高技术通讯 2022 年 第32 卷 第3 期:239-247

doi:10.3772/j.issn.1002-0470.2022.03.003

基于 FPGA 的视频实时目标检测方法研究①

陈 朋②* 何建彬** 陈 诺** 俞天纬* 宦若虹③*

(*浙江工业大学计算机科学与技术学院 杭州 310023)

(**浙江工业大学信息工程学院 杭州 310023)

摘 要 针对实时目标检测网络在图形处理器(GPU)加速器上实时性低、功耗高和成本高等问题,本文提出了一种结合通道注意力机制与深度可分离卷积的神经网络模型(AtDS-SSD),并将该网络在现场可编程门阵列(FPGA)上进行优化与部署。AtDS-SSD 网络在 SSD 模型基础上,将 VGG 16 特征提取网络部分替换成以深度可分离卷积为主体的 MobileNet 网络,并加入通道注意力模块。本文采用 8 位的定点量化方法,对网络模型参数进行量化。最后,本文将量化后的 AtDS-SSD 网络模型在 ZCU 102 平台上进行部署,并采用 PASCAL VOC 数据集进行测试。在平均精度均值只损失 0.58% 的情况下,加速器性能从 85 fps 提升到 311.7 fps,测试功耗相当于 NVIDIA RTX 2080Ti 的 11%。实验数据表明,基于 FPGA 平台结合注意力机制和深度可分离卷积的网络模型,可以提升计算实时性并降低功耗,减少网络复杂度降低导致的精度损失,从而验证了本文方法的有效性。

关键词 SSD 网络;通道注意力机制;深度可分离卷积;现场可编程门阵列(FPGA);定点量化

0 引 言

目标检测是一项计算机视觉领域中结合图像分割和图像识别的重要技术。传统目标检测的区域选择策略没有针对性,时间复杂度高,窗口冗余,对于多样性变化的特征没有好的鲁棒性。因此在边缘端设备上实现目标检测,需要实时性更好、计算复杂度更低的目标检测方法。

随着深度学习的发展,目标检测领域取得了很多突破性的研究成果,其检测的精度和速度都有了较好的效果。文献[1-3]提出了区域卷积神经网络(region-based convolutional neural network, R-CNN)和 Fast R-CNN 网络,使得神经网络在目标检测上获得了较大的突破。但 Fast R-CNN 也暴露出了区域候选的计算瓶颈问题,文献[4]在此基础上提出了 Faster R-CNN,引入一个区域候选网络(region proposal networks, RPN),并优化了算法。但是 Faster R-CNN 在实时性上仍然有所限制,其在图形处理器(graphics processing unit, GPU)上的帧速率仅有 5 fps,因此 YOLO[5] 提出了 one-stage 的概念,此方法将物体分类和物体定位在一个步骤中完成,提高了实时性,但是准确率和漏检率有待提高。SSD[6] 综合了 Faster R-CNN 和 YOLO 的优点,采用多尺度的特征图来得到准确率与实时性更高的网络模型。

另一方面,目前卷积神经网络的实现主要搭建在 GPU 上,GPU 能够使卷积神经网络的训练得到很好的加速,但是能耗较大,不易作为边缘端硬件平

① 国家自然科学基金(U1909203),浙江省自然科学基金(LY19F020032)和浙江省属高校基本科研业务费专项资金(RF-C2019001)资助项目。

② 男,1981 年生,博士,教授;研究方向:模式识别,嵌入式系统设计;E-mail:chenpeng@zjut.edu.cn。

③ 通信作者,E-mail:huanrh@zjut.edu.cn。

(收稿日期:2020-11-23)

台,限制了其应用场景。现场可编程门阵列(field programmable gate array, FPGA)是一种可编程、可定制的芯片,具有并行处理的能力,以及高性能、高灵活性等优点[7],可以被运用到 CNN 的加速中[8]。Xilinx 公司推出了用于高性能需求的异构平台 ZYNQ 系列芯片,配合 Cortex 系列的处理器,搭配可编辑逻辑部分,使得芯片架构灵活、运行功耗低、可重构性和可移植性强。同时 Xilinx 公司还推出了高层次综合工具 Vivado HLS 和 Vitis,使得卷积神经网络在 FPGA 上的开发周期大大缩短。2015 – 2019 年的 FPGA 会议[9-13]提出的各种加速器和加速器的框架,都表明 FPGA 适用于卷积神经网络的移植。文献[14]提出了全栈编译器深度神经网络虚拟机(deep neural network virtual machine, DNNVM),采用启发式子图同构算法枚举所有潜在可获利的融合机会,利用管线和数据布局进行硬件资源优化,并搜索整个计算图的最佳执行策略。文献[15]提出了一种特定于域的 FPGA 覆盖处理器(overlay processor unit, OPU),用于加速 CNN 网络。文献[16]提出了基于舍入和移位操作量化方案的 8 位优化的块浮点算法(block-floating-point, BFP),将能源和硬件效率提高了 3 倍。文献[17]提出了一种将原始网络压缩为定点形式的数据量化策略,并设计了可配置的硬件体系架构,使得网络模型在 FPGA 上具有较好的效果。

