

第十一届全国大学生光电设计竞赛（东南赛区）

技术方案报告

编 号：

参赛题目：“迷宫寻宝”光电智能小车

第十一届全国大学生光电设计竞赛东南部区组委会制

二〇二三年六月

开源方案-仅供参考
<https://github.com/plutocyy158/2023-guang-sai.git>

参赛题目：“迷宫寻宝”光电智能小车

一. 研究目标

近年来，光电技术、视觉技术与图像处理技术在多个领域成为热点应用，尤其在智能小车领域更是引发了广泛关注。本研究旨在设计一款集成视觉处理等先进技术的高精准辨识“迷宫寻宝”光电智能小车，其具备自动行驶、自主寻迹避障、图像读取和图像形状识别等功能，能够通过实时路径规划，有效提高寻宝效率和导航能力。同时，通过优化寻宝策略和智能算法，小车只需搜索 3~6 个宝藏点位置即可成功寻找到 3 个真宝藏，而无需搜索全部的 8 个宝藏点，从而避免了重复移动和检索，大幅度提高寻找宝藏的效率。此外，为进一步提升效率，小车还增添了智能复位记忆功能，避免重复移动和检索。

综上所述，该光电智能小车可实现高精准辨识、自动行驶和智能寻宝等功能。它能准确到达迷宫内宝藏点位置，并正确识别真假宝藏，高效地完成任务并顺利抵达终点。此外，这项技术可以推广到其他应用场景，如可为智能交通、智能物流以及自动导航等领域提供重要支撑和启示，为科技进步和社会发展注入新活力。

二. 研究方案

针对“迷宫寻宝”光电智能小车需具备图像识别、路径规划、自动行驶、避障循迹以及光电传感等诸多功能要求，团队对小车各功能模块进行自主设计。整个光电智能小车主要由智能车和识图装置两部分构成，系统框图如图 1 所示。

其中，智能车主要由主控模块 MCU、电机驱动模块、循迹模块、视觉模块以及电源模块组成。主控模块 MCU 采用 STM32 单片机，其作为光电智能小车的核心控制中心，负责管理和协调其余模块的正常运行；电机驱动模块采用减速电机和 TB6612，搭配姿态传感器，稳定且精准实现小车的前进和转向功能；循迹模块采用十二路灰度传感器，负责识别地面上的轨迹引导线，确保小车按预定路径行驶；视觉模块采用 Open MV，用于识别宝藏形状等关键信息；电源模块采用 12 V 电源，为各模块提供稳定的电力支持，以确保系统正常运行。

识图装置主要由摄像头、Orange pi 和显示屏组成。通过摄像头对藏宝图进行拍摄，利用 Open CV 图像处理算法将获取到的图像信息进行图像矫正等初步

处理，然后利用 YOLOv5 目标检测算法对宝藏点进行识别，并通过换算获取其位置信息。然后利用 A*路径规划算法进行实时路径规划。显示屏则用于显示拍照、识别与小车通信情况等信息。

此外，采用串口通信技术实现智能车和识图装置之间的无线通信，将处理后的图像信息传递给智能车，实现智能车与识图装置之间的即时信息共享。通过一键操作，智能车能够自动完成迷宫、宝藏等的识别、位置校正和分析。

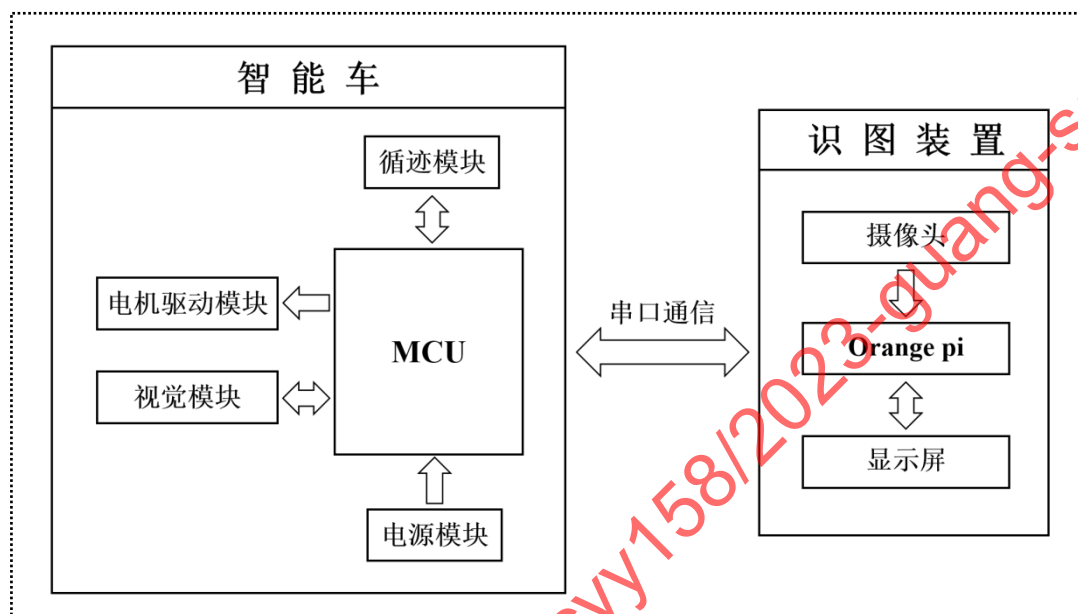


图 1 小车整体系统结构框图

三. 技术路线及可行性分析

3.1 技术路线

团队设计的“迷宫寻宝”光电智能小车的实现技术路线如下图 2 所示，主要分为识别藏宝图、建立坐标系及划分宝藏区域、路径规划、区域宝藏寻觅，判断宝藏真假等，具体介绍如下。

