

基于ZYNQ的实时水果识别系统

宋宛容, 向强*

西南民族大学电子信息学院, 四川 成都

收稿日期: 2023年1月15日; 录用日期: 2023年2月15日; 发布日期: 2023年2月23日

摘要

由于农业与科技快速发展,机器人被更多用于农业生产中,机器视觉是机器人设计中不可或缺的一部分,如何在嵌入式平台上实现实时视频捕捉与视频处理及优化是目前热门研究方向。本系统采用软硬结合方式实现ZYNQ平台上视频采集水果识别系统。硬件平台为XILINX ZYNQ 7020板卡,双目摄像头采集水果图像并通过以太网口将视频数据传输至上位机;上位机使用改进后的MobileV3-YOLOv3算法对图像进行识别。结果显示,硬件平台能够传送实时视频数据至上位机,且经过预处理IP核处理后的图像有明显的优化,改进后的算法AP50达到92%,FPS为34 f/s,模型参数数量为235.61M,并回传识别物体位置三维坐标。

关键词

可编程片上系统, 千兆以太网, 上位机, YOLO-v3, 水果识别

Real-Time Fruit Recognition System Based on ZYNQ

Wanrong Song, Qiang Xiang*

School of Electronic and Information Engineering, Southwest Minzu University, Chengdu Sichuan

Received: Jan. 15th, 2023; accepted: Feb. 15th, 2023; published: Feb. 23rd, 2023

Abstract

Due to the rapid development of agriculture and technology, robots are more used in agricultural production. Machine vision is an indispensable part of robot design. How to realize real-time video capture and video processing and optimization on an embedded platform is currently a popular research direction. The system adopts the combination of software and hardware to realize the video

*通讯作者。

collection fruit recognition system on the ZYNQ platform. The hardware platform is XILINX ZYNQ 7020 board, the binocular camera collects fruit images and transmits the video data to the host computer through the Ethernet port; the host computer uses the improved MobileV3-YOLOv3 algorithm to identify the images. The results show that the hardware platform can transmit real-time video data to the host computer, and the images processed by the pre-processing IP core are obviously optimized. The improved algorithm AP50 reaches 92%, FPS is 34 f/s, the model parameter quantity is 235.61M, and the position of the object is retrieved three-dimensional.

Keywords

System on a Programmable Chip, Gigabit Ethernet, Host Computer, YOLO-v3, Fruit Recognition

Copyright © 2023 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

水果图像识别在智慧农业领域中的重要地位,通过对水果图像的识别可以实现复合型果园果树的精准培育以及水果的自动化采摘。目前,图像分类识别技术已相对成熟并在 CPU 和 GPU 平台上实现了部署,但是高功耗的缺陷使得 CPU 和 GPU 无法应用于目前的嵌入式移动终端场景中[1],因此,如何在嵌入式平台上实现实时视频捕捉和视频处理以及优化,是目前的热门研究方向。

本文采用使用软硬协同的设计方法,结合上位机与 AX7020 开发板实现嵌入式水果识别系统。系统主要包括四大部分,即图像采集部分、图像预处理部分、图像传输部分与图像识别部分[2]。其中图像采集、图像预处理与视频数据传输功能在 FPGA 平台上完成,水果图像识别功能由上位机完成。

在 FPGA 部分,采用 OV5640 双目摄像头对水果图像进行采集,采集到的图像通过 HLS 工具设计自定义 IP 核进行锐化预处理,经过预处理后的视频数据通过千兆以太网口自定义 UDP 协议发送至上位机,该与上位机的互联操作可用于后续进一步完成图像识别算法,利用传统软件或 FPGA 完成处理均可,可操作性强。

近年来,各学者针对目标识别问题引入了不同的深度学习算法[3]。例如 Girshick 等人[4]在 2014 年首次使用 R-CNN 网络进行目标识别后,继而出现 Fast RCNN、Faster R-CNN 等目标检测模型,但由于该类算法分类速度慢、类别不均衡等问题,Redmon 等人提出 YOLO (You Only Look Once)系列算法[5][6][7],并因其检测速度与精度的优越性被广泛运用在医疗、安防、交通、农业等领域。但以往的目标检测算法仍存在速度低、无法迁移至嵌入式设备中等问题,为满足检测方法在嵌入式设备和移动设备上使用和时效性要求,本文以 YOLOv3 模型为基本框架展开对水果识别算法的研究,并对该模型的网络结构进行轻量化改进,为后续算法的移植奠定基础。本文其余部分内容如下:第二节主要介绍在 zynq 硬件系统上搭建水果图像采集与传输系统;第三节提出改进后的轻量化水果识别算法;第四节在硬件平台上搭建不同的水果识别算法,并对其实验结果进行对比。

2. 基于 zynq 的水果识别系统设计

在 FPGA 部分设计图像采集及优化并利用以太网传输硬件系统。硬件平台采用 Xilinx 公司的 zynq7020 开发板,ZYNQ7020 芯片可分成处理器系统部分(PS 端)和可编程逻辑部分(PL 端),底板为核心板扩展了

丰富的外围接口。实验采用 OV5640 摄像头模块采集视频图像, 图像预处理部分使用 HLS 工具设计图像锐化 IP 核, 将经过锐化处理后的视频数据传输至 VDMA 模块, VDMA 通过写通道将视频数据写入 PS 端 DDR3 中缓存, 最后通过千兆以太网口自定义 UDP 协议将实时视频数据传输至上位机, 在上位机采用改进后的 MobileNetV3-YOLOv3 网络进行水果识别操作并获取水果位置信息, 整体系统框图如图 1 所示。

