面向无人机端侧智能识别的多模可重构加速器

张健铭[1],周进良[1],张一涵[2],谷海宇[2]
(1.成都西南信息控制研究院有限公司 系统工程所,成都 611730)
(2.哈尔滨工业大学 航天学院,哈尔滨 150001)

摘　要:无人机端侧处理多模态目标识别任务时存在压缩模型适配性不足,缺乏混合精度计算支持及软硬件协同较差等问题,提出并实现一种面向端侧智能识别的多模可重构加速器。在硬件层面,设计支持多粒度可变位宽的运算架构,结合硬件友好型剪枝机制实现稀疏模型的低开销推理;在软件层面,基于深度学习编译框架(TVM)构建编译器,通过改进乒乓机制实现指令并行,并采用算子融合技术降低数据搬运开销。结果表明:该加速器对可见光、红外、激光雷达数据的平均识别响应时间分别为28.9、27.9、94.96 ms,平均准确率均超90%。从而证明,相应的软硬件协同设计方法显著提升了端侧设备的并行计算能力,满足了无人机在复杂环境下多模态目标实时侦察的需求。

关键词:现场可编程门阵列;加速器;软硬件协同优化;多模态目标识别;深度学习编译器
中图分类号:V279　　**文献标识码**:A

Multi-modal reconfigurable accelerator for edge-side intelligent recognition on UAVs

ZHANG Jianming[1], ZHOU Jinliang[1], ZHANG Yihan[2], GU Haiyu[2]
(1. Institute of Systems Engineering, Southwest Research Institute of Information Control, Chengdu 611730, China)
(2. School of Astronautics, Harbin Institute of Technology, Harbin 150001, China)

Abstract: To address the issues of insufficient compressed model adaptability, lack of mixed-precision computing support, and poor hardware-software collaboration in UAV edge-side processing for multimodal target recognition tasks, this paper proposes and implements a multimodal reconfigurable accelerator for edge intelligent recognition. At the hardware level, an arithmetic architecture supporting multi-granularity variable bit widths is designed, and low-overhead inference of sparse models is achieved combined with a hardware-friendly pruning mechanism. At the software level, a compiler is constructed based on the deep learning compilation framework (TVM), instruction parallelism is realized by improving the ping-pong mechanism, and operator fusion technology is adopted to reduce data transfer overhead. Experimental results show that the average recognition response time of the accelerator for visible light, infrared, and LiDAR data is 28.9 ms, 27.9 ms, and 94.96 ms respectively, with an average accuracy rate exceeding 90%. The results show that the proposed hardware-software co-design approach significantly improves the parallel computing capability of edge-side devices and satisfies the requirements of UAV-based real-time multimodal target reconnaissance in complex environments.

Key words: field-programmable gate array (FPGA); accelerator; hardware-software co-optimization; multi-modal target recognition; deep learning compiler

收稿日期:2026-04-10;　**修回日期**:2026-06-11
基金项目:国家自然科学基金(U2541268)
通信作者:谷海宇(1992—),男,博士,副教授。E-mail: guhaiyu@hit.edu.cn
引用格式:张健铭,周进良,张一涵,等. 面向无人机端侧智能识别的多模可重构加速器[J]. 航空工程进展.
ZHANG Jianming, ZHOU Jinliang, ZHANG Yihan, et al. Multi-modal reconfigurable accelerator for edge-side intelligent recognition on UAVs[J]. Advances in Aeronautical Science and Engineering. (in Chinese)

0 引 言

在城市反恐、山区巡检、洞穴科考等任务中，无人机需在未知空间开展侦察任务[1]。无人机需深入城市、山区及地下溶洞空间，其环境复杂多变，天气、光照及动态干扰会影响侦察识别效果，因此需通过激光雷达点云、可见光图像、红外多类数据，经宽带自组网回传至指挥中心，结合地图信息识别车辆、飞行物等敏感目标及人员行为姿态，为指挥控制系统提供态势支撑[2]。然而，城市环境中高大密集建筑、复杂地下空间结构易引发弱网问题[3]，导致多类数据难稳定回传，制约侦察识别效率，亟需高效端侧AI算力支撑数据本地化处理以减少数据回传量[4]。现场可编程门阵列(FPGA)具备可编程灵活性高、并行计算能力强、功耗可控性好的独特优势[5]，能适配无人机端侧多模态数据处理的动态需求[6]，因此当前已有大量研究基于FPGA开发端侧AI加速方案[7]，为无人机侦察系统的本地化智能处理提供技术基础[8]。