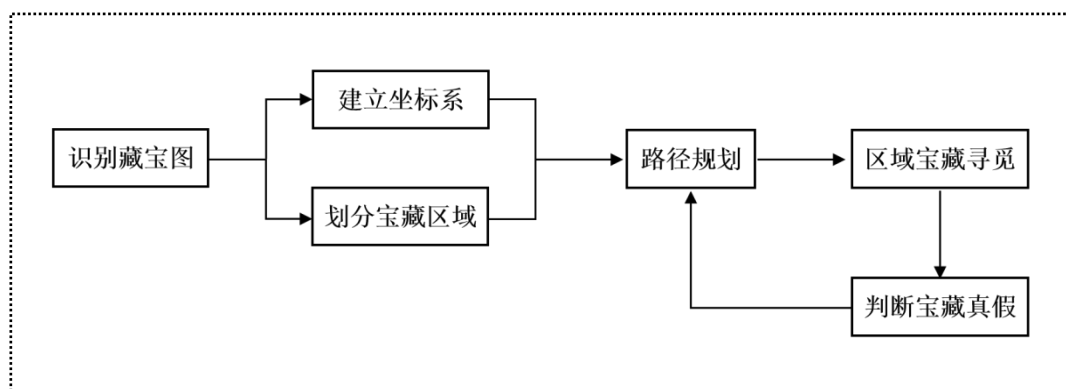


图 2 智能小车实现技术路线图

1、**识别藏宝图**：首先打开电源并启动配备摄像头的识图装置，使用该摄像头拍摄藏宝图，并通过 Open CV 算法对原始图像进行透视变换和矫正等初步处理，从而获得清晰规整的处理图像以便进行下一步识别。将处理后的图像数据传递给 Orange Pi，进行图像分析和处理。详细具体如下图 3 所示。

