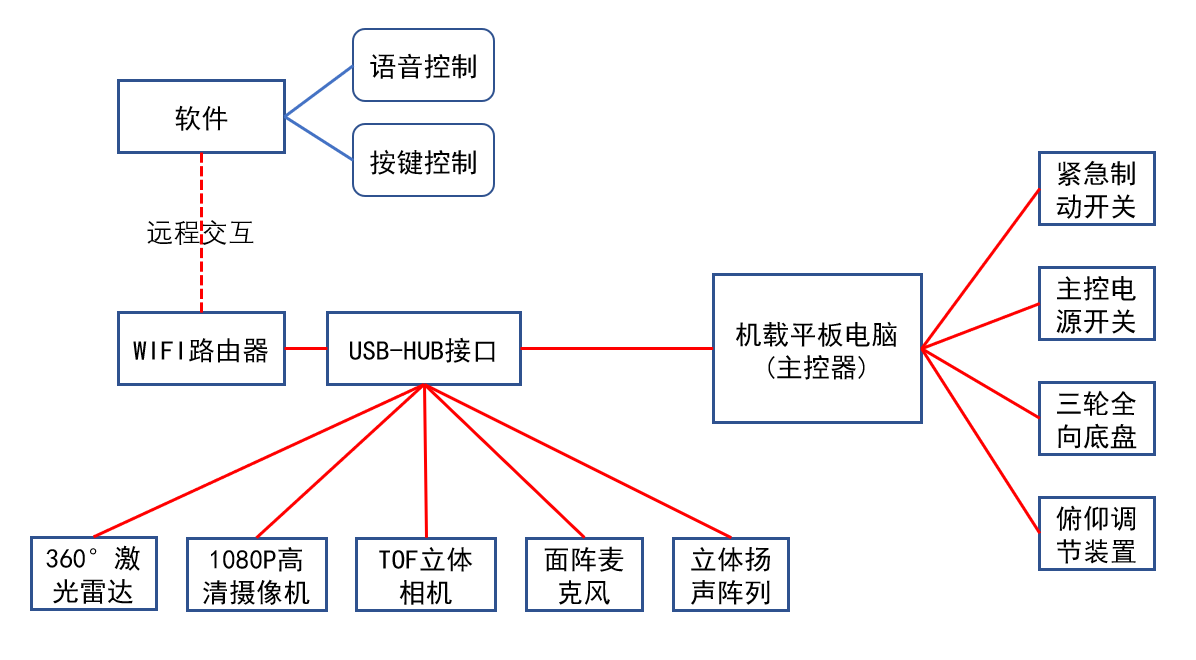


软件体系结构采用混合结构风格。

在总控模块和移动、导航、抓取模块与数据库管理间采用以数据为中心的结构模式（仓库系统风格）。基于路径规划算法、目标识别算法和动态避障算法，总控模块调度移动模块、导航模块、抓取模块执行任务，提供基本移动、自主导航、目标抓取。

在移动、导航、抓取模块内部采用调用/返回结构（主程序结构）。其中，移动模块依照总控模块传来的模式信息，通过调用运动控制单元和定位单元提供基本移动、语音控制移动功能；路径规划模块基于总控模块传来的位置信息、内部存储的地图信息以及知识源中的路径规划算法和动态避障算法，调用定位单元和运动控制单元提供自主导航功能；抓取模块基于总控模块传来的目标物体信息和知识源中的目标识别算法，调用机械臂控制单元，提供目标抓取功能。

