光电技术及应用

DOI：10.16818/j.issn1001－5868.2022111601

基于 Zynq 平台的图像目标检测系统

王利翔[1]，林珊玲[1,3]*，林志贤[1,2,3]，郭太良[2,3]

（1. 福州大学 先进制造学院，福建 泉州 362200；2. 福州大学 物理与信息工程学院，福州 350116；
3. 中国福建光电信息科学与技术实验室，福州 350116）

摘　要： 针对图像目标检测任务中采用的深度学习网络复杂的计算和规模庞大的计算参数，导致基于 ARM 架构的嵌入式系统上，目标检测任务存在着高延时和处理速度慢的问题，文章提出并设计实现了一种新型完整嵌入式道路车辆检测方案。该方案在基于 YOLOv3-Tiny 的特征提取网络中采用结构重参数化的方法提升模型检测精度，并通过 Vitis-AI 在 Zynq 嵌入式平台上部署 DPUCZDX8G 架构的加速核对卷积神经网络的并行加速，最后将改进的 YOLOv3-Tiny 网络模型经过量化、编译，以动态链接库的方式部署。实验结果表明，在 VOC2007 上测试最终实现均值平均精度（MAP）为 0.597，实时处理速度为 27.7 FPS，同时帧率功耗比为 1.49，适合边缘计算设备的低功耗要求。

关键词： 目标检测；Vitis-AI；量化；结构重参数化；Zynq

中图分类号： TP394.1；TH691.9　**文章编号：** 1001－5868(2023)01－0147－06

Image Target Detection System Based on Zynq Platform

WANG Lixiang[1], LIN Shanling[1,3], LIN Zhixian[1,2,3], GUO Tailiang[2,3]

(1. School of Advanced Manufacturing, Fuzhou University, Quanzhou 362200, CHN;
2. School of Physics and Information Engineering, Fuzhou University, Fuzhou 350116, CHN;
3. China Fujian Photoelectric Information Science and Technology Innovation Laboratory, Fuzhou 350116, CHN)

Abstract: Due to the complex computation of the deep learning network and the huge computational parameters used in the road vehicle target detection, the problem of high delay and slow processing speed exists in the target detection task on the embedded system based on ARM architecture. Aiming at the above problems, a complete embedded road vehicle target detection solution was designed and implemented in this paper. The structural re-parameterization is used in the YOLOv3-Tiny-based feature extraction network to improve the model detection accuracy, and the parallel acceleration of the convolutional neural network was deployed by Vitis-AI on the Zynq embedded platform with the DPUCZDX8G architecture acceleration core, and finally the improved YOLOv3-Tiny network model was quantified, compiled and deployed as a dynamically linked library. The experimental results show that the MAP of VOC2007 is 0.597, and the real-time processing speed is 27.7 FPS. At the same time, the frame rate power consumption ratio is 1.49, which is suitable for the low power consumption requirements of edge computing devices.

Key words: target detection; Vitis-AI; quantification; structural re-parameterization; Zynq

收稿日期：2022－11－16.
基金项目：国家重点研发计划资助项目(2021YFB3600603)；福建省自然科学基金资助项目(2020J01468)；国家青年科学基金项目(62101132).

* **通信作者：**林珊玲　**E-mail:** sllin@fzu.edu.cn

0 引言

随着物联网技术的发展，将深度学习算法部署在嵌入式边缘设备进行目标检测识别有着重要且深远的意义。这就需要嵌入式设备存储大量的数据和进行复杂的运算，这将会消耗大量的硬件资源和功耗，并且在传统 ARM 平台的嵌入式设备上部署深度学习算法进行目标检测任务通常有着较高的延时和功耗。因此实现低延时、低功耗的嵌入式边缘设备应用于图像的目标检测是十分重要的。

目标检测是计算机视觉的基本任务之一，有其重要的实际应用价值。传统的目标检测算法主要是通过手动提取特征，需要人工设计，复杂繁琐并且会出现漏检和误检等问题。然而，随着卷积神经网络(CNN)[1]的兴起，CNN 已经替代传统算法成为目标检测任务的主流算法，并且在很多场景下的识别效率超过人类。CNN 可以自动地从训练数据中学习提取特征，并且随着网络层数的增加，能提取到图像更深层次的特征，相比于传统算法有着更强大的特征学习能力。