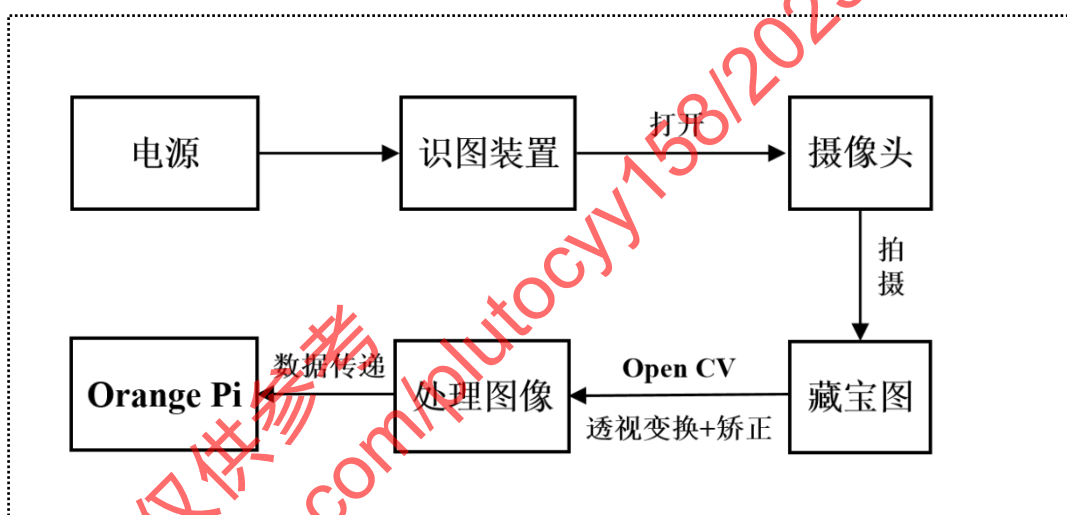


图 3 识别藏宝图流程图

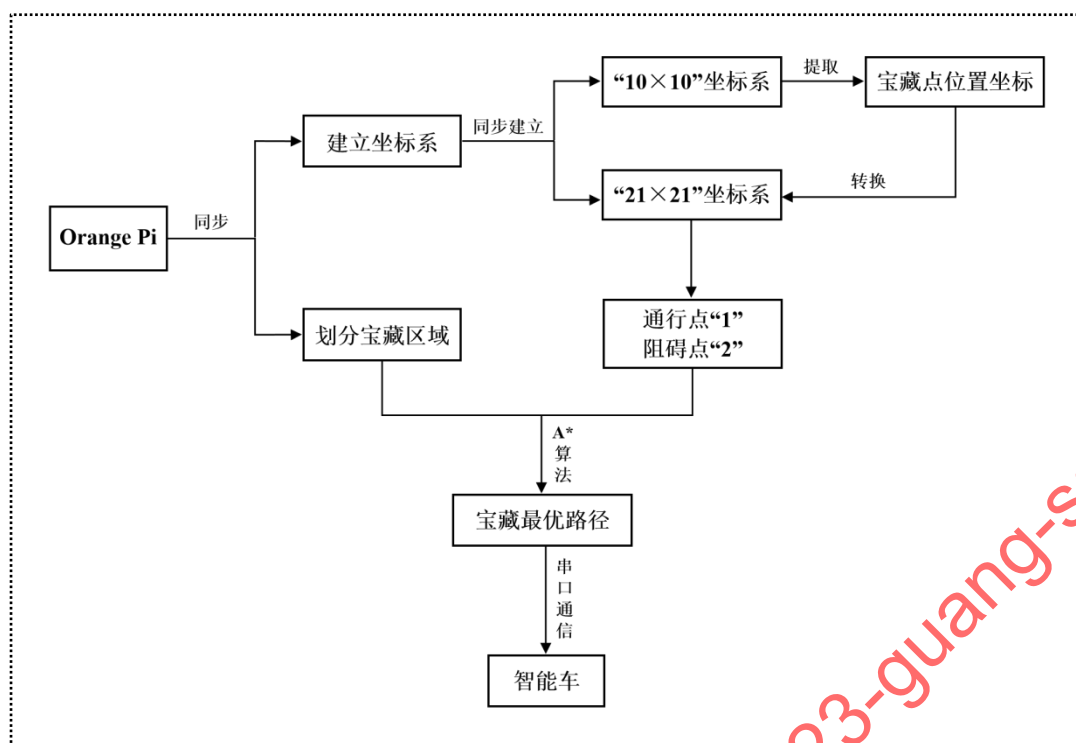


图 4 坐标系建立及路径规划流程图

2、同步建立坐标系：如图 4 所示，Orange Pi 接收到处理的图像数据信息后，利用 YOLOv5 目标检测算法对迷宫内的宝藏点进行识别，并同时建立坐标系。基于处理后的图像数据，Orange Pi 将藏宝图分割成大小相等的“10×10”坐标系（不包含障碍物）和“21×21”坐标系（包含障碍物）。然后，在“10×10”坐标系上提取宝藏点的位置坐标，并将其转换为“21×21”坐标系的像素坐标。

3、划分宝藏区域：接着，在 Orange Pi 接收并处理图像数据以及建立坐标系的过程中（如上图 4），识图装置还会对藏宝图进行区域划分，同时将每个区域内的宝藏点位置将进行先后排序，每个区域内随机摆放两个宝藏。具体的，藏宝图的左下区域被定义为第一区域，右下区域为第二区域，右上区域为第三区域，左上区域为第四区域（如图 5 所示）。

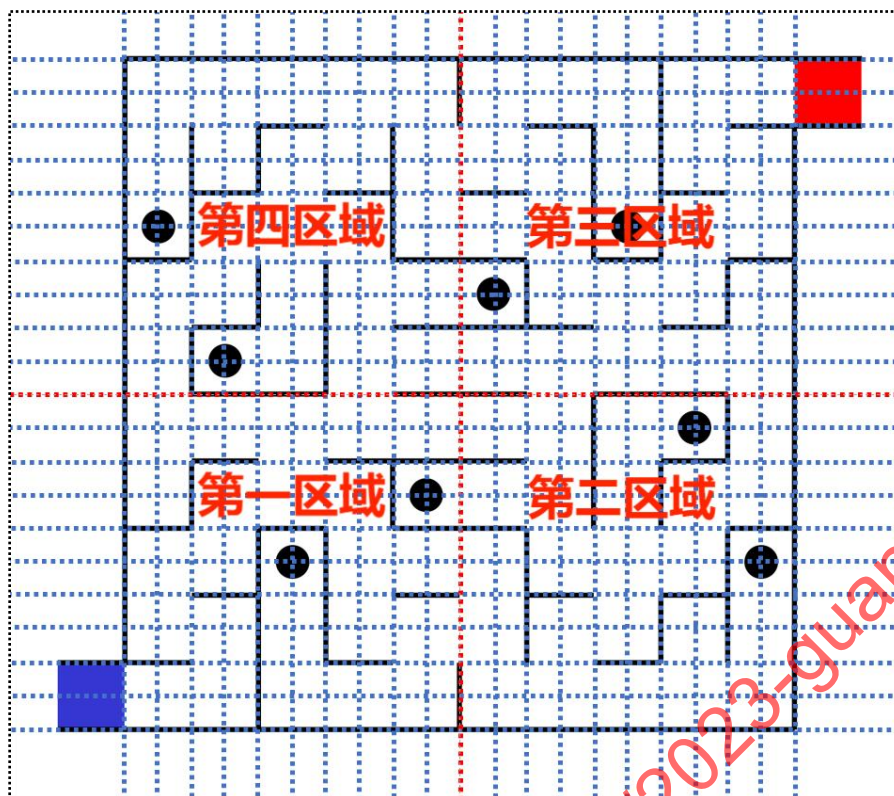


图 5 寻宝区域排序图

4、**路径规划**：在“21×21”矩阵坐标系上，根据宝藏点转化后的像素坐标，对整张藏宝图进行设定，即设定通行点“1”和阻碍点“2”，分别表示小车允许通过和不允许通过，如图6所示。同时，结合划分的宝藏区域，使用A*路径规划算法规划小车寻找宝藏的最优路径，并将其数据信息通过串口通信技术传递给智能车。具体过程如上图4所示。

