

口罩识别机器人 总结报告

组号：24

小组成员：陈星

康宇琦

王啸轩

吴晔

杨雨翔

一、项目介绍

1. 选题背景

本小组的所有成员都曾在学期初参加过班级组织的以观畴园餐厅测温岗工作体验为内容的志愿服务活动，亲身体会过学校餐厅入口处测温人员工作的不易。当下正逢炎炎夏日，食堂工作人员仍然要在门口提醒进入的同学们戴好口罩，十分辛苦。所以，我们希望可以设计并制造出一个智能机器人，它能够识别同学们戴口罩的情况并对未戴口罩的同学做出提醒，从而实现代替人工劳动的目的。

2. 作品功能

a. 在小车的综合设计中，我们设计并实现了小车+树莓派+摄像头+上位机的整体架构。各部件信息通路如下图所示：

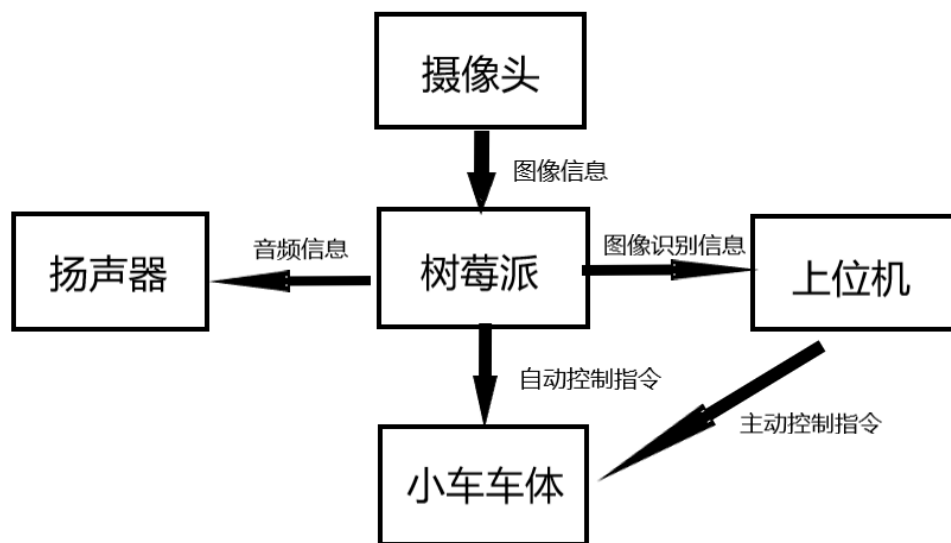


图 1. 整体框图

b. 小车实现了分辨人员是否佩戴口罩的功能。该部分由树莓派摄像头按照一定的频率循环捕捉外界影响信息，并将所拍摄的照片信息传递给树莓派进行图像识别。树莓派通过运行图像识别代码判断出图像中人脸的数目以及各自佩戴口罩的情况。

c. 小车实现了对未戴口罩的人员进行跟随，以及实现跟随过程中进行无接触避障的功能。在实现了小车的基本运动与树莓派的图像识别功能以后，我们使树莓派能够根据图片中的人脸位置，向小车发送方向信息指令。小车可以根据该指令进行趋向运动，从而实现跟随的功能。此外，在行进的过程当中，小车可以根据超声波传感器所传递的实时距离信息，进行行进路径中障碍物的躲避。

d. 小车实现了在跟随过程中通过扬声器播放提醒信息的功能。当树莓派检测到未戴口罩的人脸时，会将提前录制好的音频信息发送给树莓派扬声器，通过扬声器向外界播放语音提示信息。

e. 小车实现了通过上位机监控摄像头拍摄结果并遥控小车的功能。小组成员为树莓派编写了一个简单的图形界面。该界面能够将上位机从树莓派处接收的图像信息直观地显示在屏幕上，图像中不仅包含原始影像信息，还有人脸识别情况、面部口罩佩戴情况以及识别的置信度等信息。此外，图形界面上还设置了小车运动的控制按钮，包括前进、后退、转向、启动、停止、休眠等等，可以使得操控者能够直接利用计算机的键盘向小车发送指令，进而控制小车

