DOI: 10.13382/j.jemi.B2306585

面向 FPGA 部署的改进 YOLO 铝片表面缺陷检测系统*

戴伟杰 王衍学 李昕鸣 王祎颜
(北京建筑大学机电与车辆工程学院 北京 100044)

摘 要:在工业生产中,产品缺陷的智能检测是至关重要的。现场可编程门阵列(FPGA)是一种具有算力强、功耗低等特点的嵌入式设备,能够将小型卷积神经网络部署其中。本文基于 Xilinx Zynq 系列 FPGA 设计了一套改进 YOLOv2 目标检测算法,在模型框架中增加重排序层,对切片图进行并行计算处理后再重组,完成铝片表面缺陷的检测。该算法经过高层次设计(HLS)后,进行 RTL 转换与 IP 核封装,并导入到工程项目中完成 SoC 设计。通过综合、布局布线后生成比特流文件,导入至 PYNQ 镜像中,完成对铝片表面的工业缺陷检测。实验结果表明,本系统能够准确地检测出缺陷,并将功耗降低至 2.494 W。

关键词: FPGA;YOLOv2 算法;高层次综合设计;PYNQ;异构计算
中图分类号: TN98 **文献标识码**: A **国家标准学科分类代码**: 510.40

YOLO aluminum profile surface defect detection system for FPGA deployment

Dai Weijie Wang Yanxue Li Xinming Wang Yiyan
(School of Mechanical-Electronic and Vehicle Engineering, Beijing University of Civil Engineering and Architecture, Beijng 100044, China)

Abstract: In industrial production, intelligent detection of product defects is crucial. Field-programmable gate arrays (FPGAs) are embedded devices with features such as high arithmetic power and low power consumption that enable small convolutional neural networks to be deployed in them. In this paper, a set of improved YOLOv2 target detection algorithm is designed based on Xilinx Zynq series FPGAs, and a reordering layer is added to the model framework to complete the detection of surface defects on aluminum sheets by parallel computing processing of the slice map before reorganisation. The algorithm is designed at a high level (HLS), then RTL converted and IP cores are packaged and imported into the project to complete the SoC design. Generate bitstream files through comprehensive layout and wiring, import them into PYNQ images, and complete industrial defect detection on the surface of aluminum sheets. The experimental results show that this system can accurately detect defects and reduce power consumption to 2.494 W.

Keywords: FPGA; YOLOv2 algorithm; high-level synthesis; PYNQ; heterogeneous computing

0 引 言

在人工智能飞速发展的时代背景下,卷积神经网络(convolutional neural network,CNN)在医学图像处理、工业零件检测和自动驾驶等领域发挥重要作用。信息化下,数据开始作为人们亟待研究与处理的问题。自 20 世纪 90 年代起,各大互联网厂商纷纷开始建设自己的数据中心,以作为数据的算力载体。然而,由于摩尔定律的发展逐渐趋缓,CPU 与 GPU 的硬件条件成为了数据处理的瓶颈,数据处理开始朝着异构计算的方向发展。现场可编程门阵列(field programable gate Array,FPGA),作为一种高性能低功耗嵌入式设备,具有强大的并行计算能力与动态可重构性,能够满足卷积神经网络模型不断更新迭代的算力需求。文献[1]介绍了微软公司的 Catapult 的数据中心加速项目,通过将 FPGA 部署至自家数据中心的 1 632 台服务器中,对 Bing 搜索引擎的文件排名运算进行硬件加速,最终得到高达 95%的吞吐量提升。

收稿日期: 2023-06-01 Received Date: 2023-06-01
*基金项目:国家自然科学基金(51875032,52275079)、北京建筑大学研究生创新项目(PG2023131)资助

近年来,基于深度学习的机器视觉目标检测方法在工业领域得到广泛应用,研发人员的工作重心都用于提升算法复杂度,以完成更加复杂的工业故障诊断,但随之带来的是庞大的研发成本与时间成本,且算法的硬件实现更为困难。2016 年,Redmon 等[2]提出了 YOLO 目标检测方法,仅使用单个卷积神经网络即可对目标进行位置回归与类别预测。2022 年,王习东等[3]提出了基于 SSD 网络的人体目标跟踪系统,验证了 FPGA 能完成神经网络部署的可行性。文献[4]提出了一种目标检测算法 YOLOv2,网络结构参照 YOLO 与 SSD[5],借鉴了 SSD 对多尺度特征处理生成目标先验框的策略,对小尺寸目标的检测精度有所提高,同时解决了 YOLO 对于物体定位不够准确的缺点,满足本次工业检测系统设计的算法要求。