4.3.硬件体系结构

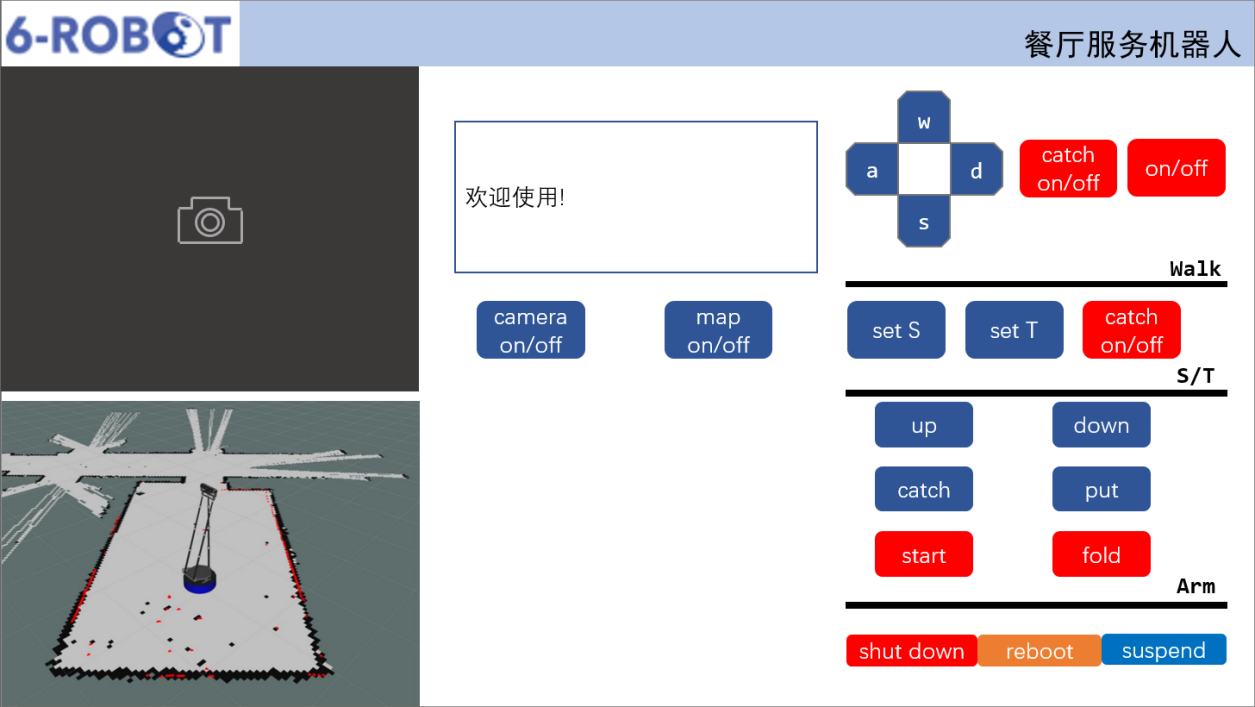


硬件体系结构主要考虑机器人自身的各个组件之间的关系和软件之间的远程交互。

用户的语音控制和按键控制可以通过远程交互控制机器人的开关和行为；同时机器人的雷达或深度相机可以构建出地图并识别障碍物并将信息传至软件界面，摄像头和麦克风捕捉到的环境情况可以通传达至用户。