唐东洋等[9]面向受限环境下的微小型无人机任务需求，提出了一种低成本、低功耗、小尺寸的室内自主搜索方案，并利用集成YOLO-v2的芯片实现目标检测与跟踪，体现了无人机平台对轻量化智能感知与实时处理能力的迫切需求；石添介等[10]针对机载嵌入式计算系统中卷积神经网络模型规模大、资源占用高的问题，提出了模型超轻量化与FPGA硬件加速相结合的方法，并完成了机载超轻量化卷积神经网络加速器原型验证，为资源受限平台上的智能识别部署提供了参考；高远等[11]将卷积神经网络引入航空领域复杂任务建模中，构建了基于卷积神经网络(CNN)的气动力预测模型，并结合多项式混沌方法实现翼型鲁棒性优化，说明了卷积神经网络在航空任务中的高效建模能力与工程应用潜力。Lee等[12]提出了一种利用时间和空间缩减来提高能量效率，从而降低数据流复杂度的DNN推理加速器体系结构，通过将稀疏权重和输入激活拼接在一起，有效地实现并行执行。Chen等[13]设计并实现了可配置节能加速器，该架构被称为Eyeriss架构，采用168个PE阵列实现四级内存层次结构；采用CNN数据流，称为Row Stationary (RS)算法，该算法具有可重构性和节能性。通过片上网络(NoC)架构，有效地利用组播技术和点对点单周期数据传输来支持RS数据流。Aimar等[14]提出了一种节能灵活的CNN加速器架构，可以有效地用于低功耗/低延迟应用领域。所提出的架构加快了计算速度，并显着减少了内存需求，并在FPGA上实现。Hu等[15]为DNN设计了一个高效的可配置加速器。提出了一种基于数组的结构，具有四个级别的处理元素。该体系结构实现了高度并行的卷积运算计算，采用混合平稳(HS)存储模式，充分利用了片外内存带宽和片上内存占用。

现有研究仍存在三大核心瓶颈：其一，压缩模型适配性不足。现有FPGA加速器对剪枝后稀疏模型支持度低，额外硬件开销大，且难以兼容不同压缩率模型配置，导致推理性能衰减明显；其二，混合精度计算支持缺失。多模态数据处理需灵活适配多种位宽，传统FPGA统一计算架构仍无法高效兼容混合精度运算，数字信号处理(DSP)资源利用率仅为理论值的57%以下，制约算力释放；其三，软硬件协同优化不足。深度学习编译器是模型与硬件的关键衔接，DNN Builder工具[16]可自动生成FPGA上的优化加速器，TVM框架[17]通过学习式成本建模优化代码并支持硬件加速器，但主流编译器与FPGA硬件适配性仍差，如TVM原生仅支持指令串行执行，算子间数据搬运开销占比超40%，且缺乏针对性调度优化，导致硬件算力无法充分发挥。

针对上述问题，本文围绕无人机端侧多模态智能识别需求开展研究，硬件层面通过多粒度可变位宽运算与硬件友好型剪枝机制，实现混合精度数据高效处理与稀疏模型低开销推理；软件层面基于TVM构建编译器，通过指令并行优化与算子融合技术降低数据搬运开销，以解决现有方案在压缩模型适配、混合精度计算及软硬件协同方面的缺陷，为无人机侦察系统端侧智能处理提供高效技术支撑。

1 多模态协同加速器软硬件架构设计

为满足无人机侦察系统端侧智能处理的需求，本文围绕多模可重构加速器的软硬件架构开展设计，旨在突破现有FPGA加速器在多模态场景下的适配瓶颈。该架构面向压缩模型适配性不足、混合精度计算支持有限等问题进行针对性设

计,并以可见光、红外和激光雷达目标识别模型的推理运算为主要应用对象。其中,不同模态任务通过加载对应的模型权重、量化参数和指令流实现切换,从而完成可见光图像模型、红外图像模型和激光雷达点云模型的分别部署与加速。对于激光雷达点云任务,本文采用"CPU预处理+FPGA张量推理"的方式进行加速,即先在CPU侧将原始点云转换为BEV特征张量。FPGA端主要加速点云识别网络中的计算部分,包括卷积、矩阵乘法、激活和结果写回等操作。多模态可重构加速器架构如图1所示,主要包含FETCH模块、LOAD模块、STORE模块、GEMM核、Tensor ALU、CTRL模块、多个数据缓存单元、指令队列及控制队列。FETCH模块负责从动态随机存取存储器(DRAM)读取指令数据,对指令数据解码后分发给对应指令队列;LOAD模块负责从DRAM向各数据缓存单元加载输入、权重等数据;STORE模块负责将加速器的计算结果存储回DRAM;GEMM核负责实现卷积计算;Tensor ALU负责实现多种激活函数;CTRL模块负责控制其他单元的运行逻辑。考虑到多模可重构加速器需支持卷积、激活、池化等多种算子的灵活调用,且需适配不同模态数据的计算需求,TVM作为开源深度学习编译框架,具备端侧硬件适配性强、支持自定义算子映射及学习式优化的优势,为此本文基于TVM搭建深度学习编译器,通过编译器与硬件模块的协同设计,实现算子指令的高效生成与调度,为后续指令并行优化、算子融合等功能提供。