目前,基于深度学习的工业缺陷诊断研究[6]大多限于验证软件算法的有效性和可行性,并未进行实际硬件部署[7]。据此,本文的研究旨在使用轻量化卷积神经网络模型 YOLOv2,并结合 FPGA 的硬件特性[8]对算法模型进行改进。同时,本文将整个模型框架部署在嵌入式边缘计算设备 FPGA 上[9-11],以实现对铝片表面工业缺陷的检测与诊断。此外,通过 FPGA 设备板载的 RJ-45 以太网接口将其连入局域网,使用局域网内的 PC 上位机登录设备,即以可视化界面操作的方式完成工业诊断。这种设备不仅具备轻量化和便携化的优势,而且能够实现高效的检测性能。

1　算法原理与硬件选型

1.1　卷积神经网络基本概念

卷积神经网络一般包含卷积层、池化层和全连接层。卷积层的计算主要将权重 w_{ij} 构成的卷积核在输入特征层 x_j^{in} 上滑动,相乘并累加得到输出特征层 x_i^{out}。卷积计算需要大量的乘累加运算,公式如下:

$$x_i^{out} = \sum_{j=1}^{N} x_j^{in} * w_{ij} + b_i \cdots, 1 \leqslant i \leqslant M \tag{1}$$

式中:b_i 为偏置,N 为输入特征层个数,M 为输出特征层个数。

池化层一般使用最大值池化对输入特征进行降采样,减少数据量的同时增加鲁棒性。

全连接层对输入进行特征空间的线性转换,对输入特征进行加权从而得到输出,全连接层可由卷积层操作实现:

$$x^{out} = \sum_{j=1}^{N} x_j^{in} w_{ij} + b \tag{2}$$

1.2　YOLOv2 目标检测算法

基于深度学习的物体检测与识别常用的网络有 R-CNN、YOLO 系列、SSD 等,其中 SSD 的应用较为广泛,但结构复杂且参数数量多,速度上不如 YOLO。图 1 展示了在同一数据集下各检测算法的速度与精度情况。

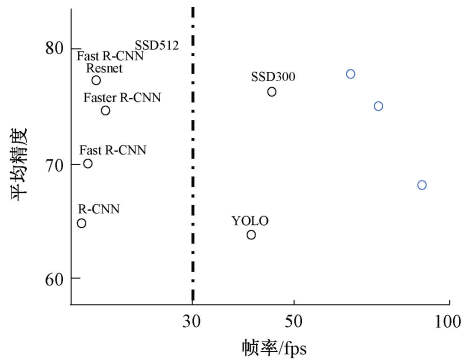


图 1　检测算法速度与精度对比

Fig. 1　Comparison of algorithm speed and accuary

传统 YOLOv2 使用的主干框架为改进的 Darknet-19,其中包含 19 个卷积层、1 个平均池化层、5 个最大值池化层和一个 softmax 层组成[12]。改进后 YOLOv2 的网络结构模型框架如图 2 所示,包含 4 种类型的层:卷积层、最大池化层、路由层和重排序层。卷积层的功能为特征提取,得到特征图,模型的大部分运算量来源于卷积层。池化层负责像素抽样,即实现特征图的降采样。路由层完成多层次特征融合,最后由重排序层对特征抽样排序。模型网络结构由 23 个卷积层(conv)、5 个最大池化层(maxpool)、1 个重排序层(reorg)、2 个路由层(route)及一个输出层(detection)组成,共计 32 层。

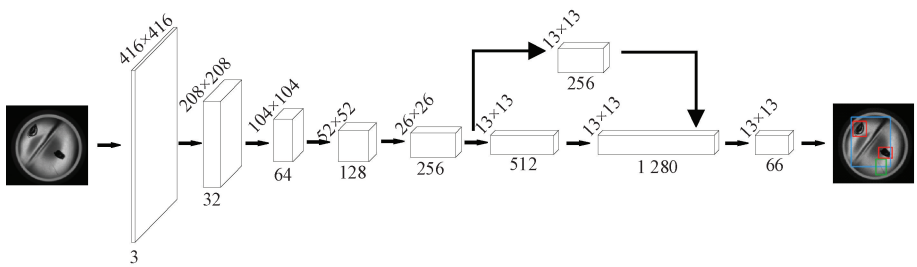


图 2　模型框架

Fig. 2　Model framework

总的来说,改进 YOLOv2 算法主要分为如下 3 步来完成:

1)图片预处理:对于任意图片尺寸的输入,首先要进行图片大小调整,转换为 416×416×3 的尺寸后,再进行归一化 RGB 处理,同时由于该算法最终部署在嵌入式设备端,还需再做一次数据量化,将 32 位浮点数转换为 16 位定点数,完成数据精度压缩。

2)网络检测:对图片预处理部分 416×416×3 的输出作为该部分的输入,使用了卷积层降采样,使得输入卷积网络的 416×416 图片最终得到 13×13 的卷积特征图(416/32=13),即对原始图片的行列进行 13 等分,单通道上 13 pixels×13 pixels 里的每一点都代表原先 416×416 图片中的一块切片区域。

3)图片后处理(预测):对这些切片进行特征计算后再重组为整张图片,完成整个特征提取后进行目标框选和输出。

1.3　Zynq FPGA 异构平台

本次系统设计的硬件为一款基于 Xilinx FPGA Zynq-7020 的开发板,Zynq 采用片上系统(system on chip,SoC)设计,集成双核 ARM Cortex A9 处理器和 Xilinx Artix-7 系列 FPGA,即 ARM+FPGA 的异构计算方案,Zynq-7000 SoC 架构如图 3 所示。

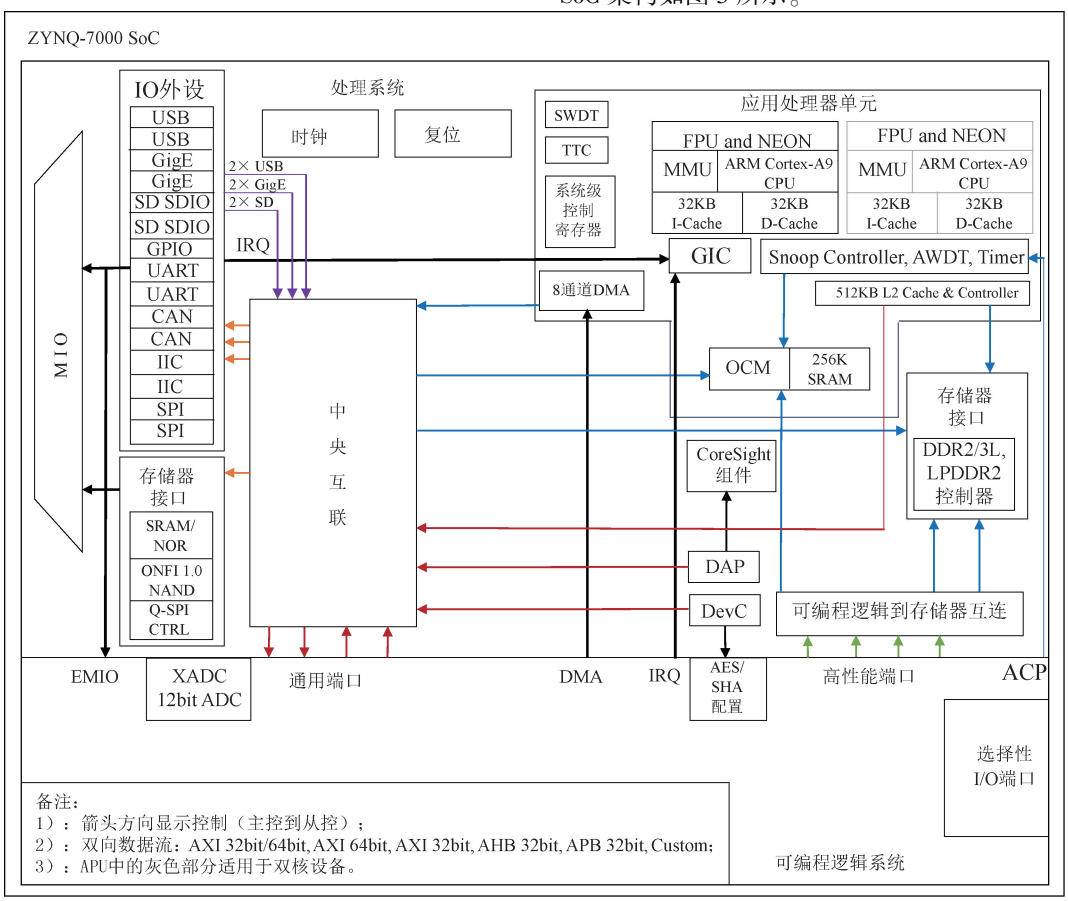


图 3　Zynq-7000 SoC 架构

Fig. 3　Zynq-7000 SoC overview

Zynq 中的 ARM 处理器与部分片内存储资源,通常被称为处理器系统(processor system,PS)端,FPGA 控制的逻辑资源端被称为可编程逻辑(programable logic,PL)端。芯片内部 PS 端与 PL 端之间采用 AXI 总线进行数据交互。AXI 总线是由 ARM 公司提出的一种高性能、高带宽且低延时的片内总线,在高性能 SoC 设计中被大量使用。

