

[人工智能普及教育]

把机器人小车升级为无人驾驶小车

文_谢作如 / 温州科技高级中学 王海涛 / 上海人工智能实验室

问题提出

智慧交通是人工智能应用的一个重要场景，无人驾驶则是一种利用各种传感器、控制系统和人工智能算法实现对汽车自主控制的技术，是智慧交通的核心技术。在中小学开设无人驾驶课程，让机械结构、智能控制和深度学习等技术融合起来，是一门很有价值的跨学科课程。但无人驾驶课程开设难度较大，除了对授课教师要求较高外，还需要一系列硬件支持，比如内置支持模型推理能力芯片的小车价格往往达数千元。

考虑到大部分中小学校的创客空间或机器人实验室有现成的可编程机器人小车，而这些小车一般都能支持 2.4G、蓝牙和 Wi-Fi 等遥控。那么，能否给小车配上无线摄像头增加视觉能力，用较低的成本开展无人驾驶课程呢？

技术分析：无人驾驶的实现

常见无人驾驶小车功能分析

据不完全统计，为中小学开发无人驾驶课程核心器材（小车）的企业不多。笔者从转向、感知、控制等角度对中小学无人驾驶小车的技术进行了分析。最常见的转向方式是两轮差速和麦克纳姆轮，部分高端的小车会采用阿克曼转向，即类似真实汽车，用方向盘（舵机控制）形式转向。摄像头是必备的感知设备，外加一些红外避障传感器，高端的小车还会配置激光雷达，使用 SLAM 技术。高端小车会使用 ROS 控制系统，而对于低龄段学生，用 Python 结合 GPIO 或 pinpong 库，也是常见的选择。一般都会使用具备一定算力的主板，以树莓派和 Jetson Nano 为主，有的会用高端国产芯片；相对低端的产品会使用 Arduino、ESP32 类主板，使用智能摄像头推理，但大家一般也不会把这些小车看作无人驾驶小车。

无人驾驶的低门槛实现

从任务执行角度看，无人驾驶小车的控制主要包括车速、方向和车灯（有些会加鸣笛）。而环境感知部分除去高端 SLAM 建图（需要激光雷达），仅需对车道、交通标志和行人进行识别。正是这些识别功能对小车主板提出了较高的要求，如果对小车实时反馈要求不高，则可用远程推理结合遥控方式实现，可显著降低“无人驾驶”的门槛。其中，远程推理是指用电脑（服务器）对摄像头画面进行推理。如图 1 所示，可以选择无线摄像头将画面传输到普通电脑，根据推理结果以无线方式发送控制信号，无线摄像头既可以装在小车上，也可以装在小车运动的场地上方。如装在小车上，可以做第一视角的无人车，跟现有的无人车功能非常类似；如装在小车运动场地上方，则可返回小车的位置信息。当然，两种方式也可同时存在。

