文章编号:1671-4598(2025)11-0050-08 DOI:10.16526/j.cnki.11-4762/tp.2025.11.006 中图分类号:TP391 文献标识码:A

基于 FPGA 的高能效 YOLO 目标检测系统

刘 达,李付帅,弋庆龙
(福州大学 电气工程与自动化学院,福州 350108)

摘要:依据 YOLO 系列网络在目标检测领域的出色表现,提出了一种基于 FPGA 的高能效 YOLO 目标检测系统;采用层融合优化设计目标检测网络 YOLOv5n 降低模型训练层数,并接着对优化训练的 YOLOv5n 模型采用量化感知训练量化权重数据;通过优化模型不同网络层的硬件子模块设计了一种混合流可配置的硬件加速器,并采用乒乓双缓存与层间流水线协同作用方式实现模型加速推理;采用软硬件协同设计实现目标检测硬件加速系统,通过软核处理器合理调度硬件子模块实现硬件加速器高效并行工作;目标检测硬件加速系统在 Xilinx VC707 FPGA 开发板上实际测试以 100 MHz的工作频率,且功耗仅为 2.88 W 的情况下实现了 27.15 GOPS 的吞吐量,能效高达 9.43 GOPS/W,满足目标检测系统的能效需求。

关键词:目标检测;优化量化;YOLOv5n;FPGA;硬件加速

Detection System of High-Efficiency YOLO Targets Based on FPGA

LIU Da, LI Fushuai, YI Qinglong
(School of Electrical Engineering and Automation, Fuzhou University, Fuzhou 350108, China)

Abstract: Based on the excellent performance of YOLO series networks in the field of target detection, a high-efficient YOLO target detection system based on FPGA is proposed. The target detection network YOLOv5n is designed by layer fusion optimization to reduce the number of model training layers, and then the optimized YOLOv5n model is trained with quantization aware training to quantize the weight data. By optimizing the hardware submodules of different network layers of the model, a hybrid flow configurable hardware accelerator is designed, and the ping-pong double buffer and inter-layer pipeline synergy are used to achieve model acceleration reasoning. The target detection hardware acceleration system is realized by using software and hardware collaborative design, and the hardware submodules are reasonably scheduled by the soft core processor to realize efficient parallel operation of the hardware accelerator. The target detection hardware acceleration system was tested on a Xilinx VC707 FPGA development board and achieved a throughput of 27.15 GOPS at a 100 MHz operating frequency and a power consumption of only 2.88 W, with an energy efficiency of up to 9.43 GOPS/W, meeting the energy efficiency requirements of the target detection system.

Keywords: target detection; optimization quantization; YOLOv5n; FPGA; hardware acceleration

0 引言

电气化铁路交通邻域近年来持续拓展相关技术,通过弓网系统实现电能传输的方式被广泛应用于交通行业中。弓网系统的正常运作对高速动车的安全行驶起着重要的保障作用。因此,研究弓网燃弧检测系统具有实际意义。

目标检测网络作为卷积神经网络(CNN, convolutional neural network)的一种典型网络,常用于在图像中自动定位目标,与图像分类[2-3]不同,它不仅能获取目标的类别信息,还能够通过绘制边界框来获取到位置信息,因此被广泛地应用在智能识别[4]、图像处理[5-6]、医学图像分析[7]等领域[8]。目标检测网络根据处理方式可以划分为单级式检测网络和两级式检测网络。单级式检测网络相比两级式检测网络,检测精度与速度之间的平衡性更优,其中单级式检测网络的经典代表为 YOLO(You Only Look Once)[9]。文献[10-11]采用传统的检测方式来对目标进行识别,但大多数都是通过人工的方式进行处理,将采集到的图像信息上传至云平台,由于平台计算能力的限制,该方法存在传输时延迟过高、系统复杂以及识别精确度低等问题,因此,采用目标检测网络中更有优势的 YOLO 网络实现对燃弧目标图像的快速识别。

基于 CNN 的目标检测网络在实际硬件设备部署的

收稿日期:2025-03-27; **修回日期**:2025-05-09。
作者简介:刘 达(2000-),男,硕士研究生,工程师。
引用格式:刘 达,李付帅,弋庆龙.基于 FPGA 的高能效 YOLO 目标检测系统[J].计算机测量与控制,2025,33(11):50-57,96.

主流方案可分为四类：中央处理器单元（CPU，central processing unit）、图形处理单元（GPU，graphics processing unit）、专用集成电路（ASIC，application-specific integrated circuit）以及现场可编程逻辑门阵列（FPGA，field programmable gate array）。从实际应用分析，CPU 与 GPU 部署 CNN 时普遍面临功耗过高的问题；ASIC 集成度高，但其研发与制造时间长，限制了大规模应用。相较之下，FPGA 凭借高并行计算、低功耗、硬件可重构三大核心特性，能更灵活地适配 CNN 的运算需求，成为当前极具优势的硬件加速方案。文献［12］通过模块化设计对 CNN 进行分层，对每层模块进行专用化设计，实现了高能效的 CNN 硬件加速器。文献［13－14］通过将网络模型层融合轻量化和内部权重数据量化至低位宽方式来节省 FPGA 硬件资源消耗和片内外存储数据的交互量，从而实现高能效的 FPGA 硬件加速器。文献［15］通过对模型进行卷积循环展开，并采用脉冲矩阵阵列对计算过程进行并行加速，优化硬件推理，来实现 CNN 硬件加速器。文献［16］在逻辑上通过设计移位方式来代替乘法器进行计算，节省了硬件资源，得到了一种运行效率非常高的 CNN 硬件加速器。文献［17］通过系统软硬件协同设计，实现了运行速度快、资源消耗少的 YOLOv2 硬件加速器。