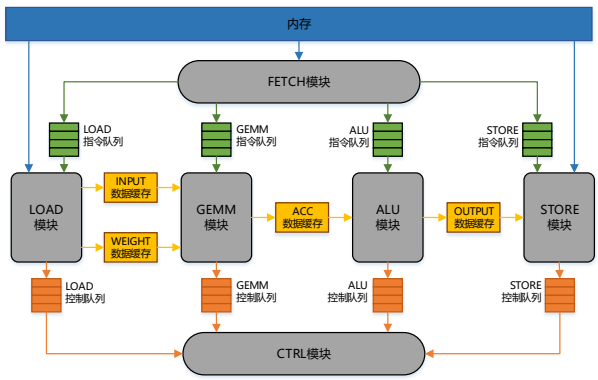

图1 多模态可重构加速器架构

Fig. 1 Architecture of multimodal reconfigurable accelerator

1.1 基于端侧智能的无人机识别加速

复杂环境下(如微光、雨、雪等)中对多模态感知数据的高效处理,本文设计的多模可重构加速器可实现对可见光、红外及激光雷达点云等数据的智能计算加速,显著提升无人机端侧平台的计算性能。

多模可重构加速器主要包括指令读取单元、数据加载单元、数据存储单元、矩阵乘法计算单元、算术逻辑计算单元、多个数据缓存单元和队列单元。首先,可见光和红外图像数据在CPU中进行通道拼接,并将拼接后的数据和融合模型的指令数据存储到相应的DRAM地址。接着,CPU通过寄存器映射的方式向FPGA发送指令数据的基地址信息、指令长度和加速器的启动标志信号。指令读取单元根据获取到的基地址信息和指令长度从DRAM中读取指令数据,在解码后分发给指令队列。数据加载单元从其对应的指令队列中读取指令,根据解码后的信息将DRAM中的输入数据、权重数据等加载到数据缓存单元中。计算模块根据数据缓存单元中的数据完成卷积、激活函数、量化等运算。最后,数据存储将最终的计算结果存回双倍速率同步动态随机存储器(DDR),当前卷积层计算完成后,FPGA通过寄存器映射的方式向CPU发送计算完成信号。融合模型和检测模型可以完全拆分成卷积层,通过编译器的控制顺序将对应层的权重数据和指令数据送入DRAM,在加速器中完成计算。对于卷积算子,本文采用卷积到矩阵乘法的映射方式。具体而言,输入特征图首先按照通道和空间维度进行分块,卷积核权重被展开为适合矩阵乘法的数据形式,随后由GEMM核完成主要乘加计算。LOAD模块负责将输入特征块和权重块从DRAM加载到片上缓存,GEMM核完成卷积对应的乘加运算,Tensor ALU负责激活、截断和重量化等逐元素操作,STORE模块将输出特征块写回DRAM。

通过上述数据流,普通卷积、1×1卷积、量化卷积以及点云BEV特征网络中的规则卷积均可统一映射到LOAD、GEMM、Tensor ALU和STORE流水线中执行,从而提高不同模态识别模型在同一硬件架构上的复用能力。

1.1.1 多粒度可变位宽运算

本文通过优化DSP乘法计算流程,提升DSP

资源的利用效率，实现了智能可重构计算加速器对混合精度数据处理的支持，加速器的DSP优化方式如图2所示，在FPGA中，DSP属于特殊运算资源，支持27×18位乘法运算，且具备累加、移位等功能，最大支持48位输出。当数据位宽为16 bit时，无需对数据进行预处理，仅根据指令配置对数据执行读取操作即可。权重共享量化方法如图3所示，当输入数据位宽为8 bit，且两个输入特征图数据位宽与权重数据位宽之和小于27 bit时，可在同一片DSP资源上对两个特征图输入数据进行拼接，同时得到两个乘法运算结果。编译器会将输入特征图数据分割为上半图与下半图两部分，通过按通道拼接上下半图数据（即将两个8 bit输入特征图数据拼接为一个16 bit数据），送入特征图输入存储单元。运算后的数据从输出缓存单元输入至PU模块时，在PU模块中拆分为两组数据执行并行运算。根据指令配置处理后的数据，分别通过AXI4接口写回DDR，且数据在DDR中的存放地址分别对应输出特征图的上半图与下半图。

