

智能送药小车（F 题）

项目报告

题目

智能送药小车（F 题）

团队

钟梓轩 宋宜航 李稚凌

摘要

使用 Arduino Mega2560 作为主控芯片，使用电机驱动板 1298n 来驱动电机，使用两节 3.8V 电池供电，并使用一块 LM2596DC-DC 模块来实现电源升压。使用三块 TCS230 颜色传感器进行巡线，并通过 OpenMV Cam H7 Plus 进行数字识别。通过 OpenMV 机器视觉模块将采集到的图片集进行本地训练，获得自己的数字检测模型。检测模型的精准度会随着训练图片集数目和训练时间不断提高。两轮的小车结构更有利于小车的转向，减小了小车所占空间。经测试，智能送药小车的技术指标达到了设计要求。

1 引言

该系统需要将 200g 药品送到指定病房，小车在利用视觉模块识别到病房号，并感应到药品装载后，自动开始运送，根据走廊上的标识信息自动循迹；在到达指定药房并且卸载药品之后，小车自动返回。

为实现以上功能，我们将系统分为几个模块，分别是巡线模块，数字识别视觉模块，药品检测模块，电机控制模块和供电模块。其中视觉模块使用 OpenMV 进行数字识别，在识别到数字之后，传输给 Arduino 来控制电机运动，在小车运行时，OpenMV 持续检测，在路口处检测目标病房的数字，判断左右转。在到达目标病房并且已经卸载完药品之后，依然通过识别数字返回原位。未完成拓展部分，主从小车通信。

2 系统方案

本系统主要由循迹模块、数字识别模块、药品检测模块组成，下面分别论证这几个模块的选择。

2.1 循迹模块的论证与选择

方案一：使用灰度传感器循迹

我们尝试了灰度传感器进行寻迹，但是在实际场地实验时发现用来循黑白线的灰度传感器并不适用于此题的红白巡线。具体原因是，红色和白色对灰度传感器常用的 850 纳米至 950 纳米的红外光吸收率基本相同，因此无法检测并区分红色和白色线路。

方案二：使用 OpenMV Cam H7 Plus 模块进行循迹

OpenMV 模块可用于红白巡线，具体方案为，利用 OpenMV 官方提供的“`image.find_blobs()`”函数，将屏幕捕捉画面的水平居中部分均分为五块区域，识别区域中是否存在红色色块，通过对存在值的检测及五块区域加权计算，得出行驶路线是否偏移及其偏移程度。

方案三：使用颜色传感器 TCS230 检测红线

考虑使用颜色传感器 TCS230 来检测红白线，该传感器通过发射白光，并检测反射光中被吸收的部分，来实现对现实区域中颜色的检验。

经过对比，使用灰度传感器循迹对于红色路径的识别率较低，而使用颜色传感器 TCS230 检测路线的效果较好，且不占用视觉模块资源，因此我们选用使用颜色传感器 TCS230 检测路线的方案。

2.2 数字识别模块的论证与选择

方案一：使用 OpenMV Cam H7 Plus 机器视觉模块进行数字识别

通过配置环境、采集图像数据、制作数据集并进行本地神经网络训练，以及对实时图片的滤波及仿射处理，可以实现利用 MobileNet V2 神经网络训练进行的 OpenMV 数字识别。随着数据量的增加，该模型的识别精度也随之升高。且 OpenMV 作为经典的开源开发板，经过发售公司在 STM32H7 芯片上对 MicroPython API 控制硬件底层的烧录，可以使用简单的 Python 包等工具进行程序编写，环境配置简单，开发方便。

方案二：使用 NVIDIA Jetson Nano B01 模块进行数字识别

利用 Jetson Nano 所具有的 CORTEX-A57 处理器，执行神经网络、对象检测及分割等功能。通过配置 ROS 系统及相关环境，编写滤波及 NC 利用代码，以及制作数据集进行模型训练，可以实现对待分类物体较为准确的识别。

相较于 OpenMV 数字识别，NVIDIA Jetson Nano 环境配置较为繁琐，且对于此项目的任务需求，芯片算力有较大冗余；同时，其使用时的电源需求配置高，且发热量较大，需要额外配备散热组件。考虑到以上因素，我们选择使用 OpenMV 模组来进行数字识别。

2.3 药品检测模块的论证与选择

方案一：使用应变片类传感器

将药品放置在应变片类压力检测薄膜上，当检测出有物体被放置，立即向主控芯片发送电平信号，控制小车启动行驶。

方案二：使用超声测距传感器

将药品放置在普通药箱上，超声测距传感器发送一组特殊的脉冲信号，并在障碍物表面反射后返回，通过传感器接收端进行接收，并通过声音传播时间及声速常数计算得出障碍物距离。当检测到障碍物被放置时，超声测距传感器将电平信号传递给 Arduino Mega2560 对应引脚，在软件层面控制小车启动。