2. 上位机

在上位机端我们制作了一个简单的图形界面，包括一些功能按钮和图片显示窗口。这个程序通过 Socket 协议和树莓派进行通信。一方面操作者可以通过功能按钮向树莓派及小车发送特定的指令，例如启动、休眠、前进后退等；另一方面也可以接收经过树莓派处理的图像，上面用方框标注出了目标脸部的位置，并且用颜色标注了目标是否佩戴了口罩。

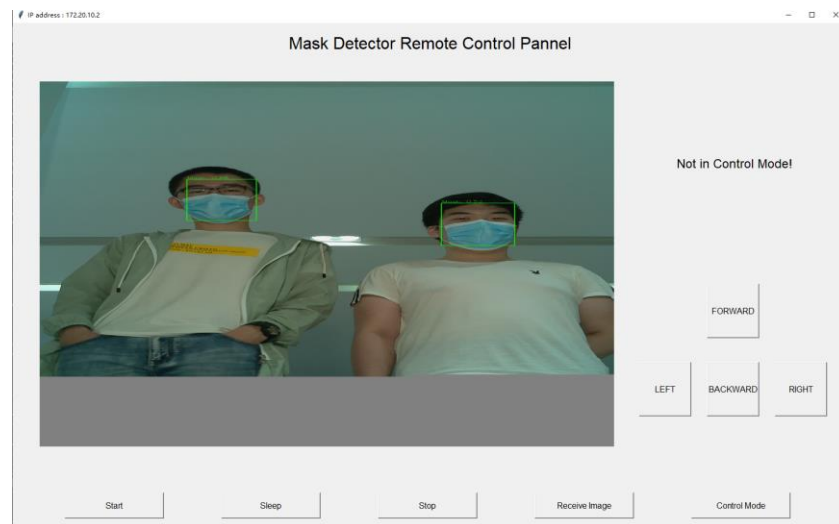


图 4. 上位机界面示例

3. 传感器

我们小车部分使用的传感器主要包括超声波传感器和加速度传感器。超声波传感器主要用于避障，我们通过自己封装的一个 Sensors 库来对其进行调用。

加速度传感器主要用于小车的转向，我们通过检测其变化的角度来控制小车转过的角度。由于小车转弯速度较快，传感器读数可能有较大误差，因此我们打算采用多个传感器来共同检测，以减少角度的漂移。同时由于单片机引脚有限，因此我们购买了支持 SPI 协议的 MPU6500，通过 SPI 来读取传感器的读数。

4. 机器人运动

- 小车实现了基于负反馈的直行。在小车直行的控制逻辑上，我们并未选用通过设定经验参数而实现的静态直行，而是设定了基于负反馈的动态直行方案。在直行指令被下达之前，小车会通过陀螺仪设定一个参考方向。而后在小车的行进过程中，会动态检测是否偏离此方向，若偏离则通过左转或右转来回到参考方向。如此的设计方案使得小车的直行更加稳定，代码的兼容性也更强。
- 小车的运动采取了状态机的设计方案，包含停止、巡逻、手动控制和跟随四种状态。小车能够根据上位机发送的信号在四种状态之间切换。
- 小车的避障采取了状态机的设计方案，包含直行、转向前进、左转、右转和受控转向五种状态，状态转换图如下图所示：