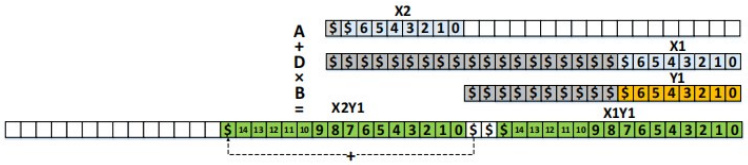

图2　加速器的DSP优化方式

Fig. 2　DSP optimization methods of the accelerator

加法器可用于累加包含单独高位乘积项与低位乘积项的45位乘积，累加单个45位积时，需对高位项与低位项进行校正累加。若最终累加结果无溢出，可通过简单运算分离，该方法的局限在于每个DSP Slice可累加的乘积项数量。由于高位项与低位项之间始终保持两位间隔，可保证在低位不溢出的情况下累加多达7个项；超过7个乘积项后，需额外使用DSP Slice以突破这一局限。

在INT8计算模式下，本文并非简单地将每个INT8乘法独立映射到一个DSP Slice，而是利用DSP Slice输入位宽较宽的特点，将两路低位宽数据按照高低位方式进行打包，使其在同一个DSP Slice中完成并行乘法计算。为避免两路乘积在累加过程中发生位域重叠，打包时在高位数据和低位数据之间预留保护位，并在累加结束后通过移位和截取操作恢复两路计算结果。由于卷积计算需要对多个乘积项进行累加，因此进一步考虑了累加过程中的位宽增长和溢出风险。实验设计中，每组打包乘积最多支持有限数量的安全累加；当累加项数量超过该范围时，需要额外DSP Slice完成结果校正和分离。因此，8个DSP Slice可实现14路INT8乘法累加，相比常规INT8乘法映射方式可提高DSP资源利用效率。因此，8个DSP Slice可执行14个INT8乘法与加法运算，相较于拥有相同数量乘法器的竞争型器件，INT8深度学习运算效率提升1.75倍。

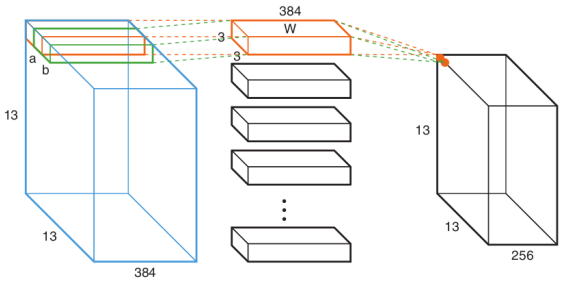

图3　权重共享量化方法

Fig. 3　Weight-shared quantization method

1.1.2　结构化剪枝与加速器架构协同优化

为实现智能可重构计算加速器对剪枝的支持，项目组通过轻量化算法与加速器协同优化，提出一种硬件友好型剪枝方法。载入模型为剪枝后的稀疏模型，加速器读取所有分组数据，即对多个通道的数据进行选择。

该剪枝方法通过保留局部最突出参数，可有效保留模型整体重要性，进而实现较高压缩率；同时，该方法符合脉动阵列工作模式，实现剪枝功能的额外开销较小。剪枝原理如下：在传入PE运算阵列的$N\times P$组特征图数据中，可对每P个相邻通道进行筛选，选择其中1个通道作为输入。PE运算阵列剪枝流程如图4所示，以$P=2$为例，可对每2个通道进行筛选，选择其中1个通道作为输入：即在通道1与通道2中选择1个通道数据作为PE运算阵列第1行输入数据，在通道3与通道4中选择1个通道数据作为第2行输入数据，在通道$2N-1$与

通道2N中选择1个通道数据作为第N行输入数据。上述通道选择剪枝策略同样可扩展至每4个通道选择1个通道或其他分组粒度。当采用每4个通道选择1个通道的方式时,加速器在每4个相邻输入通道中选择重要性较高的1个通道送入PE运算阵列,其余通道在当前层不参与乘加计算。与每2个通道选择1个通道相比,该方式能够获得更高的通道压缩率和更低的计算量,但也可能带来更明显的精度下降。因此,在实际部署时,可根据不同网络层的冗余程度选择不同分组方式:对浅层或特征敏感层采用不剪枝或较小分组粒度,对冗余度较高的中深层采用较大分组粒度。从硬件实现角度看,该方法仅需扩展通道选择器并增加相应配置位,不改变PE阵列和通用矩阵到矩阵乘法(GEMM)计算单元的主体结构,因此仍具有较好的硬件友好性。