综上所述,本文基于 FPGA 对视频实时目标检测算法进行优化实现。本文主要工作如下。

(1) 提出了结合通道注意力机制与深度可分离卷积的神经网络模型(attention-based depthwise seperable single shot multibox detector,AtDS-SSD),减少了计算量,增强了高层特征图的语义信息。

(2) 提出一种基于 FPGA 的算法网络量化编译方案,将本文算法移植到 FPGA 上,在保证其对目标检测准确率的基础上减少算法的复杂度,降低功耗。

1 系统总体设计

本文系统总体框架如图 1 所示,由 AtDS-SSD 网络模型的生成、量化、编译以及部署 4 个部分构成。

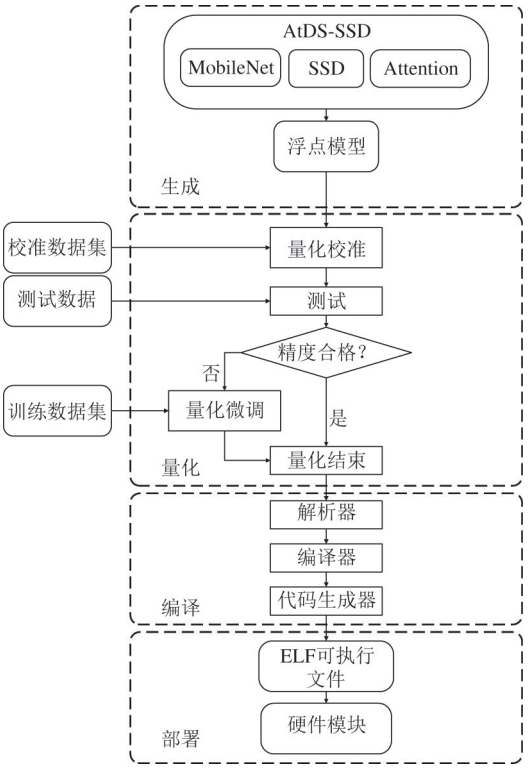


图 1 系统总体框架

1.1 AtDS-SSD 神经网络

SSD 采用回归方法获取目标对象的位置,并根据目标对象位置周围的特征进行目标分类,因此需要将特征图分割成若干个相同大小的网格,对每个网格分别进行预测分类,并通过非极大值抑制方法得到最终的检测结果。

标准的 SSD 神经网络运行时间较久,不满足实时性需求,并且模型参数计算量较大。为了满足实时性需求,并减少参数计算量,本文使用深度可分离卷积替换原有的常规卷积层,将常规卷积分离成深度卷积和点卷积两部分,使得计算复杂度更适合边缘设备。深度可分离卷积是轻量级神经网络 MobileNet 的重要组成部分,所以使用 MobileNet 作为主体网络替换原始 SSD 网络中的 VGG 16。同时本文结合通道注意力机制增强高层语义特征信息,补偿了由于模型参数计算量减少与实时性提升导致的精度下降,具有重要意义。

图 2 展示了 AtDS-SSD 卷积神经网络的总体结构,包含 3 部分:第 1 部分为 MobileNet 基础网络,通过深度可分离卷积减少基础网络的计算量;第 2 部分为通道注意力机制,通过增加很小的计算消耗,提