相较而言，方案二所使用的超声测距传感器的反应速度高，安装结构简单便捷，稳定性较高，且在成本较低的情况下，能够满足药品检测的全部需求，所以选择方案二。

3 系统结构与电路设计

3.1 智能送药小车结构设计

由于题目要求小车的长宽高不能超过 25cm*20cm*25cm，再加上赛道的宽度只有 30cm，为了保证转向的时候能够避免碰到黑色实线，我们将电机双驱动置于小车车身中间位置，在底盘左右两侧固定，并在底盘前后固定安装万向轮。这样能够方便小车移动的调试，简化代码量。直行时只需控制两轮 PWM 输出相同，转向时给两轮极性相反的 PWM 输出即可完成小车移动控制。

根据设计要求和实际情况，我们将颜色传感器巡线模块安装在底盘的车头位置，紧靠红线位置但位于红线两侧，识别到白色输出前进信号，一旦一个传感器识别到红色则输出一个调整信号使小车走直线。

我们搭建三层平台，最底层放置走线和电池，第二层放置主控、降压模块和电机驱动并完成接线，最上层我们设置为药品的放置区域，通过超声波检测药品是否已经放置。

OpenMV 模块通过安装支架固定在车头前端，由于识别数字时数字宽度较大，OpenMV 无法全部识别到数字，所以搭配舵机，实时调整摄像头的位置来达到扩大视野范围的目的。

3.2 智能送药小车电路设计

我们的智能送药小车使用 Arduino Mega2560 作为主控芯片，使用电机驱动板 l298n 来驱动电机，使用两节 3.8V 电池供电，并使用一块 LM2596DC-DC 模块来实现电源升压。使用三块 TCS230 颜色传感器进行巡线，并通过 OpenMV Cam H7 Plus 进行数字识别。