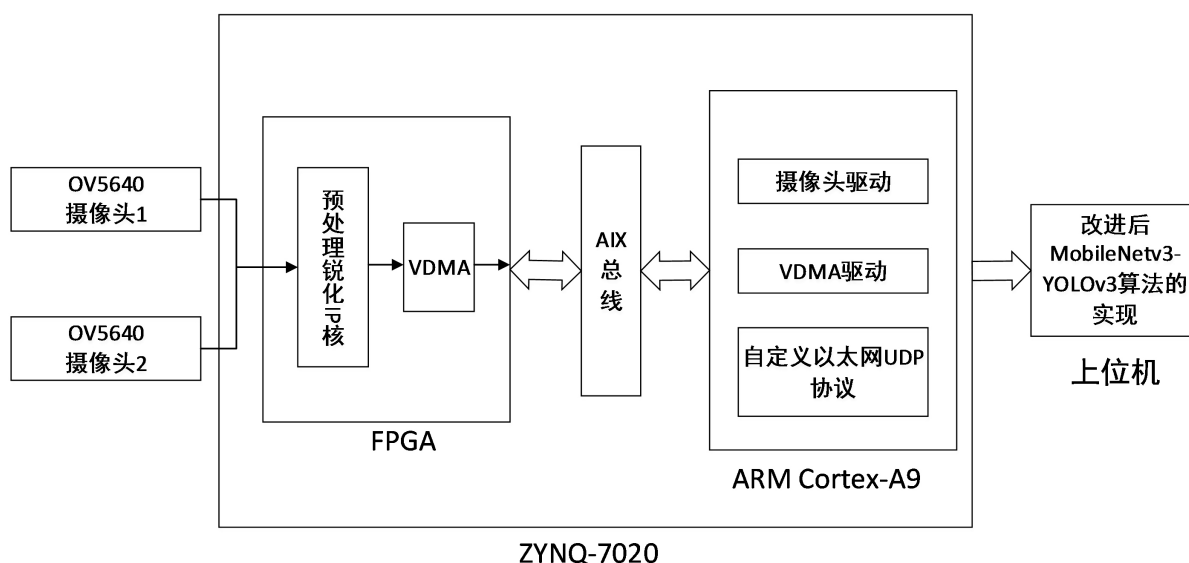


Figure 1. The overall system block diagram of fruit recognition

图 1. 水果识别整体系统框图

2.1. 图像锐化 IP 核设计

为了实现水果的准确识别, 增强图像的边缘和细节, 提高图像的对比度, 本系统在 FPGA 部分进行图像的锐化预处理操作, 使用 VIVADO HLS (High-Level Synthesis) 工具实现图像预处理算法功能, HLS 支持 C/C++ 语言设计图像预处理 IP 核, 克服传统硬件描述语言设计难度大、效率低、开发周期长等缺陷。

图像锐化通过增加图像中的高频分量来减少模糊, 同时增强图像的细节边缘与轮廓, 增强灰度反差, 与图像平滑为相反的操作, 有助于后期对目标进行识别和处理。主要的图像锐化方法有微分法和高通滤波法。本系统中选用拉普拉斯锐化法设计图像预处理 IP 核。拉普拉斯锐化法的基本原理为: 当邻域中心像素灰度低于它所在的领域内其它像素的平均灰度时, 此中心像素的灰度应被进一步降低, 当邻域中心像素灰度高于它所在的领域内其它像素的平均灰度时, 此中心像素的灰度应被进一步提高, 以此实现图像锐化处理[8]。二维图像函数拉普拉斯变换定义式如(1)所示:

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (1)$$

其对应的离散方程表示如(2)所示:

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y-1) + f(x, y+1)] - 4f(x, y) \quad (2)$$

拉普拉斯算子具有旋转不变性。即针对某个角度时, 利用拉普拉斯算子计算得出的图像再进行旋转和先进行旋转后再进行滤波得到的结果相同。旋转不变性使得该算子在各个方面的锐化增强效果平滑, 不突兀。

使用 HLS 工具自定义拉普拉斯锐化预处理 IP 核资源评估如图 2 所示。由图 2 可知, 拉普拉斯锐化预处理 IP 核在 ZYNQ 上的 BRAM_18K 占用率为 2%, DSP48E、FF 占用率为 1%, LUT 占用率为 3%。

Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	-	0	7
FIFO	0	-	60	384
Instance	6	4	1584	1473
Memory	-	-	-	-
Multiplexer	-	-	-	6
Register	-	-	2	-
Total	6	4	1646	1870
Available	280	220	106400	53200
Utilization (%)	2	1	1	3

Figure 2. Image sharpening IP core resource consumption
图 2. 图像锐化 IP 核资源耗用

使用 LCD 屏测试预处理 IP 核对图像优化功能对比如图 3 所示, 由对比可知, 在经过锐化预处理后, 采集图像相对于预处理前对比度增强。



(a) 预处理前效果



(b) 预处理后效果

Figure 3. Comparison of image preprocessing effects
图 3. 图像预处理效果对比

2.2. VDMA 的介绍与配置

AXI VDMA (AXI Video Direct Memory Access), 是 Xilinx 提供的软核 IP。其功能和 AXI DMA 类似, 都可以为存储器或者 AXI4-Stream 类目标外设之间提供高带宽直接存储器存取。和 DMA 相比, VDMA 增加了帧缓存的缓冲机制和同步锁相(GenLock)等功能, 是为了针对视频图像应用而做的升级版的 DMA。VDMA 中集成了如帧同步和 2D DMA 传输等视频专用功能, 有助于在 ZYNQ 平台架构上进行图像和视频处理, 极大缩短了开发周期[9]。DMA 和 VDMA 的对比如表 1 所示。