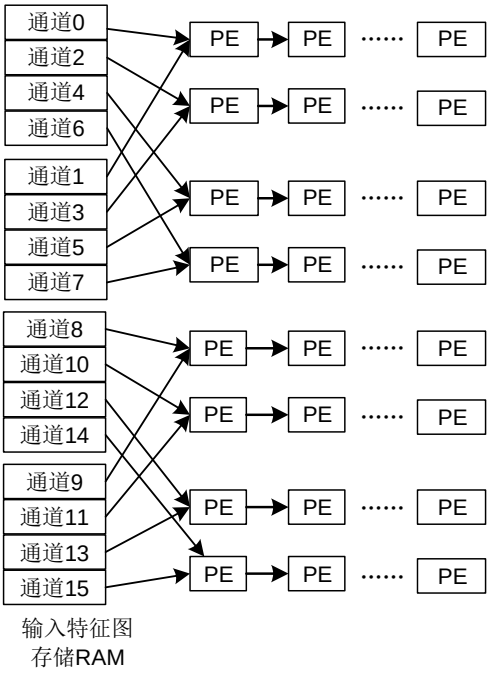

图4 PE运算阵列剪枝流程

Fig. 4 Pruning flow of PE computing array

特征图由多个二维特征点组成,对于输入特征图中某一特征点的数据,需根据该特征点的输入通道,将其存放至对应BRAM(1个BRAM对应1个通道)。在每一层神经网络开始计算前,需对剪枝模式进行配置,配置内容主要包含两部分:一是通道选择模式配置,可选择相邻2个或4个通道数据中的1个输入至PE运算阵列的1个PE单元,也可选择将每个通道的数据均输入至PE运算单元;二是滤波器分组模式配置,执行卷积运算时,输入特征图数据需分别与不同滤波器进行点乘操作,对于不同分组的滤波器,可从之前配置的多个通道备选数据中选择不同通道数据输入至该组内的PE单元,且相同分组的滤波器应选择相同通道数据,每一层卷积网络的滤波器尺寸均保持一致。

在硬件实现时,结构化剪枝产生的通道选择结果会在编译阶段转化为对应层的通道选择配置,并随该层指令一同写入指令流。加速器执行时,LOAD模块根据配置结果加载被保留通道对应的输入特征和权重数据,被剪枝通道不再参与后续乘加计算。这样既减少了无效计算,也降低了片上缓存访问和数据搬运开销。

与非结构化剪枝相比,该方法不需要复杂的稀疏索引解析和不规则访存控制,更适合FPGA中的规则阵列计算结构。因此,本文采用结构化通道选择方式,使剪枝模型能够以较低硬件开销映射到PE阵列和GEMM计算单元中。

1.2 深度学习编译器

未经优化的原始神经网络直接用于推理时,会产生大量冗余计算与数据传输开销。本文引入TVM作为端到端编译框架,并针对多模态数据的量化需求,解决了原生TVM QNN机制仅支持uint8类型而导致与加速器硬件量化不兼容的问题。本文提出一种基于“量化参数-计算图结构”协同优化的数值体系,通过构建动态缩放因子(Scaling Factor)与偏移量(Offset)的线性映射模型,将量化后的int8数据直接映射至硬件GEMM核的计算位宽中。该机制允许编译器在编译阶段预计算重量化(Re-quantization)参数,将其以常数形式植入指令流,从而避免了在端侧推理时进行复杂的浮点运算,有效降低了多模态数据转换的耗时。

TVM支持两种运行量化模型的方式,一种是导入全精度模型,使用TVM自带的基于KL散度的量化工具,这种方法的量化过于简单,虽然对模型的兼容性好,但是精度下降很大,无法满足精度要求。另一种是导入预量化的模型,利用TVM已有的量子神经网络(QNN)机制将量化模型转换到现有的算子调度上。然而现有机制只支持导入量

化为uint8的类型，算子的计算也是根据uint8来设计的，导致与加速器的量化方式不兼容。针对上述问题，本文基于“量化参数—计算图结构”协同优化的思路，提出定制化QNN数值体系构建方法，设计动态可配置的量化位宽与参数调节机制，构建适配多模态数据的QNN数值体系，量化卷积算子的流程图如图5所示，一个量化模型的卷积算子，相较无量化模型的卷积输入输出参数，多了每个参数分别对应的缩放尺度和偏移量，对应的计算模式也有所变化。该方法可实现QNN对多模态数据的精准适配，在保证激光雷达点云目标定位精度、可见光与红外图像目标识别精度的同时，降低模型转化耗时与端侧计算开销，为后续多模可重构加速器的高效推理提供适配性良好的量化模型基础。

