

基于图像处理的辨识智能人形机器人控制系统设计

叶禹铖¹, 张智闻²

¹盐城工学院化学化工学院, 江苏 盐城

²盐城工学院电气工程学院, 江苏 盐城

收稿日期: 2024年12月31日; 录用日期: 2025年2月25日; 发布日期: 2025年3月5日

摘 要

在多样化的机器人应用场景中, 智能人形机器人控制系统已成为当前研究的热点。通过结合数字图像处理技术、计算机视觉原理以及模式识别方法, 本文实现了一款高效能、自动化程度高的辨识智能人形机器人控制系统设计。在硬件设计方面, 本文结合了伺服电机的精密控制与力矩输出特性, 实现了机器人关节的平滑连贯动作。在软件设计方面, 软件整体程序采用模块化函数的编程思想, 根据硬件设计按照功能不同分好模块进行相应子函数模块编程, 主要分为三个时期: 系统各模块的初始化、读取MPU6050传感器的数据、动作组程序控制、辨识功能实现(颜色识别、人脸识别等)。测试结果显示, 该系统在对颜色目标的追踪和人脸识别上均达到了90%以上的准确率。该智能人形机器人具备一定精度的辨识能力, 运行稳定, 功能完善, 具有一定的应用前景。

关键词

人形机器人, 伺服电机, 无线WIFI技术, 颜色追踪, 人脸识别

Design of Identification Intelligent Humanoid Robot Control System Based on Image Processing

Yucheng Ye¹, Zhiwen Zhang²

¹School of Chemistry and Chemical Engineering, Yancheng University of Technology, Yancheng Jiangsu

²School of Electrical Engineering, Yancheng University of Technology, Yancheng Jiangsu

Received: Dec. 31st, 2024; accepted: Feb. 25th, 2025; published: Mar. 5th, 2025

文章引用: 叶禹铖, 张智闻. 基于图像处理的辨识智能人形机器人控制系统设计[J]. 人工智能与机器人研究, 2025, 14(2): 247-258. DOI: 10.12677/airr.2025.142025

Abstract

In diverse robot application scenarios, intelligent humanoid robot control systems have become a current research hotspot. By combining digital image processing technology, computer vision principles, and pattern recognition methods, this paper realizes the design of an efficient and highly automated identification intelligent humanoid robot control system. In terms of hardware design, this article combines the precision control and torque output characteristics of servo motors to achieve smooth and coherent movements of robot joints. In terms of software design, the overall software program adopts the programming concept of modular functions. According to the hardware design, modules are divided into different functions for corresponding sub function module programming. It is mainly divided into three stages: initialization of system modules, reading data from MPU6050 sensors, control of action group programs, and implementation of recognition functions (color recognition, face recognition, etc.). The test results show that the system has achieved an accuracy rate of over 90% in tracking color targets and facial recognition. This intelligent humanoid robot has a certain degree of accuracy in identification, stable operation, complete functions, and has certain application prospects.

Keywords

Humanoid Robot, Servo Motor, Wireless WIFI Technology, Color Tracking, Facial Recognition

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在机器人技术迅猛发展的当下,智能人形机器人在工业、服务等领域扮演着越来越重要的角色[1]-[3]。图像处理技术作为机器人感知环境的关键技术之一,通过对环境中的视觉信息进行处理和分析,实现对目标的识别、追踪与交互[4][5]。目前,利用单目视觉传感器进行目标识别和追踪已经在实际应用中展现了低成本与便捷性,例如通过高斯滤波、图像空间变换来提高识别准确度,以及结合模型特征提取技术进行环状标志物的检测[6]。尽管已有研究取得了一定的成果,如利用机器人执行乒乓球击打任务、采摘机器人的避障系统设计等,但在实际操作中仍存在诸多挑战。例如,机器人在作业过程中可能会面临移动与障碍物动态变化带来的叠加效应,这要求机器人控制系统不仅要准确识别目标,还要实时调整自身的动作以适应环境变化。此外,精度和响应速度的提升也是现有研究亟待解决的问题。

本文提出了一款高效能、自动化程度高的辨识智能人形机器人控制系统设计。测试结果显示,该系统在对颜色目标的追踪和动作上均达到了超过 90%的准确率。该智能人形机器人具备一定精度的辨识能力,运行稳定,功能完善,具有一定的应用前景。

2. 系统总体设计

本文设计将控制系统分为两部分设计,一是硬件部分设计,二是软件部分设计。本文硬件部分选择了体积小、质量轻的材料搭建,以确保整个机器人系统的重心稳定和灵活性。对于软件架构来说,本文采用了模块化设计,便于根据不同应用场景调整和优化功能模块。系统的软件部署在实时操作系统之上,确保了各项任务在严格时间约束下的有效执行。通过高速总线技术与分布式处理模式,每个模块可以在最短的时间内完成数据交换和任务同步,确保了整个系统的流畅与高效。总体方案设计流程图如图 1 所示。