Table 1. Comparison of DMA and VDMA**表 1.** DMA 与 VDMA 对比

对比项	DMA	VDMA
功能	为存储器或 AXI4-Stream 类目标外设提供高带宽存取	为存储器或 AXI4-Stream 类目标外设提供高带宽存取, 帧缓存和帧同步等多种缓存机制
资源占用	较低	较高
应用场合	提供高速数据的传输, 如高速 ADDA 的数据收发等	主要针对视频类数据的高速传输, 如摄像头图像显示等

在本系统中, 使用一个 VDMA 模块。摄像头采集视频数据后, 传送至预处理 IP 核, 将处理后的数据通过读通道传输至 VDMA 模块, VDMA 的写通道再将视频数据写入 DDR 内存中缓存, 为了完成视频数据的读写功能, 使能 VDMA 的读通道和写通道, 并同时使能不对齐传输。Memory Map 接口采用 AXI4 接口, 与 ZYNQ HP 口进行数据交互, 读取 PS 端 DDR 中的图像数据。使能 Frame Buffers 为 3, 即 3 帧图像缓存, 避免采集后的图像割裂, 由于本系统需要对帧缓存进行频繁的写入和读出, 为了避免读通道和写通道同时访问同一帧, 我们将 VDMA 配置成动态同步锁相模式。VDMA 模块配置如下图 4 所示。

Figure 4. VDMA IP core configuration**图 4.** VDMA IP 核配置情况

2.3. 图像采集控制模块

2.3.1. 图像采集系统硬件设计

本系统采用 ZYNQ 开发板上的摄像头扩展接口, 连接 OV5640 摄像头模块, 由于需对摄像头进行初始化后才能正常使用, 因此将 CMOS_SCL 信号和 CMOS_SDA 信号连接上拉电阻, SCCB 端口是通过 EMIO 连接至 PS 中, SCCB 的驱动由 PS 实现。原理图如图 5 所示。

OV5640 摄像头数据的图像数据由 PS 端向 VDMA IP 核的帧缓存中写入, 采用 Xilinx 提供的 Video in to AXI4-Stream IP 核实现数据转换, 通过 VDMA IP 核中的写通道将 AXI4 Memory Map 格式的数据流转换成 AXI4-Stream 类数据流并写入帧缓存中。系统数据流向如图 6 所示。

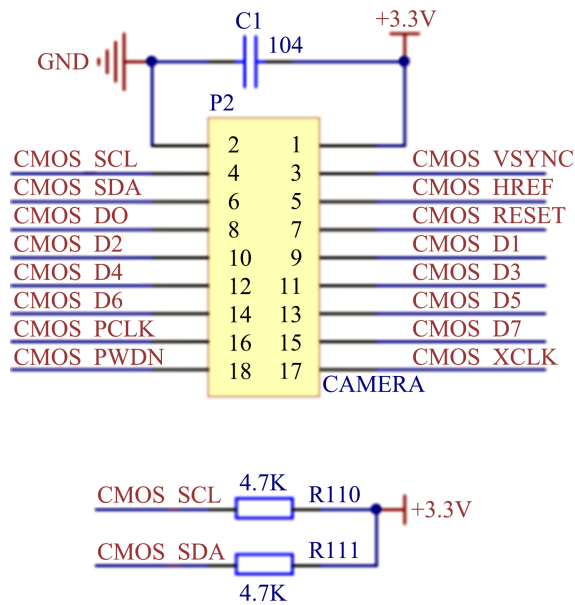


Figure 5. OV5640 hardware schematic
图 5. OV5640 硬件原理图

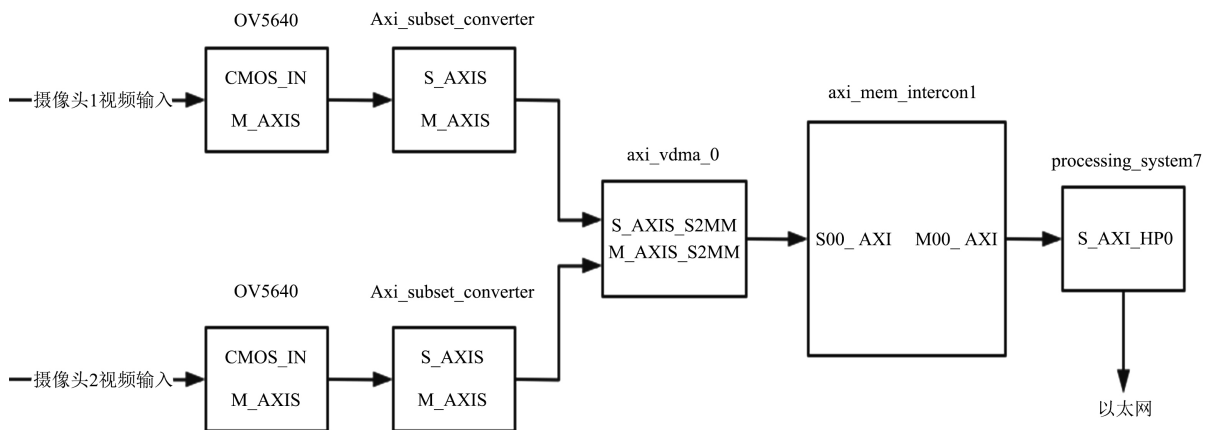


Figure 6. System data flow process diagram
图 6. 系统数据流向过程图

系统数据流向过程可总结如下:

- 1) OV5640 采集模块为自定义 IP 核, 负责将数据转换成视频流数据;
- 2) 视频流数据经过 Video in to AXI4-Stream IP 核转换成 AXI4-Stream IP 格式数据流, 然后通过 VDMA 的写通道转成 AXI4-Stream Map 格式, 并最终写入 DDR 内存中;
- 3) VDMA 通过 AXI Smartconnect IP 核与 AXI-HP 端口进行连接, 从而高效访问 DDR3, 进行视频数据的读写[9];
- 4) 自定义 udp 协议将视频数据打包为数据包通过以太网口传输至上位机。