1. 接口设计

5.1.用户界面设计



如图，用户界面左部为机器人高清相机传来的画面和实时地图界面。界面中部为状态框和相机、地图的开关。界面右部为机器人行走、设置自动寻路、机械臂控制、总开关等按钮。

5.2.硬件接口

1、360°激光雷达：

可捕捉360°的深度信息，并使用SLAM算法构建地图以及识别障碍物，并将信息传递至用户界面和运动控制模块、障碍回避模块、地图管理模块、路径规划模块。

2、底盘：

可捕捉机器人的实时姿态，包括朝向、倾角、速度等信息，并将信息传递至用户界面和运动控制模块、障碍回避模块。

3、1080P高清摄像头：

可实时捕捉机器人所朝向的画面，并将画面传至用户界面。

4、TOF摄像头：

可以识别目标物体并将信息传至目标特征管理模块，帮助机器人识别并抓取目标物体。

5、面阵麦克风、立体扬声阵列：

可以使机器人所在环境和用户终端产生语音交互，帮助用户了解机器人所处环境。

6、USB接口：

连接各个硬件接口和网卡，与用户进行远程交互。

1. 详细设计

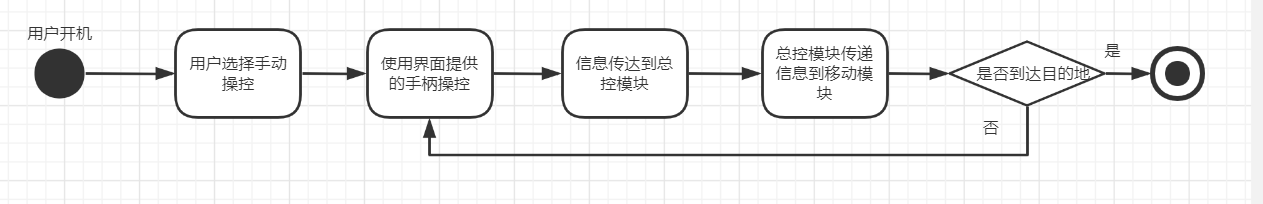
**6.1** UI界面

**功能：**作为交互界面用于用户操控机器人系统，用户通过交互界面传递给机器人系统任务指令。现界面提供有服务自主路径规划功能、手动控制功能与识别与抓取功能。

**界面图像：**UI界面简略图在见第5节接口设计5.1节图像展示。

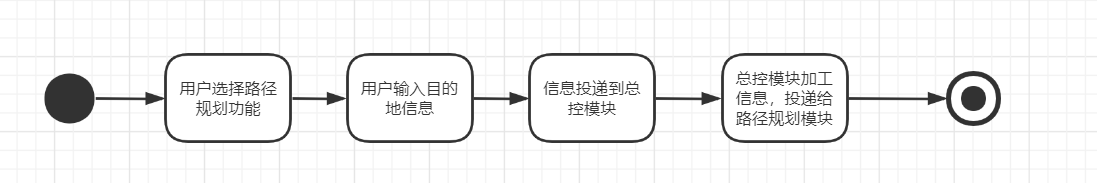
**交互分析：**

1. 手动操控功能顺序图



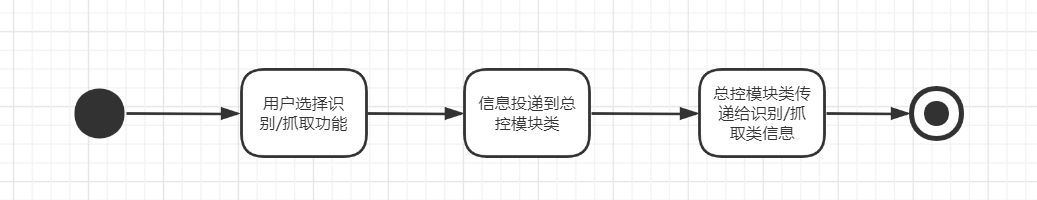
用户通过操控盘(包含方向盘、转向盘、速度控制盘)输入的信息会投递到总控模块类，信息标记为手动控制，总控模块将信息按照功能预定路线传递信息到下属模块类。总控模块类与移动模块类内部工作详情见后。

1. 路径规划功能顺序图



用户选择路径规划功能后，弹出对话框指定目的地位置，信息标记为路径规划交付给总控模块加工处理，总控模块处理后交付给路径规划模块类控制运动。总控模块类与路径规划模块类内部工作详情见后。