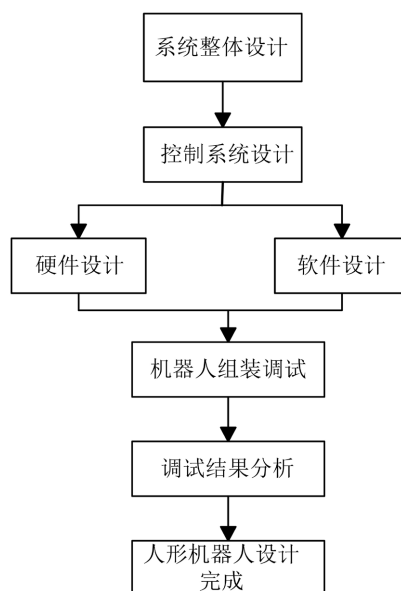


Figure 1. Overall scheme design flow chart

图 1. 总体方案设计流程图

3. 系统硬件设计与实现

辨识智能人形机器人的硬件部分分为主控单元模块、舵机、姿态传感器、图像处理模块、蜂鸣器、电源电路、USB 下载电路等模块。

3.1. 主控单元模块

根据实际项目需要, 本文主控单元模块选择一款多功能树莓派扩展板[7]。树莓派扩展板各接口功能如图 2 所示。

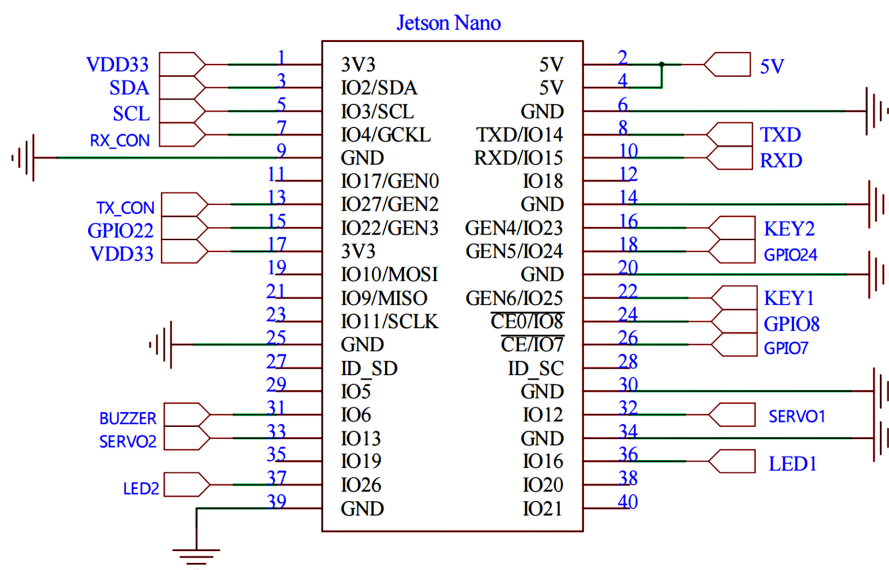


Figure 2. Schematic diagram of raspberry Pi extended edition

图 2. 树莓派拓展版原理图

3.2. 舵机

本文舵机主要分为两种：PWM 舵机和串行总线舵机。人形机器人的四肢，采用 LX-824 HV 串行总线舵机，走线美观，控制精准。机器人的头部采用 LFD-01M 微型 PWM 舵机，可以实现头部上下左右转动，增加视觉探测范围。控制器将 PWM 信号发送给舵机的信号调制芯片，获得直流偏置电压。然后，舵机内部会产生基准脉冲信号，与直流偏置电压比较得到电压差。最后，电压差输出到舵机驱动芯片决定舵机的正反转，而脉冲高电平部分的持续时间控制舵机转动的角度。机器人将先点头后摇头并且终端上将会打印对应的提示。控制器与舵机之间采用串行通讯，每个舵机通过 ID 号进行区分。在控制舵机转动前，先按需求设置好对应 ID。然后通过串口发出包含 ID 号、转动角度、时间等信息的指令，当对应舵机接收到后即可执行。舵机电路设计如图 3 所示。