本文提出了一种基于 FPGA 的高能效 YOLO 目标检测系统，利用 YOLOv5n 网络进行准确地、自动化地检测燃弧图像。在模型训练前将批归一化层（BN，batch normalization）融合到相邻的卷积层当中，减少训练层数与参数量，降低模型复杂度，并将 YOLOv5n 网络模型原有的 Silu（Sigmoid linear unit）激活函数更改为 Relu（Rectified linear unit）激活函数。训练完轻量化后的模型再采用量化感知训练（QAT，quantization aware training）算法对优化设计好的 YOLOv5n 网络模型进行量化压缩，将权重数据位宽量化为 8 bit，为后续 FPGA 硬件部署做铺垫。利用高层次综合（HLS，high level synthesis）工具在可编程逻辑（PL，programmable logic）端上设计了一种混合流可配置的加速器架构，并设计特征图与权重数据重利用、乒乓双缓存与层间流水线机制加快系统运行速度，同时在处理系统（PS，processing system）端对硬件加速器中的模块进行任务调度与参数配置，控制整个系统的运行，并将前向推理识别到的图像检测框和坐标进行显示。最终在 Xilinx VC707 开发板上通过软硬件协同设计的方式实现了整个目标检测系统。

1　目标检测网络算法及优化

1.1　目标检测网络

YOLOv5[18] 模型对比之前的 YOLO 版本和其他目标检测模型，其大小和速度的平衡可以适应不同的计算能力和应用需求。在检测过程中通过使用 Mosaic 和 MixUp 等图像数据增强技术提高模型在不同情况下目标的泛化能力，使得 YOLOv5 在保持高平均精度均值的同时，显著地提高了目标检测的效率。

YOLOv5 根据网络的宽度和深度来划分为 YOLOv5n、YOLOv5s、YOLOv5m、YoLOv5l、YOLOv5x 五个版本。YOLOv5n 因其模型小、低复杂性、处理速度快等特点特别适合部署在 FPGA 平台上。因此，本文采用了 YOLOv5n 作为硬件加速器目标检测模型，并对 YOLOv5n 算法结构进行改进，将原模型中的激活函数由 Silu 替换为 Relu 激活函数。激活函数通过对卷积层输出数据进行变换，将线性关系转化为非线性关系，从而增加了神经网络的表达能力，同时可以将输出限制在特定范围内，能够更好地适应不同类型数据，加速模型推理过程。YOLOv5n 原有的 Silu 激活函数计算过程包括指数、乘法、除法和加法运算，复杂度高，在硬件逻辑上实现会消耗较多的逻辑资源，对模型的部署不利。因此，将其更换为逻辑上更容易实现的 Relu 激活函数，推理过程简单，可通过数据选择器完成。微调后的 YOLOv5n 网络模型结构如图 1 所示。

YOLOv5n 模型由 3 个主要组件组成：骨干（Backbone）网络、颈部（Neck）网络以及检测头（Prediction）网络。骨干网络由卷积批归一化激活（CBR，ConvBNRelu）层、跨阶段局部网络（CSP，cross stage partial）和快速空间金字塔池化（SPPF，spatial pyramid pooling-fast）模块组成，用于从输入图像中提取特征的卷积神经网络部分，通过多个模块来提取图像特征，并逐渐降低图像的空间维度，同时增加特征图的深度。颈部网络进一步处理和聚合特征图，以便更好地捕捉目标图像的不同尺度信息，并通过自顶向下和自底向上的路径增强特征传播，将高层特征图通过上采样和低层特征图实现多尺度的有效融合。检测头网络负责对骨干网络和颈部网络提取的特征图进行多尺度的目标检测来预测目标的类别标签、位置坐标和置信度分数。

1.2　网络模型层融合优化

在对模型微调后，YOLOv5n 目标检测网络中的 CBR 模块计算过程包括卷积、BN 和 Relu 三部分，在训练网络模型时，BN 层能够加速网络收敛，控制过拟合，一般设置在卷积层之后，能够有效解决梯度消失与梯度爆炸问题，加快收敛速度，提高训练效率和网络稳定性，但在网络模型前向推理中会增加运算量，导致中间计算的数据结果占用更多的内存，影响模型的性能，将其部署在 FPGA 中会额外消耗一定的硬件资源。因此，为了减少模型加速过程中不必要的计算工作量和数据传输，将 BN 层融合进相邻的卷积层当中，从而达到