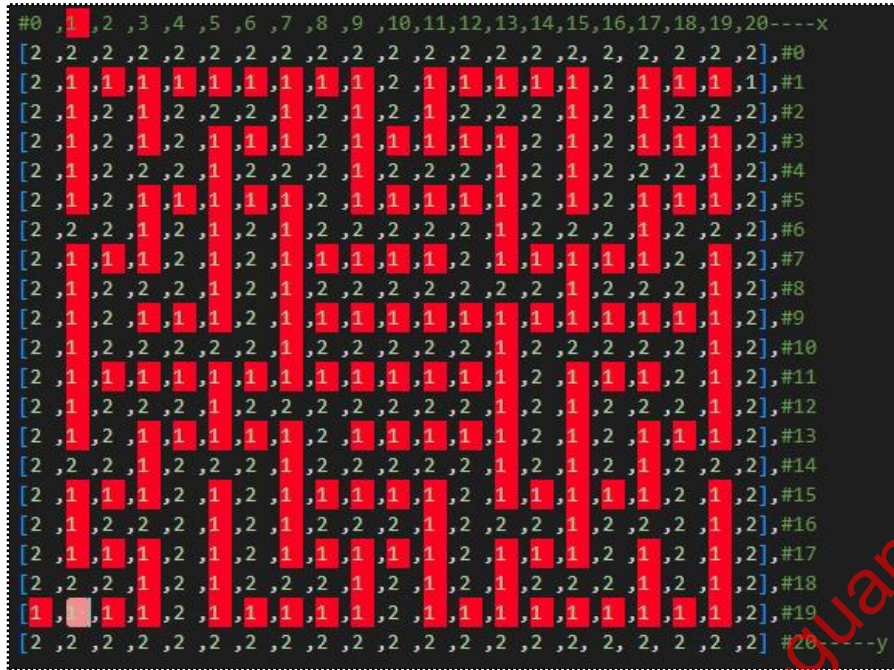


图 6 通行点与阻碍点设定图

5、区域宝藏寻觅及宝藏真假判断：如图 7 所示，智能车接收到数据后，将按序前往各个宝藏区域寻找宝藏。如果在当前区域找到真宝藏点，根据“一个区域只有红蓝两个宝藏”原则，小车不再继续寻找下一个宝藏点，而是直接进入下一个区域。同时，根据“己方与对方宝藏、两种伪宝藏在藏宝图内各自呈对称分布”原则，根据已找到的真宝藏点位置，将对称位置的宝藏点设定为阻碍点“2”，智能车之后不在前往该对称的宝藏点位置，从而提高寻宝效率。

接着，小车在按序前往各个宝藏区域寻找宝藏时，采用 Open MV 及 TensorFlow Lite 深度学习模型对宝藏形状进行识别和真假判定，智能车通过串口通信技术将宝藏真假信息发送给 Orange Pi，当 Orange Pi 接收到真宝藏信息后，会重新规划下一个区域的宝藏点路径，并通过串口通信技术将路径信息传递给智能车。当 Orange Pi 接收到假宝藏信息后，会重新规划同一区域另外一个宝藏点的路径，并通过串口通信技术将路径信息传递给智能车。通过不断循环执行上述过程，直到找到 3 个真宝藏，小车结束寻宝。

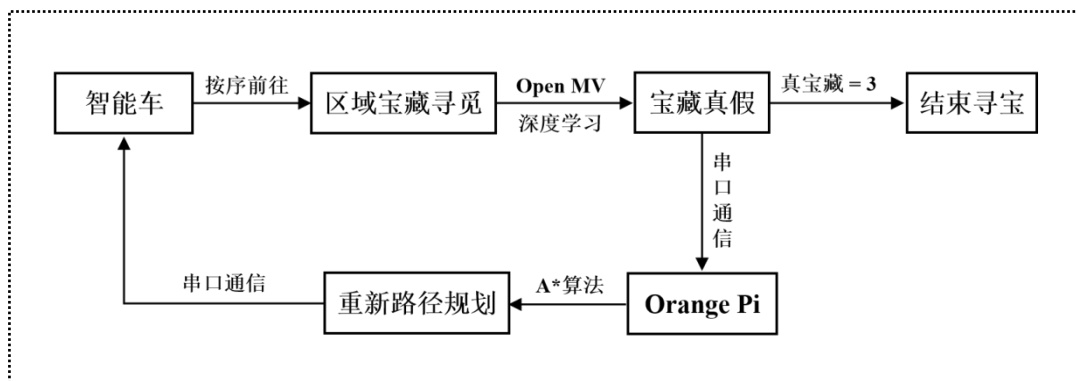


图7 区域宝藏寻觅及宝藏真假判定流程图

综上所述，该“迷宫寻宝”光电智能小车实现技术路线过程详细如下图8所示。

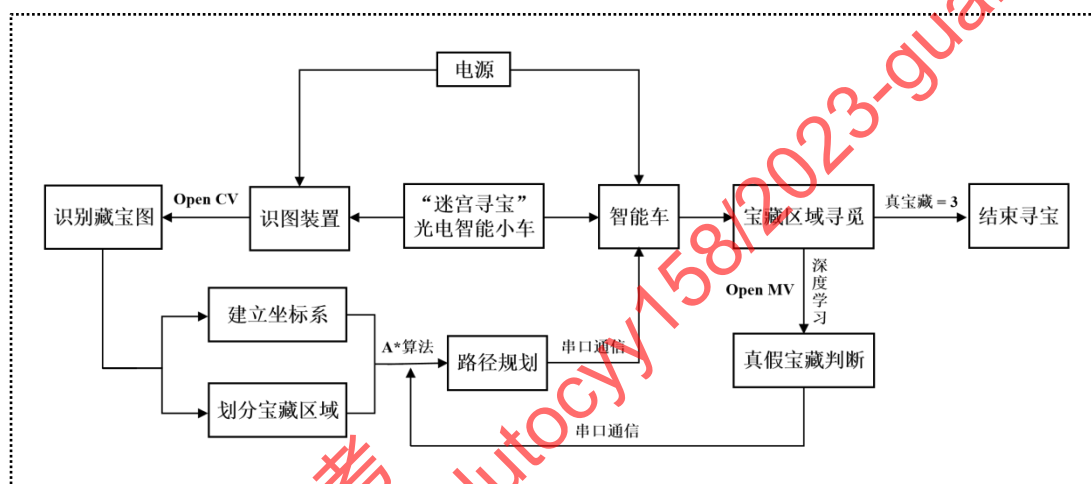


图8 智能小车实现技术路线详细流程图

3.2 可行性分析

本方案综合运用了图像处理、路径规划和实时通信等技术。通过智能车端的图像处理来识别迷宫中的宝藏点，并利用路径规划算法找到最优路径以实现寻找宝藏点的目标，其实现具备一定的可行性。详细分析具体如下。

1、**图像处理：**利用摄像头及 Orange Pi 对迷宫图像进行处理和颜色块识别。这些技术已在许多实际应用中得到验证，具备较高的可行性。

2、**宝藏点识别：**采用 YOLOv5 目标检测算法对宝藏点进行识别，以确定宝藏点的位置。该算法在计算机视觉领域有着出色的表现，通过训练和调优，能够在迷宫中准确地识别出宝藏点的位置。