技术人员可通过 Zynq 进行软硬件协同开发,相比于传统 FPGA 芯片,能胜任更加复杂的嵌入式系统任务[13-20]。

1.4　PYNQ 开发框架

PYNQ(Python on Zynq),是 Xilinx 公司推出的一种开放源代码框架,PYNQ 的原理是在 Zynq 设备上的 ARM 处理器中运行 Linux 系统,并且利用 C-Python 作为解释器,实现 Python 代码的支持。同时,PYNQ 在设备的 ARM 处理器中运行了 Notebook 服务器,用户可通过 Web 程序连接到 PYNQ 开发环境完成交互式计算。

这种全新的开发框架使得设计人员能够使用 Python

进行快速的 FPGA 部署,利用 Zynq 中可编程逻辑 PL 和微处理器 PS 的优势来快速构建高性能嵌入式应用程序,并且在部署过程中无需研究硬件的实现细节,降低了软件开发者对于硬件的开发难度。

2　系统设计与实现

2.1　硬件架构设计

根据 Zynq 平台的异构特性,将本次设计分为硬件设计与软件设计 2 部分。硬件端主要通过 HLS 进行 YOLOv2 IP 核构建与封装,再将该 IP 核导入硬件工程中完成整个 Block Design 设计与验证,最终生成比特文件供软件端调用。软件端主要对铝片表面工业数据集进行参数训练与权重文件导出,最后在 PYNQ 镜像中完成软硬件协同测试。

本次设计的硬件型号与软件开发环境如表 1 所示。

表 1　硬件开发环境与工具

Table 1　Hardware development environment and tools

名称	型号
硬件平台	ZYNQ7020
芯片型号	XC7Z020-CLG400-
开发工具	Vivado 2019. 2
软件开发平台	Jupyter Notebook
IP 设计平台	Vivado HLS 2019. 2
功耗测试	Vivado 自带功耗测试工具

首先使用 C++在 Vivado HLS 中进行 YOLOv2 IP 核设计,该软件平台可以将高级编程语言如 C、C++、System C 代码自动转换为硬件描述语言(Verilog 或 VHDL 文件),对源文件进行代码的正确性验证以及添加约束后,将其封装为 IP 核,以供硬件端调用,HLS 的开发流程如图 4 所示,最后封装完成的 IP 核如图 5 所示。

完成 IP 核设计后,接下来进行整个系统的硬件搭建过程。使用 Vivado 2019. 2 创建 Zynq7020-clg400-2 的开发板工程,在工程中使用 Block Design 硬件设计平台进行架构设计。首先添加 Zynq IP 核并配置对应参数,如时钟频率、复位信号、DDR、外围 IP 引脚、PS-PL 交互等,再加入由 HLS 导入的 YOLO IP 核,将其与 Zynq IP 核进行连接。连接使用了 Xilinx 官方提供的 AXI SmartConnect IP 核,ZYNQ 系统通过该 IP 核进行 PL 端通信和数据交互,将 S_AXI_HP 端口与 m_axi_DATA_BUS 端口互通,即建立与 YOLOv2 IP 核的连接。最终硬件搭建如图 6 所示。

完成 Block Design 搭建后,下一步进行模型验证,检查是否存在逻辑错误与布线错误。当验证通过后即可使用 Create HDL Wrapper 进行 HDL 文件转化,然后才能对