整体系统框图如下图 7 所示。

至此, 系统硬件部分设计完毕, 完成硬件设计后生成比特流文件, 导出到 SDK, 进行 SDK 程序开发。在 SDK 工程中主要分为图像控制部分和以太网数据传输部分。系统资源占用情况如下图 8 所示。

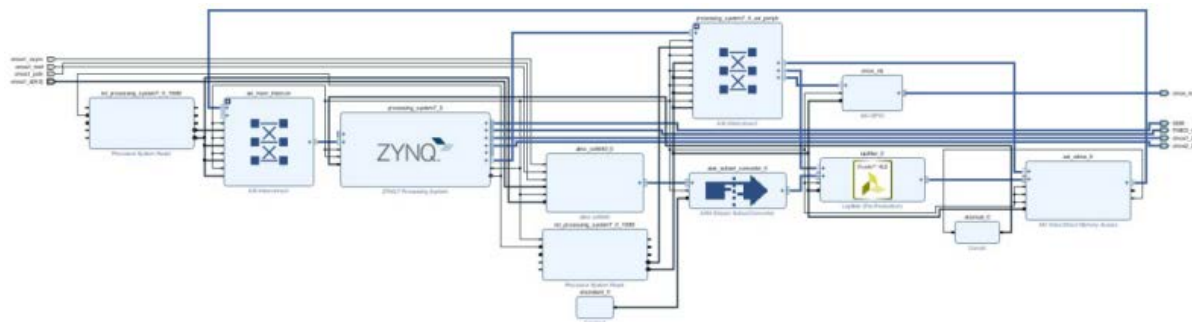


Figure 7. FPGA system block diagram
图 7. FPGA 系统框图

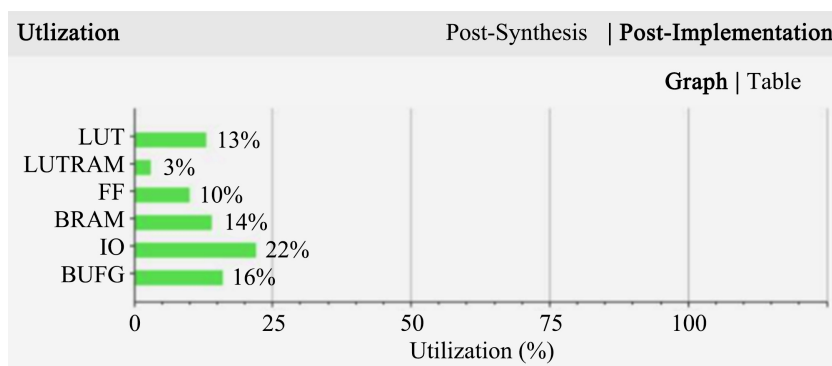


Figure 8. ZYNQ7020 resource occupancy
图 8. ZYNQ7020 资源占用情况

2.3.2. LWIP 以太网视频传输

以太网是一种应用广泛的局域网，也是目前最常用、最通用的通信协议。由于以太网拥有通信速率高、使用成本低、抗干扰性强等一系列优点，被广泛应用在交换机、网络远程监控、工业自动化等对通信速率要求较高的场合。如今最常用的两种以太网为千兆以太网(1000 Mbit/s)与万兆以太网(10 Gbit/s)，本系统实现的为千兆以太网。

以太网通信中数据是以数据包的形式传输，单包数据量可达到几十上百甚至上千个字节，而在串口通信中，单次数据传输仅为一个字节。UDP (User Datagram Protocol, 用户数据报协议)为以太网数据传输协议的一种，协议中数据包对各层协议的逐层封装来实现数据传输。数据格式如图 9 所示。

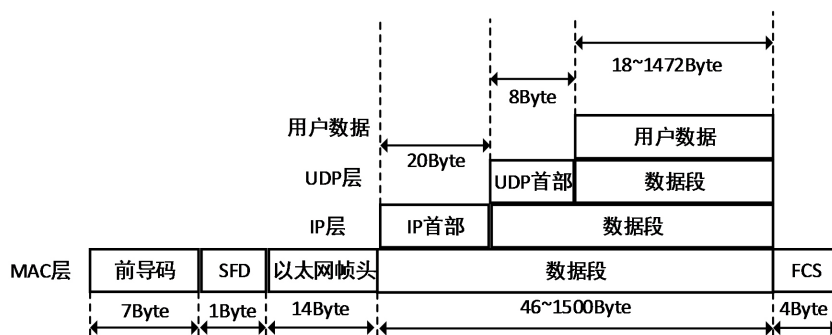


Figure 9. Ethernet packet data format
图 9. 以太网包数据格式

为了实现视频数据的实时传输, 本系统自定义 UDP 数据包协议, 主要步骤为获取板卡信息和获取数据两部分, 如表 2, 表 3 所示。

Table 2. Get board information

表 2. 获取板卡信息

询问命令(字节数)	命令信息
1	header
4	0x00020001
应答命令(字节数)	命令信息
1	Header 0x01
4	0x00020001
6	板卡 MAC 地址
4	板卡 IP 地址
1	0x02

Table 3. Get data

表 3. 获取数据

(a) 控制命令	
控制命令(字节数)	命令信息
1	Header
4	0X00020002
6	板卡 MAC 地址
1	摄像头通道选择
1	启动信号
(b) 应答命令	
应答命令(字节数)	命令信息
1	Header 0x01
3	0x000200
1	通道标识, 2 代表摄像头 1, 3 代表摄像头 2
3	序列号, 以太网包序号, 用于上位机识别
N	图像数据