3、**路径规划**：采用 A* 路径规划算法对迷宫地图进行路径规划，以寻找小车到达宝藏点的最优路径。A* 算法是一种启发式算法，它利用启发信息寻找最优路径，对环境反应迅速，搜索路径直接，因此被广泛应用于路径规划问题。其已在许多应用中得到验证，具备较高的可行性。

4、**实时通信**：使用串口通信技术进行识图装置与智能车之间的实时数据传输。串口通信具有低功耗、简单易用、可靠稳定等特点，已经广泛应用于无线通信领域，因此该方案的可行性较高。

5、**区域宝藏寻觅**：根据预设顺序，智能车依次前往每个区域寻找宝藏点。通过设定通行点“1”和阻碍点“2”，并使用 A* 算法寻找最优路径，可以有效地寻找宝藏点。该策略在理论上是可行的，但实际操作中可能需要根据不同的迷宫结构进行调整和优化。

6、**宝藏真假判断**：采用 Open MV 及 TensorFlow Lite 深度学习模型对宝藏形状进行识别和真假判定，通过合理的技术选型、数据集构建和训练优化，能够提高宝藏识别和真假判定的准确性和实时性，为智能车寻宝任务提供可靠的支持。

综上，该方案整体上具备可行性。在实施过程中需要考虑硬件设备、算法优化和系统稳定性，并进行相应测试和调试以确保方案有效。

四. 解决的关键问题

1、利用自反馈循迹算法实现精准稳定的小车运行控制

光电智能小车沿规定的宝藏路径稳定循迹时，往往会受到诸如环境光线、地面不平等因素影响，因此团队采用十二路灰度传感器控制小车运动方向和循迹。其设计框图如图 9 所示。

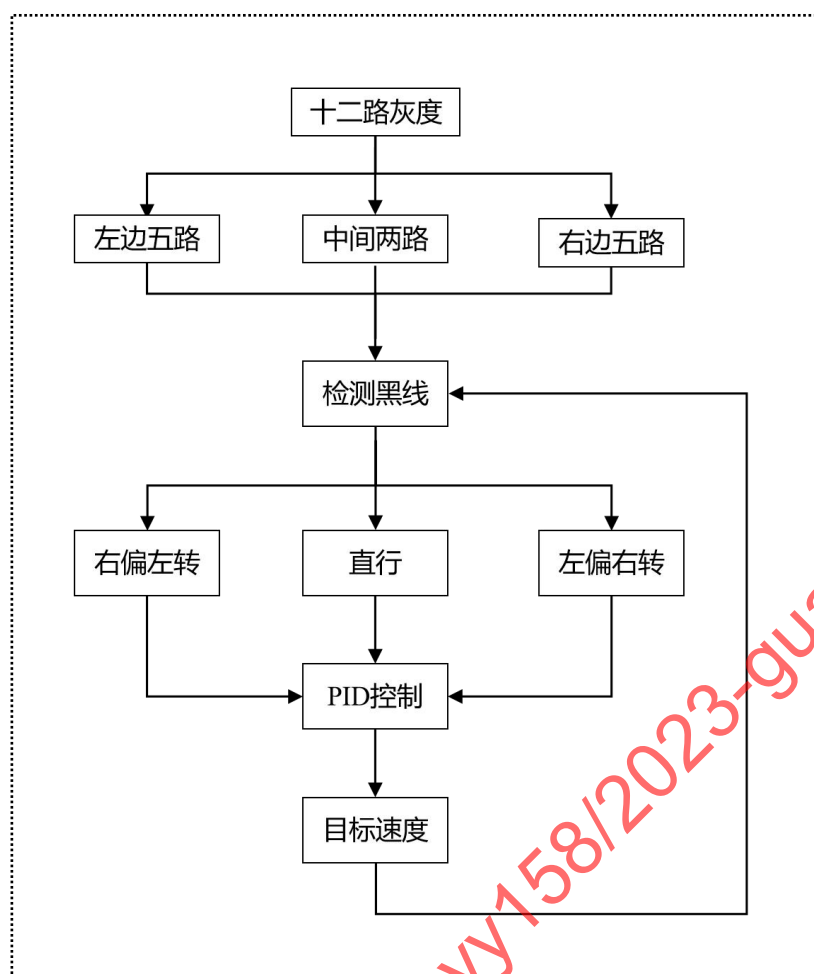


图 9 灰度循迹设计原理框图

2、利用 TensorFlow Lite 深度学习模型精准识别宝藏真假

团队采用 Edge Impulse 提供的 TensorFlow Lite 深度学习模型方案，自行构建神经网络。通过大量的训练数据和轮数，神经网络能够更好地学习和理解不同宝藏图像的特征，从而实现高精度的宝藏真假识别。如图 10 所示，其识别成功率达到了 95.3%。