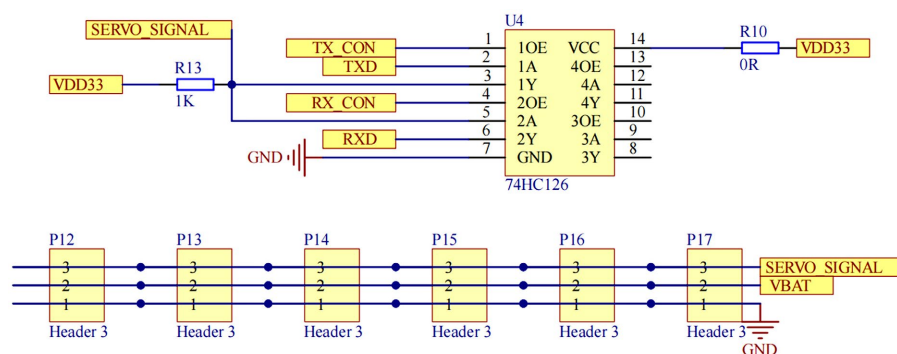


Figure 3. PCA9685 circuit diagram

图 3. PCA9685 电路图

3.3. 姿态传感器模块

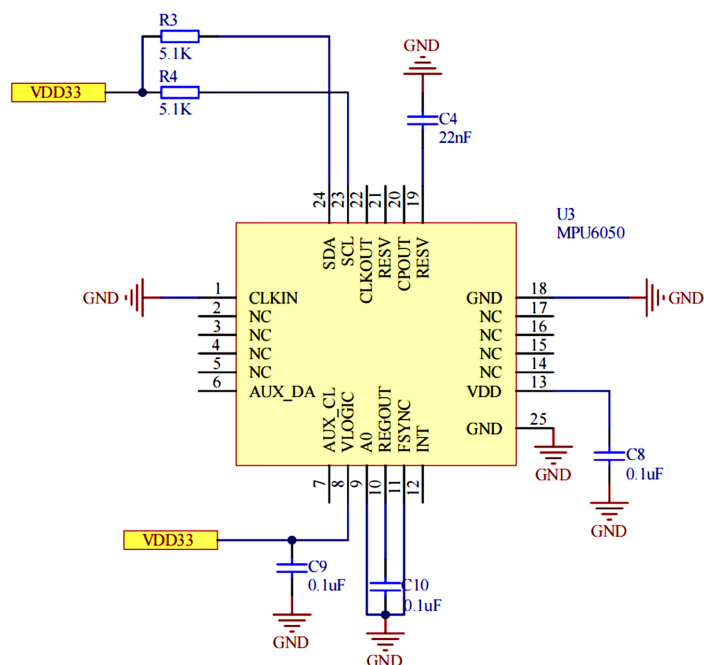


Figure 4. Circuit diagram of MPU6050

图 4. MPU6050 电路图

姿态传感器是一种用于测量物体姿态的设备, 它可以用于测量物体的角度、方向和加速度等参数来确定物体的姿态。本文选取 MPU6050 芯片[8], 内置运动数字处理引擎, 可减少复杂的融合演算数据, 减轻处理器的负荷, 拥有更低的功耗, 通过 I2C 通讯就能获取数据。姿态传感器模块电路设计如图 4 所示。

3.4. 图像处理模块

图像处理模块作为本文控制系统中至关重要的部分。该模块负责捕获、分析并解释视觉信息, 指导机器人完成相应的动作。通过深入的研究和应用先进的图像处理技术, 可以实现对人体动作的精确识别, 从而提升机器人对复杂环境的适应能力和执行任务的精准度。针对环境光照变化大、背景复杂等实际情况, 设计出的图像处理模块具备强大的环境适应性和鲁棒性。本文在图像处理模块设计中, 采用了由深度卷积神经网络构建的特征提取器, 能够从大量的图像样本中自动学习到人体关键部位和动作特征。DCNN 具备多层次、非线性的图像表征能力, 能有效提取人体关节位置及其相对关系, 为后续动作判定提供强有力的数据支持。结合空间向量和平面投影技术, 模块能够精确计算人体关节在空间中的角度变化。通过实时输入的视觉信息, 机器人能够动态调整自身关节状态, 以模拟人类的动作序列。

在图像的预处理阶段, 采用了高效的光照校正算法和背景抑制方法, 有效提升了图像质量。光照校正算法通过对原始图像的光照部分进行分析, 按照预设参数实时调整图像亮度, 使机器人能够在不同光照环境下保持图像识别的高准确性。背景抑制方法利用深度感知机制, 从背景中分离出人体目标, 极大降低了背景杂乱对于关键特征提取的干扰, 增强了系统识别的准确性和稳定性。数据传输环节, 模块设计采用了 AT 指令构建的局域网络, 并通过 WIFI 模块实现无线数据传输[9]。在传输协议方面, 设立了一套新的数据传输标准, 保证了指令传输的高效和稳定, 从而使得伺服电机能按照精准的时序响应控制信号, 实现机器人动作的高精度同步执行。总体上, 图像处理模块在设计上注重实用性与先进性的平衡, 采用了多样化的图像优化技术, 以适应各种复杂的应用场景。