1. 识别/抓取功能顺序图



用户选择识别/抓取功能后，信息标记为识别/抓取投递到总控模块类，总控模块传递给识别/抓取模块类自动进行后续操作。识别/抓取内部工作细节设计见后。

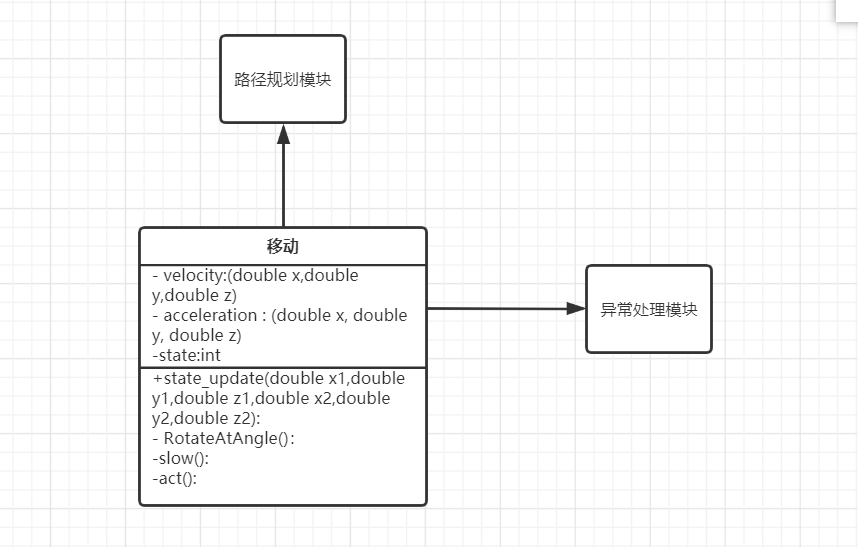
* 1. 移动控制模块

**功能：**负责机器人实现基本的运动，主要是旋转转向与直线平移运动。接收由路径规划模块提供的运动速度与运动方向为输入参数，移动模块需要时刻接收雷达的距离测量信息，当预定运行方向出现障碍物时会采取减速处理。

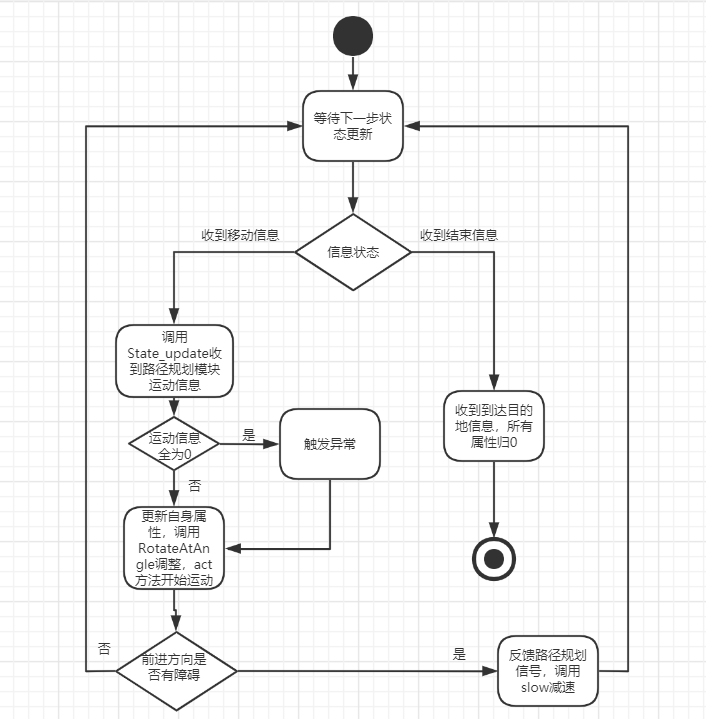
**输入：**移动速度、移动方向、运动状态标记

**输出：**控制底盘运动的信息指令，调用ROS的内置模块

**模块示意图：**



**模块工作顺序图：**



其中信息状态为state属性，1为收到移动信息，0为收到结束信息。

**模块具体属性与方法详细说明如下：**

velocity : (double x, double y, double z) 表示机器人当前速度, x表平移前进速度，负值代表后退；y表示平移向左运动，负值代表向右运动；z 代表机器人自转速度，正值左转，负值右转；x,y的单位为”米/秒”，z的单位为“弧度/秒”

acceleration : (double x, double y, double z) 表示机器人当前加速度，对应项与velocity相同

State\_update() : (double x1,double y1,double z1,double x2,double y2,double z2) 用于得到机器人的下一步的状态信息，x1,y1,z1代表velocity的属性值，x2,y2,z2代表acceleration属性值。