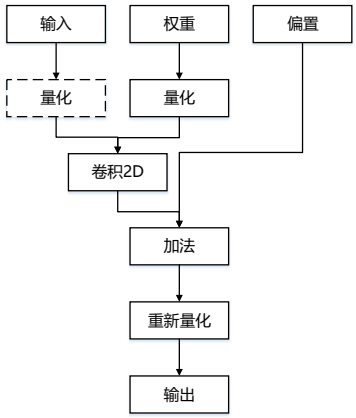

图5 量化卷积算子的流程图

Fig. 5 Flowchart of quantized convolution operator

1.2.1 指令并行执行优化

TVM原有的指令调度仅支持GEMM与算术逻辑部件(ALU)指令串行执行，导致硬件模块在数据搬运或非卷积计算时出现空闲。本文通过改进乒乓机制并引入乱序执行控制逻辑来提升并行度，优化前、后ALU与GEMM的并行方式分别如图6～图7所示，可直观观察到ALU与GEMM实现了计算并行，充分利用硬件模块。为实现该优化，首先改进乒乓地址：将原有相邻的两块乒乓地址调整为第二块乒乓起点位于静态随机存取存储器(SRAM)中点，便于硬件对SRAM进行分区管理；其次改写指令流，在虚拟线程执行方式改变后，将指令改写为对应模式；最后为实现对各模块的更细粒度管理，在指令流生成后扫描并生成指令乱序执行单元所需的各类同步标志位。

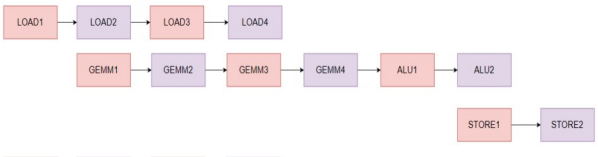

图6 优化前各模块并行方式

Fig. 6 Parallel mode of each module before optimization

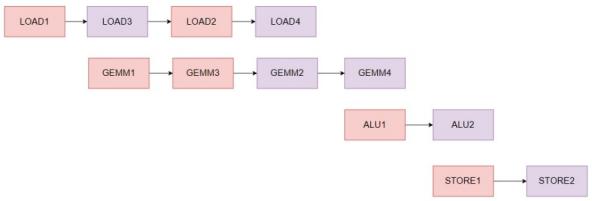

图7 优化后各模块并行方式

Fig. 7 Parallel mode of each module after optimization

1.2.2 算子融合支持

TVM原本的计算图包装和优化方式都不适配网络状况预测模型，所以需要进行定制化的计算图优化。首先，需要确定在加速器上执行的部分，确定后对图包装的配置进行设置，生成标志位；在加速器上执行前需要进行维度拆分，由于在加速器上会进行计算循环分块，所以对输入通道数有要求——必须是最小通道数的整数倍，所以对于不满足要求的数据，使用0值进行填充，填充后进行维度拆分(四维变为六维)，满足循环分块的需求；最后，为了保持输出一致，能通过编译器的形状检查，计算结束后需要进行切片，恢复原本的输出形状。

除了维度拆分的图优化，在加速器上运行的部分还可以进行算子融合优化。每个算子进行计算时都包含了三个操作“读数据—计算—存数据”，数据的搬运开销随算子数量增加而增加，所以为了减少数据搬运的开销，让该块数据的操作尽可能在DRAM中一次性完成，实现的方式就是算子融合，例如卷积和偏差加，算子融合前的计算过程是“取数据—卷积—存数据—取数据—偏差加—存数据”，若将两个算子融合，计算过程就变为“取数据—卷积—偏差加—存数据”，节省了“存数据—取数据”的开销，过程如图8所示。具体地，为了实现算子融合，需要手动在模型可融合的层后插入标志位，在编译器进行指令生成时，先生成整个融合算子的指令再生成数据搬运指令。在优化过程中发现TVM对于ReLU的优化不够完善。ReLU本质上是数值截断指令，而卷积层在计算完

毕后每一层也会插入一个数值截断指令,因为算子的输入和输出虽然是int8类型,但中间的计算数据类型是int32(为了防止溢出),所以结果也需要截断以满足条件。两个数值截断指令可以融合为一个,直接消去了relu的计算过程,减小了计算量。

在编译流程中,TVM首先对输入模型进行算子识别、张量形状分析和量化参数解析,然后根据加速器支持的指令格式生成LOAD、GEMM、Tensor ALU和STORE等硬件指令。对于不同模态识别模型,编译器根据模型结构和输入张量形状生成对应的指令流、权重布局和数据搬运顺序,从而实现不同模型在同一加速器上的部署。