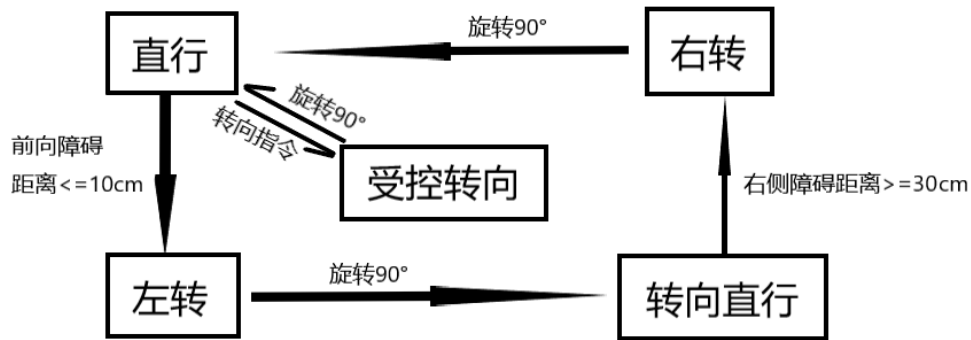


图 5. 小车运动状态转换图

5. 通信

通信部分除了上述树莓派和上位机的通信,还包括树莓派和 Arduino 以及 Arduino 和调试电脑之间的通信。

树莓派和 Arduino 使用 UART 通信协议,利用有线连接进行通信。而 Arduino 和调试电脑之间通过蓝牙模块 HC-05 进行通信,主要用于在调试过程中输出一些调试信息。

三、问题与解决

1. 系统延迟

这个项目最大的问题就是系统的延迟过高,尤其是树莓派处理的部分,其延迟拖慢了整个系统的运行速度。

在开发过程中,我们遇到的第一个较大的延迟就是树莓派的图像处理部分。我们开始时使用了 PyTorch 框架进行模型推理,尽管对网络结构进行了缩减,但每帧用时仍在 2~3 秒左右。

为了优化推理速度,我们将模型移植到了一个边缘推理框架——tensorflow lite 上,每帧用时缩短到了 0.3 秒左右,基本上满足了实时性的要求。

另一个延迟较大的部分就是摄像头的拍摄。如果按原始分辨率进行拍摄,从预热到得到图片大约需要 2 秒左右的时间。由于这个延迟主要来自于摄像头的感光元件,因此难以从软件上进行优化。我们最后在准确度和速度上进行了取舍,适当降低了拍摄的分辨率,将拍摄时间缩短到了大约 1 秒。

如果后续需要在延迟上进行优化,就可以从上述两个方面来考虑。首先是模型的推理速度,我们一方面可以对模型本身进行优化,另一方面也可以使用一些边缘推理加速的硬件来加快数据的处理。其次是摄像头的拍摄速度,我们可以设计出一款速度更快、同时精度损失较少的感光元件,也可以选用更高效的信息传输协议,对摄像头的驱动进行修改,以满足我们对实时性的要求。

开发过程中遇到的另一部分延迟就是传感器的延迟。首先在每次读取超声波测距的数据时,我们都需要等待回传的信号进行计算,在这个过程中主循环被暂停。由于我们有六个超声波传感器,因此这一读取过程累加起来,对处理速度造成了极大的影响。其次是陀螺仪的延迟,由于硬件的特性,每次上电后陀螺仪的示数都需要 10 秒左右才能稳定,给使用造成了不便。

为了解决超声波测距的延迟问题,我们在程序中加入了提前判断,当超声波距离符合条件时立刻停止测量,进入下个状态。这种方法一定程度上减少了延迟的影响,但效果还是不够理想。后续如果需要改进,我们可以使用多线程的办法,为超声波测距单独创建一个线程。这样可以将测量的延迟独立于主程序之外。

2. 模型精度

开发时遇到的另一个问题是模型的精度并不理想。在初期测试时，我们将摄像头放在桌子上，得到的测试效果较好。但是实际安装后才发现小车拍摄到的图片大多光线不足，得到的脸部信息较为模糊，而且相较于桌子上的平视视角，小车对来往行人是仰视视角，这都对模型准确率造成了影响。另外在上一步优化推理速度的同时，模型的精度也相应有所损失，影响了推理的结果。

在后续的开发过程中我们发现如果人为进行补光，小车的识别率可以保持在一个能接受的水平，再加上时间限制，我们没有对这个问题进行进一步地解决。如果要从根本上解决这一问题，我们可以利用小车拍摄一些不同的人脸照片，整理出树莓派专用的数据集，并在训练过程中使用一些数据增强技巧，这样就可以得到更适合于这个项目的识别模型。

