**实验文档**

在“2025 具身智能机器人目标检测算法大赛”的“农业蘑菇收割识别”赛道中，本人需针对农业场景下的蘑菇收割需求，设计并实现高效精准的目标检测算法，并模拟智能机器人操控，参与到目标检测技术在智慧农业领域的发展道路探索中。该赛道聚焦于解决农业生产中蘑菇识别与定位的实际问题，要求算法能够在复杂的农田环境中准确检测出蘑菇的位置、大小等关键信息，为后续机械臂自动化收割提供可靠的坐标数据支撑。这不仅需要算法具备优秀的目标检测性能，还需考虑农业场景下光照变化、蘑菇形态多样性、无关物品干扰等实际挑战，旨在通过技术创新提升农业生产的智能化与自动化水平，推动目标检测技术在智慧农业领域的应用与发展。

**一、项目概述**

1.1项目背景与目标

本项目旨在实现一个基于香橙派Orange-Pi-AIpro(20T)的物体识别与模拟机械臂抓取系统。在硬件部署前，通过软件模拟实现以下核心功能。

①图像中物体的检测与识别

②计算物体在图像中的坐标位置

③支持批量图像处理，为后续机械臂控制提供坐标数据

1.2 项目特点

①端侧部署优化：针对香橙派Orange-Pi-AIpro(20T)设计，支持低资源环境运行

②多尺度检测：通过不同分辨率图像处理提升小目标检测效果

③坐标校准：针对机械臂抓取需求优化边界框坐标，提高抓取精度

④可扩展架构：采用工厂模式与策略模式，便于后续功能扩展

**二、系统架构**

2.1整体架构

2.1.1目标识别与定位

搭建项目，通过OpenCV读取指定文件夹下面的jpg图片，加载预训练模型基础检测器yolov5s，对图片进行初步的处理，为图片的识别做好准备。

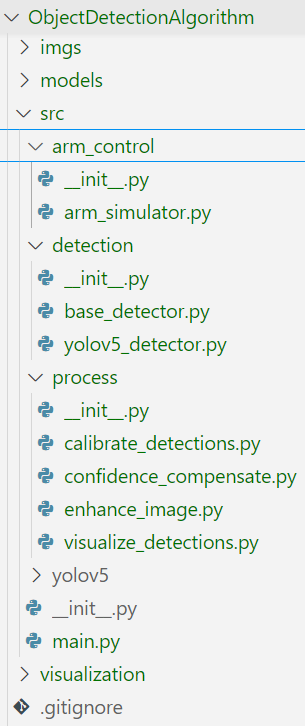
2.1.2 物体分析模块

该模块借助目标检测模型对检测到的农作物进行关键尺寸计算，采用动态分辨率策略优化检测流程：基于OpenCV实现图像轮廓检测，针对小蘑菇与大蘑菇的尺寸差异执行差异化检测策略，通过自适应调整分辨率提升检测速率与准确性。同时，依托工厂模式创建YOLOv5多尺度检测框架，并结合单例模式管理配置参数，对检测结果进行坐标校准与NMS后处理，进一步优化检测精度与处理效率，形成从目标识别到尺寸计算的完整分析链路。

2.1.3机械臂控制决策

该模块基于物体检测所得坐标生成虚拟抓取指令，驱动模拟机械臂执行抓取动作（鉴于蘑菇处于静止状态且无物理机械臂，采用逻辑模拟实现）。在完成虚拟抓取操作后，通过字符串模拟机器人响应信号，触发系统进入下一轮循环，自动加载并处理下一张待检测图片，形成“识别-定位-模拟抓取-循环迭代”的完整自动化流程。

