

硬件方面：

1. 使用 **ROS2** 系统控制小车运动
2. 使用 **香橙派5 / 树莓派4B / Jetson Nano / N100 等** 作为主控（兼容ROS2高效运行）

软件方面：

1. 多模型配合进行用户情绪识别：
 - 应该选择什么模型 - 考虑针对一下方案可以找到什么开源模型/库
 - **语音转文本 + 文本情绪分析** + 人脸情绪分析 - 考虑 **Google Speech-to-Text（语音识别） + BERT / RoBERTa（文本情绪分析） + OpenCV + dlib / FER（Facial Expression Recognition）（面部情绪分析）**。
 - **语音情绪分析** + 人脸情绪分析 - 考虑 **VGGish / OpenSMILE（语音情绪分析）**
2. agent 进行人机交互
 - 一方面处理与用户的**语音交互**，一方面接收**情绪分析结果** - 考虑 自然语言处理（NLP）库 + 对话系统（如 Rasa 或 Dialogflow）来创建智能的交互系统 + **强化学习**
 - 输出：语音交互 + UI图像变化指令
3. 多模型配合方法：
 - 各个模型 **预处理** 机制应一样
 - **注意力机制**（自注意力/多头注意力）
 - **多层感知机**
4. 小车运动控制：
 - 小车简单的运动
 - 自动防跌落 - 考虑 **激光雷达/防避障传感器/超声波传感器**
 - 自动避障
 - 其他传感器 - 考虑 **摄像头 + 温度（体温）传感器 + 麦克风 + 音响设备**
5. 哪些计算在云，哪些计算在本地
 - 本地：实时性要求较高的：**自动避障+防跌落** + 传感器数据（人脸信息，语音信息）上传到云
 - 云：接收信息进行情绪结果**分析 + 记录 + 学习**
 - 兼顾隐私问题 - 考虑 **将人脸信息/语音信息在本地预处理** 后再上传到云
6. 小车与云的通讯方法
 - MOTT / WebSoket - 考虑 **速度要快**

外观设计

1. 交互界面设计：
 - 总体需要**简洁**，传递温暖与美好
 - 注意**亲和力**和**可互动性** - 考虑 移动应用/网页界面

2. 外形设计：

- 避免过于复杂的机械设计
- 考虑 **圆润+柔和+可爱** 的元素
- 要让用户有**交互的欲望**

3. 模块化设计：

- 鉴于传感器等模块较多，设计成**可拆卸模块**方便后续更改

附人员安排

2人：负责硬件+外观设计（damo）+通讯方法+云服务器 的 选择+理由（5、6）—— lyj+brx

1人：模型选择 + 给出理由（1）—— yzb

2人：agent + 多模型配合方法（2、3）—— gyx+cyx