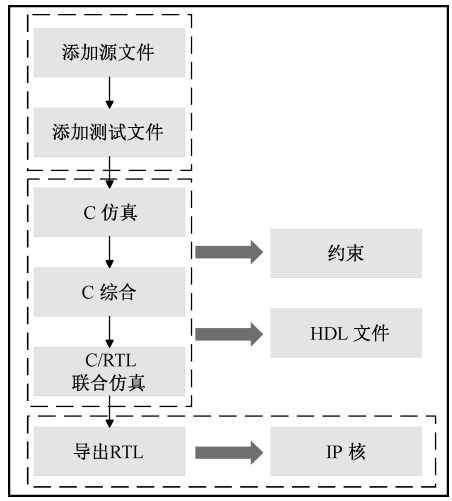


图 4　HLS 设计流程

Fig. 4　HLS design process

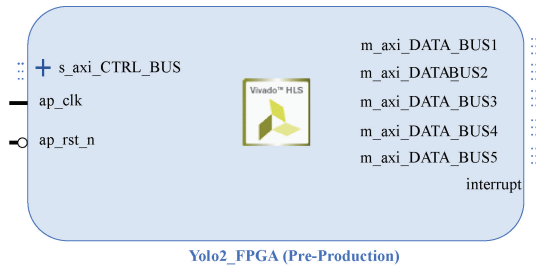


图 5　YOLOv2 IP 核

Fig. 5　YOLOv2 IP core

该工程进行综合,将所设计电路代码翻译成门级网表的形式,最终生成和导出 bit 文件。值得注意的是,PYNQ 系统在进行硬件解析时,除 bit 文件外还需 tcl 文件(硬件工程中. runs 目录下)且两者命名要一致,才能完成整个硬件电路的映射工作。

2.2　检测系统实现

检测系统框架图如图 7 所示。

使用 balenaEtcher 工具将 PYNQ-Z2 系统镜像烧入到 SD 卡中,通过跳针将板卡设置为 SD 卡启动,待板卡上电后将自动启动 PYNQ 系统,上位机可通过 Putty 串口助手访问开发板搭载的 Linux 系统,同一局域网下的 PC 可通过 IP 地址(Linux 系统输入 ifconfig 命令查询网络 IP)及对应端口号登陆的方式进入该系统,进行相应的程序开发与文档编写工作。

本次训练采用的数据集来自 Baidu Paddle AI Studio,如图 8 所示,通过海康工业相机采集铝片表面工业缺陷,包含了针孔、脏污、褶皱、划伤 4 个类别的缺陷目标,整个数据集中缺陷数量达到 1 000 个以上。