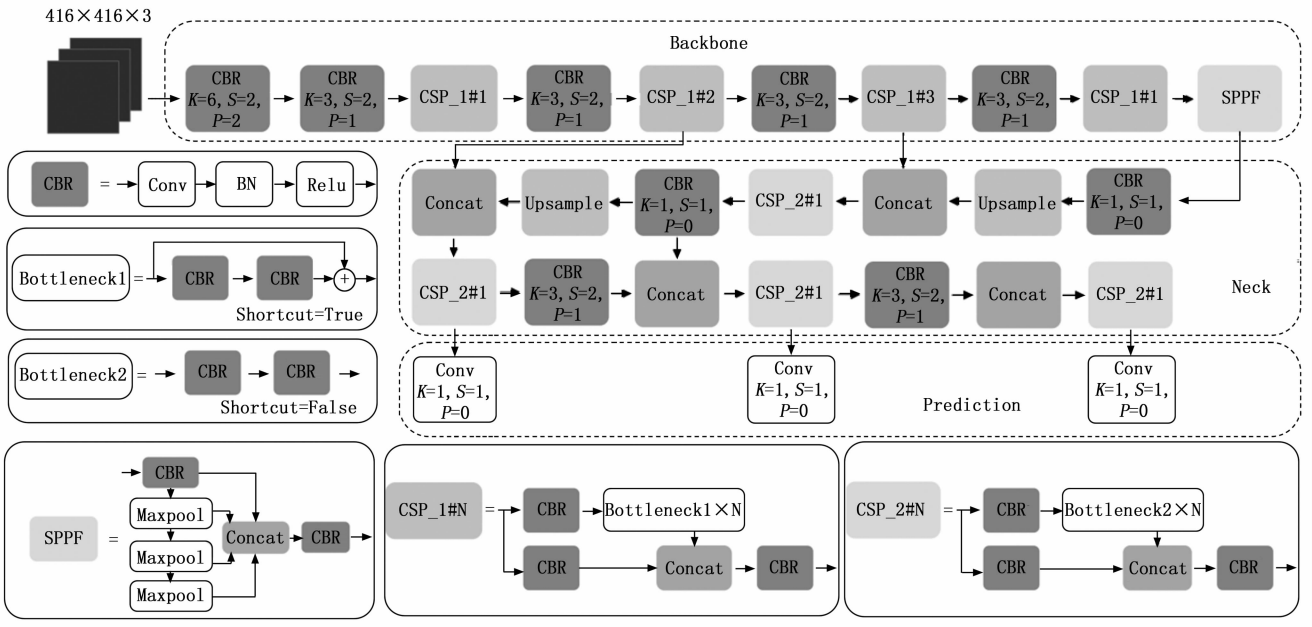


图 1 YOLOv5n 网络模型

降低模型计算量和提高系统目标检测性能的效果。

模型中的卷积层有助于在前向推理过程中找到特定的局部图像特征，提取输入图像中的重要信息，其计算公式如公式（1）所示：

$$a_i = \sum_i^N w_i x_i + b \tag{1}$$

式中，w 表示卷积层的权重，b 表示偏置，a 和 x 分别表示输出数据和输入数据，N 表示输入特征图中元素的个数。

卷积层输出数据的均值 μ_B 和方差 δ_B^2 如式（2）与式（3）所示：

$$\mu_B = \frac{1}{N}\sum_{i=1}^N a_i \tag{2}$$

$$\delta_B^2 = \frac{1}{N}\sum_{i=1}^N (a_i - \mu_B)^2 \tag{3}$$

通过均值和方差可以将原本随机分布的数据转为正态分布的数据，从而使得输入到 BN 层的数据分布较近，有利于网络的迭代。BN 层的计算公式如公式（4）所示：

$$y_i = \frac{a_i - \mu_B}{\sqrt{\delta_B^2 + \varepsilon}} * \gamma + \beta \tag{4}$$

式中，γ 和 β 分别为尺度因子和偏移因子，尺度因子优化了特征数据分布的宽窄，偏移因子优化了数据的偏移量，在网络模型训练中，是两个可学习参数，训练结束后数值固定。ε 为大于 0 且极小的值，用于防止 δ_B^2 为 0 的异常情况出现。将公式（1）代入到公式（4）可得融合后的卷积计算公式如公式（5）所示：

$$y_i = \sum_{i=1}^N \frac{w_i * \gamma}{\sqrt{\delta_B^2 + \varepsilon}} x_i + \left(\frac{b - \mu_B}{\sqrt{\delta_B^2 + \varepsilon}}\gamma + \beta\right) \tag{5}$$

式中，融合后的卷积核权重 W 和偏置 B 如公式（6）与公式（7）所示：

$$W = \frac{w * \gamma}{\sqrt{\delta_B^2 + \varepsilon}} \tag{6}$$

$$B = \frac{b - \mu_B}{\sqrt{\delta_B^2 + \varepsilon}}\gamma + \beta \tag{7}$$

1.3 网络模型量化

目标检测网络 YOLOv5n 在电脑端经过训练迭代后的模型，其内部权重和偏置数据位宽为 32 位，直接部署在 FPGA 上会消耗大量的硬件逻辑资源和存储空间，FPGA 硬件平台上一般采用定点类型数据进行运算。因此，为了实现更高效的硬件加速器，本文对 YOLOv5n 网络模型进行 QAT 定点量化[19]处理。通过在网络模型中的关键层前后插入伪量化节点，在实际不改变数据精度的情况下，仅模拟量化和反量化操作，让模型提前适应量化带来的精度损失，最终得到可直接量化且精度高的模型，无需训练后再校正，并将网络模型权重数据压缩至 8 位定点整数。

QAT 量化中浮点数和定点整数之间的换算公式如公式（8）所示：

$$r = S(q - Z) \tag{8}$$

式中，r 表示浮点实数，q 表示量化操作后得到的低精度定点整数，S 为缩放因子，决定了量化后的数值范围和精度，Z 为零点，用于确保零点的精确表示，可使量化后的数值更好地映射原始浮点数值。

将公式（8）代入卷积计算公式中可得出卷积层输出数据的量化公式（9）：

$$q_a = \frac{S_w S_x}{S_a}\sum_i^N (q_w - Z_w)(q_x - Z_x) + \frac{S_b}{S_a}(q_b - Z_b) + Z_a \tag{9}$$

由于模型训练时采用对称量化模式，权重和偏置的

零点 Z_w、Z_b 取为 0。在大部分实际应用工程中，偏置数据的缩放系数取为 $S_b = S_w S_x$，输入和权重数据的缩放因子数值位宽为 8 位，因此 $S_w S_x$ 在数值上最多缩放至 16 位，用 32 位位宽存储偏置是完全足够的，但带来的误差会在量化训练中不断传播，影响最终的精度，因此卷积的量化公式（9）调整为公式（10）：

$$q_a = 2^{-n}\left[2^n \frac{S_w S_x}{S_a}\left(\sum_i^N (q_w - Z_w)(q_x - Z_x) + 2^n q_b\right)\right] + Z_a \tag{10}$$