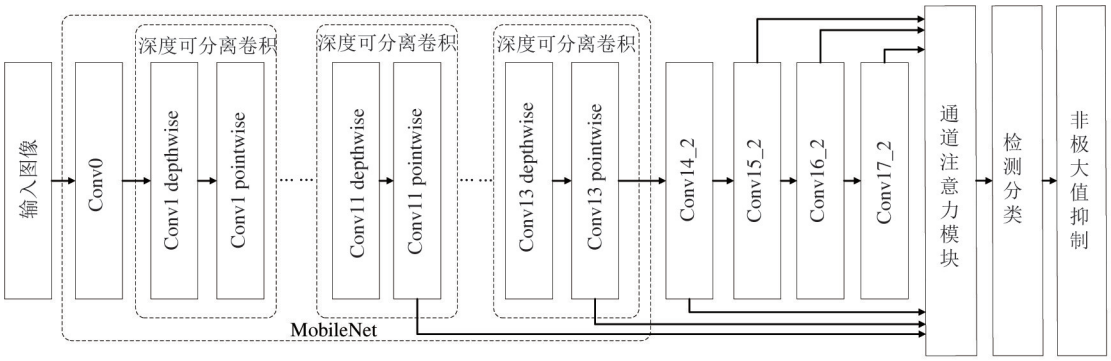


图 2　AtDS-SSD 卷积神经网络结构示意

升网络性能;第 3 部分为 SSD 的类别预测与位置回归。本文对网络结构的优化在保证实时目标检测需求的同时,减少边缘端设备的计算量,有助于将网络模型移植到资源有限、低功耗和低成本的嵌入式应用场景。

1.1.1　深度可分离卷积

深度可分离卷积是 MobileNet 的重要组成部分,将标准卷积核分离成一个逐通道处理的深度卷积核和一个跨通道处理的点卷积核,有效缩小模型参数计算量的同时仍然保持较高的准确率。

标准卷积示意图如图 3 所示,其中 $\boldsymbol{F}$ 为输入,维度为 $D_f \times D_f$, 通道为 M;将 $\boldsymbol{F}$ 映射到 $\boldsymbol{G}$ 作为输出,维度为 $D_g \times D_g$, 通道为 N。以常规卷积的卷积核进行卷积,需要 N 个卷积核,每个卷积核的维度为 $D_k \times D_k$, 通道为 M,总体计算复杂度为

$$CC_{\text{conv}} = D_k \times D_k \times M \times N \times D_f \times D_f \tag{1}$$

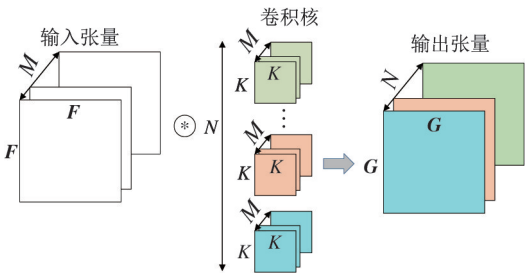


图 3　标准卷积

深度可分离卷积示意图如图 4 所示,由深度卷积部分与点卷积部分组成。深度卷积部分有 M 个 $D_k \times D_k \times 1$ 的卷积核,将产生 M 个输出张量,作为点卷积部分的输入。点卷积部分有 N 个 $1 \times 1 \times M$ 的卷积核,生成 N 个 $D_g \times D_g$ 的输出张量。该方法总体计算复杂度 CC_{Depth} 为

$$CC_{\text{Depth}} = D_k \times D_k \times M \times D_f \times D_f + N \times M \times D_f \times D_f \tag{2}$$

深度可分离卷积计算量 CC_{Depth} 与常规卷积计算量 CC_{conv} 的比率为

$$\frac{CC_{\text{Depth}}}{CC_{\text{conv}}} = \frac{1}{D_k^2} + \frac{1}{N} \tag{3}$$

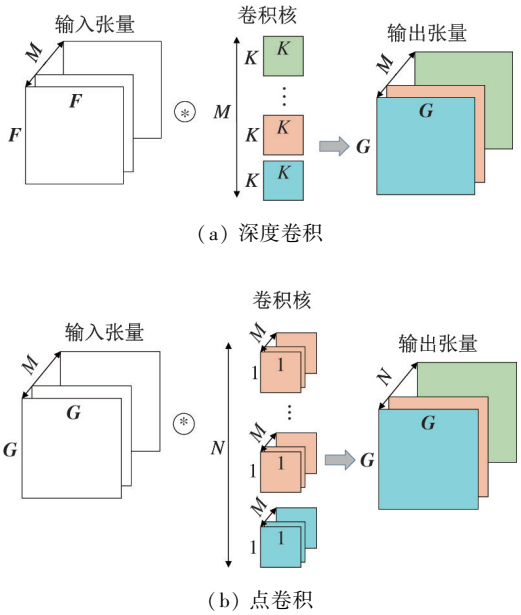


图 4　深度可分离卷积

以 AtDS-SSD 中的 CONV11 层举例,输出 N = 1024 通道的特征图,卷积层卷积核的尺寸为 3 × 3,则模型的参数计算量仅为标准卷积参数计算量的 11.21%,大幅减小了模型参数的计算量。