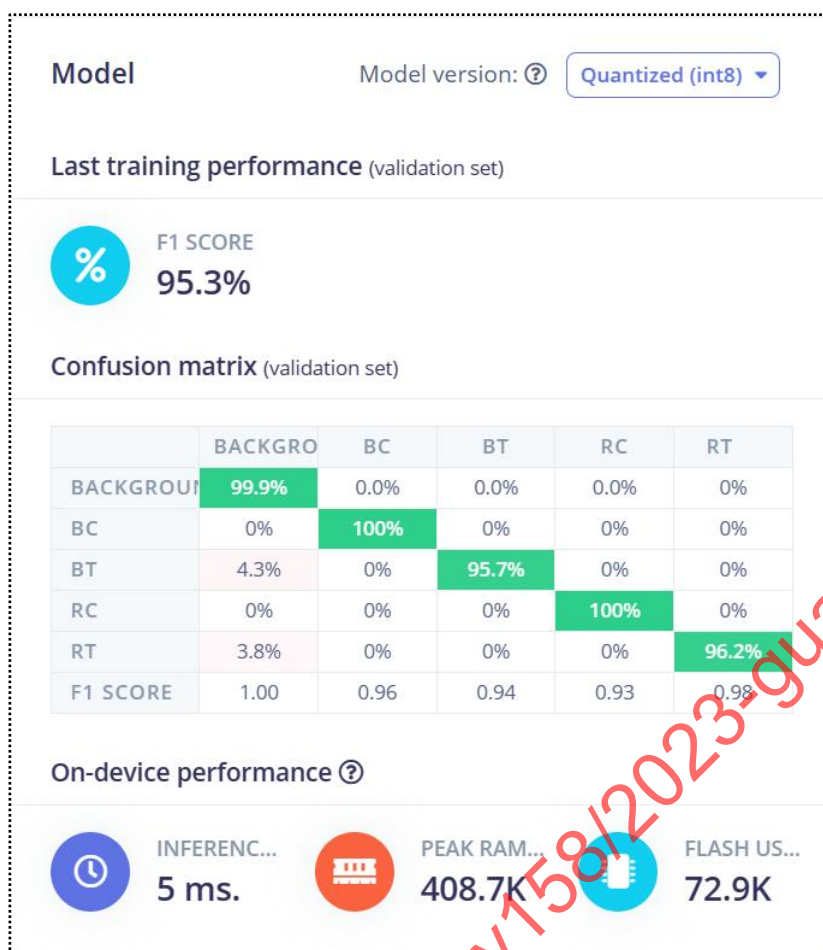


图 10 TensorFlow Lite 深度学习模型识别测试结果图

五. 特色与创新点

5.1 特色

团队设计的“迷宫寻宝”光电智能小车由智能车和识图装置两部分组成，具有先进的循迹、识别和任务执行能力。智能车采用模块化设计，包括主控模块 MCU、电机驱动模块、循迹模块、视觉模块和电源模块，这种设计使得各个功能模块紧密结合，可大大提高系统的灵活性和可维护性。识图装置由摄像头、Orange pi 和显示屏构成，其可为小车提供强大的图像处理和分析能力，并显示拍照、识别与小车通信情况等信息。这些优秀特色使得该小车不仅能够快速而准确地完成迷宫寻宝任务，还具备广泛的应用潜力。

5.2 创新点

1、**高精度辨识**：采用 Edge Impulse 提供的 TensorFlow Lite 深度学习模型自行构建神经网络，并搭配 Open MV，智能车能够在行驶过程中快速而准确地识别各类宝藏的真假，从而节省寻找宝藏的时间。

2、**实时路径规划**：采用 A* 路径规划算法对宝藏进行实时路径规划。将识别到的宝藏定为阻碍点，只有前往寻找的宝藏点才会定为通行点，避免前往某一宝藏点过程中经过另一宝藏点，出现误撞的情况，智能车会根据剩下的宝藏重新规划路径，以确定小车到达剩余宝藏点的最优路径，有效提高小车寻宝效率和导航能力。

3、**智能复位记忆功能**：智能小车配备一键复位功能。当小车在未找到全部宝藏而重新出发时，其将前往尚未寻找的宝藏点，而已找到的宝藏点将不再重复前往，从而有效避免重复移动和检索，提高寻找宝藏的效率。

4、**智能化宝藏优选**：通过优化寻宝策略及智能算法，小车仅需前往 3~6 个宝藏点位置，而非全部 8 个。这种优化的选择策略也可避免重复移动和检索，显著提高小车寻找宝藏的效率。