当前，业界部署卷积神经网络模型的加速方法主要有以下 3 种：1)使用基于单指令流多数据流或者多指令流多数据流的图像处理器(GPU)[2]进行模型的加速计算，但是存在高功耗和设备庞大等不足，限制了它在嵌入式系统中的应用，只适用于云端大型计算平台或者服务器；2)使用专用集成电路(ASIC)[3]实现模型的加速计算，具有优良的性能和能源效率，不过它面临着灵活性低、开发成本高、研发时间长等问题；3)使用 FPGA[4]实现模型的加速计算，有着与 GPU 相当的计算性能，又有着更高的能量效率和灵活性，能够根据卷积神经网络的结构和特点进行优化，所以 FPGA 正逐渐成为硬件加速的研究热点。

基于上述问题，本文提出了一种基于 Zynq 平台实现的交通目标检测嵌入式系统，以满足目标检测任务实时性的需求。该设计采用改进 YOLOv3-Tiny 作为目标检测模型，并且将参数数据格式量化为 8 位定点数，在较小精度损失的情况下，进一步压缩模型大小。充分利用 Zynq 异构计算系统的优点，采用 ARM 内核对输入图片的预处理和输出结果的计算，在 FPGA 端上部署 DPUCZDX8G 架构加速核对模型中卷积操作进行并行加速计算，最终实现运算速度的提升。

1 目标检测网络

1.1 模型网络结构

YOLOv3-Tiny 模型是在 YOLOv3 模型的基础上经过轻量化设计的目标检测网络，适合部署在存储空间有限的嵌入式平台。但是由于 YOLOv3-Tiny 的特征提取网络仅仅包含 7 层卷积层和池化层，导致模型识别精度有待提高。针对这一问题，我们通过对 Yolov3-Tiny 模型的特征提取网络进行改造，通过改造 3×3 卷积层，用多分支并行卷积的方法对卷积层进行重构，分出一条 1×1 卷积的旁路。并且通过结构重参数化[5]的方法在推理阶段将 1×1 的卷积分支合并入 3×3 卷积核中，使其在保持直通网络推理速度上的优势的同时，进一步提高检测精度。本文采用的目标检测网络模型结构如图 1 所示，该结构包含前端的特征提取网络、融合特征金字塔(Feature Pyramid Network，FPN)[6]和最后作为输出的 YOLO 层。其中，经过改进的特征提取网络包含 7 个卷积层、6 个池化层。前 4 个卷积层块由 3×3 卷积和 1×1 卷积并行组合而成。

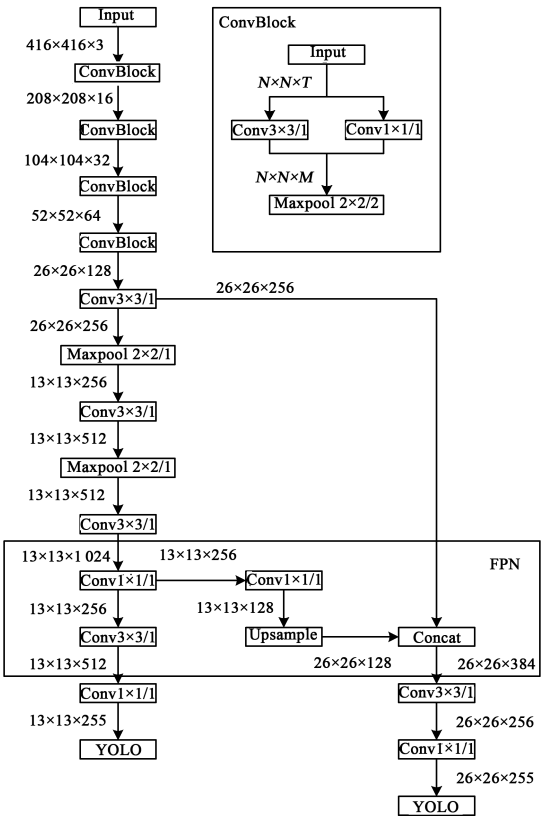


图 1 改进后的 YOLOv3-Tiny 网络结构

1.2 结构重参数化

结构重参数化是指在训练的时候使用多分支卷

积结构，在推断时融合为直通式网络。将训练阶段产生的模型通过结构和参数的变化得到推断模型。并且，推断模型的转化方法在数学上经过证明等价，采用该方法可以大幅压缩模型参数量，提升模型检测效果。下面将解释如何将训练时的多分支卷积结构转化为推断时的直通型结构。首先，是 BN 层和卷积层的融合。在图 1 中，每个卷积块(ConvBlock)为一个多分支卷积块，由卷积层和批归一化(BN)构成，可以由以下两个公式表达：

$$conv = \omega x + b \tag{1}$$

$$BN_{\gamma,\beta}(conv) = \gamma \frac{conv - \mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} + \beta \tag{2}$$