3. 电机运动控制

由于本小组所领取到的小车有一个车轮电机损坏，而补领的小车恰好又是对向电机损坏，所以小组成员将两个小车未损坏的电机组装到了一个车体之上。这导致了我组小车的两轮电机转速相差过于悬殊，在调试的过程中遇到了很多的困难。尽管小组成员从理论和经验两方面入手，努力克服了这些困难并实现了最终功能，但是小车在运动过程中仍时有如转向不精准之类的不稳定现象发生。此外，由于小车采用的电机为直流电机而非步进电机，车轮的转速无法直接测量。而若想借助外部元件实现又会因小车车体空间过小导致霍尔编码器无处安放，于是最终我们放弃了对车轮电机转动的精密度更高的 PID 控制。或许，找到一个合适的码盘将会是一个有效的解决方案。

四、创新点

1. 结合了多种传感器、数字模块和主控来实现小车的控制。在实现小车的控制时，我们结合了多种不同的信号，例如超声波的探测信号、树莓派的处理结果、上位机的指令等，小车的运动和每一部分的信息都有关系。

在文献调研时我们发现了识别口罩的模型、树莓派实现的小车跟随、小车超声避障等不同的项目，但各自没有什么联系。我们通过口罩检测跟随这样一个功能将各个模块结合到了一起，共同组成了一个完整的电子系统。

2. 选题贴近时事，实用价值比较高。而且我们在设计时考虑到了实际的需求，例如在选择上位机时没有用蓝牙模块进行通信，而是选用了适合长距离传输的方式。从上位机，到树莓派再到 arduino，我们建立了一个比较完善的使用场景。

3. 针对模型在树莓派上的推理速度进行了优化，使得其可以满足小车这样一个较为实时性的系统的需求。

五、工作日志

6月28日上午：

1. 硬件方面，组员们领取小车后进行了一系列的安装工作。首先是拆除了小车已经被搭建好的结构，然后为小车安装上了其主控板。由于套件中的行程开关线缆已不翼而飞，所以还进行了行程开关连线的焊接工作。
2. 软件方面，组员们研究了开发板的数据文档，下载并安装好了小车主控板的 IDE：CCS 平台。随后进行了预习报告和 PPT 的展示，领取了一些所需器件并登记了所缺芯片。

6月28日下午：

1. 组员们尝试启动树莓派并安装了计算机视觉库 opencv, 并搜索到了与面部口罩识别相关的程序在树莓派上进行编译尝试。
2. 组员们针对项目中需要使用到的各模块的原理、功能、使用方法等进行了学习, 同时对 CCS 编程进行了学习与尝试。经过一下午的学习, 组员们已经能够根据相关教程, 自己动手编程和连接电路, 最终在小车的主控板上实现一些简单的功能。

6 月 29 日上午:

1. 硬件方面, 小组成员检查并确定了小车左轮电机的故障, 领取了一个右轮故障的小车并将原车电机拆卸、焊接、组装到了新的车体之上。此外, 小组成员检查并确定了小车主控板出现烧毁的故障, 于是更换了小车主控板, 并安装了对应的 IDE。
2. 软件方面, 小组成员学习了 PWM 控制电机的原理, 并尝试编写代码控制电机的运转。经过组装与调试, 小车车轮能够按照设定的方向转动。

6 月 29 日下午:

1. 小组成员编写代码, 成功实现了超声波传感器的测距功能。
2. 小组成员编写代码, 成功实现了 OLED 模块的显示功能。
3. 小组成员尝试在树莓派上运行插件。
4. 小组成员完成了新主控板与小车的连接, 编写代码, 成功利用 PWM 通过控制工作电压的占空比控制了小车车轮电机的转速。
5. 小组成员完成了行程开关的焊接与连线工作, 编写代码, 实现了利用行程开关控制电机的运转。经测试, 在小车接触到障碍物, 行程开关闭合后, 小车的两车轮能够从直行状态转换到碰撞处理状态。
6. 为小车添加了蓝牙控制的电机电源开关, 便于后续的调试工作。

6 月 30 日上午:

1. 小组成员尝试编写代码进行小车的转向控制工作。
2. 小组成员尝试编写代码对小车的电机进行 PID 控制。
3. 小组成员在树莓派上成功安装了 opencv 库。

6 月 30 日下午:

1. 小组成员实现了粗略的方向控制。随后, 通过方向控制, 小组成员基本能够实现小车的直线行走。
2. 小组成员在树莓派上成功安装了 pytorch 库, 调试了树莓派的 UART 串口, 学习了与 GPIO 有关的知识。
3. 小组成员尝试通过调制电机 PWM 波的频率与占空比控制小车的平稳行进。