每个 UDP 包都包含由 Header, 在第一个字节, 其格式如表 4 所示。

Table 4. UDP header information
表 4. UDP 包头信息

比特位	值(0)	值(1)
bit0	查询或控制	应答
bit0~bit7	随机数据	

开发板与上位机通信工作流程为:

- 1) 上位机发送询问命令;
- 2) 开发板应答询问;
- 3) 上位机发送控制命令请求数据;
- 4) 开发板发送数据;
- 5) 步骤 3 与 4 循环。

在完成硬件设计后,生成比特流文件,导出到 SDK,进行 SDK 程序开发。在 SDK 工程中主要分为图像控制部分和以太网数据传输部分。

图像控制部分操作主要包括初始化中断控制器、初始化三帧缓存、初始化中断、配置分辨率、初始化 vdma 通道和设置回调参数。

以太网数据传输部分代码编写过程大致如下:

- 1) 进行 lwip 的初始化,设置板卡的 MAC 地址,IP 地址,子网掩码,网关信息,并进行 DHCP 设置;
- 2) 填入 IP 地址,绑定回调函数,回调函数接受回调函数后判断命令种类并发送 udp 数据;
- 3) 利用 sendpic 发送图像,在主函数中产生循环,至此,图像数据的发送完成。

至此,FPGA 部分图像的采集、预处理与传输功能完成。开发板硬件连接如图 10 所示。



Figure 10. ZYNQ7020 hardware connection diagram
图 10. ZYNQ7020 硬件连接图

3. 基于 YOLO-v3 的水果识别算法

YOLO (you only look once)是由 Redmon [5]等人在 2015 年提出的端到端的目标检测系列模型,自此

之后学界陆续推出更新换代的数个版本, 包括 YOLOv1、YOLOv2 [6]和 YOLOv3 [7]等, 其中 YOLOv3 以 YOLOv2 算法为基础, 对多个部分融入了改进内容。借鉴特征金字塔网络的思路, 利用三级尺度做预测[3], 能较好地满足实时检测任务的精度和速度要求, 检测小物体的能力也得到很大提升。

3.1. YOLO-v3 识别算法研究

YOLOv3 检测模型主要由骨干网络和检测网络两部分构成, 引用 Darknet-53 网络的前 52 层作为用于特征提取的骨干网络, 去掉 Darknet-53 的全连接层, 构成全卷积网络, 引入了 ResNet 残差网络结构, 缓解深层次神经网络中梯度消失的问题, 去除全连接层, 降低池化对梯度的影响, 下采样采用步长为 2 的卷积层实现。采用 Faster R-CNN 中的特征金字塔结构作为检测网络, 尽可能减少特征损失, 提高检测精度。YOLOv3 的网络结构如图 11 所示, 相对于 YOLOv2 具有更深的卷积层。其中 Convolutional 由卷积层、批量归一化和 leaky RELU 激活层构成, residual 表示残差结构, conv2d 表示卷积操作, up sampling 表示上采样, concatenate 表示两组进行拼接操作[3]。