3.2.1 系统总体框图与实物图

系统框图

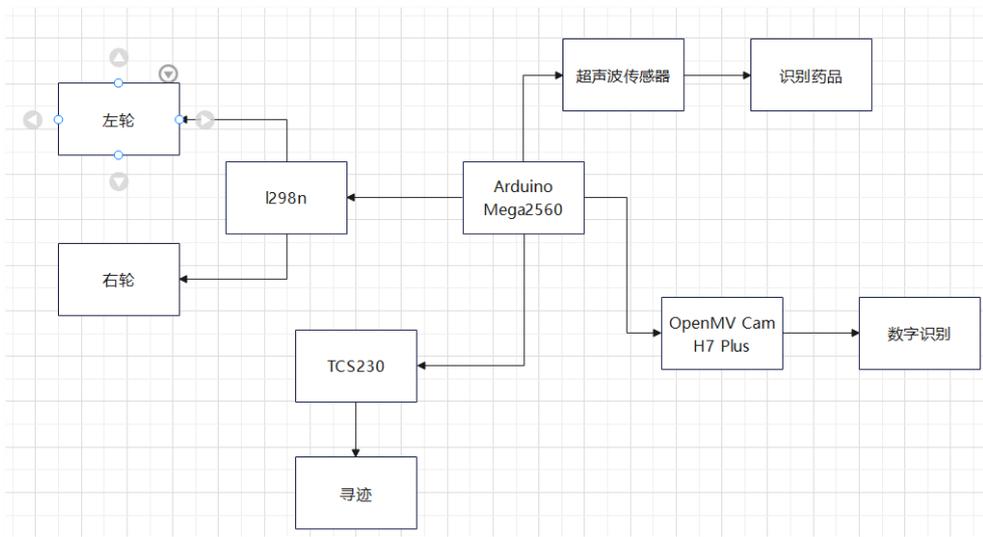


Figure 1 系统框图

实物图

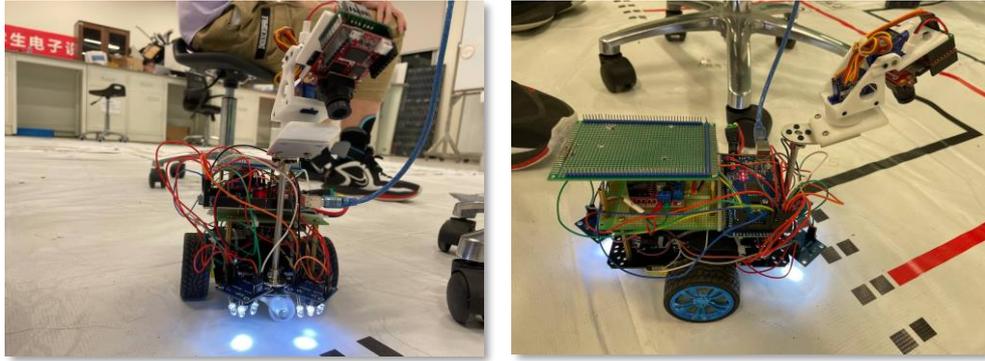


Figure 2 实物图

3.2.2 电机控制电路原理图

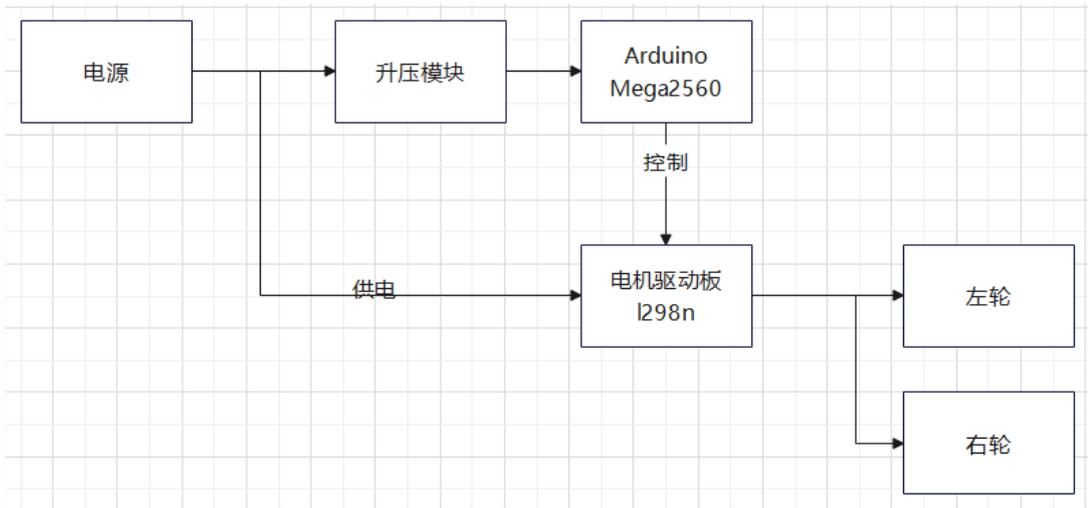


Figure 3 电机控制电路原理图

3.3 程序的设计

在主控的函数主体部分，车子会通过 RGB 颜色传感器对当前所处位置进行识别，判断需要进行直行、左右转弯、暂停等操作。在需要进行上下位机交互时，由主机控制从机进行识别，并接收相应的返回值控制信号。主程序的流程是：首先识别目标病房号数字，在感应到药品放上小车后，开始运送，同时利用巡线功能保持直行，通过与 OpenMV 进行交互获取转向，到达目的地；在感应到药品卸载后，小车后转；之后，按照预定的轨迹进行移动和返回。

4 测试方案与测试结果

4.1 测试方案

4.1.1 软件调试

OpenMV 的视觉识别代码采用软件调试的方法，通过串口发送识别到的数字，在串口调试软件上直接打印出接收到的数字，确保代码的正确性。其中，Python 和 C 通讯，使用了 “str.encode()” 函数，将 str 数据编码为 bytes 数据使 MEGA 能够顺利解读。

4.1.2 硬件软件联调

在搭建好小车的硬件结构之后，安装 OpenMV，Arduino 之后，上位机和下位机通信，上位机传输识别的数字和位置坐标，下位机进行相应动作。

4.2 测试条件

测试条件：黑色和红色电工胶带用于测试巡线和停止；数字字模用于测试数字识别功能。

4.3 测试结果及分析

4.3.1 测试结果

单个小车运送药品到指定的病房并返回到药房测试结果：暂无

4.3.2 结论

根据上述测试数据可以得出以下结论：

1、本次设计采用模块化思想，先后完成了数字检测、电机驱动、巡线识别等模块的设计，较好地完成了题目的基本要求，实现了单个小车运送药品到指定的病房。

2、影响电机驱动的稳定性和控制精度的主要因素是两电机电流分配不均匀，为有效减少电机转速不同影响，采用双电机驱动模块分别给两个电机供电，从而有效地提高了控制精度。

3、为保证识别精度，OpenMV 在训练时要保证数据集中图片包含多个角度，每个数据包包含多张图片。随着数据集增大，识别精度也逐渐提高。在训练过程中，从最初 160 张的低精度到最后约一千余张有较高的实际识别精度，数据识别更准确，系统更加稳定。