式中，非定点整数的部分只有 $\frac{S_w S_x}{S_a}$，通过合理的 n 位变换可将卷积层输出数据量化公式转为定点计算，$2^n q_b$ 同样在公式上对 32 位的偏置数据进行移位量化，通过这种方式可以有效地减少数值损失，在硬件逻辑上也容易实现。

在 CSP 模块中引入残差结构可以增加了层与层之间的梯度值，避免了因网络加深而导致的梯度消失问题。在浮点类型的网络模型计算中，假设输入到残差结构中的两个实数值分别是 r_1 和 r_2，相加后得到的和用 r_{add} 表示，计算公式如公式（11）所示：

$$r_{\text{add}} = r_1 + r_2 \tag{11}$$

在量化到定点整数后残差结构的过程计算往往需要考虑两个数据量化后的数值范围。因此，在数据进行相加后需要重新量化以减少模型计算上的误差，量化的计算公式如公式（12）所示：

$$q_{\text{add}} = \frac{S_1}{S_{\text{add}}}(q_1 - Z_1) + \frac{S_2}{S_{\text{add}}}(q_2 - Z_2) + Z_{\text{add}} \tag{12}$$

拼接（Concat）层主要用于网络模型中不同层之间的特征图融合，在量化时可采用与残差结构类似的操作，对输入的两个数值中的一个先进行量化后再拼接，最后再对拼接的数据进行缩放输出。

2　硬件加速器框架设计

2.1　混合流可配置硬件加速器架构设计

YOLOv5n 网络模型可以根据基本模块将网络层分为卷积层、池化层、激活层、上采样层、拼接层。通过分析网络模型各层组成成分和运算特点，硬件加速器中子模块的参数配置和部署情况如表 1 所示。

表 1　硬件加速器模块配置

模块名	参数配置	部署
Conv1	$K=6,S=2,P=2$	On PL
Conv2	$K=3,S=2,P=1$	On PL
	$K=3,S=1,P=1$	On PL
Conv3	$K=1,S=1,P=0$	On PL
MaxPool	$K=5,S=1,P=2$	On PL
Relu	N/A	On PL
Upsample	Nearest	On PL
Concat	N/A	On PS

Conv1～Conv3 是具有不同卷积核尺寸大小、步长和补零数的标准卷积层，其中 K，S，P 分别代表卷积核尺寸，步长和补零数。为避免 PL 端与 PS 端之间频繁的数据交互，最大池化层、激活层和上采样层在 PL 端进行单独设计模块。拼接层模块根据运算特点可在 PS 端进行地址映射实现两层特征图之间的连接。

CNN 加速器中有两种常用的硬件架构，一种是设计通用的模块处理单元，不同网络层共用通用的处理单元，在运行前通过调配参数实现驱动，该加速器设计方式节省硬件资源，但会导致系统运行速度十分缓慢，出现高延迟情况。另一种方法是对每个网络层单独设计专用的模块处理单元，实现系统快速运行，该架构能够实现低延迟，但能效低，并且需要消耗大量的硬件资源，对模型的部署不利。为解决这些问题，本文设计了一种混合流可配置硬件加速器架构，如图 2 所示，该架构在延迟、能效和硬件资源利用率之间取得了良好平衡。

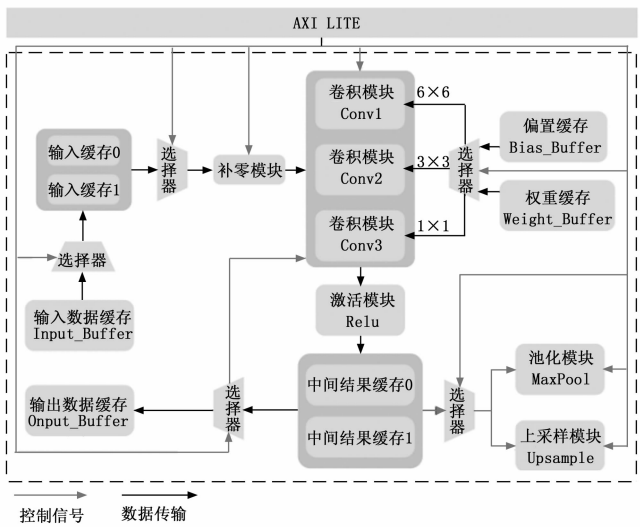


图 2　混合流可配置硬件加速器架构

2.2　卷积模块架构设计

在 YOLOv5n 网络模型中，卷积层的占比最多，因此需要频繁地进行加速处理。由表 1 可知，卷积层大致可分为 3 种类型，本文为 3 种不同的卷积层分别设计了单独的卷积处理模块，每个卷积加速模块中都设有专用的处理单元，以实现最佳的加速性能。本文主要依据卷积模块的计算特点，采用输入特征图通道和输出特征图通道并行循环展开的方式，以 Conv2 为例，卷积模块架构设计如图 3 所示。

输入特征图尺寸大小由行数 C，列数 R，通道数 N 表示。考虑到 FPGA 的静态随机存储器（BRAM，block random access memory）十分有限，无法在计算前直接将整个输入特征图数据全部存储到 FPGA 的片上存储资源中。因此，本文采用输入特征图数据循环分块