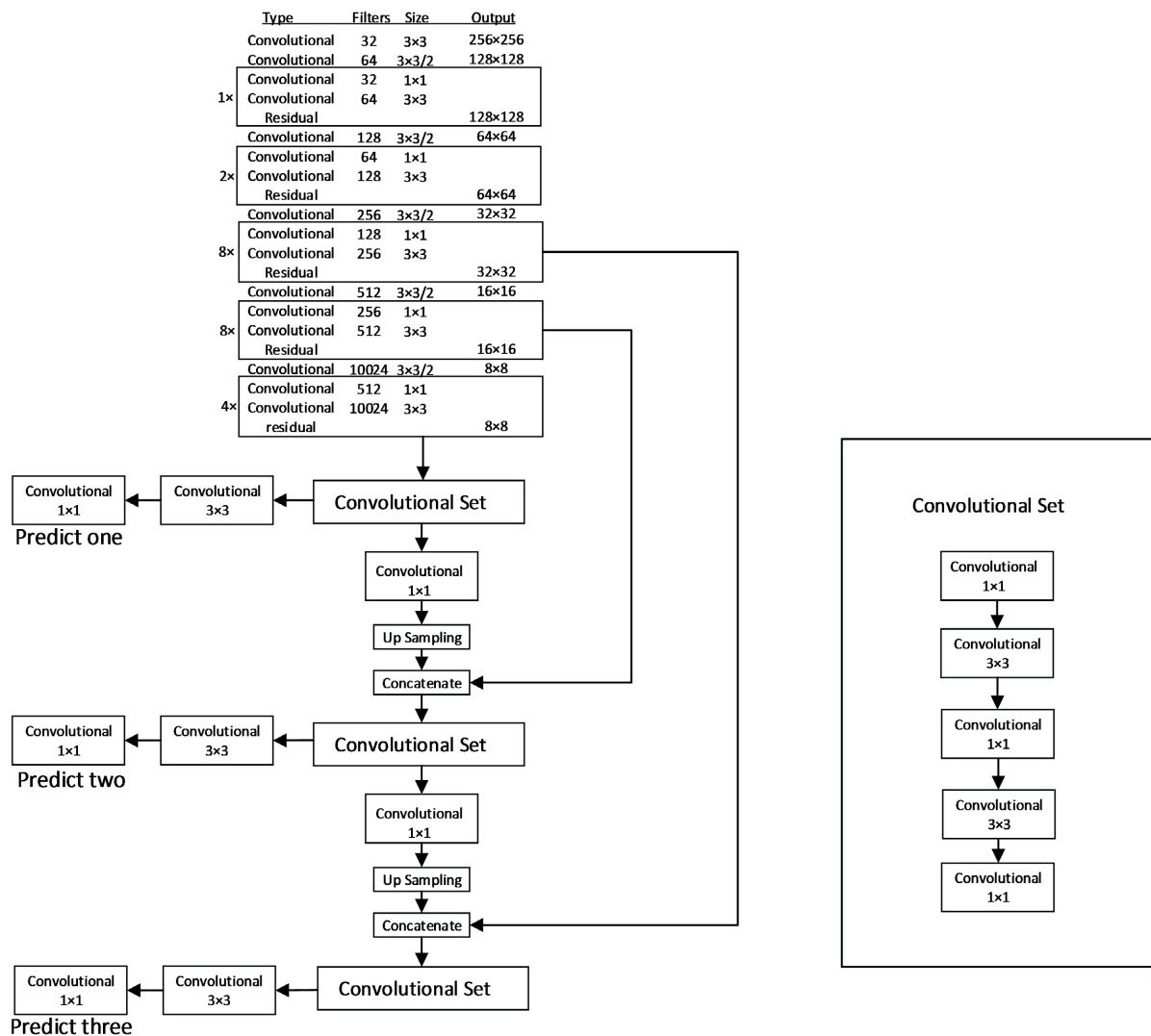


Figure 11. YOLOv3 network structure diagram

图 11. YOLOv3 网络结构图

3.2. 轻量级网络 MobileNetV3

轻量级网络的核心是对卷积神经网络设计出更加高效的网络计算方式, 通过此方法可在不损失网络性能的情况下减少网络参数数量[3], 提升网络识别速度, 以便后续研究中实现在移动终端或嵌入式设备中的应用。

目前主流的轻量级网络有 SqueezeNet, MobileNetV3 以及 ShuffleNet 网络, 能够很好的在移动端和嵌入式设备中运用, 通过比较目前主流的轻量级网络, 选择了 MobileNetV3 网络对 YOLOv3 算法的网络架构进行修改。MobileNetV3 网络是由 MobileNetV1 和 MobileNetV2 改进所得到的网络, 主要技术包括以下六点:

- 1) 引入深度可分离卷积;
- 2) 使用激活函数 RELU6;
- 3) 引入线性瓶颈的倒残差结构;
- 4) 引入基于 SE 结构的轻量级注意力模型;
- 5) 使用激活函数 h-swish(x);

6) 利用了关于网络结构搜索的方法, 在此方法中可发现由于中间几层开销较大, 针对此问题将平均池化层进行前移, 将特征图升维, 节省了大量计算时间, 并避免了瓶颈层中的升维操作, 减少计算量。修改后的结构如图 12 所示。

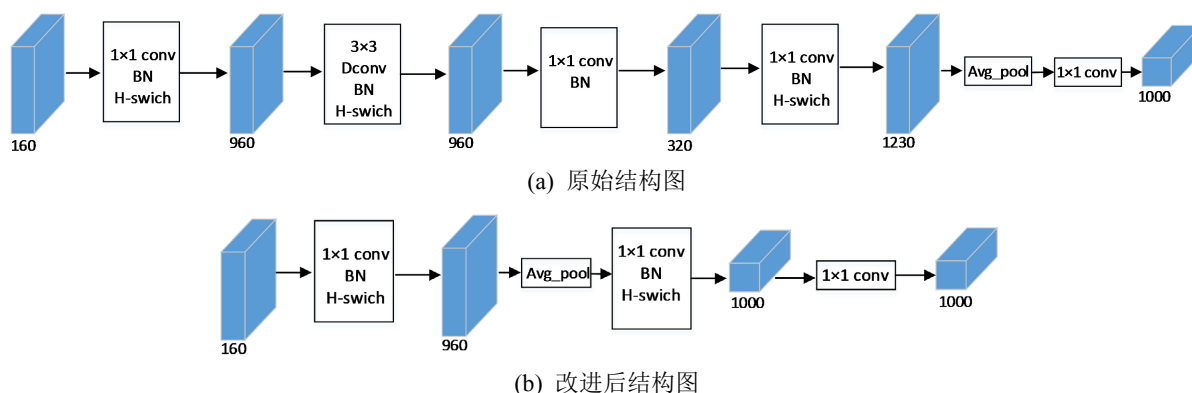


Figure 12. Network structure diagram comparison

图 12. 网络结构图对比

3.3. YOLO-v3 算法的改进

本系统采用 YOLOv3 算法作为水果识别技术的基础算法, 但由于 YOLOv3 模型体积大且参数量多, 仍然存在速度较低、无法迁移至嵌入式设备中等问题。为适应水果识别机器人的应用场景, 在实际应用中根据实际抓取过程中获得果实的真实坐标, 解决目前水果识别与分类算法的局限性, 本设计通过对比目前主流的轻量级网络, 选择将 MobileNet 轻量级网络引入 YOLO-v3 目标检测网络中, 得到 MobileNetV3-YOLOv3 模型, MobileNetV3 使用深度可分离卷积来构建模型, 采用 SE 注意力机制提升模型的特征表达能力, 因此 MobileNetV3-YOLOv3 模型不仅具有 YOLOv3 模型的优点, 模型计算量与参数量也大大减少, 能够应用在多数嵌入式设备中。