在指令并行优化方面,编译器会分析不同硬件模块之间的数据依赖关系,在保证前后算子依赖正确的前提下,使数据加载、矩阵计算、逐元素运算和结果写回尽可能重叠执行。具体而言,通过实现GEMM与ALU的乱序并行,可有效消除硬件空闲时间,从而降低推理延迟。以ResNet-18的INT8计算为例,原始未经优化的加速器推理时延为60.03 ms,经过指令并行优化后可降至50.12 ms,时延降低约16.5%。

在算子融合优化方面,编译器将卷积、偏置加、激活和重量化等可连续执行的算子合并为一个计算阶段,使中间结果尽量保留在片上缓存中,避免频繁写回和再次读取DRAM,从而降低数据搬运开销。同时,该优化还可减少诸如ReLU等逐元素算子的冗余计算。以ResNet-18的INT8计算为例,经过算子融合后,加速器推理时延可进一步降至45.74 ms,相较于未经优化的60.03 ms,时延降低约23.8%。

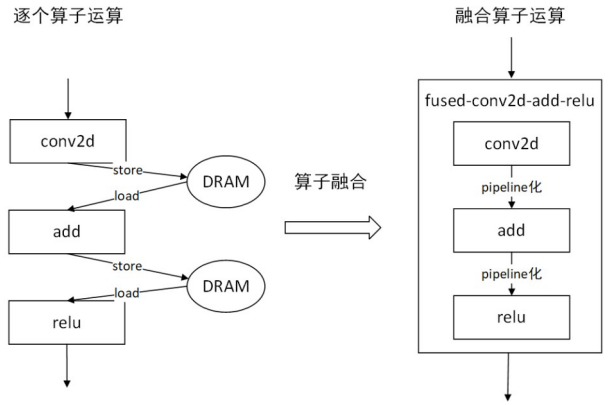

图8 算子融合过程

Fig. 8 Operator fusion process

2 实 验

2.1 实验设置

为从多维度综合评估多模可重构加速器的性能表现,基于Xilinx UltraScale+系列现场可编程门阵列(具体型号为XCZU19eg)开发板搭建实验环境,用于测试该加速器对可见光图像、红外图像及激光雷达点云数据的计算性能;同时,搭建基于Jetson Nano的对比平台,以凸显本加速器的性能优势,如图9所示。

图9 Jetson Nano开发板

Fig. 9 Jetson Nano development board

2.2 验 证

本研究的目标识别性能验证基于多源感知任务数据集开展。数据采集平台选用六自由度并联机构平台(如图10所示),用于模拟无人机在空中飞行时的俯冲、盘旋等姿态,其载荷平台集成高清可见光摄像头与非制冷红外热像仪,两类传感器通过时间戳同步模块实现数据采集时序对齐。采集区域选取室内和室外综合试验场,包含车辆、行人和无人机三种识别对象。激光雷达点云数据选用KITTI公开数据集[18]的3D目标检测子集。该子集源自德国卡尔斯鲁厄理工学院采集的真实交通场景数据,包含Velodyne HDL-64E激光雷达生成的点云数据(点云密度约100万点/s,测距范围0.5~120.0 m),涵盖城市道路、乡村公路等与试验场具有场景互补性的环境,包含车辆、行人、骑行者等7类标注目标。

为了验证多模可重构加速器的性能,测试流程首先如下:分别启动三种不同模态的目标识别程序并开始计时,将试验场目标识别测试数据集

和KITTI公开数据集中的500张图像送入到多模可重构加速器和Jetson Nano中处理，当所有模态图像处理完成时结束计时。根据计时的时间、数据集中图片数、标识框的结果即可算出三种不同模态目标识别的响应时间和准确率。为保证实验结果具有可比性，本文对各项评价指标作如下定义：响应时间指单帧数据完成预处理并写入DRAM后，FPGA执行神经网络推理并将结果写回DRAM所需的平均时间，不包括传感器采集、CPU端预处理、点云体素化/BEV转换以及NMS后处理时间；mAP指各目标类别平均精度AP的均值，用于评价目标识别精度；功耗指模型连续推理过程中的板级平均功耗；FPS/W表示单位功耗吞吐率，其计算方式为FPS与平均功耗之比。