1.1.2　通道注意力机制

通道注意力机制通过对通道间的依赖关系进行建模,可以自适应调整各通道的特征响应值,仅在牺

牲少量计算量的情况下,可以极大地提升网络性能。

注意力机制模块由压缩(squeeze)、激励(excitation)及注意(attention)操作 3 部分组成,如图 5 所示。

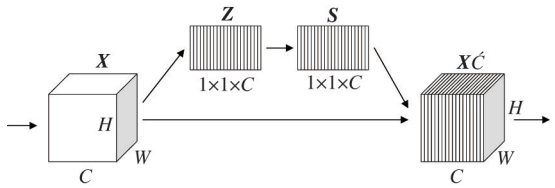


图 5　通道注意力机制

Squeeze 操作是对输入 $\boldsymbol{X}$(维度为 $C \times H \times W$)进行压缩,使用全局平均池化将输入特征图的全局信息压缩为通道描述符,具体的计算公式如式(4)所示。

$$\boldsymbol{Z} = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\boldsymbol{X}(i,j) \tag{4}$$

其中,(i, j) 为输入 $\boldsymbol{X}$ 的坐标,输出 $\boldsymbol{Z}$ 为 $C \times 1 \times 1$ 的矩阵。

Excitation 操作是对各通道的依赖程度进行建模,本文使用 ReLU 非线性激活函数和 Sigmoid 激活函数的门限机制来实现。其中为了限制模型的复杂度,增强模型的泛化能力,使用了 2 个全连接层去学习,根据输入数据可以调节各个通道特征的权重。具体计算公式如式(5)所示。

$$\boldsymbol{S} = \text{Sigmoid}(\boldsymbol{W}_2 \cdot \text{ReLU}(\boldsymbol{W}_1\boldsymbol{Z})) \tag{5}$$

其中,$\boldsymbol{W}_1$ 和 $\boldsymbol{W}_2$ 为通道权重,$\boldsymbol{W}_1$ 的维度是 $C' \times C$,$\boldsymbol{W}_2$ 的维度是 $C \times C'$,$C' = C \times \frac{1}{4}$,通过 ReLU 激活函数和 Sigmoid 函数进行训练学习,最终得到的 $\boldsymbol{S}$ 的维度为 $C \times 1 \times 1$。

Attention 操作为特征加权的过程,将原始的输入 $\boldsymbol{X}$ 替换为经过注意力模块获得的特征 $\boldsymbol{X}'$,并将其引入到原网络中进行目标检测。通过对各个通道的数据乘上不同的权重,从而增强对关键通道数据的信息。具体计算公式如式(6)所示。

$$\boldsymbol{X}' = \boldsymbol{X} \times \boldsymbol{S} \tag{6}$$

1.2　模型量化

由于嵌入式平台的资源有限,将卷积神经网络移植到嵌入式平台需要进行模型压缩,其中量化模型是一种常用方法。与 GPU 及中央处理器(centrol processing unit, CPU)相比较,FPGA 在模型量化上可以更为灵活,故本文采用模型量化的网络压缩方法。

由于卷积神经网络中不同层的数据动态范围通常很大,因此,对所有层进行统一的定点量化可能会导致很大的性能损失。为了解决这个问题,本文对每一层都单独进行定点量化,将 32 位浮点型数据转换为 8 位整型数据。

量化的方案如图 1 中的量化部分所示,将训练好的浮点模型和校准数据集输入到量化校准模块中,获得定点模型。量化结束以后得到的定点模型不一定是最优状态,所以对初步量化好以后的模型进行数据准确性测试。将测试结果与浮点模型进行精度对比,当精度损失较大时,需要进行量化微调;当精度损失较小时,即可得到最终的定点模型。

其中量化校准模块如图 6 所示,本文将每层的特征图和网络参数收集到量化校准模块中,对所有参数进行对数取整,得到取整后的数据直方图。根据直方图中连续八位占比最大的区间,可以不同地在每一层中找到最佳零点的位置,随后根据零点位置对所有参数进行移位操作,并进行八位定点量化。在上述步骤中浮点参数过大或过小容易造成八位定点数据溢出,对于过大的数据,保留符号,将其绝对值设置为最大值;对于过小的数据,将其设置为 0。本文逐层确定每一层的零点位置,得到最佳量化结果的定点模型,用测试数据集进行测试,根据测试结