使用 Samba 共享文件夹的方式,将检测系统所需的

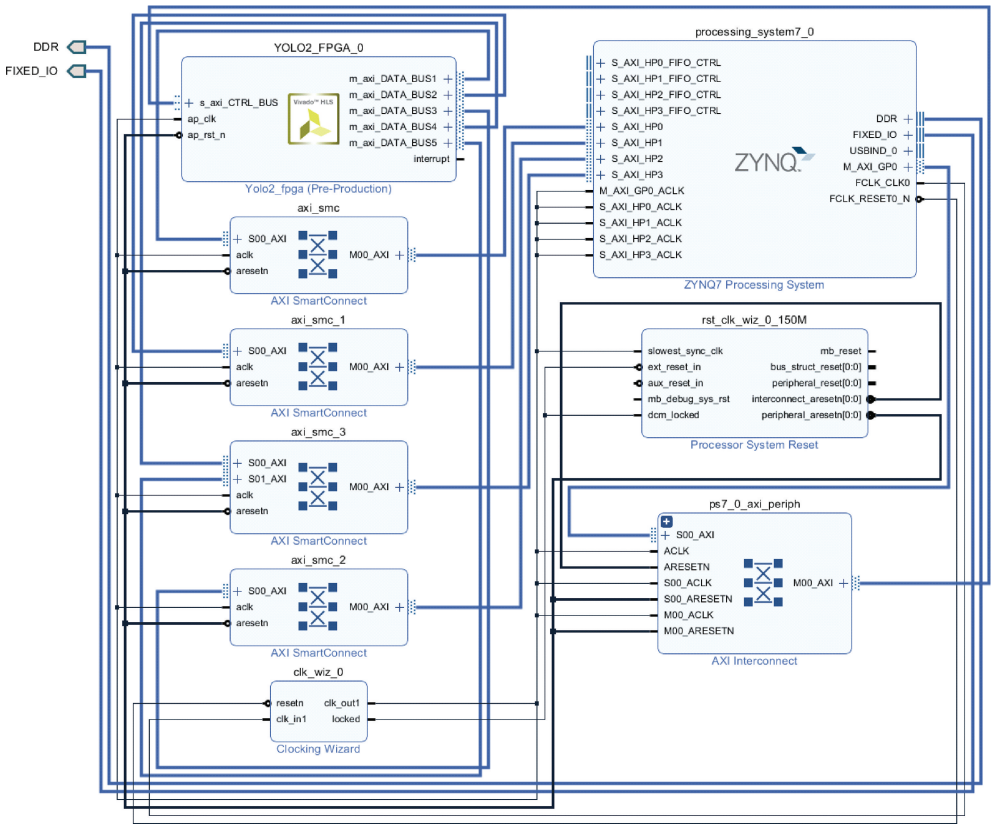


图6　Block Design 搭建

Fig. 6　Block Design construction

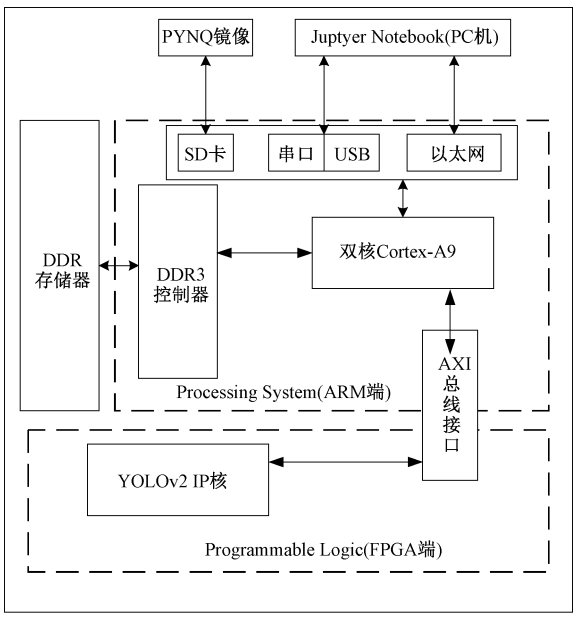


图7　检测系统框架图

Fig. 7　Framework diagram of detection system

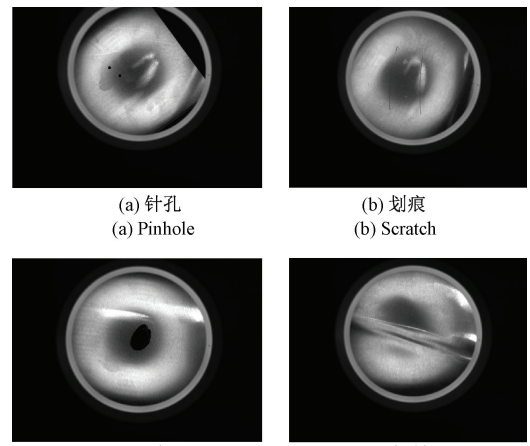


图8　数据集展示

Fig. 8　Dataset presentation

硬件配置、训练权重、检测集等文件放入 PYNQ，在 Jupyter Notebook 中登陆已配置完成的开发板后，运行该缺陷检测系统，检测结果如图9所示，可以看出该系统能准确识别出缺陷并进行标注。

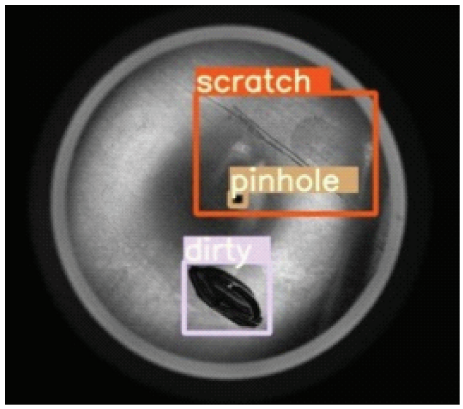


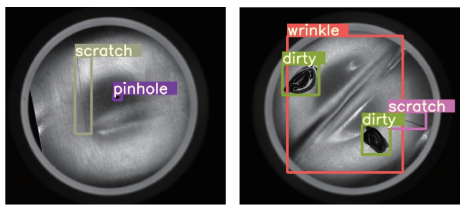
图 9　系统检测结果

Fig. 9　System detection results

3　实验分析

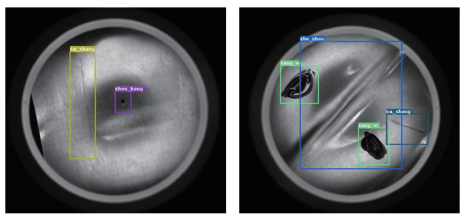
3.1　对比实验

使用该缺陷系统与基于 Paddle 的 SSD 算法进行对比检测，结果如图 10 所示，可以看出本系统能较好地识别出各类缺陷并进行标注。



(a) FPGA_YOLO检测效果

(a) FPGA_YOLO detection effect



(b) Paddle_Detection SSD检测效果

(b) Paddle_Detection SSD detection effect

图 10　各系统效果对比

Fig. 10　Comparison of system effects

3.2　性能分析

基于 PYNQ 框架实现的铝片缺陷检测系统，其逻辑资源池消耗如表 2 所示，由于算法的复杂性及图像数据传输，导致 LUT、BRAM、DSP 资源占用较大，后期仍有较大优化空间。