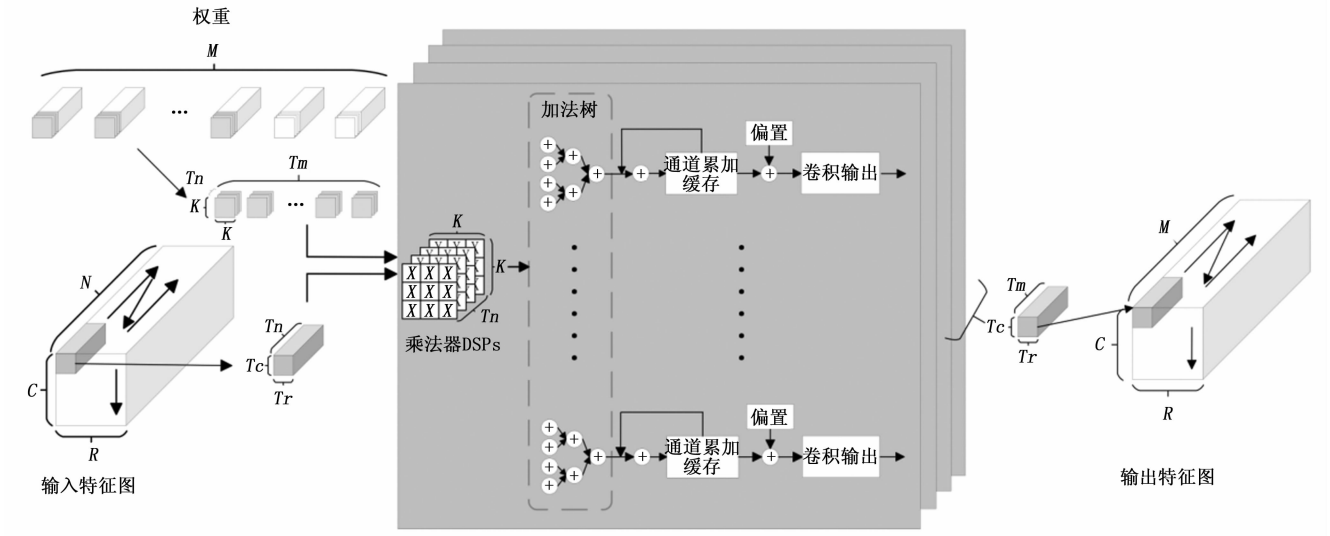


图 3 卷积模块架构

的设计方式，预先将图像数据存储在双倍速率数据存储器（DDR，double data rate）中，在卷积模块计算前进行分块输入到 BRAM 中保存。一次循环读取的数据量大小主要由分块后的输入特征图和权重卷积核尺寸所决定。将分块后相对应的输入特征图和权重数据块传输至卷积计算单元中，并进行充分的循环利用。

输入特征图的补零操作在卷积计算单元运算前进行，根据网络层的补零参数将四周地址位空出，用数据 0 填充即可。补零后的分块输入特征图数据与对应的权重数据送入乘法器单元进行卷积乘法运算。由于采用的数据分块技术，一次卷积操作只计算 T_n 个输入通道数据，所得到的累加数据和并不是最终的结果，将其存回片外 DDR 会增加内存的访问次数，导致运行速度减慢，因此卷积计算的部分累加和数据先经过三级流水线加法树进行累加，再通过输出通道数并行计算的方式同时得到 T_m 个部分通道和结果，将其存入到中间缓存区中与后续所计算得到的通道数据进行求和。在计算完一次分块数据后，输入特征图数据保持不变，卷积核权重数据沿着输入通道和输出通道再次更新相同个数的数据后与输入数据进行乘累加运算，计算完的结果与中间缓存区的数据进行累加后再存入，待到滑动立方体循环遍历完所有通道数的输入数据后，最终得到的数据和与该层偏置数据进行相加，即为卷积输出数据，将其输送至片外 DDR 中存储。

2.3 乒乓双缓存与层间流水线设计

为进一步提高整个加速器系统的数据吞吐量，减少总线传输与内存访问之间的延时，本文设计了一种双缓存区存储方式，即乒乓双缓存。通过使用两个相同大小的缓存区，读取、计算和写回操作可以同时循环进行，从而减少延时。乒乓双缓存的核心就是相邻两层之间的分时操作，在网络层操作一块缓存区的同时另一网络层操作另外一块缓存区，以此往复。此外，在网络模型中卷积层存在大量的乘累加运算，传统的流水线中，层与层之间需要完全等待上一层的计算完成之后，将结果传输到下一层才能开始计算，数据的停滞很大程度上会浪费大量的模块运算时间。这种传统的流水线方式会占用大量的带宽，增加额外的功耗和推理时间。为优化数据传输过程，减少数据的总体传输时延，本文设计了乒乓双缓存与层间循环流水线方法，如图 4 所示。

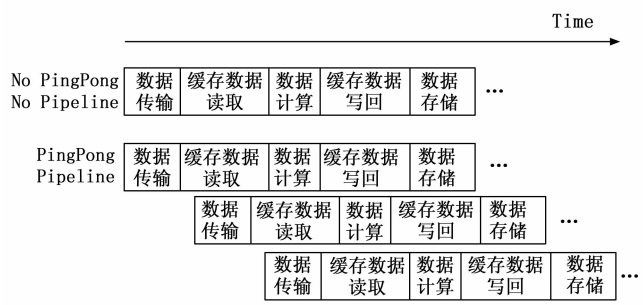


图 4 乒乓双缓存与层间循环流水线架构

利用 FPGA 的并行计算特点和流水线运行的特点，以硬件资源换取时间，提高整个系统加速速度。将整个推理过程划分为多个子阶段，每个阶段的平均处理速率基本相同，使得相邻两层之间无需等待上一层数据全部计算完即可开始下一层模块数据运算，每个时钟周期的输出都对应于固定时钟周期间隔之前的输入。因此，乒乓双缓存与层间循环流水线设计架构可以并行地执行每一级的操作，从而提高系统数据的吞吐率。同时可以避免中间缓存结果频繁地访问片外存储器，缩短流水线启动延时，加快系统的运行速度。