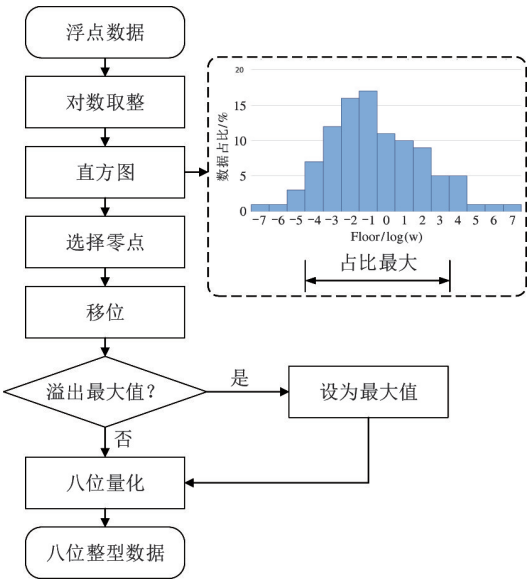


图 6　量化校准模块

果可以选择是否进行下一步的量化微调。

量化微调模块是将量化以后的网络模型转换回浮点格式进行微调,期间需要使用到训练数据集,且中间参数如梯度、权重、激活等浮点数将会重新训练。将重新训练的结果再次量化为定点数据,量化微调后的定点模型再与最初的浮点模型进行精度对比。重复上述步骤,直到量化之后的网络模型的精度损失在可接受范围内。

校准数据集的主要作用是定义模型动态输入的范围,因此本文选取的校准数据集包含了模型输入的所有类别。

1.3　模型编译

模型编译使用的是 Xilinx 的 Vitis AI 编译工具,该编译工具是编译器系列的统一接口,用于优化 DPU 的神经网络计算。每个编译器都将网络模型映射到高度优化的 DPU 指令序列中。Vitis AI 编译工具如图 1 中的编译模块所示,主要由解析器、优化器和代码生成器 3 个部分组成。

解析器将模型中的网络描述符映射到指令中。Vitis 编译工具可以根据不同的 FPGA 型号选择相应的指令集。通过指令调度 FPGA 上的资源,进行块分区和内存映射。块分区的主要作用是将网络模型和网络参数在片上存储,对每一层的计算都进行分区,充分利用卷积神经网络的数据本地化并减少数据输入输出,实现高效且减少功耗的作用。内存映射主要作用是将外部内存空间分配用于主机和网络加速器之间的通信。块分区和内存映射结束后,FPGA 便可以通过指令集完成网络模型的计算。

优化器优化网络模型,其中包括计算节点的融合(例如将 BN 层融合到预卷积中),充分复用 FPGA 上的数据,通过固有的并行性进行有效的指令调度或数据的充分利用,可用于处理 CNN 的高存储复杂性。

最后通过代码生成器生成可执行文件,该文件包含了网络模型、参数与权重等信息,可将其部署到 FPGA 上。

2　实验结果分析

本文完成了以下 2 组实验。(1) VGG-SSD、MobileNet-SSD 以及 AtDS-SSD 3 种卷积网络模型在 GPU 上目标检测的平均精度均值比较以及运行时间比较;(2) AtDS-SSD 网络模型在 GPU 与 FPGA 上的功能验证以及性能比较。通过上述实验来验证基于 FPGA 结合注意力机制与深度可分离卷积的网络模型在边缘端设备进行目标检测的综合优势。

2.1　网络模型训练

本文涉及到的 3 种卷积网络模型都由服务器训练生成,硬件平台的处理器为 Intel i9-10900X,显卡为 NVIDIA RTX 2080Ti,部署 TensorFlow 深度学习框架,通过 NVIDIA CUDA 运算平台调用显卡进行卷积神经网络学习训练。本文采用 PASCAL VOC 2007 和 PASCAL VOC 2012 训练数据集进行训练,PASCAL VOC 2007 测试数据集进行测试,该数据集包括 20 个类别,即 aeroplane、bicycle、bird、boat、bottle、bus、car、cat、chair、cow、diningtable、dog、horse、motorbike、person、pottedplant、sheep、sofa、train、tvmonitor,共 22 163 张训练图片和 4952 张测试图片。