改进后的 MobileNetV3-YOLOv3 算法

通过将 YOLOv3 中 Darknet-53 网络修改为 MobileNetV3 网络, 模型识别速度得到一定提升, 但由于

存在部分信息丢失的情况, 因此继续对 MobileNetV3-YOLOv3 网络做进一步的改进。

根据 MobileNetV3-YOLOv3 的网络结构, 增加更深层的网络, 提高果实识别的准确率。利用大特征图可以更好的检测小目标的特点, 将第三级卷积结果继续进行上采样, 将尺度由 52×52 扩大为 104×104 , 再将得到的结果与 104×104 尺寸的特征图拼接, 从小目标中得到更细粒度的特征及位置信息[3]。增加的部分作为第四级预测, 其中包含 104×104 , 52×52 , 26×26 , 13×13 四个预测尺度, 通过多尺寸预测, 能够加强对果实的识别与定位效果。同时通过对先验框策略进行选择, 得到适合果实数据集的先验框, 并对学习率、批大小进行调优, 以获得最佳的训练效果[3]。

input	operator	exp size	#out	SE	NL	s	output
416, 416, 3	conv2d		16		HS	2	208, 208, 16
208, 208, 16	bneck, 3×3	16	16		RE	1	208, 208, 16
208, 208, 16	bneck, 3×3	64	24		RE	2	104, 104, 24
104, 104, 24	bneck, 3×3	72	24		RE	1	104, 104, 24
104, 104, 24	bneck, 5×5	72	40	✓	RE	2	52, 52, 40
52, 52, 40	bneck, 5×5	120	40	✓	RE	1	52, 52, 40
52, 52, 40	bneck, 5×5	120	40	✓	RE	1	52, 52, 40
52, 52, 40	bneck, 3×3	240	80		HS	2	26, 26, 80
26, 26, 80	bneck, 3×3	200	80		HS	1	26, 26, 80
26, 26, 80	bneck, 3×3	184	80		HS	1	26, 26, 80
26, 26, 80	bneck, 3×3	184	80	✓	HS	1	26, 26, 80
26, 26, 80	bneck, 3×3	480	112	✓	HS	1	26, 26, 112
26, 26, 112	bneck, 3×3	672	112	✓	HS	1	26, 26, 112
26, 26, 112	bneck, 5×5	672	160	✓	HS	2	13, 13, 160
13, 13, 160	bneck, 5×5	960	160	✓	HS	1	13, 13, 160
13, 13, 160	bneck, 5×5	960	160		HS	1	13, 13, 160
13, 13, 160	conv2d, 1×1		960		HS	1	13, 13, 960

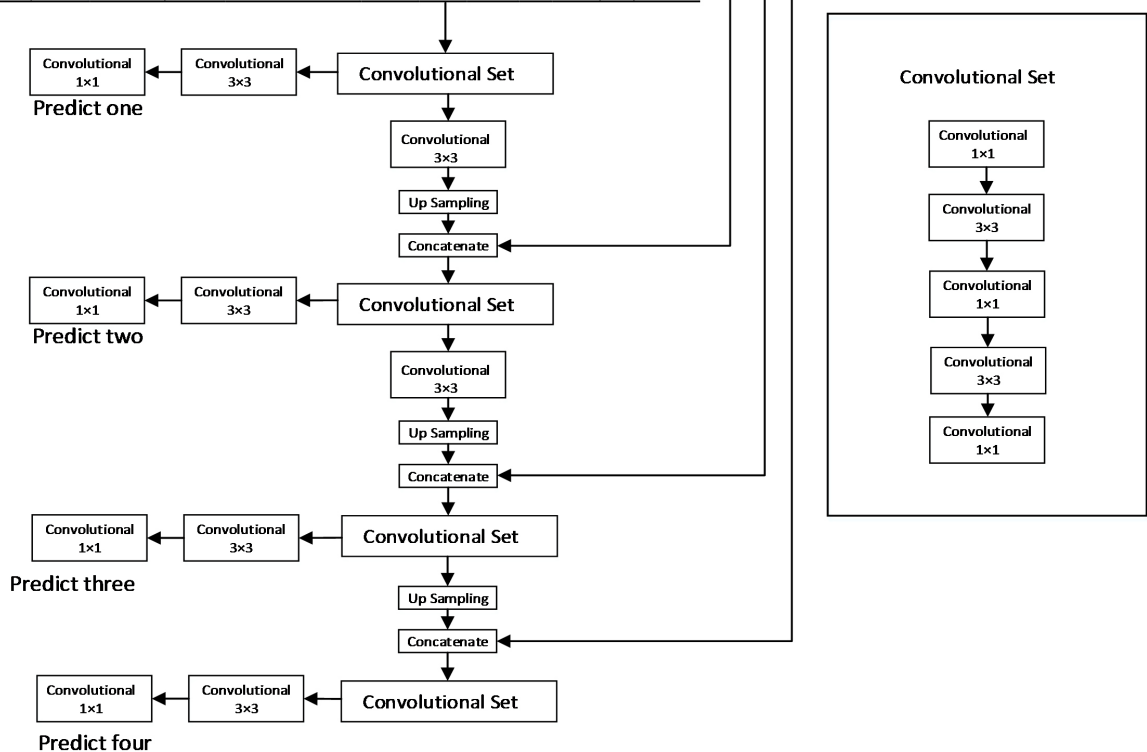


Figure 13. The improved MobileNetV3-YOLOv3 network structure

图 13. 改进后的 MobileNetV3-YOLOv3 网络结构

最终改进后的网络架构不仅模型参数大量减少, 均值识别精度达到 92%, 相较于原算法有所提升, 识别帧数达到 34 f/s, 且利用摄像机拍摄果实得到内参、外参、像素坐标深度信息, 转化为水果的三维坐标, 实现果实的精准定位, 改进后的网络结构图如图 13 所示。

4. 实验与结果分析

收集苹果、橘子、香蕉三类水果图片, 并以 8:2 的比例划分训练集, 验证集, 制作成 COCO 模式, 并利用 COCO 评价指标对模型效果进行评价。在 COCO 数据集上进行预测并搭建基于卷积神经网络的 YOLOv3 模型, Faster R-CNN 模型和 MobileNetV3-YOLOv3 模型, 并对改进后网络精度、网络大小和识别速度与改进前网络、传统识别网络进行对比, 其中 mAP50 为 IOU 取 0.5 时的 AP 值。由表 5 可知, 改进后的模型 AP50 可达到 92%, 识别帧数可达 34 f/s, 网络参数量为 235.61 M, 相较于传统目标识别网络不仅检测精度与识别速度有所提高, 且减少模型参数量, 有助于后续实现在嵌入式平台的部署。