State: int 记录当前状态，0为停止运动状态，1为运动状态。在运动状态下， 如果状态更新时收到的速度、加速度全为0，判定为异常。

RotateAtAngle()：使机器人以某一个角度进行旋转，以达到左右转，向后转等等功能。

Slow():当机器人运行前方有障碍物时，减缓速度

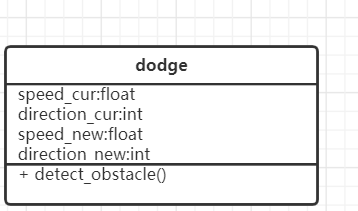
Act():按现有运动状态信息执行机器人运动

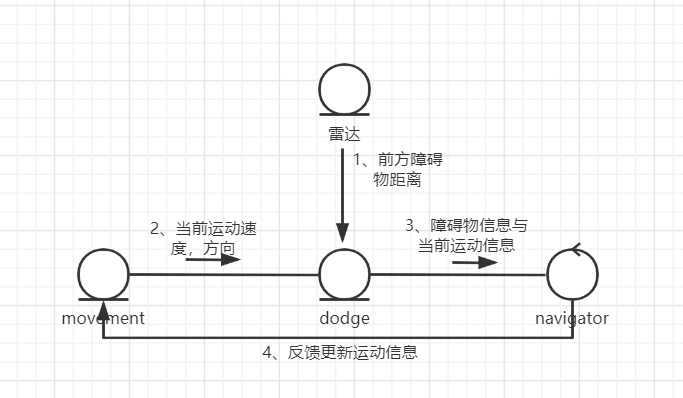
* 1. 障碍回避模块（待修改）

**输入**：预定移动速度与移动方向、前方障碍物距离

**输出**：经过修改的移动速度、移动方向

**功能**：当机器人检测到前进方向上的障碍后，修改机器人当前的速度与前进方向。以实现避障。类图以及交互图如下：



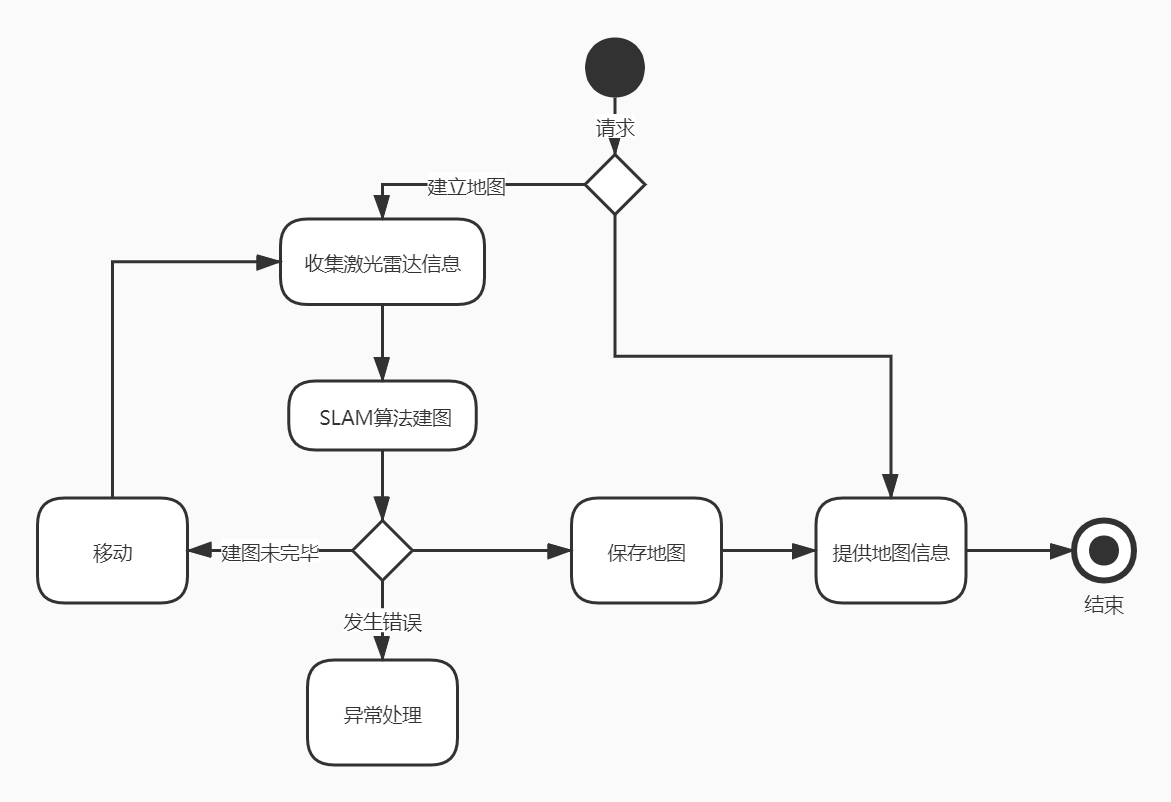


**说明**：当dodge模块接收雷达的障碍物距离信息，当距离达到一个阈值后，请求movement模块的当前运动信息。整合当前运动信息与障碍物距离信息后发送给规划模块，随后接收规划模块更新后的运动信息，将此运动信息传递给movement执行更新功能。