其中，x 为输入，ω 为卷积权重参数，b 为卷积偏置，γ 和 β 是经过训练得到的参数，μ_{B} 和 σ_{B} 是计算得到的训练数据的均值和方差。公式(1)为卷积运算，公式(2)为 BN 操作，将公式(1)代入公式(2)得到输入经过卷积层和批归一化后的结果，可表示为

$$BN_{\gamma,\beta}(x) = \frac{\gamma \cdot \omega}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} \cdot x + \frac{\gamma}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} \cdot (b - \mu_{\mathrm{B}}) + B \tag{3}$$

$$\hat{\omega} = \frac{\gamma \cdot \omega}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} \tag{4}$$

$$\hat{b} = \gamma \frac{conv - \mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} + \beta \tag{5}$$

$$BN_{\gamma,\beta}(x) = \hat{\omega} \cdot x + \hat{b} \tag{6}$$

将公式(3)的两个系数整理由公式(4)和(5)表示得到公式(6)，其中 $\hat{\omega}$ 和 $\hat{b}$ 表示融合后的卷积权重参数和偏置。由此可以得到卷运算和 BN 操作融合的结果，这样由训练模型转为推断模型，极大减少了推断过程的耗时。

多分支卷积结构的合并如图 2 所示。将 1×1 卷积通过边缘填充(padding)成 3×3 卷积，在填充后的 3×3 卷积中，卷积核中心为原来 1×1 卷积核的权重参数，其余位置都为零值。根据卷积运算的齐次性，卷积运算在卷积核尺寸一致的情况下，可以通过对应元素相加融合为一个 3×3 卷积核。

1.3　量化方法

量化就是将高精度的浮点数表示的权重和偏置用低精度的整数来近似表示，在量化到低精度之后就可以应用到嵌入式移动平台上对计算过程进行加速，并且带来的好处是模型容量的减少，使得能够很好地应用到嵌入式移动环境中。本设计采用对称量化的方法，将原本模型 32 bit 浮点数量化为 8 bit 定点数，并且采用训练后量化(Post Training Quantization)的方法。对称量化公式如下：

$$x_{\mathrm{int}} = round\left(\frac{x}{\Delta}\right) \tag{7}$$

$$\Delta = \max(abs(r\max), abs(r\min))/128 \tag{8}$$

其中，x 和 x_{int} 分别表示量化前和量化后的数值，Δ 为量化的缩放因子。式(7)代入参数后求得经过量化后的压缩率大约为 4 倍。

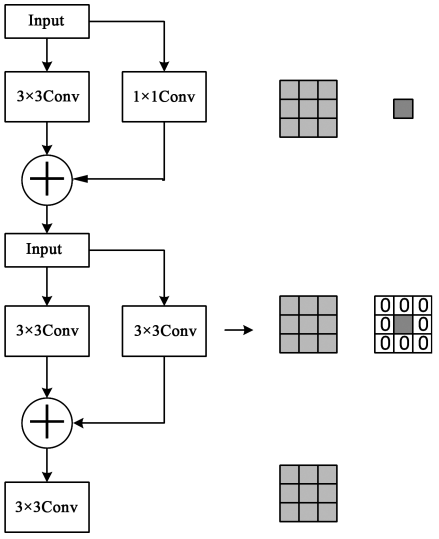


图 2　结构重参数化

2　系统设计与实现

2.1　模型搭建

根据以上所述，本文采用 TensorFlow 深度学习框架来进行开发改进的 YOLOv3-Tiny 网络模型并进行训练。设置初始输入图片尺寸为 416×416，优化器为随机梯度下降法(SGD)，设置 Batchsize 为 32。训练选择 LeakyReLU 激活函数。另外，对 VOC 训练集和测试集进行 10 次 K-mean 聚类算法得到以下 6 个先验框：(32.44，39.96)，(53.40，106.99)，(110.90，81.81)，(124.56，205.52)，(260.31，136.90)，(298.46，276.98)。