系统总功耗为 2.494 W，主要分为动态功耗和静态功耗，各参数见表 3 所示。其中动态功耗占 90%以上，这是由于在整个前向推理计算中，涉及大量的 PS 与 PL 端通信、相关硬件库调用、PS 与存储器之间数据交互。而静态功耗仅为晶体管物理特性泄露、时钟单元、I/O 引脚等，是避不可免的一小部分。综合来看，本次系统设计相较于 GPU 而言，还是大大降低了系统功耗，满足嵌入式硬件系统低功耗的要求。

表 2　逻辑资源池消耗

Table 2　Logical resource pool consumption

资源类型	消耗量	总量	消耗率/%
LUT	32 396	53 200	60.89
LUTRAM	6 156	17 400	35.38
FF	27 486	106 400	25.83
BRAM	87.50	140	62.50
DSP	151	220	68.64

表 3　系统功耗

Table 3　System Power consumption

功耗参数	功耗/W
系统静态	0.178
系统动态	2.316
总片上	2.494

以 Zynq7020 硬件平台部署的改进 YOLOv2 铝片表面缺陷检测系统，在 PYNQ 框架的实现过程中，关于处理速度的性能报告如表 4 所示。

表 4　系统运行时间报告

Table 4　System runtime report

系统运行	耗时/s
图像预处理	0.165 3
加载内存	0.018 5
FPGA 处理	1.008 1
网络检测	0.475 2
图像后处理	1.083 3
总计	2.750 4

4　结　论

本文基于 FPGA 设计了一套改进 YOLOv2 目标检测算法，根据 FPGA 的并行计算特性，对输入图像进行切片处理，再由网络模型中增设的重排序层进行重组并输出，最终成功部署于 Zynq7020 嵌入式设备端，完成铝片表面缺陷检测。通过 Xilinx 官方的 HLS 开发套件对检测算法进行 IP 核封装，完成整个硬件架构的设计与验证。检测系统选用基于 Linux 的 PYNQ 框架，可通过 PC 登陆网络 IP 的方式，实现对工业相机拍摄抓取图片的缺陷检测与

定位。实验结果表明,该系统在准确完成检测的同时能将功耗大幅度降低,系统功耗2.494 W,能耗比仅为GPU(Nvidia 1650super 4 G,75 W)平台的0.033,整个系统的运行时间为2.75 s,后期仍有较大优化空间,优化后可考虑将该缺陷检测系统部署于工业生产线中,提供实时故障诊断。

参考文献

[1] ANDREW P, ADRIAN M, CAULFIELD, et al. A reconfigurable fabric for accelerating large-scale datacenter services[J]. Communications of the ACM, 2016, 59(11):13-24.

[2] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2016:779-788.

[3] 王习东,王国鹏,王保昌,等. 基于FPGA与退化YOLO的手机镜片缺陷检测系统[J]. 电子测量技术,2022,45(18):10-17.
WANG X D, WANG G P, WANG B CH, et al. Mobile phone lens defect detection system based on FPGA and degraded YOLO [J]. Electronic Measurement Technology, 2022, 45 (18): 10-17.

[4] REDMON J, FARHADI A. YOLO 9000: Better, faster, stronger[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2017: 6517-6525.

[5] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single shot multibox detector[C]. European ConferenSce on Computer Vision, 2016:21-37.

[6] 蔡东吟. 基于深度学习的铝型材表面缺陷检测方法研究[D]. 重庆: 重庆交通大学,2021.
CAI D Y. Research on surface defect detection method of aluminum profile based on deep learning [D]. Chongqing:Chongqing Jiaotong University,2021.

[7] 张训飞. 基于语义分割的钢材缺陷检测算法研究[D]. 杭州:浙江大学,2023.
ZHANG X F. Research on steel defect detection algorithm based on semantic segmentation [D]. Hangzhou: Zhejiang University,2023.

[8] CROCKETT L H. The Zynq Book: Embedded Processing with the ARM Cortex-A9 on the Xilinx Zynq-7000 All Programmable SoC [M]. Strathclyde Academic Media, 2014.

[9] 吴健,顾明剑,曾长紊,等. 基于ZYNQ的卷积神经网络加速器设计[J]. 计算机工程与设计,2022,43(6):1572-1581.
WU J, GU M J, ZENG CH W, et al. Accelerated design of convolutional neural network based on ZYNQ[J]. Computer Engineering and Design, 2022, 43 (6): 1572-1581.

[10] 陈辰,柴志雷,夏珺. 基于Zynq7000 FPGA异构平台的YOLOv2加速器设计与实现[J]. 计算机科学与探索,2019,13(10):1677-1693.
CHEN CH, CHAI ZH L, XIA J. Design and implementation of YOLOv2 accelerator based on Zynq7000 FPGA heterogeneous platform[J]. Journal of Frontiers of Computer Science and Technology, 2019, 13(10):1677-1693.