2.2整体架构



**三、算法原理**

3.1 物体检测原理

YOLOv5检测流程涵盖图像预处理（将图像尺寸调整至模型输入要求）、特征提取（通过CNN网络提取图像特征）、区域预测（同步预测物体边界框坐标与类别概率）以及后处理（借助NMS算法过滤重叠检测框，保留高置信度预测结果）；而多尺度检测策略则通过对同一图像按960x960和1280x1280两种尺寸分别缩放后进行检测，其中小尺度（960x960）可提高推理速度，大尺度（1280x1280）能增强小目标检测能力，最终将不同尺度的检测结果还原至原图尺寸并合并，保留置信度较高的检测框，以此兼顾检测效率与精度。

3.2 坐标校准算法

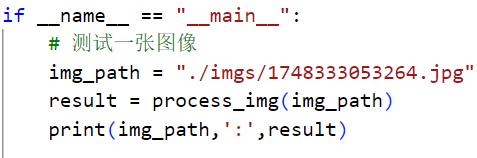
3.2.1偏移修正与3.2.2尺寸扩展 在坐标校准过程中，偏移修正通过将x坐标左移（x = x - w \* 15%）以补偿机械臂抓取时的位置偏移，同时将y坐标上移（y = y - h \* 12%）来校正摄像头视角偏差；尺寸扩展则对宽度进行w = w + w \* 32%的扩展以增加抓取容错空间，高度按h = h + h \* 5%进行微调，并通过最小尺寸限制确保w/h≥50像素，避免产生无效检测框，从而形成一套针对机械臂抓取需求的坐标优化机制。

3.3 非极大值抑制(NMS)

非极大值抑制(NMS)的作用是消除重叠的检测框并保留最优结果，其实现流程为：先按置信度对检测框排序，再计算框间交并比(IOU)，将重叠度大于0.5的检测框过滤，其中IOU的计算公式为交集面积与并集面积的比值，通过该算法可有效提升检测结果的准确性与唯一性。

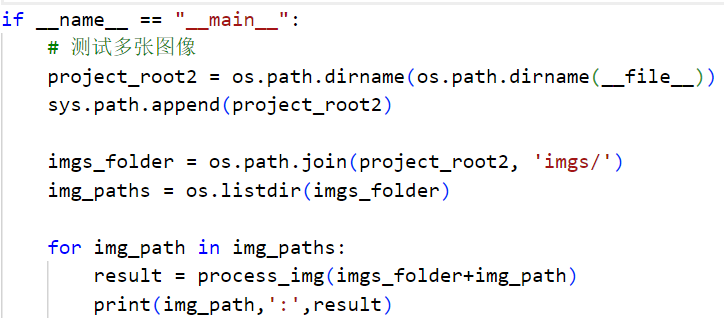
**四、使用方法**

4.1单图检测（测试单张图片）



执行命令python src/main.py

4.2批量检测（自动处理所有测试图片）



执行命令python src/main.py

4.3输出结果说明

检测结果以列表形式返回，每个物体对应一个字典。



**五、依赖关系**

5.1硬件依赖

开发测试环境：支持 Linux 系统的设备（如 PC、香橙派 - OrangePi AIpro），推荐 OrangePi AIpro（20T 算力，24G 内存）用于端侧部署。

输入设备：USB 摄像头（后续硬件对接时使用，当前测试可用图片模拟）。

输出设备：机械臂（可选，用于实际抓取，当前阶段支持模拟控制）。

5.2软件依赖

基础环境：Python 3.8+、pip 包管理器。

第三方库：

opencv-python：图像处理（读取、增强、轮廓检测）。

ultralytics：集成 YOLOv5 目标检测模型。

numpy：数值计算与数据结构处理。

**六、安装配置步骤**

6.1.克隆项目仓库

https://github.com/RiverMelanie/ObjectDetectionAlgorithm.git

cd ObjectDetectionAlgorithm

6.2创建并激活虚拟环境

python -m venv venv

source venv/bin/activate # Linux/Mac系统

venv\Scripts\activate # Windows系统

6.3安装依赖库

pip install -r requirements.txt

安装完成后，可通过以下命令验证环境是否正确配置：

pip list | grep -E "opencv|ultralytics|numpy"

6.4准备测试数据

mkdir -p imgs

6.5运行代码

python src/main.py