项目测试：做一辆无人驾驶小车

笔者在学校创客空间选择了一款现有的小车进行测试。其中，小车用“麦昆”，无线摄像头选择了 ESP32-CAM。

可编程遥控小车的选择

麦昆小车套件仅仅提供了底盘，控制板可采用 micro:bit 和掌控板，性价比高。其中，micro:bit 支持 2.4G

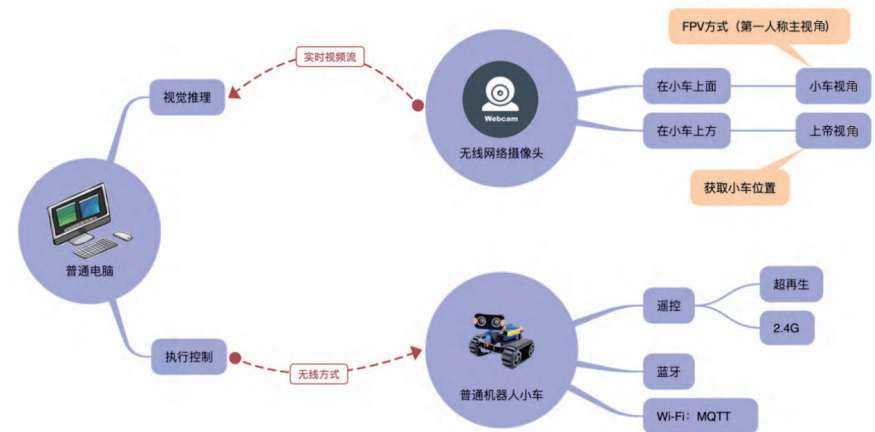


图 1 普通小车实现无人驾驶工作流程

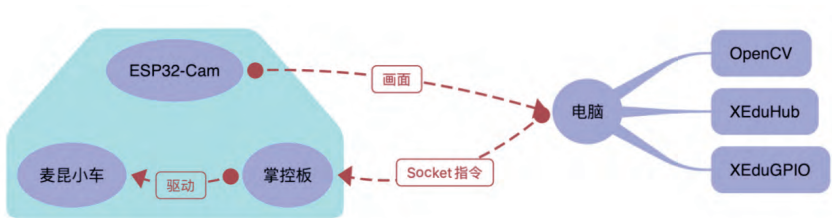


图2 无人驾驶的技术实现方案

的控制，可以与其他 micro:bit 板子通讯。掌控板则支持 Wi-Fi，可采用 MQTT、HTTP 或 Socket 通讯，使用更加灵活。

ESP32-CAM 是基于 ESP32 芯片设计的一款开源硬件，价格不到 30 元，融合了 Wi-Fi 和蓝牙通讯功能，适用于需要拍照、录像、图像处理等应用场景。笔者还特意刷写了 1 个固件，能通过串口配置 ESP32-CAM 的 Wi-Fi 信息，然后以 HTTP 方式就能获取摄像头的画面信息。

无人驾驶技术实现

为了用最简洁的方式控制小车，笔者放弃了常见的 MQTT 协议，选择了 Socket，因为 MQTT 还需单独部署 MQTT 服务器。具体的工作流程如图 2 所示，电脑端借助 OpenCV 获取 ESP32-CAM 的画面，经过 XEduhub 模型推理后，借助 XEduGPIO 库发送指令给掌控板，掌控板再驱动小车，麦昆小车就成了一辆拥有强悍算力的智能小车。

流程图中的 XEduGPIO 库是为活动专门编写的，内置了 Socket 连接的各种协议。核心类 CarComm 提供了 speed、servo、led 和 stop 这 4 种最基本的方法，分别用于驱动小车前进、舵机转动、改变 Led 状态和停止运动。在 XEduGPIO 库的支持下，4 行代码即可实现小车的驱动。当然，掌控板也需进行相应的 Socket 编程，写好相应的固件，刷入即可运行。

```
Python
from xedugpio import *      # 导入库
host = "192.168.31.40"      # TCP 服务端的 IP 地址
car = CarComm(host)         # 实例化小车对象
car.speed(speedL,speedR)    # 设置小车速度，驱动小车
car.servo(90)               # 设置小车舵机的角度
car.led(1)                   # 设置小车 led 的状态
car.stop()                   # 小车停止运动
```

在电脑端读取 ESP32-CAM 画面需要用 OpenCV，参考代码如下。画面推理则需先训练模型，再借助 XEduHub 推理。因 MMEdu 从数据标注到模型训练已经提供了完整的操作流程，这里不再展开，请参考 XEdu 文档。

```
Python
import cv2,urllib.request
import numpy as np
url = 'http://192.168.50.50/320x240.jpg' # 修改摄像头 IP 地址
cap = cv2.VideoCapture(url)
```

```
# 检查 IP 摄像头流是否成功打开
if not cap.isOpened():
    print(" 无法打开 IP 摄像头流 ")
    exit()
while True:
    img_resp=urllib.request.urlopen(url)
    imgnnp=np.array(bytearray(img_resp.read()),dtype=np.uint8)
    # 将图像水平翻转
    frame = cv2.imdecode(imgnnp,-1)
    frame = cv2.flip(frame, 1)
    cv2.imshow("Capturing",frame)
    cap.release()
    cv2.destroyAllWindows()
```

活动设计：从人车合一到无人驾驶

2024 年 1 月，全国青少年人工智能创新实践活动在上海南洋中学举办，以上述无人驾驶小车为基础，设计了 1 个线下活动——八段锦巡线推物挑战赛。参加活动的学生需独立完成小车的机械拼装、配置 Wi-Fi 信息，然后测试“人车合一”任务，再挑战“无人驾驶”任务。小车活动地图如图 3 所示。

“人车合一”任务指学生在电脑摄像头前，通过自己的人体姿态控制小车移动和铲子起落，按跑道轨迹将物体分别移至 3 个目的地。“无人驾驶”任务则要求使用电脑读取小车上的摄像头画面，根据推理结果控制小车自主完成八段锦姿态分类任务，小车能合理规划路径，完成寻找地图中与目标八段锦姿态相同的立牌（竖立呈现的图片）并转动铲子。活动现场如图 4 所示。



图3 无人驾驶小车地图设计

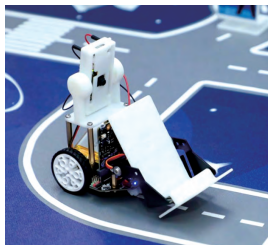


图4 无人驾驶活动现场

总结与反思

50 多名来自全国各地的学生参加了上述活动，他们中大部分是第一次接触无人驾驶，表现出了巨大的热情，边学习边探究，收获颇丰。一些现场观摩的教师也颇为惊喜，他们没有想到无人驾驶还可以用这种低成本的方式实现，并表示要尽快实施并设计一套完整的人工智能课程。实际上，这套低成本方案不仅可用于无人驾驶，还可用于无人机的智能巡检、智慧农场的虫害监测、保护区的生态监测等有趣的课程内容。