本文在训练过程中将输入图片的分辨率调整为 300×300,批尺寸(Batchsize)设置为 32。训练完成后对各个网络模型的目标检测准确性和损失值进行对比,各个网络结构训练过程中损失值的变化如图 7所示,其中纵坐标为损失值,横坐标为训练的迭代次数,各个网络结构训练过程中准确率的变化如图 8所示,其中纵坐标为准确率,横坐标为训练的迭代次数。

从图 7 中可以看出,loss 值随着训练的迭代次数增加逐渐减少,一直到 loss 值几乎不变的时候,表明训练已经达到最优值,可以停止训练。图 8 可以看

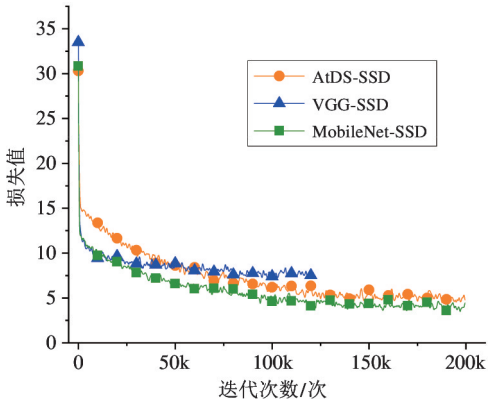


图 7　网络的损失值的变化

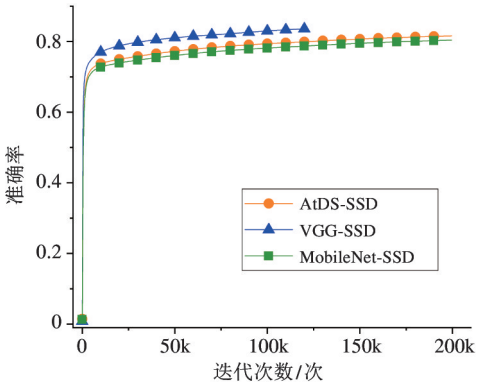


图 8　网络的准确率的变化

出 MobieNet-SSD 与 AtDS-SSD 的准确率都低于 VGG-SSD 网络，是因为 VGG-SSD 进行检测分类的最大特征张量的尺寸是 30×30，而 MobieNet-SSD 与 AtDS-SSD 进行检测分类的最大特征张量的尺寸是 19×19，所以对小目标分类检测更加弱，准确率有所下降，但实时性增强，速率更快。同时 AtDS-SSD 相比较于 MobileNet-SSD，通过结合通道注意力模块增强了高层特征语义消息，补偿了由于参数计算量的减少带来的精度损失。

2.2　模型目标检测对比

本实验在 NVIDIA RTX 2080Ti 上分别使用 VGG-SSD 神经网络、轻量级神经网络 MobileNet-SSD 和结合通道注意力机制和深度可分离卷积的 AtDS-SSD 神经网络对 PASCAL VOC 2007 测试数据集进行目标检测并比较。在目标检测中，通常采用平均精度均值(mean average precision，mAP)指标对精度进行评估，实验结果如表 2 所示。从检测结果可以看出，本文提出的 AtDS-SSD 网络模型在目标检测的平均精度均值上相较于 VGG-SSD 网络降低了 11.02%，但是相较于 MobieNet-SSD 网络，AtDS-SSD 的准确率提升了1.2%。

表 2　VOC 测试集中部分目标检测的平均准确率

算法	mAP/%	部分类别检测精度/%							
		plane	table	plant	boat	bird	bus	person	cat
VGG-SSD[6]	77.72	83.26	78.91	55.35	70.31	75.53	85.29	79.70	87.67
MobileNet-SSD[17]	65.50	65.45	71.58	38.20	54.26	57.79	77.65	70.96	79.52
AtDS-SSD	66.70	67.74	71.32	38.68	56.81	58.30	78.38	71.83	80.39

此外，为检测算法的实时性，本文对比了 VGG-SSD、MobileNet-SSD 和 AtDS-SSD 卷积神经网络的检测速度，具体检测结果如表 3 所示。由于 VGG-SSD 在实时性效果上远低于其他两个网络，因此不适合将其移植到 FPGA 上。而结合注意力机制与可分离卷积的 AtDS-SSD 卷积神经网络在检测速度上可以满足实时性需求，且在精度上相较于 MobileNet-SSD 网络略有优化，使其实时性和检测精度达到了更好的平衡，适用于边缘端设备进行目标检测。