2.2　平台搭建

Vitis-AI 是 XILINX 公司推出的嵌入式 AI 开发工具，在目标检测卷积神经网络模型部署中起到编译器端和后端的作用，硬件开发流程如图 3 所示。

将前端 TensorFlow 框架训练后的网络参数进行冻结(freeze)得到浮点 32 位网络模型 YOLOv3-Tiny.pb 文件，并且从训练集中选取 500 张无标注的图片经由编写好的脚本 Input.py 进行量化校准。

网络参数量化工具对 YOLO 模型进行基线模型分析，经过多次对设定的阈值进行优化，去除冗余部分，使得神经网络模型轻量化，得到评估模型和部署模型。然后，将部署模型(deploy_model.pb)送入编译器映射为 DPU 指令流，在 FPGA 上调度 DPU 内核可降低计算量和硬件功耗，在保证精度的前提下提高运算速度。最后，DPU 驱动库由 AI 库、云端运算和软件交叉编译组成，可以调用 AI 库中的算法库和基本库；通过构建 YOLOv3-Tiny 网络模型，在 Vitis-AI 中运行 YOLOv3-Tiny 的可执行驱动库(.so)和推断主程序(run.py)。

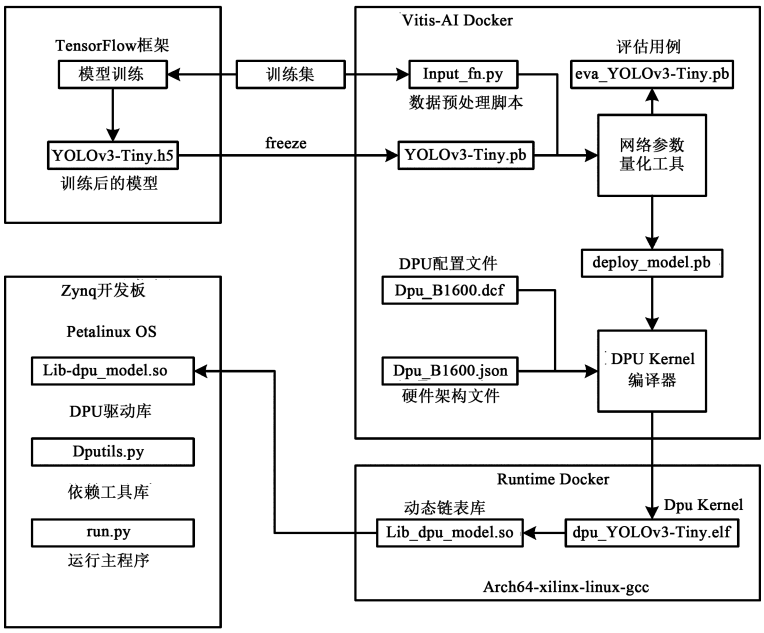


图 3 Vitis-AI 开发流程图

目标检测硬件环境要求，整个工程采用 vivado2018.2 软件搭建系统。本实验系统主要由图像采集部分、可编程逻辑端(PL 端)、操作系统端(PS 端)和 HDMI 接口构成。摄像头采用 OV5640 模块，采集到的 1080P 的图像数据，本系统以 RGB565 的格式，经过处理器接口传递到控制器，然后经过颜色空间转换，完成对图像数据的转换。整个处理部分分为以 DPU 为核心的图像并行计算部分和以 ARM 处理器为核心的板载系统部分。其中，ARM 处理器完成系统的应用层处理；DPU 完成图像的实时处理，通过 AXI 接口完成数据流的传输和存储。本设计采用 PetaLinux 编译环境进行板载系统的设计，并且内部通过搭建 Vitis-AI 的运行环境，完成目标检测系统所需要的嵌入式系统环境。

3 实验结果与分析

在 Ubuntu18.4 系统下用 TensorFlow 框架进行网络模型的构建和训练。其中，改进后的 YOLOv3-Tiny 模型的特征提取网络使用在 ILSVRC-2012 数据集上预先训练的方式，其中预先训练的主干网络在数据集上 TOP1 准确率达到 58.97%，这样做的好处是可以减少训练和加速收敛。训练模型采用 VOC2007 和 VOC2012 数据集，使用 18 993 张训练图片和 2 510 张测试图片。设置训练轮数为 300 轮，随着训练次数的不断增多，误差逐渐减小。在第 231 次训练时，网络参数获得最佳 mAP 为 59.7%。