3.5. 蜂鸣器

主控单元模块中板载的是有源蜂鸣器, 采用直流电压供电。有源蜂鸣器内部带震荡源, 当通电后即可直接发声, 其电路设计如图 5 所示。

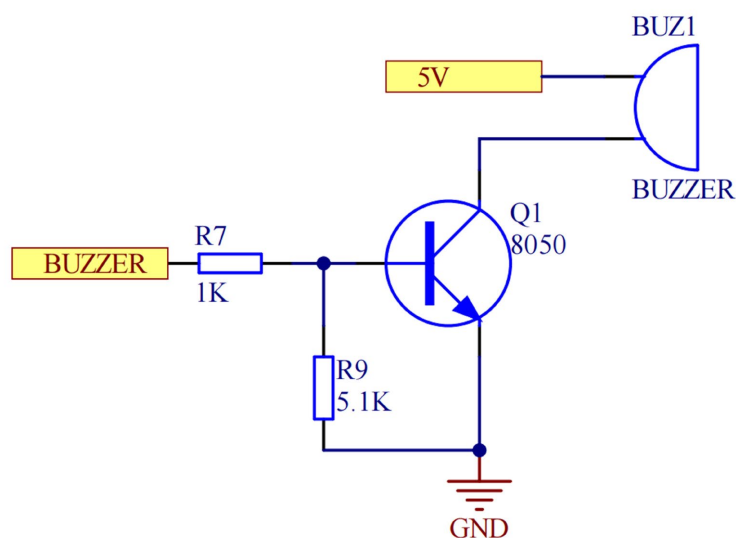


Figure 5. Buzzer circuit diagram
图 5. 蜂鸣器电路图

3.6. 电源电路模块

本文主要有两种供电方式，一种是采用锂电池供电(额定电压为 8.4 V)，另一种是采用 USB 接口线进行供电。采用锂电池供电时，则需要将电压转换为 6 V 和 3.3 V (6 V 舵机供电，3.3 V 主控和传感器供电)。

4. 系统软件设计

整体程序采用模块化函数的编程思想，根据硬件设计按照功能不同分好模块进行相应子函数模块编程。总体的软件流程，如图 6 所示，分为三个时期：系统各模块的初始化、读取 MPU6050 传感器的数据、动作组程序控制、辨识功能实现(颜色识别、人脸识别等)。

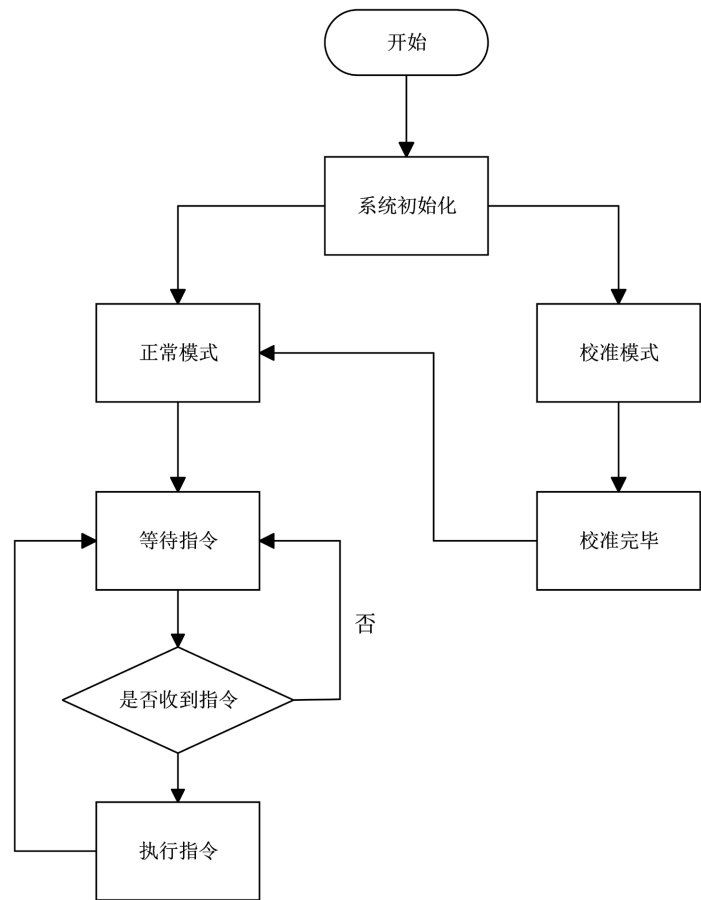


Figure 6. Main program flowchart
图 6. 主程序流程图

4.1. 读取 MPU6050 传感器的数据

首先需启动 MPU6050 传感器，并设定传感器的量程，其中“set_gyro_range”是设定陀螺仪的工作范围，“set_accel_range”是设定加速度计。

```
mpu = Mpu6050.mpu6050(0x68)# 启动 Mpu6050
mpu.set_gyro_range(mpu.GYRO_RANGE_2000DEG)# 设置 Mpu6050 的陀螺仪的工作范围。
mpu.set_accel_range(mpu.ACCEL_RANGE_2G)# 设置 Mpu6050 的加速度计的工作范围。
```


然后便可获取传感器的数值, 并且将获取的数值转换为角度值, 由于该项目仅包含向前和向后跌倒的情形, 因此仅需关注 Y 轴上的角度, 采用 “degrees” 函数即可实现。

```
accel_date = mpu.get_accel_data(g=True) # 获取传感器值。
```

```
angle_y = int(math.degrees(math.atan2(accel_date['y'], accel_date['z']))) # 将获得的数据转化为角度值。
```

4.2. 动作组

动作组动作组程序控制的相关程序保存在 “ActionGroupControl” 模块中, 为了简洁美观, 本文提前导入该模块, 并在后续的程序中用 “AGC” 来表示。

```
import hiwonder.ActionGroupControl as AGC。
```

接下来便可通过设定角度阈值来让机器人执行对应动作。本小段以机器人向后倒下的程序为例。设定 “angle_y” 的绝对值大于 160 时, 即判定机器人向后倒下。此时, 在终端打印 “stand up back !”, 并且调用后倒起立的动作组 “stand_up_back”。

```
if abs(angle_y) > 160:
```

```
    print("stand up back ! ")# 打印执行的动作名
```

```
    AGC.runActionGroup('stand_up_back')# 执行动作
```