Table 5. Comparison of accuracy and speed of different network models
表 5. 不同网络模型精度与速度对比

网络名称	mAP ₅₀ (%)	参数(M)	FPS (帧/秒)
MobileNetV3-YOLOv3	88.1	231.92	37
Faster R-CNN	88.2	384.02	22
YOLOv3	89.2	353.82	34
本算法	92.4	235.61	34

最后, 我们以苹果作为实验对象以实现水果的识别效果, 上位机接收板卡采集视频图像, 进行识别处理, 实验结果及水果坐标展示如图 14, 图 15 所示。

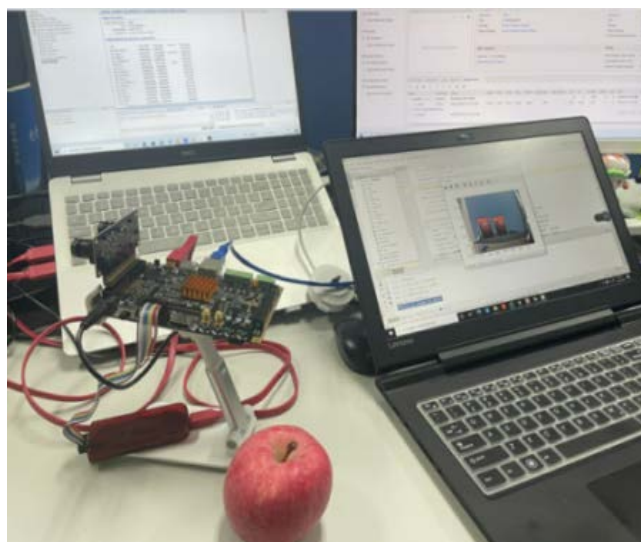


Figure 14. The experimental results of the video acquisition and transmission system

图 14. 视频采集传输系统实验效果

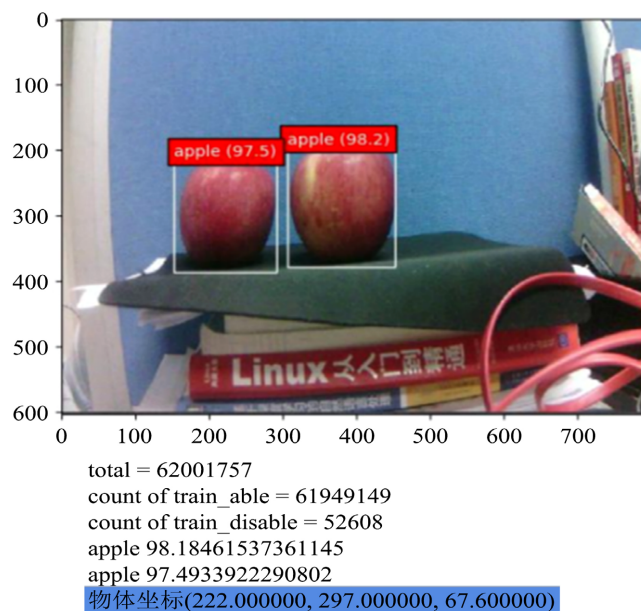


Figure 15. Fruit recognition and coordinate return effect
图 15. 水果识别与坐标回传效果

5. 结语

本文介绍了一种基于 ZYNQ 的水果识别系统, 通过软硬协同的方式, 在 FPGA 部分完成对水果图像的采集与传输, 在上位机使用改进后的 YOLOv3 网络进行水果图像的种类与位置信息检测与识别。实验结果显示, 本系统可实现实时的水果种类及位置信息识别, 且改进后的网络模型 AP50 可达到 92.4%, 模型大小仅为 235.61 M。本文研究联合软件算法与硬件平台实现水果识别, 并对算法进行轻量级改进, 未来的研究中, 可尝试对系统进行改进并移植至嵌入式设备中, 实现水果图像的实时采集与识别。

基金项目

西南民族大学中央高校基本科研业务费专项资金项目(No.2021NYYXS81); 四川省科技厅重点研发项目(No.2022YFG0377)。

参考文献

- [1] 杨雨诺. 基于 PYNQ 的图像分类识别技术研究与实现[D]: [硕士学位论文]. 南京: 南京邮电大学, 2020.
- [2] 刘培. 人员视频检测智能前端系统设计与实现[D]: [硕士学位论文]. 太原: 中北大学, 2022.
- [3] 尼留溪. 枣采摘机器人的目标识别技术研究[D]: [硕士学位论文]. 保定: 河北农业大学, 2021.
- [4] Girshick, R., Donahue, J., Darrel, T., et al. (2014) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 23-28 June 2014, 580-587. <https://doi.org/10.1109/CVPR.2014.81>
- [5] Redmon, J., Divvala, S., Girshick, R., et al. (2016) You Only Look Once: Unified, Real-Time Object Detection. 2016 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, 27-30 June 2016, 779-788. <https://doi.org/10.1109/CVPR.2016.91>
- [6] Redmon, J. and Farhadi, A. (2017) YOLO9000: Better, Faster, Stronger. 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, 21-26 July 2017, 7263-7271. <https://doi.org/10.1109/CVPR.2017.690>
- [7] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement. arXiv e-prints, 321-342.
- [8] 刘海敏. 基于 Zynq 的 MIMO 无线传输系统的设计与研究[D]: [硕士学位论文]. 呼和浩特: 内蒙古大学, 2018.
- [9] 车启谣, 严运兵. 基于改进 YOLOv3 的行人检测研究[J]. *智能计算机与应用*, 2022, 12(8): 8-13.