2.4 池化模块架构设计

池化模块作为 CNN 的核心组件，核心作用是降低

特征图维度、保留关键信息。最大池化是最常用的池化方法，它在二维窗口上运行，并输出窗口的最大值。YOLOv5n 网络模型中的最大池化模块均存在于 SPPF 模块中，采用步长为 1，滑动窗口大小为 5×5，补零为 2 的池化窗口来对图像进行下采样处理。最大池化层操作与卷积层操作类似，不同在于池化层没有权重数据，因此无需访问外部存储器 DDR，且进行的计算为比较运算。本文设计的池化模块操作如图 5 所示。

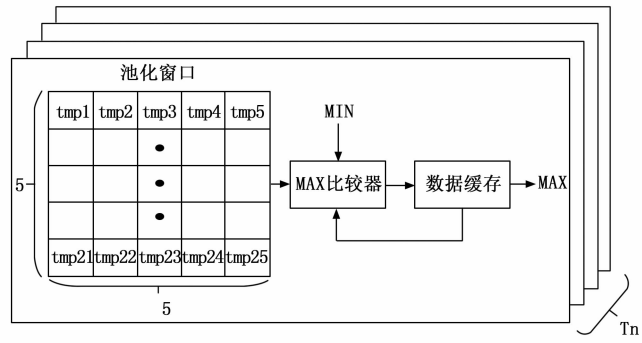


图 5　池化模块架构

在进行池化运算之前，按照池化窗口大小从输入缓存区中取出 5×5 个输入特征图数据，并将数据按顺序依次输入至最大值比较器中。首位数据与预先设置的最小值进行比较，得到的结果放入数据缓存区中存储，待到第二位数据输入至比较器中再将缓存区中的数据重新放入比较器中得出最大值，以此往复，直到运算完整个池化窗口的数据得出最后的最大值，再将最大值存入输出缓存区。在一次池化窗口操作结束后，池化窗口根据步长向前移动来进行下一窗口的池化操作，对输入特征图数据重复上述操作，直至所有数据均执行完最大池化操作。

2.5　插值型上采样模块架构设计

YOLOv5n 网络模型中，位于颈部网络的两个上采样层为了使得图像符合显示区域的大小，提升其特定尺寸特征图的空间分辨率，以恢复原始特征图的细节信息或匹配其他特征图维度。上采样层与卷积层不同，只需在原有图像像素的基础上在像素点附近插入新的数据。因此，本文设计了一种插值型上采样模块架构，如图 6 所示。

上采样模块初始化后，从第一行第一列数据顺序开始，列数据连续输出两个周期的数据后再进行下一列数据，重复此操作直到一行所有列数据输出完后再回到该行首个数据地址位，以相同的方式再次读取该行的所有列数据，此时，第一行所有数据均完成了上采样设计，可以将地址跳转至下一行，重复所有行所有列数据即可完成整个特征图数据的上采样操作。

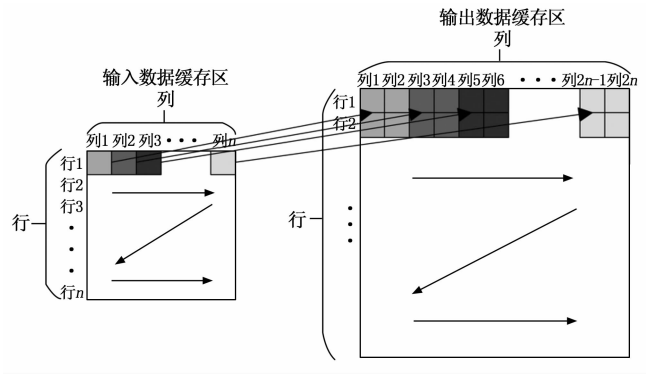


图 6　插值型上采样模块架构

3　目标检测硬件加速系统

目标检测硬件加速系统的框架如图 7 所示，采用软硬件协同设计的方式来实现高能效 YOLOv5n 目标检测神经网络，整个加速器系统框架可分为 PS 和 PL 两部分。

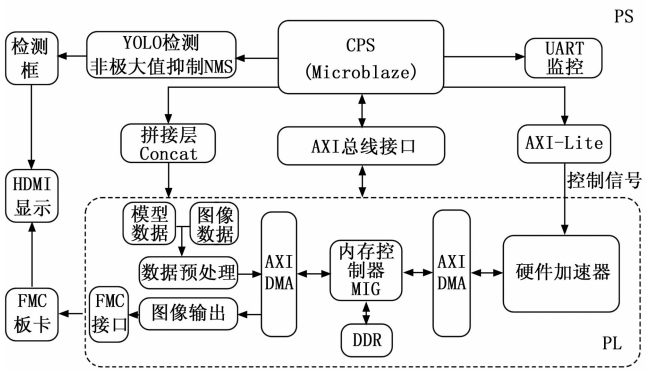


图 7　目标检测硬件加速系统

PS 端处理器 CPU 通过 AXI（Advanced eXtensible Interface）总线配置加速器参数实现前向推理，同时调度内存控制器（MIG，memory interface generator）来控制片外存储器 DDR 缓存数据。CPU 软件端通过操作数指定的地址位和数据传输长度实现数据交互。UART 串口用于监控计算整个加速器系统的推理时间。模型加速推理完后通过非极大抑制（NMS，non maximum suppression）算法找出最符合目标的检测框进行显示，并输出坐标位置信息和目标置信度数值。