[11] 周彦臻,吴瑞东,于潇,等. 面向FPGA部署的CNN-SVM算法研究与实现[J]. 电子测量与仪器学报,2021,35(4):90-98.
ZHOU Y ZH, WU R D, YU X, et al. Research and implementation of CNN-SVM algorithm based on FPGA[J]. Journal of Electronic Measurement and Instrumentation, 2021,35(4):90-98.

[12] 张烈平,李智浩,唐玉良. 基于迁移学习的轻量化YOLOv2口罩佩戴检测方法[J]. 电子测量技术,2022,45(10):112-117.
ZHANG L P, LI ZH H, TANG Y L. Light-YOLOv2 mask wearing detection method based on transfer learning[J]. Electronic Measurement Technology, 2022, 45 (10): 112-117.

[13] 周铮杰. 基于PYNQ的神经网络加速器研究[D]. 哈尔滨:哈尔滨工程大学,2021.
ZHOU ZH J. Research on neural network accelerator based on PYNQ [D]. Harbin: Harbin Engineering University, 2021.

[14] 高振. 基于FPGA的目标检测加速器设计[D]. 杭州:杭州电子科技大学,2022.
GAO ZH. Target detection accelerator design based on FPGA [D]. Hangzhou: Hangzhou Dianzi University,2022.

[15] 缪丹丹,张鹏,张鑫宇,等. 基于ZYNQ平台的通用卷积加速器设计[J]. 国外电子测量技术, 2022, 41(11): 72-77.
MIAO D D, ZHANG P, ZHANG X Y, et al. Generalized convolutional accelerator design based on ZYNQ platform [J]. Foreign Electronic Measurement Technology, 2022, 41 (11): 72-77.

[16] 卫建华,刘润利,许佳豪,等. 基于PYNQ框架的人体目标跟踪系统[J]. 国外电子测量技术, 2021, 40(12): 89-95.
WEI J H, LIU R L, XU J H, et al. Human target tracking system based on PYNQ frameword[J]. Foreign

Electronic Measurement Technology, 2021, 40(12): 89-95.

[17] 郑翔文. 基于 FPGA 的智能舌诊系统设计[D]. 西安: 西安电子科技大学,2022.
ZHENG X W. Design of intelligent tongue diagnosis system based on FPGA [D]. Xi′an: Xidian University,2022.

[18] 徐言. 基于 FPGA 云加速的智能小车设计及其在特定场景下的应用研究与实现[D]. 南京: 东南大学,2019.
XU Y. Design and implementation of intelligent car based on FPGA cloud acceleration and its application in specific scenes[D]. Nanjing: Southeast University,2019.

[19] 吴帅校. Faster R-CNN 目标检测网络算法压缩和 FPGA 实现[D]. 北京: 北京交通大学,2021.
WU SH X. Implementation with FPGA and compression of Faster R-CNN object detection network algorithm [D]. Beijing: Beijing Jiaotong University,2021.

[20] 欧俊宏. 基于 FPGA 车载计算平台的目标检测算法设计及优化[D]. 成都: 电子科技大学,2021.
OU J H. Design and optimization of object detection algorithm on FPGA computing platform for autonomous vehicle [D]. Chengdu: University of Electronic Science and Technology of China, 2021.

作者简介

戴伟杰,2021 年于重庆交通大学获得学士学位,现为北京建筑大学硕士研究生,主要研究方向为 FPGA 开发、图像与信号处理。
E-mail: davi_30@163.com

Dai Weijie received his B. Sc. degree from Chongqin Jiaotong University in 2021. Now he is a M. Sc. candidate at Beijing University of Civil Engineering and Architecture. His main research interests include FPGA development, image and signal processing.

王衍学(通信作者),2009 年于西安交通大学获得博士学位,2010~2011 年加拿大渥太华大学博士后,现为北京建筑大学教授、博导,主要研究方向为装备故障诊断与智能维护、剩余寿命与健康管理及信号处理与特征提取等。
E-mail: wyx1999140@126.com

Wang Yanxue(Corresponding author) received his Ph. D. degree from Xi′an Jiaotong University in 2009 and postdoctoral fellow at the University of Ottawa in Canada from 2010 to 2011. Now he is a professor and Ph. D. supervisor of Beijing University of Architecture. His main research interests include equipment fault diagnosis and intelligent maintenance, RUL prognosis and health management, signal processing and feature extraction etc.