六. 其他说明

本团队设计的“迷宫寻宝”光电智能小车搭配有六个功能按键，以方便小车进行调试，具体功能介绍如下表 1 所示。

表 1 小车按键具体功能介绍表

按键	具体功能
1	开启/关闭 OLED 显示屏
2	上下移动选择菜单栏
3	确认操作
4	返回上一界面
5	一键启动小车
6	复位操作

七. 结论

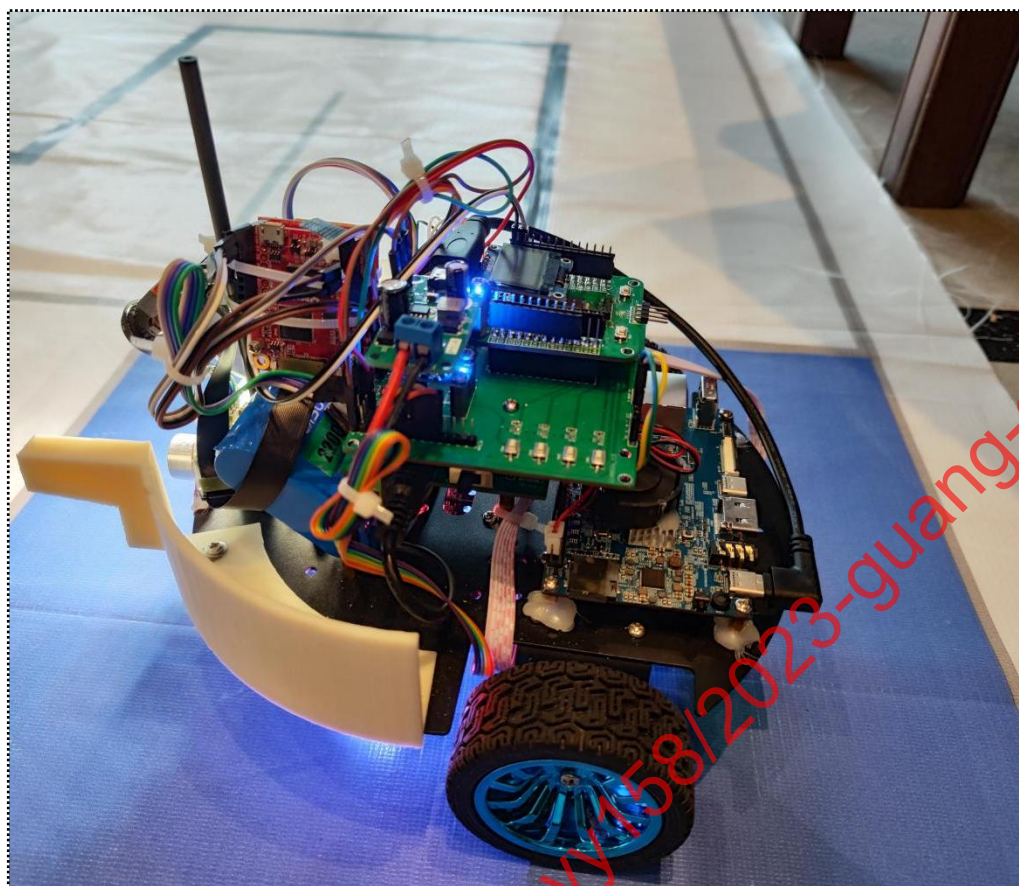
本团队设计了一款集成视觉处理等先进技术的高精准辨识“迷宫寻宝”光电智能小车，其具备自动行驶、自主寻迹避障、图像读取和图像形状识别等功能，分智能车和识图装置两部分。具体而言，智能车采用模块化设计，包括主控模块MCU、电机驱动模块、循迹模块、视觉模块和电源模块。而识图装置由摄像头、Orange pi 和显示屏构成，为小车提供强大的图像处理和分析能力。

总而言之，该光电智能小车不仅具有高精准辨识、实时路径规划、智能复位和智能寻宝等特点，而且该项技术还可以推广到其他应用场景，如可为智能交通、智能物流以及自动导航等领域提供重要支撑和启示，为科技进步和社会发展注入新活力。