* 1. 地图管理模块（待修改）

**功能**：记录地图建模信息，并给路径规划模块提供地图信息

**说明**：当初次使用或重置系统后需要对机器人进行相关的配置，用户通过手推或手柄控制机器人遍历实地场景，使用激光雷达及相关SLAM算法对地图进行建模并保存。当地图建模完成后，提供自主路径规划模块查询地图的接口。

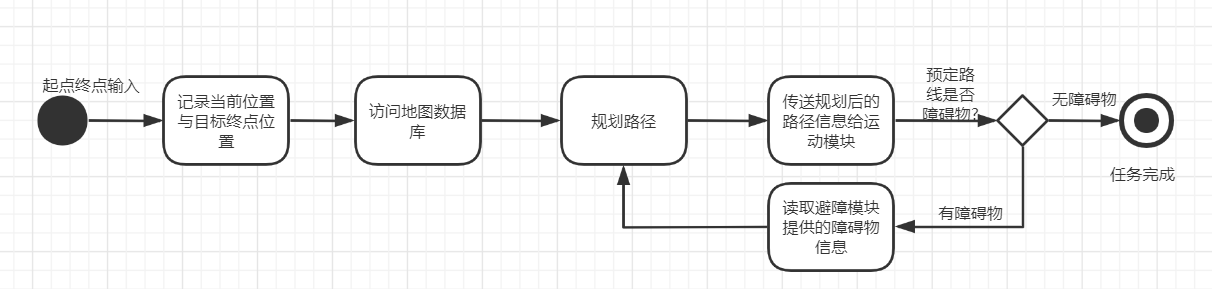


* 1. 自主路径规划模块（待修改）

**输入**: 机器人起点位置(没有输入则默认为当前位置)与终点目标位置。

**输出:** 机器人的移动速度、移动方向，运动前进路径。

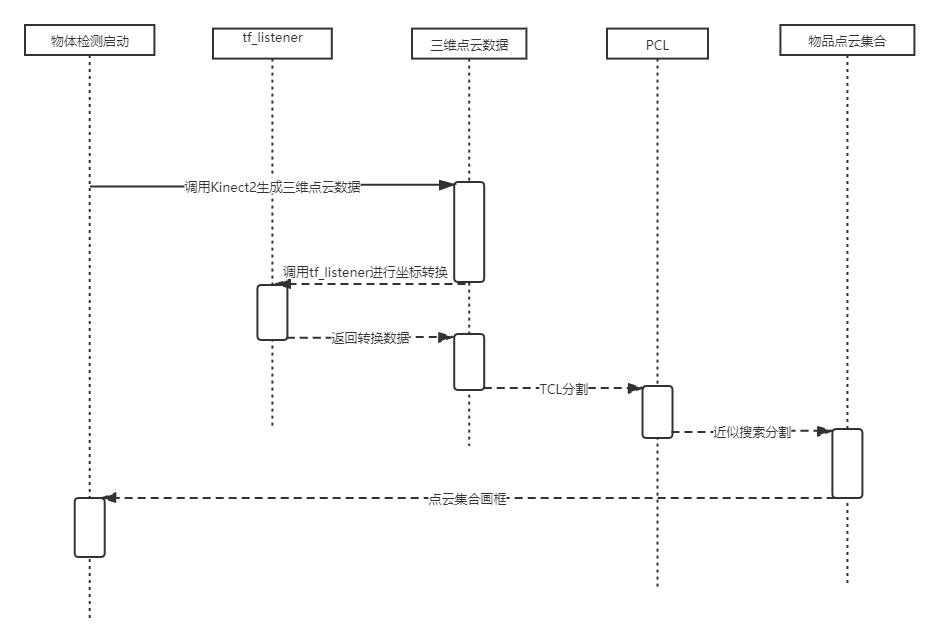
**功能：**当依据机器人已经建立的地图数据库信息，根据用户设置的起点与终点自动规划最短且不会碰撞障碍物的路径。



**说明**：路径规划模块活动图如上，模块收到输入的目的地与起始位置信息后，访问地图数据库信息，开始路径规划，并将计算好的信息传送给运动控制模块，若运动过程中避障模块识别到障碍物，说明此障碍物为不在地图数据库之中的障碍，需要加入考虑传感器收集的障碍物信息，重新规划路径，直到抵达目的地。