使用 “runActionGroup” 函数来调取动作组, 其中的参数为对应的动作组名, 因此必须与保存时的动作组名一致。为避免手持机器人平放时, 误伤人体, 同时也为增加检测准确性, 应等待一段时间再执行动作。因此可以添加标志位 “count1”, 程序每检测到一次 “angle_y” 角度大于 160, 即计数一次。计数达到一定值, 方可执行动作。当满足 “count1” 大于或等于 5 时, 应执行 “stand_up_back” 动作, 并将标志位清零, 等待下次计数。

4.3. 辨识(颜色处理)

首先设定红、绿、蓝为可识别颜色, 在 “range_rgb” 中新增颜色及颜色 RGB 值[10]。由目标颜色来决定机器人是点头还是摇头, 使用的仍然是 “setPWMServoPulse” 函数。目标为红色时的程序时, 程序对舵 1 进行处理, 让机器人连续点头两次, 再回到中位, 改变第二个参数(PWM 信号值)即可, 当其为 1500 时, 表示舵机回中。

对图像进行处理, 所谓对图像的处理即对数字的处理。先导入程序, 其次, 初始化并开启摄像头, 进行图像获取, 然后对图像进行复制、重映射、显示。接着, 需要将 BGR 格式图像信息转换为灰度图像信息, 通过 cv2.cvtColor 函数来实现。程序代码如下:

```
gray=cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

在图像处理完毕后, 程序需要对标签进行检测处理。通过调用 tag 的库文件对获取的图像进行检测。代码实现如下所示:

```
detections = detector.detect(gray, return_image=False)在检测后将会获取标签码四个角点的信息: corners = np.rint(detection.corners)。
```

接下来需要绘制 Tag 标签的轮廓。在 OpenCV 中程序通过 cv2.drawContours 函数来实现, 程序如下所示:

```
cv2.drawContours(im, [np.array(corners, np.int)],-1, (0, 255, 255))。
```

这个函数一共有五个参数, 各个参数的含义如下所示: 第一个参数 “img”: 代表一张图片; 第二个参数 “[np.array(corners,np.int)]”: 代表轮廓; 第三个参数 “-1”: 代表轮廓的索引, -1 表示绘制所有轮廓; 第四个参数 “(0, 255, 255)”: 代表轮廓的颜色; 第五个参数 “2” 代表轮廓的厚度。

程序通过直接调用 Apriltag 库来进行编码与解码, 以获取标签的信息。根据编码方式的不同, 生成不同的内点坐标。在确定的四边形中明确点阵坐标, 为了进一步确定编码是否可靠, 要同时与已知的编码库进行匹配。在经过筛选和验证后, 便可以计算该标签的 ID 和旋转角。

4.4. 辨识(人脸)

本文让机器人模仿人类进行人脸识别, 就必须细化到每一步进行模拟。总体来说从流程上会经过几个步骤:

(1) 人脸检测, 比如在计算机图像或视频中找到人脸的位置。当发现有人脸出现在图片上时, 标记出人脸的坐标信息, 或者将人脸切割出来。OpenCV 和 Dlib 是两个常用的算法库, OpenCV 中可以使用 Haar 级联分类器来做人脸检测, 比如人脸和脸上眼睛的位置[11]。Dlib 中是通过计算图像局部区域的梯度方向直方图来提取特征, 这种方法的本质在于梯度的统计信息, 而梯度主要存在于边缘的地方。

(2) 图像处理, 比如先将图片灰度值转换, 计算图像中像素的梯度, 获取人脸的位置。最后对人脸进行处理, 比如通过分析其中的特征, 与数据库中存储的已知人脸特征进行比较, 确定人脸。机器人的头部会来回转动, 搜索视野范围内的人脸。当识别到人脸后, 机器人会执行打招呼的动作。

4.5. 上位机软件设计

上位机的软件设计基于 VNC viewer 软件[12]设计完成, 最终完成的用户界面如图 7 所示。其目的主要是电脑应用程序的制作, 制作电脑端软件应用的目的是为了获取机器人的相关数据且可通过该应用直接控制机器人的运行, 利用先进的编程技术和图像处理算法, 构建了功能强大的上位机界面。通过该界面, 用户可以直观地操作和监控机器人的行为, 实时获取图像信息, 并进行分析和处理。

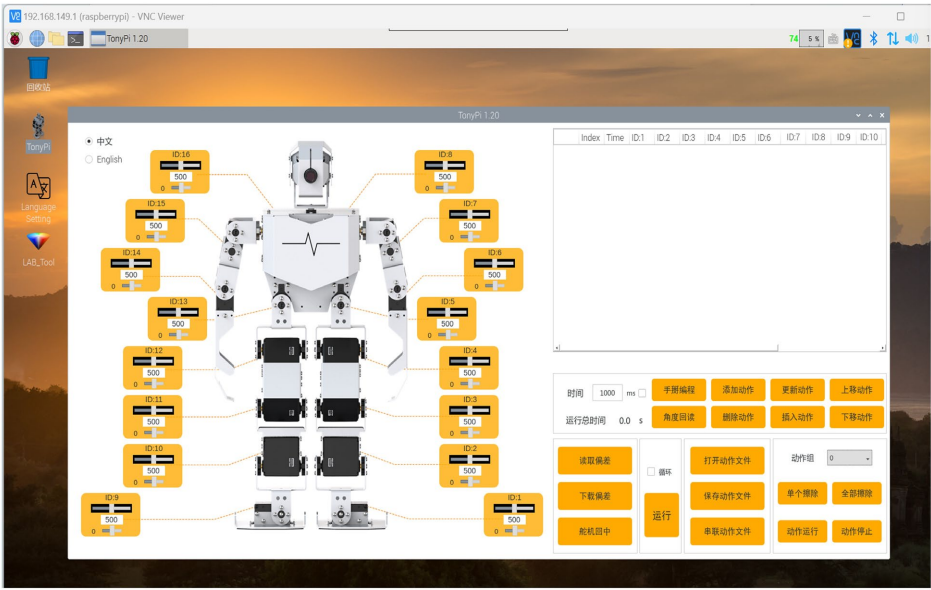


Figure 7. Schematic diagram of upper computer interface
图 7. 上位机界面示意图

5. 系统测试

首先打开 VNC, 与机器人进行连接, 完成网络热点共享, 然后通过上位机的小程序控制人形机器人的动作。

5.1. 辨识(颜色)

对于一个 BGR 图像来说, “image [0, 0, 0]” 元素的第一个值表示像素的 Y 轴坐标或者行号(0 代表顶部); 第二个值代表像素的 X 轴坐标或列号(0 表示最左边); 第三个值表示颜色通道。这些记录图像的数组和普通的 Python 数组一样可以单独访问, 从而可以获得某个颜色通道的数据, 或截取图像某个区域的图像。本文可以运行程序“sudopython3TonyPi/Camera.py”, 随后在弹窗中可看到当前拍摄的实时画面, 如图 8 所示。

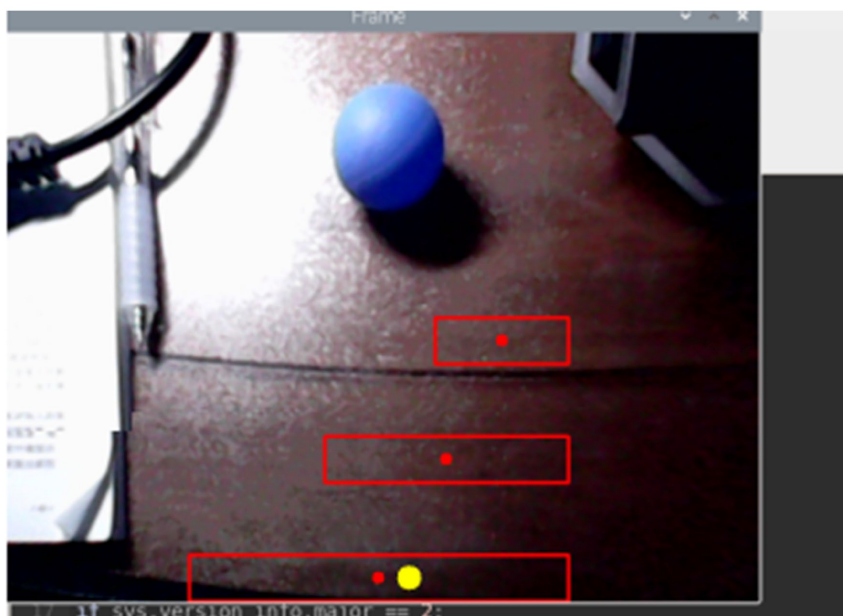


Figure 8. Real time footage captured
图 8. 拍摄实时画面

通过初步处理的图像, 只有再经过轮廓提取和位置确定, 信息才能被有效利用。因此, 程序需要对物体和背景的颜色进行区分。OpenCV 中的 “inRange” 函数可以将图像中的所有像素以 0 和 1 来表示, 将值为 0 的像素点用黑色显示, 值为 1 的用白色显示。这种方法即为图像的二值化处理。处理语句如下所示:

```
frame_mask = cv2.inRange(frame_lab,
    (lab_data['red']['min'][0],
    lab_data['red']['min'][1],
    lab_data['red']['min'][2]),
    (lab_data['red']['max'][0],
    lab_data['red']['max'][1],
    lab_data['red']['max'][2]))# 对原图像和掩模进行位运算
frame_lab 为图像信息;
```

(lab_data['red']['min'][0],lab_data['red']['min'][1],lab_data['red']['min'][2]) 为红色的颜色下限。

(lab_data['red']['max'][0],lab_data['red']['max'][1],lab_data['red']['max'][2]) 为红色的颜色上限。

该语句的含义是将二值化后的图像信息存储到 “frame_mask” 变量中。当像素点的 L、A、B 值处于红色的颜色上、下限之间时, 用 1 表示; 其余像素点用 0 表示。

为了降低干扰, 让图像更加平滑, 课题将二值化处理后得到的灰度图像, 依次进行腐蚀和膨胀处理。先腐蚀后膨胀的运算称为开运算, 它能够消除亮度较高的细小区域, 更细致地分离物体; 对于较大物体, 可以在其面积不明显改变的情况下平滑其边界。腐蚀处理的目的是去除图像边缘的毛刺, 需要使用 OpenCV 库中的“erode”函数。该函数的第一个参数“frame_mask”是二值化图像, 第二个参数代表的是进行操作的内核。下面的函数中设置的内核形状为矩形, 尺寸为 3×3 矩阵。膨胀处理会将图像的边缘扩大, 用以填充目标物体边缘或内部的非目标像素点。其使用的是“dilate”函数, 第一个参数是腐蚀处理后的图像信息, 第二个参数与腐蚀函数相同。本文所述控制系统对颜色识别追踪测试图, 如图 9 所示。

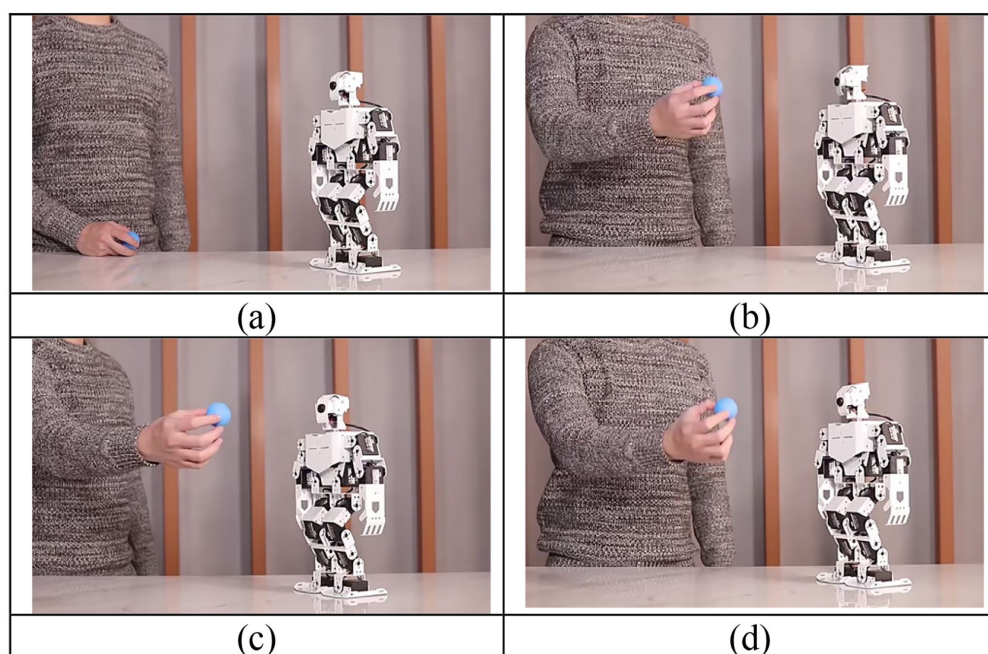


Figure 9. Robot performing color tracking
图 9. 机器人进行颜色追踪

5.2. 辨识(人脸)

开启程序后, 机器人会检测前方是否出现人脸。当检测到后, 程序将在回传画面中圈出识别到的人脸, 并发出“滴滴”警报声。所谓图像处理就是对数字的处理, 初始化并开启摄像头后进行图像获取, 然后对图像进行复制、重映射、显示。

本文使用人脸识别库训练好的卷积神经网络人脸模型, 把模型放置到指定的路径下, 通过 OpenCV 的函数进行调用。接下来, 本文定义一个“conf_threshold”变量, 用来存放阈值。再计算当前检测到的人脸与人脸识别库的所有用户面部特征对比的相似度, 得到置信度值最后将该值和“conf_threshold”进行比较, 进而判断机器人下一步的辨识程度。当置信度值“confidence”大于阈值时, 本文将人脸坐标进行转换为未缩放前坐标, 将人脸框出, 并运行子线程设置蜂鸣器响一声作为反馈。机器人的头部会来回转动, 搜索视野范围内的人脸。当识别到人脸后, 机器人会执行打招呼的动作。人脸辨识效果图, 如图 10 所示。

通过上述测试结果显示: 该系统在对颜色目标的追踪和动作上均达到 90%以上的准确率。即, 本文所设计控制系统使得人形机器人具备一定精度的辨识能力, 具有一定实际应用推广价值和效益。