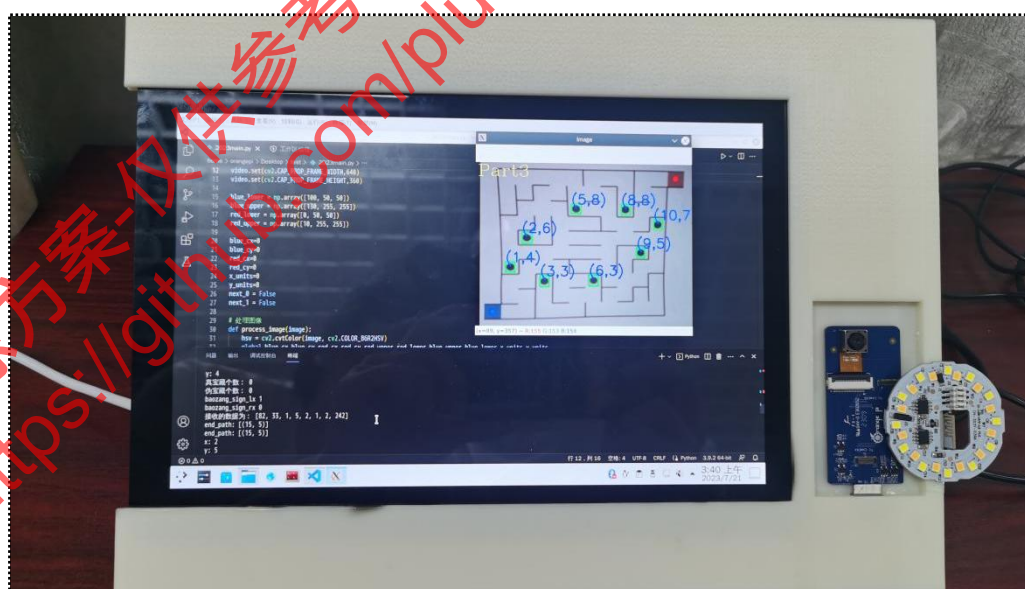
开源方案-仅供参考

<https://github.com/plutocyy158/2023-guangzhou-sai.git>

附录 1 光电智能小车整体实物图

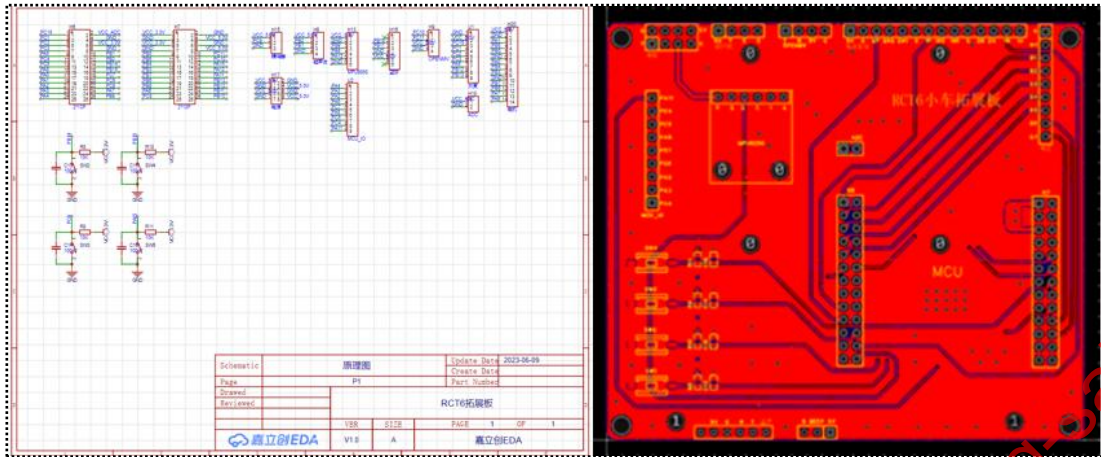


附图 1 智能车+识图装置整体实物图

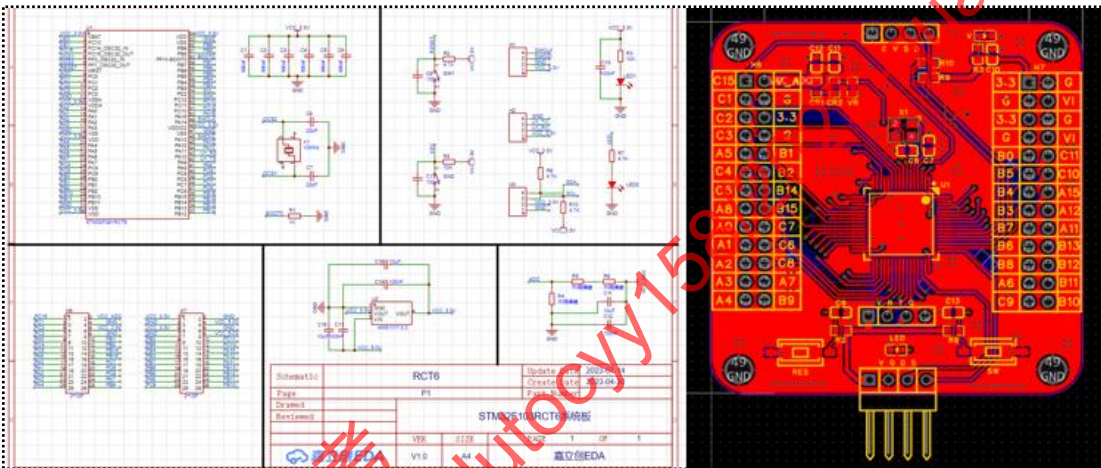


附图 2 识图装置部分外设实物图

附录 2 寻宝小车硬件设计图



附图 3 RCT6 拓展板原理图及 PCB



附图 4 STM 最小系统板原理图及 PCB

附录 3 智能车主控程序代码

```
if (start_flag == 1)    //到时候这一段代码。放在定时器里面，会不会效果更好
{
    switch(task_flag)    //初始为0
    {
        case 0:
            TASKKS();    //自动开始循迹到(3,4)
            break;

        case 1:
            TASK_XY();
            break;

        case 2:
            TASK();
            break;

        case 3:
            TASK_ZZ();    //(9,7)自动出去
            break;

        case 4:    //没有识别到宝藏
            TASKBACK ();
            break;

        case 5:    //拐角处没有识别到宝藏
            TASKBACK_Turn ();
            break;

        default:
            Line_flag = 0;
            Stop_Flag = 0;
            set_motorL_disable();    //关闭电机?
            set_motorR_disable();
            break;
    }
}
```

附图 5 智能车部分主控程序代码图

附录 4 识图装置程序代码

```
while open:
    ret, image = video.read()
    if image is None:
        break
    if ret == True:
        # 处理图像
        processed_image = process_image(image)

        # 显示图像和坐标
        new_image = get_perspective_transform(processed_image)
        new_image = process_image(new_image)
        huaxian_image = np.copy(new_image)
        # 绘制直线和坐标点
        cv2.line(processed_image, (blue_cx-x_units, red_cy-y_units), (blue_cx-x_units, blue_cy+y_units), (0, 0, 0), 3)
        cv2.line(processed_image, (blue_cx-x_units, red_cy-y_units), (red_cx+x_units, red_cy-y_units), (0, 0, 0), 3)
        cv2.line(processed_image, (red_cx+x_units, blue_cy+y_units), (red_cx+x_units, red_cy-y_units), (0, 0, 0), 3)
        cv2.line(processed_image, (red_cx+x_units, blue_cy+y_units), (blue_cx-x_units, blue_cy+y_units), (0, 0, 0), 3)
        # 画坐标点
        #cv2.line(huaxian_image, (blue_cx, blue_cy), (red_cx, blue_cy), (0, 0, 0), 3)
        #cv2.line(huaxian_image, (blue_cx, blue_cy), (blue_cx, red_cy), (0, 0, 0), 3)
        #cv2.line(huaxian_image, (blue_cx, red_cy), (red_cx, red_cy), (0, 0, 0), 3)
        #cv2.line(huaxian_image, (red_cx, red_cy), (red_cx, blue_cy), (0, 0, 0), 3)

        for b in range(0,11):#横
            cv2.line(huaxian_image, ((blue_cx+int(x_units/2)), (red_cy-int(y_units/2))+(b*y_units)), ((red_cx-int(x_units/2)), (red_cy-int(y_units/2))+(b*y_units)), (0, 0, 0))
        for v in range(0,11):#竖
            cv2.line(huaxian_image, ((blue_cx+int(x_units/2))+(v*x_units), (blue_cy+int(y_units/2))), ((blue_cx+int(x_units/2))+(v*x_units), (red_cy-int(y_units/2))), (0, 0, 0))
        cv2.putText(huaxian_image, "Q" -> "paishe", (10,40), cv2.FONT_HERSHEY_COMPLEX, 1.0, (0,255,255), 2)

        #results = model(new_image)
        #results_ = results.pandas().xxy[0].to_numpy()

        cv2.namedWindow('Image', 0)
```

附图 6 识图装置部分程序代码图

开源方案-仅供参考
<https://github.com/plutocyy158/2023-guang-sai.git>