* 1. 目标检测模块（待修改）

**功能：**记录目标特征信息，并给目标抓取模块提供三维坐标

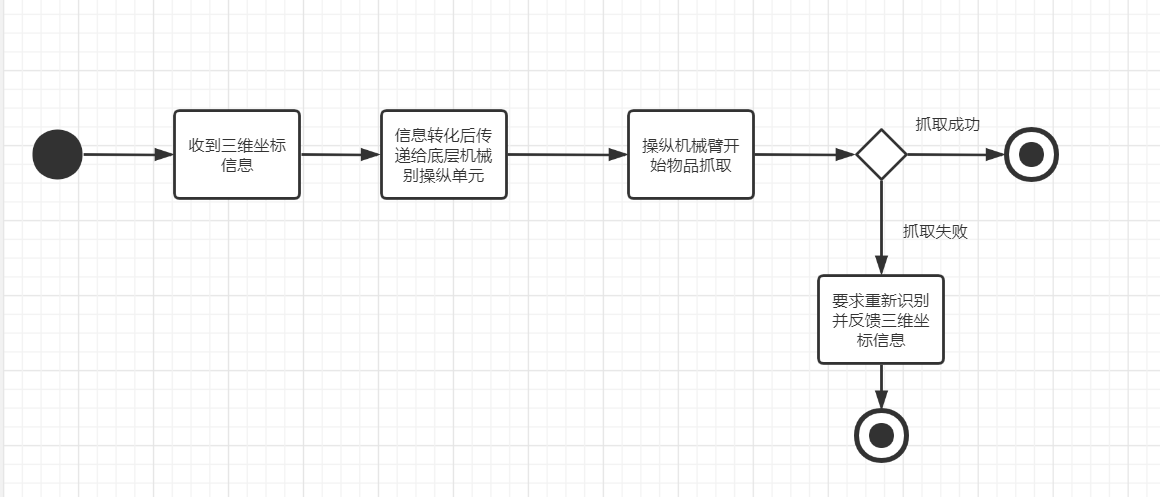


* 1. 目标抓取模块（待修改）

**输入:** 待抓取物品的三维坐标

**输出:** 抓取是否成功的结果

**功能**：按接收目标特征管理模块传送的物品三维坐标，将信息转化后传递给底层机械臂控制单元，控制机械臂抓取目标物品。



1. 运行与开发环境
   1. 运行环境
      1. 硬件运行环境

本系统运行的计算机配置：

CPU：建议 1.2GHz Pentium III 处理器及以上。最低700MHz

内存：建议 RAM 要求2GB。

硬盘：建议20 GB 以上的可用空间。

显示器分辨率：最低Super VGA (800x600) 或更高分辨率的显示器（颜色设

置为32位真彩色）

其他：键盘、鼠标、2个以上USB接口。

* + 1. 软件运行环境

本系统运行的软件环境为Ubuntu 16.04的支持kinetic版本的ROS系统。

* 1. 软件环境

软件开发环境是 RoboWare Studio，软件开发语言为C++，工具使用基于 Visual Studio Code 开发的 ROS 专用 IDE。

1. 需求可追踪性说明

8.1 功能需求

本项目作为一款用于餐饮行业辅助运送菜品的机器人，主要功能为帮助将一定量的物品自动从一个地点运送到指定地点，或者通过用户手动操控或者语音控制完成这一过程。

（1）本系统要求机器人对地图进行建模并保存。实现这一目标的方法是使用启智ROS机器人自带的激光雷达，搭配SLAM算法完成建模工作。本系统中的地图管理模块将实现此功能，由数据库建立模块保存地图数据。

（2）本系统要求用户与机器人通过移动设备进行交互，并使用手柄对机器人进行操控。启智ROS机器人自带多种控制手柄支持，拥有丰富灵活的设置参数。由外部通信管理模块完成交互功能。