表 3　VOC 测试集中检测速率对比

网络	mAP/%	检测速度/fps
VGG-SSD[6]	77.72	46
MobileNet-SSD[18]	65.50	90
AtDS-SSD	66.70	85

2.3　AtDS-SSD 在不同平台的性能比较

本实验对 AtDS-SSD 在 NVIDIA RTX 2080Ti 和 ZCU 102 上的运行结果进行讨论。

首先在不同硬件平台上对 AtDS-SSD 神经网络进行功能验证。功能验证主要是将 ZCU 102 上计算得到的预测数据与 NVIDIA RTX 2080Ti 上计算得到的预测数据进行对比，保证网络模型的输出能达到目标检测的基本功能。

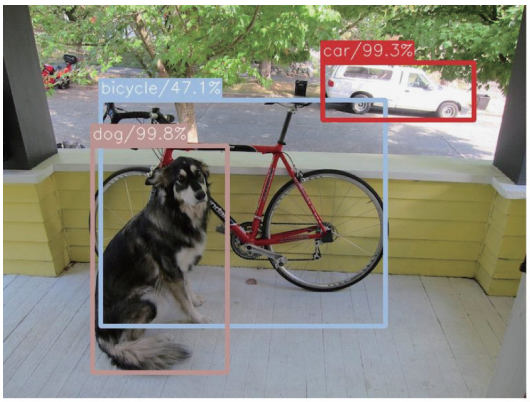
本实验针对单目标检测和多目标检测都进行了功能验证。单目标结果如图 9 所示。图 9(a)为单目标在 NVIDIA RTX 2080Ti 上的检测结果，目标位置检测精准且分类正确，置信度为 87.0%；图 9(b)为单目标在 ZCU 102 上的检测结果，目标位置检测精准且其分类正确，置信度为 80.8%，结果与 NVIDIA RTX 2080Ti 相差不大。多目标检测如图 10 所示。图 10(a)为多目标在 NVIDIA RTX 2080Ti 上

的检测结果,目标位置检测精准且其分类正确;图 10(b)为多目标在 ZCU 102 上的检测结果,目标位置检测精准且其分类正确,但是置信度略低于 NVIDIA RTX 2080Ti 的运行结果。上述结果表明,在 NVIDIA RTX 2080Ti 与 ZCU 102 上运行最后的检测结果产生的偏差不影响最终结果的呈现。

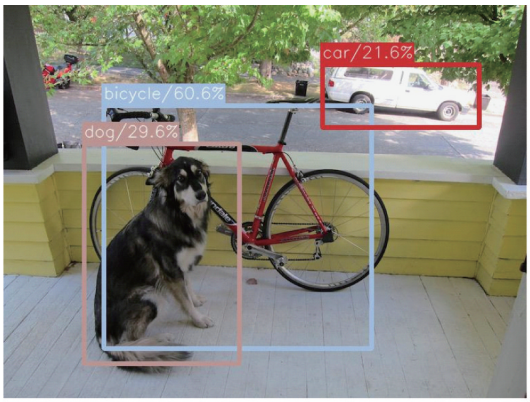
(a) GPU下运行的结果

(b) FPGA下运行的结果

图 9　单目标检测结果



(a) GPU下运行的结果



(b) FPGA下运行的结果

图 10　多目标检测结果

本实验对 AtDS-SSD 神经网络在 NVIDIA RTX 2080Ti 和 ZCU 102 上的性能和功耗进行测量与对比,其结果如表 4 所示。在 NVIDIA RTX 2080Ti 上进行目标检测需要的功耗为 77 W,而在 ZCU 102 上进行测试,功耗为 8.56 W,设计功耗低,非常适合用于边缘端设备处理实时目标检测。而且在 ZCU 102 上使用多线程模式对输入图像进行测试时,帧率达到 311.7 fps,高于 NVIDIA RTX 2080Ti 平台。

表 4　GPU 和 FPGA 性能对比

平台	mAP/%	功耗/W	帧率/fps
RTX 2080Ti	66.70	77.20	85.0
ZCU 102	66.12	8.56	311.7

3　结　论

本文提出了结合通道注意力机制与深度可分离卷积的 AtDS-SSD 网络,减少了计算复杂度,增强了高层特征图的语义信息。提出了一种对基于 FPGA 的算法网络原始模型进行量化编译方案,将本文算法移植到 FPGA 上,相较于现有边缘实时目标检测系统,综合兼顾了目标检测的实时性和准确性,使得