3.1 Zynq 平台部署

该系统基于 Xilinx 的 AUX3EG 嵌入式异构平台进行，开发板搭载主频为 1.2 GHz，型号为 ARM Cortex-A53 的 CPU 内核和型号为 XCZU3EG-1SFVC784I 的 FPGA 芯片，并且该开发板带有 SD 卡接口用来读取提前烧录好的 Petalinux 系统镜像。同时扩展板上配有串口通信接口和 MiniDP 接口，我们通过串口通信软件 PuTTY 在 PC 端读取开发板传来的信息，并且通过 MiniDP 转 HDMI 转接线连接液晶显示屏实时检验目标检测的效果。

3.2 实验结果比较

本文采用 Vitis-AI 工具部署的图像目标检测系统与其他硬件平台的比较结果如表 1 所示。FPGA

硬件设计的资源使用情况如表 2 所示。实验结果表明经过重参数化改造后的模型在相较于原始的 YOLOv3-Tiny 模型的检测精度提升了 6.1%，在实验数据集上的图片检测效果大幅提升。此外，实际部署后的图像目标检测结果如图 5 所示，相较于图 4 改进前的检测效果，本文提出的改进后的目标检测算法在同样的图片下对于小目标的检测有着更为细腻的检测效果。由于重参数化改进的模型使用了多分支并行卷积结构，导致在推理速度上相较于改进前的模型有着略微增加，但是在不同的计算平台下处理一张图片的耗时相差不大，说明改造后的轻量化网络在提升检测精度的同时没有增加计算耗时。同时，因为在模型量化方面将 32 位浮点数转化为 8 位定点数，这样做的好处是减少了硬件推理时间，但是使得经过数据量化后的模型损失了 1.2% 的检测精度。

表 1　模型在不同平台上的测试结果

模型	识别精度/mAP	计算时间/ms
CPU-Xeon-E52690	0.597	795
ARM-Cortex-A53	0.597	1 330
ZYNQ-XCZU3EG	0.585	36

表 2　PL 端硬件资源使用情况

资源	已使用	总资源	占用量/%
LUT	34,663	70 560	49.13
FF	63,453	141,120	44.96
BRAM	127.5	216	59.03
DSP	311	360	86.38
LUTRAM	2,273	28,800	7.89