PL 端部分主要是实现硬件加速器的前向推理。在系统上电后，CPU 软件端首先通过 FMC（FPGA Mezzanine Card）板卡配置显示器初始化参数。燃弧图像和 YOLOv5n 模型参数通过数据预处理模块进行数据重排序后保存到存储器中，并由 AXI-DMA（Direct Memory Access）模块传输到硬件加速器中。整个系统在硬件加速器中通过专用加速模块对输入的数据进行高效的前向推理处理。在完成硬件加速器前向推理后，PL 端会向

PS 端发送中断信号并传输结果信息。

整个系统由 AXI 总线接口进行数据交互和配置模块参数，从而实现目标检测系统中不同网络层的前向推理。AXI-DMA 控制器作为枢纽，承担着 PL 端硬件加速器模块与 DDR 之间的数据传输工作。硬件加速器在完成一次加速运算后通过总线向 CPU 软件端发送中断信号，软核处理器经过数据分析后向 AXI-Lite 总线下达模块控制命令重新配置硬件加速器中的网络层参数，如此反复直到完成整个检测网络的前向推理。

系统软硬件协同设计运行方式如图 8 所示，硬件 PL 端与软件 PS 端并行化的架构可充分缩短系统延迟，同时最大化发挥软件端的任务调度能力与硬件端加速推理性能。

4 结果分析

4.1 模型训练结果与分析

在电脑端深度学习框架下对 YOLOv5n 网络进行全精度训练与 QAT 量化训练。本文选用由高速红外摄像头捕捉到的 3 000 张真实场景的燃弧图像进行实验训练。

为了准确评估量化前后的模型检测性能，本文将采用检测算法中的精确率（P，precision）和平均精度均值（mAP，mean average precision）作为模型检测效果的评估指标，计算公式如式（13）～（16）所示：

$$P = \frac{TP}{TP + FP} \tag{13}$$

$$R = \frac{TP}{TP + FN} \tag{14}$$

$$AP = \int_0^1 P(R)\mathrm{d}R \tag{15}$$

$$mAP = \frac{\sum_{n=1}^{N} AP_n}{N} \tag{16}$$

其中：TP 为真正例，FP 为假正例，FN 为假负例，R 为召回率，表示模型识别到的真正例占所有检测量的比例。平均精度 AP 用于衡量模型对目标的检测质量。

将 YOLOv5n 检测网络分别基于浮点全精度模式和 QAT 量化模式下训练迭代 300 轮，P 和 mAP 的数值变化曲线如图 9 所示。

由模型指标变化对比曲线图可得出数据收敛后，浮点全精度训练模式下的模型 P 和 mAP 分别为 84.4%、82.5%，QAT 模型 P 和 mAP 分别为 82.2%、80.7%，量化后的性能指标都有所下降，但损失值均在允许误差范围内，对部署在 FPGA 开发板上的加速器系统检测效果的影响不大。

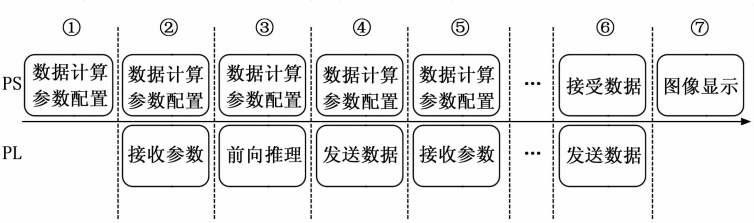
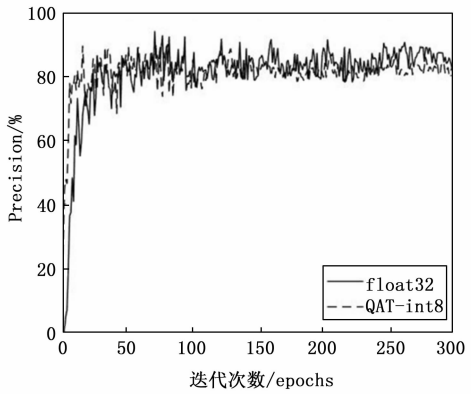


图 8 系统软硬件协同运行方式



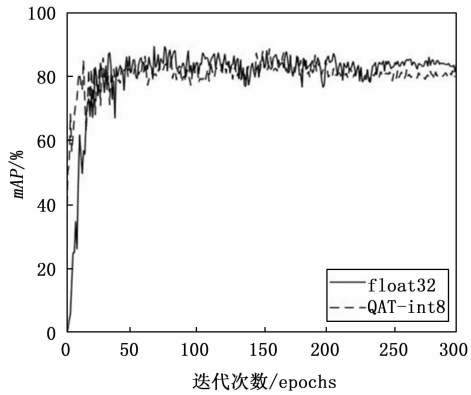


图 9 模型指标变化对比曲线

QAT 量化后的数据类型由浮点 32 位转为 8 位定点整数，网络模型大小缩减至浮点全精度模型的 1/4，如表 2 所示，降低了硬件加速器中各个模块的参数计算量，极大地减少了硬件资源和存储资源的消耗，使系统能够实现更大的并行度计算，加快运行速度。

表 2 浮点全精度模型与 QAT 模型对比

训练模式	模型数据类型	模型大小/M
浮点全精度	Float 32	7.047
QAT	Int 8	1.831

4.2 模型实现结果

本文选择 Xilinx Virtex-7 VC707 作为目标检测硬件加速系统的部署平台，在表 3 中具体列出了硬件加速器资源的利用情况。

表 3 硬件加速器资源利用率

资源	LUT	FF	BRAM	DSP
消耗	79 397	80 545	126.5	221
总数	303 600	607 200	1 030	2 800
利用率/%	26.15	13.26	12.28	7.89