（3）本系统要求机器人对不同待抓取物品进行特征学习，以便后续物品的识别抓取。启智ROS机器人头部装备kinect2视觉传感器，彩色图像分辨率高达1920\*1080，用以图像获取；使用PCL的平面检测算法，可以准确识别物体，进行学习。由目标特征管理模块实现此功能。

（4）用户操控机器人时，机器人的动作主要为移动与抓取。启智ROS机器人拥有三轮全向底盘，可以安装机械臂满足抓取物品的需求。由目标抓取模块完成此功能。

（5）用户可以使用语音指令指定机器人的操作。启智ROS机器人自带阵列麦克风，可以用于采集正前方的实验数据；使用PocketSphinx语音识别包可以进行语音指令识别。由语音控制模块完成指令处理。

（6）机器人有自动规划路径移动到指定地点的功能。启智ROS机器人带有navigation导航功能，通过对ROS包修改增加控制速度，多点导航的功能，满足用户规划路径的需求。

8.2 非功能需求

部分非功能需求依托于机器人的具体硬件参数而满足，如功耗，可移植性，系统可用性等。以下仅说明需要通过系统设计方案满足的需求。

8.2.1 性能指标

（1）处理能力：

  可以处理简单的语音指令，包括识别定义指令、识别并提示未定义指令。可以在平坦的地形条件下完成相对短的路径规划，规划过程限定在10s内。可以判断、躲避障碍并重新规划路线。

此要求将由语音控制模块，自主路径规划模块满足，我们将通过优化路径规划算法的方式，提升算法的速度，满足性能要求。

8.2.2 质量属性

（1）完整性：

SRS要求实现的完整功能包括以下几点：

1.物品抓取与放下：自主识别目标、判断距离；

2.路径规划：路径规划应在10s内完成，且路径为较优路径；

3.语音识别：识别固定的几条简单指令，并能识别出未定义指令；

4.避障功能：可识别阻挡运动的障碍物、停止并在10s内重新规划路线。

本系统将以上功能分割到不同模块中，由目标抓取，自主路径规划，语音控制，障碍回避模块分别完成，由总控模块对这些模块进行控制。

（2）健壮性：SRS要求机器人能够识别并提示出未定义指令。设置最大执行时长，避免没有可操作目标、地面情况导致的运动受阻或路径死循环。

该要求将由语音控制模块和总控模块完成。发生错误时，以上模块将捕获错误，停止工作，向用户发送提示，人工排查错误。