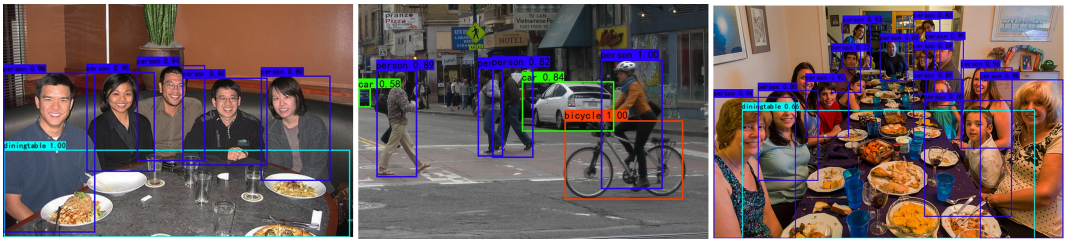


图 4　改进前检测效果图

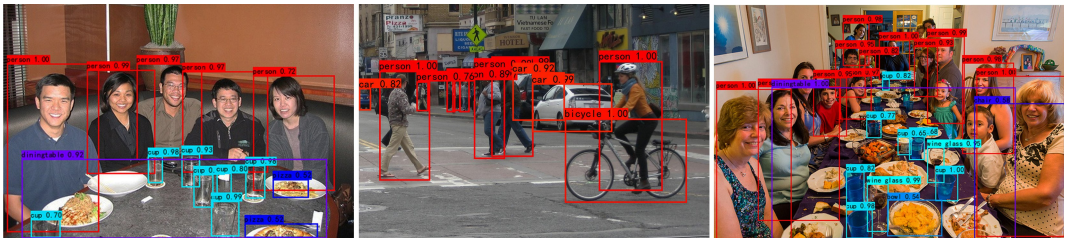


图 5　改进后检测效果图

该设计相较于基于 ARM 的嵌入式平台目标检测神经网络模型运算速度有着显著的提升。与其他文献方案的比较结果如表 3 所示。采用 Zynq 异构嵌入式系统检测每张图片速度为 36 ms，分别是 ARM 内核的嵌入式系统和通用 CPU 平台的 36 倍和 22 倍。但是在识别精度上有 1.2% 的损失，这是由于在转化模型的过程中采用了定点量化的方式造成的。同时，相较于文献[7]所提出的方案在相同输入图片尺寸的前提下，处理速度上也有明显的提升，比原方案的耗时减少近 4 倍。并且与文献[8-10]的方案相比在速度上的提升也是明显的。并且，本文设计的系统在功耗方面同样比其他 FPGA 工作上，每处理一帧图片所需的功耗更低，帧率功耗比为 1.49，满足边缘计算设备部署实现深度学习交通目标检测的低延时、低功耗的性能要求。

表 3　与其他的 FPGA 工作的对比(单位:ms)

输入图片尺寸	本文	文献[7]	文献[8]	文献[9]	文献[10]
	416×416	416×416	256×256	1 280×720	416×416
目标网络	YOLOv3-Tiny	YOLOv3-Tiny	YOLO-Tiny	YOLOv3	YOLOv2
耗时	36	530	220	711	980
帧率功耗比	1.49	0.74	0.79	0.77	0.20

4　结束语

本文针对图像目标检测任务中深度学习算法复杂和庞大的计算参数导致耗时过长的问题，设计并实现了一种嵌入式图像目标检测系统。该系统在目标检测的精度、计算平台处理图片的速度和功耗方面均有很大改善，实现了处理速度和功耗的平衡，且部署方便，适合在移动端场景使用。

参考文献：

[1] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[2] Gong L, Wang C, Li X, et al. MALOC: A fully pipelined FPGA accelerator for convolutional neural networks with all layers mapped on chip[J]. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 2018, 37(11): 2601-2612.

[3] Martinez B, Valstar M F. Advances, challenges, and opportunities in automatic facial expression recognition[J]. Advances in Face Detection and Facial Image Analysis, 2016: 63-100.

[4] Venieris S I, Bouganis C S. fpgaConvNet: Mapping regular and irregular convolutional neural networks on FPGAs[J]. IEEE Trans. on Neural Networks and Learning Systems, 2018, 30(2): 326-342.

[5] Ding X, Zhang X, Ma N, et al. Repvgg: Making vgg-style convnets great again[C]// Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition, 2021: 13733-13742.

[6] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]// Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, 2017: 2117-2125.

[7] 陈浩敏，姚森敬，席 禹，等. YOLOv3-tiny的硬件加速设计及FPGA实现[J]. 计算机工程与科学，2021，43(12)：2139-2149.
Chen Haomin, Yao Senjing, Xi Yu, et al. Design and FPGA implementation of YOLOv3-tiny hardware acceleration[J]. Computer Engineering and Science, 2021, 43(12): 2139-2149.

[8] 周锡雄，钟 胜，张伟俊，等. 基于异构FPGA的卷积网络加速器[J]. 模式识别与人工智能，2019，32(10)：927-935.
Zhou Xixiong, Zhong Sheng, Zhang Weijun, et al. Heterogeneous FPGA based convolutional network accelerator[J]. Pattern Recognition and Artificial Intelligence, 2019, 32(10): 927-935.

[9] 雷小康，尹志刚，赵瑞莲. 基于FPGA的卷积神经网络定点加速[J]. 计算机应用，2020，40(10)：2811-2816.
Lei Xiaokang, Yin Zhigang, Zhao Ruilian. FPGA-based convolutional neural network fixed-point acceleration[J]. J. of Computer Application, 2020, 40(10): 2811-2816.

[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 Chen, Chai Zhilei, Xia Jun. Design and implementation of YOLOv2 accelerator based on Zynq7000 FPGA heterogeneous platform[J]. J. of Frontiers of Computer Science and Technology, 2019, 13(10): 1677-1693.

作者简介：

王利翔(1996—)，男，福建省福州市人，硕士研究生，主要从事图像处理、计算机视觉及其硬件加速的研究；

林珊玲(1991—)，女，福建省泉州市人，博士，讲师，硕导，主要从事新型显示技术、印刷电子显示、电子纸显示和平板显示驱动以及图像处理技术研究。