其中，查找表（LUT，look up table）通过存储预设真值表来实现任意组合逻辑运算。触发器（FF，filp flop）主要用于时序电路中的时序逻辑实现，并且还可用于存储逻辑状态。BRAM 是硬件加速器平台主要的片内存储资源，主要用于中间数据的缓存。数字信号处理器（DSP，digital signal processor）主要用于实现乘法和加法等计算功能。

为了评估硬件加速系统的能效，本文分别在 Intel I3-12100F、GTX 1650 和 Virtex-7 VC707 平台实现了 YOLOv5n 网络模型。表 4 中列出了本文设计的系统与 CPU、GPU 平台实现性能的对比。在电脑端的深度学习环境框架下，可在 CPU 与 GPU 平台上对 YOLOv5n 网络进行模型训练，并记录推理时间、吞吐量及功耗指标。本文在 FPGA 开发板上设计实现的系统功耗仅为 2.88 W，在远低于 CPU 与 GPU 的情况下，实现的能效 9.43 GOPS/W，优于 CPU 平台和 GPU 平台上的模型实现。因此，将 YOLOv5n 部署在 FPGA 上具有极大的优势。

表 4　不同平台性能比较

平台	Intel I3-12100F	GTX 1650	Virtex-7 VC707
推理时间/ms	25.9	7.2	151
吞吐量/GOPS	158	569	27.15
功耗/W	58	67	2.88
能效/(GOPS/W)	2.72	8.49	9.43

表 5 为本文设计的 YOLOv5n 目标检测硬件加速器与其他文献在不同 FPGA 硬件平台实现的不同硬件加速器的性能比较，主要通过吞吐量与功耗的比值能效来衡量不同目标检测系统之间实现的性能差异。相比之下，本文在各方面性能都优于文献［13］和文献［21］设计的硬件加速系统。吞吐量对比文献［20］还有一定的差距，但功耗低，总体能效上更优。本文在追求系统高运算性能的同时还能够兼顾检测速度和功耗的平衡，更加适用于实际应用场景。

表 5　硬件加速器性能比较

	文献[13]	文献[20]	文献[21]	本文
网络模型	YOLOv5s	YOLO v3-Tiny	YOLO v3-Tiny	YOLO v5n
硬件平台	AX 7350	Ultra96 V2	Zed board	VC 707
精度	Fixed 16	Int 8	Fixed 16	Int 8
时钟频率/MHz	100	250	100	100
吞吐量/GOPS	10.80	31.50	10.45	27.15
功耗/W	3.43	4.26	3.36	2.88
能效/(GOPS/W)	3.15	7.40	3.11	9.43

如图 10 所示为目标检测硬件加速系统的检测效果示意图。在右侧电脑端将程序烧录进 FPGA，系统检测出的结果显示在左侧电脑屏幕上。

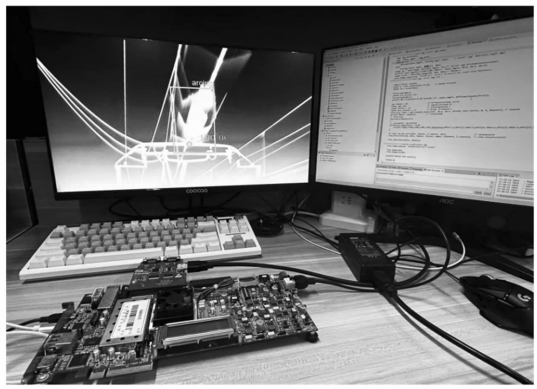


图 10　检测效果示意图

5　结束语

本文提出了一种基于轻量化网络模型 YOLOv5n 的硬件加速方法，通过优化设计，将其应用在 Xilinx VC707 FPGA 硬件平台上实现燃弧目标检测。通过网络模型层融合进行优化以及 QAT 对 YOLOv5n 网络进行数据量化，降低了模型参数量和计算复杂度，减少了硬件资源的消耗。在硬件模块之间采用乒乓双缓存机制和层间流水线设计提高了数据吞吐量，加快系统运行速度。利用 HLS 开发工具设计了一种混合流可配置的硬件加速器架构，并通过软核处理器配置参数控制专用的加速模块来实现检测网络的前向推理。实验得出数据，本文设计的硬件加速系统单张图片的推理时间为 151 ms，数据吞吐量为 27.15 GOPS，功耗仅为 2.88 W，能效高达 9.43 GOPS/W，明显优于 CPU、GPU 和其他 FPGA 加速系统，在性能、功耗和资源利用率上达到了更好的平衡。

参考文献：

[1] HAO J，GAO G，WU G. Dynamics of pantograph-catenary arc during the pantograph lowering process［J］. IEEE Transactions on Plasma Science，2016，44：2715－2723.

[2] PEI Y，HUANG Y，ZOU Q，et al. Effects of image degradation and degradation removal to CNN based image classification［J］. IEEE Transactions on Pattern Analysis and Machine Intelligence，2021，43：1239－1253.

[3] WANG J，ZHENG Y，WANG M，et al. Object scale adaptive convolutional neural networks for high-spatial resolution rmote sensing image class-ification［J］. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing，2021，14：283－299.

[4] HUANG Q，CAI Z，LAN T. A single neural network for mixed style license plate detection and recognition［J］. IEEE Access，2021，9：21777－21785.

（下转第 96 页）