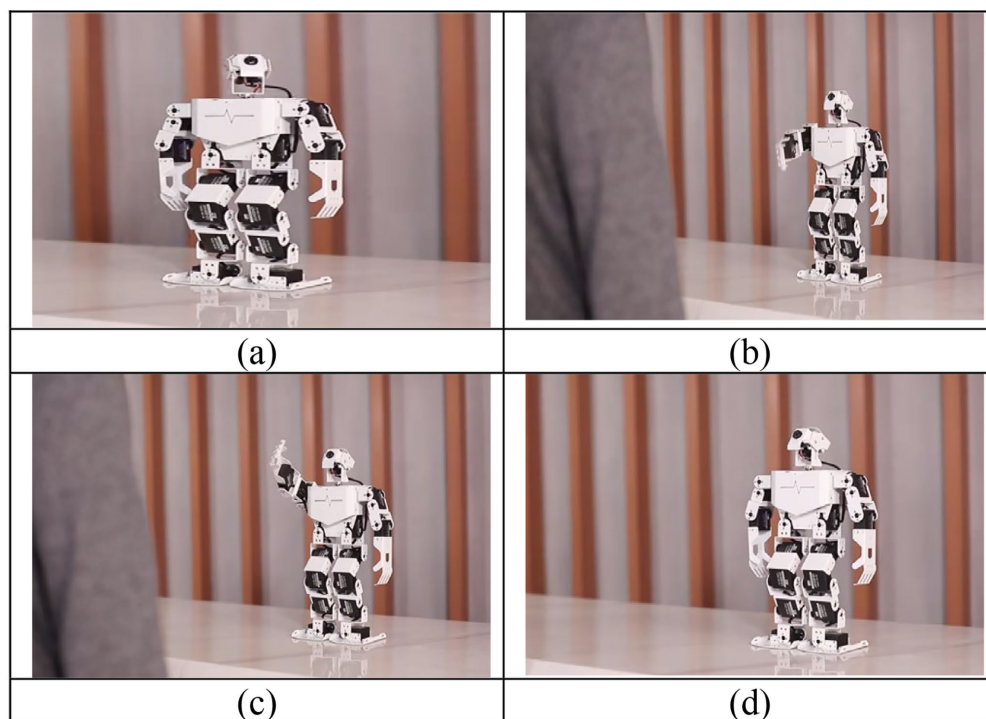


Figure 10. Robot performs facial recognition and responds

图 10. 机器人进行人脸识别并反应

6. 结语

本文基于图像处理的辨识智能人形机器人控制系统设计, 系统整体划分为硬件设计和软件设计两大部分, 两者都采用了模块化的思想进行模块化设计, 硬件设计模块上主要功能实现集中在主控单元模块、舵机、姿态传感器、图像处理模块、蜂鸣器、电源电路、USB 下载电路等模块。软件整体程序采用模块化函数的编程思想, 根据硬件设计按照功能不同分好模块进行相应子函数模块编程, 主要分为三个时期: 系统各模块的初始化、读取 MPU6050 传感器的数据、动作组程序控制、辨识功能实现(颜色识别、人脸识别等)。在完成了各个元器件的组装之后的实物测试, 主要测试了人形机器人的辨识能力, 主要以颜色追踪与人脸识别为例。由测试结果图可知, 该系统在对颜色目标的追踪和动作上均达到 90%以上的准确率。经实验验证分析可得: 本文所设计人形机器人具备一定精度的辨识能力, 运行稳定, 功能完善, 具有一定的应用前景。

参考文献

- [1] 王苑丞, 文靖豪, 张祺睿, 等. 基于 Jetson Nano 的智能双足人形机器人[J]. 智能计算机与应用, 2024, 14(4): 177-179.
- [2] 王宏民, 莫敦成, 江励, 等. 5G 网络人形机器人远程控制实时性能研究[J]. 制造业自动化, 2023, 45(7): 155-159, 163.
- [3] 冷晓琨. 双足机器人参数设计及步态控制算法研究[D]: [博士学位论文]. 哈尔滨: 哈尔滨工业大学, 2020.
- [4] 蒋正涛. 工业机器人视觉系统在电气自动化中的应用探究[J]. 信息记录材料, 2024, 25(8): 70-72.
- [5] 李响, 陆锋, 陈钰婷, 等. 图像识别技术及其在架空线检查机器人中的运用[J]. 黑龙江科学, 2024, 15(4): 144-147, 151.
- [6] 王文胜, 李世蛟. 基于单目视觉传感器障碍物躲避的智能车设计[J]. 传感器与微系统, 2023, 42(4): 119-122.
- [7] 汪文彬, 陈德祥, 马玉春. 树莓派开关监控软件的实现与教学示范[J]. 电子元器件与信息技术, 2023, 7(1): 114-

- 116, 127.
- [8] 胡海洋, 王淑成, 杨旭, 等. 基于 STM32F427 四旋翼无人机控制系统设计[J]. 仪器仪表与分析监测, 2024(2): 24-27.
 - [9] 陆增洁, 黄雄健, 汪诗怡, 等. 基于机器学习的电力系统语音指令识别算法研究[J]. 电力与能源, 2024, 45(4): 486-489.
 - [10] 赵灵芝, 赵柳, 张小清, 等. 智能手机辅助双探针比色法检测 α -葡萄糖苷酶[J]. 分析测试学报, 2024, 43(6): 905-912.
 - [11] 谢强, 张华梅, 王宇才. 基于 OpenCV 的手势识别技术在机电系统人机交互中的应用与研究[J]. 轻工科技, 2024, 40(5): 119-121, 159.
 - [12] 侯勇. VNC 在展项控制中的应用与实践[J]. 设备管理与维修, 2023(17): 70-71.