2 种参数得到了更好提升,且降低了功耗,提高了计算能效。实验结果表明,本文对视频实时目标检测的优化与实现满足了边缘端设备计算实时性的要求,同时也解决了功耗问题。

参考文献

[1] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, USA, 2014:580-587

[2] GIRSHICK R, DONAHUE J, DARRELL T, et al. Region-based convolutional networks for accurate object detection and segmentation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 38(1): 142-158

[3] GIRSHICK R. Fast R-CNN[C]//2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015:1440-1448

[4] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137-1149

[5] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA, 2016:779-788

[6] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot multiBox detector[C]// European Conference on Computer Vision, Amsterdam, Netherlands, 2016:21-37

[7] 原魁, 路鹏, 邹伟. 自主移动机器人视觉信息处理技术研究发展现状[J]. 高技术通讯, 2008, 18(1):104-110

[8] 赵然, 常轶松, 刘波, 等. SoPC FPGA 云平台软硬件协同交互框架[J]. 高技术通讯, 2020, 30(4):342-347

[9] ZHANG C, LI P, SUN G, et al. Optimizing FPGA-based accelerator design for deep convolutional neural networks [C]//Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Monterey, USA, 2015: 161-170

[10] QIU J, WANG J, YAO S, et al. Going deeper with embedded FPGA platform for convolutional neural network [C]//Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Monterey, USA, 2016: 26-35

[11] ZHANG J, LI J. Improving the performance of OpenCL-based FPGA accelerator for convolutional neural network [C]// Proceedings of 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Monterey, USA, 2017: 25-34

[12] ZENG H, CHEN R, ZHANG C, et al. A framework for generating high throughput CNN implementations on FPGAs[C]//Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Monterey, USA, 2018: 117-126

[13] DING C, WANG S, LIU N, et al. REQ-YOLO: a resource-aware, efficient quantization framework for object detection on FPGAs[C]//Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Seaside, USA, 2019:33-42

[14] XING Y, LIANG S, SUI L, et al. DNNVM: end-to-end compiler leveraging heterogeneous optimizations on FPGA-based CNN accelerators[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020, 39(10):2668-2681

[15] YU Y, WU C, ZHAO T, et al. OPU: an FPGA-based overlay processor for convolutional neural networks[J]. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2020, 28(1):35-47

[16] LIAN X, LIU Z, SONG Z, et al. High-performance FPGA-based CNN accelerator with block-floating-point arithmetic[J]. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2019, 27(8):1874-1885

[17] GUO K, SUI L, QIU J, et al. Angel-Eye: acomplete design flow for mapping CNN onto embedded FPGA[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2018, 37(1):35-47

[18] NIKOUEI S Y, CHEN Y, SONG S, et al. Real-time human detection as an edge service enabled by a lightweight CNN[C]//2018 IEEE International Conference on Edge Computing (EDGE), San Francisco, USA, 2018:125-129

Research on real-time FPGA-based video target detection method

CHEN Peng[*], HE Jianbin[**], CHEN Nuo[**], YU Tianwei[*], HUAN Ruohong[*]

([*]College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023)

([**]College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023)

Abstract

In order to solve the problems of low real-time performance, high power consumption and high cost of real-time target detection network on graphics processing unit (GPU) accelerators, a neural network model named attention-based depthwise seperable single shot multibox detector (AtDS-SSD) that combines channel attention mechanism and depthwise separable convolution is proposed, and the network is optimized and deployed on field programmable gate array (FPGA). Based on the SSD model, the AtDS-SSD network adds an attention module, and replaces the VGG 16 network with the MobileNet network which is mainly composed of depthwise separable convolution. An 8-bit fixed-point quantization method is used to quantify the network model parameters. The quantified AtDS-SSD network model is deployed on the ZCU 102 platform and tested by using the PASCAL VOC data set. The accelerator performance has increased from 85 fps to 311.7 fps, and the power consumption is equivalent to 11% of NVIDIA RTX 2080Ti, with only 0.58% drop of meanaverage precision. The experimental results show that the FPGA platform combined with the attention mechanism and the depthwise separable convolution network model can improve the real-time performance, reduce the power consumption, and reduce the accuracy loss caused by the reduction of network complexity, which verifies the effectiveness of the method proposed in this paper.

Key words: single shot multibox detector (SSD) network, channel attention module, depthwise separable convolution, field programmable gate array (FPGA), fixed-point quantization