7 月 1 日上午:

1. 小组成员测试了树莓派蓝牙的功能。
2. 小组成员尝试构建类来实现对小车运动控制部分代码的封装。
3. 小组成员取回了订购的树莓派摄像头和扬声器, 将摄像头连接到树莓派上并调试, 目前摄像头已经能够正常拍摄实物图片并传输到树莓派上。

7 月 1 日下午:

1. 小组成员将扬声器连接到树莓派上并调试, 经测试扬声器能够正常播放音频。
2. 小组成员尝试用树莓派拍摄人脸并运行代码进行口罩识别。经过测试, 树莓派能够处理图像, 并能正确识别出画面中戴口罩的人的数量。至此, 树莓派部分的功能已经基本实现。
3. 小组成员重新架构小车运行的代码, 目前小车已经能够完成前进、后退、左右转弯、向左右方向行进等多种运动状态。
4. 小组成员在小车车体上组装了超声波传感器, 并编写简要的小车避障代码来综合测试小

车运动方面的性能。经过实际测试，小车已能够实现前向的避障功能。

7月2日上午：

1. 小组成员将树莓派中口罩识别的模型移植到了移动计算的框架中，使得模型计算的速度有了显著提高。
2. 小组成员优化代码，尝试控制小车使其避障过程的行为更加稳定。

7月2日下午：

1. 小组成员优化代码，尝试控制小车使其避障过程的行为更加稳定。
2. 小组成员就如何将目前已完成的小车与树莓派两部分的工作整合为一体，进而实现最初设定的目标功能进行了相关讨论，初步得出了实行方案并据此制定了下周的工作计划。

7月5日上午：

1. 小组成员将树莓派与小车连线后尝试进行数据通信，通过编写代码和修改调试，小车能够接收到树莓派进行图像识别后发送的信号，并能够针对不同信号执行特定的指令。
2. 小组成员设计电源管理电路。

7月5日下午：

1. 小组成员拆除了此前为调试而搭建的小车结构，进行了实现实际功能所需要的小车结构的组装。
2. 小组成员通过改变图片数据的加载方式，省去了图片的存储与读取环节，优化了图像识别代码的运行速度。
3. 小组成员尝试编写代码，为小车树莓派部分的功能搭建可视化图形界面。
4. 小组成员尝试为小车运动的代码增添更复杂的逻辑，以迎合预期实现的功能的需求。

7月6日上午：

1. 小组成员优化了小车运动的代码并进行了漫长的调试，现在小车的运动更加稳定、精确。
2. 小组成员完善树莓派部分代码和图形化遥控界面，为将小车部分与树莓派部分的结合做准备工作。

7月6日下午：

1. 小组成员修改了小车避障代码的逻辑，以更高效地适应预期实现的功能。
2. 小组成员尝试编写代码，将树莓派拍摄并处理过的图片信息发送到PC端。

7月7日上午：

1. 小组成员完善树莓派部分代码和图形化遥控界面，树莓派部分结合前的准备工作已经完毕。
2. 小组成员完善了小车的新版避障代码，经过重复调试，小车已能够正常实现避障功能。

7月7日下午：

1. 小组成员经过漫长的调试，成功解决了小车执行实际功能时遇到的多种问题。
2. 小组成员将小车与树莓派两部分结合为一体，测试其整体功能的实现情况。目前已完成了小车一部分功能的测试，发现均能够正常执行。

7月8日上午：

1. 小组成员对小车进行了综合调试，目前已基本实现了小车的行进与避障、面部口罩识别、对未戴口罩者进行跟随等基本功能。
2. 小组成员将电源管理电路焊接完毕。

7月8日下午：

1. 小组成员继续对小车进行综合调试，对小车执行目标功能的完整流程进行了测试。经过多次的修改，小车已基本能够完整实现目标功能。
2. 小组成员为答辩时需要展示的视频的拍摄工作进行了准备。

7月9日上午：

1. 小组成员拍摄了答辩时播放的功能演示视频。
2. 小组成员在助教的协助下进行了现场的实物验收。

六、项目代码(见附件)