图10 实验平台

Fig. 10 Experimental platform

2.2.1 性能验证

为了客观评估本加速器的性能，我们在Nvidia Jetson Nano开发板上部署了相同的目标识别模型作为对比基准。可见光、红外、激光雷达目标识别测试结果如表1所示，可以看出：可见光、红外、激光雷达目标识别测试的平均响应为28.90、27.90、94.96 ms，与之相对应的Jetson Nano上的测试分别为140.00、142.00、522.28 ms，说明多模可重构加速器实现了多模目标识别加速。

表1 性能验证结果

Table 1 Performance verification results

模态	多模可重构加速器平均时间/ms	Jetson Nano平均时间/ms
可见光	28.90	140.00
红外	27.90	142.00
激光雷达	94.96	522.28

2.2.2 精度验证

精度验证结果如表2所示，可以看出“可见光、红外、激光雷达目标识别测试中各分类和平均准确率mAP均超过90%，说明多模可重构加速器在加速的同时保证了计算的准确性。

表2 精度验证结果

Table 2 Accuracy verification results

模态	车mAP	人mAP	无人机mAP	平均mAP
可见光	0.950	0.920	0.910	0.925 0
红外	0.960	0.910	0.900	0.924 1
激光雷达	0.915	0.892	0.905	0.906 8

综上所述，激光雷达点云数据的处理时间(94.96 ms)明显高于可见光与红外图像，这是由于3D点云数据存在无序性和稀疏性，在编译打包时的预处理开销较大。但得益于算子融合优化，其推理时间仍满足10 Hz以上的实时性要求。

2.2.3 加速器资源消耗分析

实验过程中对加速器的硬件资源占用情况如表3所示，其中块随机存取存储器(BRAM)与数字信号处理器(DSP)为主要资源消耗部件：BRAM的已使用数量为203.5个，占开发板BRAM总资源的65.22%；DSP的已使用数量达1 165个，在开发板DSP总资源中的占比为67.42%。查找表(LUT)利用率为23.11%、触发器(FF)利用率为18.65%、UltraRAM(URAM)利用率为16.67%、全局缓冲器(BUFG)利用率仅为1.10%，均处于较低水平。BRAM作为数据缓存核心部件，需为多模态数据的高速存取提供支撑，而DSP则是实现多粒度可变位宽运算、保障混合精度计算效率的关键硬件单元，二者的高利用率充分体现了其在加速器硬件架构中的核心作用，也验证了硬件设计对多模态数据处理需求的针对性适配。

表3 资源占用情况

Table 3 Resource utilization

类型	已使用数量	总资源数量	利用率/%
LUT	53 247	230 400	23.11
LUTRAM	6 100	101 760	5.99
FF	85 951	460 800	18.65
BRAM	203.5	312	65.22
URAM	16	96	16.67
DSP	1 165	1 728	67.42
BUFG	6	544	1.10

2.2.4 功耗和能效验证

功耗和能效验证如表4所示,在可见光、红外和激光雷达三种模态任务中,多模可重构加速器与Jetson Nano的功耗处于相近水平,但前者能效更高。具体来看,多模可重构加速器在可见光、红外、激光雷达任务中的功耗分别为8.3、8.1、9.2 W,对应能效分别为3.5、3.5、0.87 FPS/W;Jetson Nano的功耗分别为7.6、7.3、8.5 W,对应能效分别为1.1、1.0、0.22 FPS/W。结果表明,虽然多模可重构加速器功耗略高,但由于响应时间显著降低,其单位功耗下可完成更多目标识别任务,因此整体能效明显优于Jetson Nano,更适合资源受限条件下的端侧部署。

表4 功耗和能效验证

Table 4 Verification of power consumption and energy efficiency

模态	多模可重构加速器		Jetson Nano	
	功耗/W	能效/(FPS/W)	功耗/W	能效/(FPS/W)
可见光	8.3	3.50	7.6	1.10
红外	8.1	3.50	7.3	1.00
激光雷达	9.2	0.87	8.5	0.22

2.3 典型侦察场景分析

在无人机遂行“搜索与跟踪”任务过程中,端侧加速器不仅需维持高吞吐量,还必须应对航拍环境下极端复杂的视觉挑战。本文配套的无人机飞行监控界面如图11所示,该系统集成了自主研发的航线规划与飞行监控功能,能够实时显示无人机的坐标、姿态、速度及载荷传感器状态。当无人机在50~1 000 m高度巡航时,端侧加速器需对下传的实时图像进行毫秒级处理,以确保地面对目标的锁定与跟踪指令能够准确下达。

图11 无人机飞行监控与侦察任务界面

Fig. 11 UAV flight monitoring and reconnaissance mission interface

针对实际侦察任务中的算法模块设计,本文验证了加速器在处理复杂航拍图像时的鲁棒性。如图12所示,无人机端侧识别面临3大核心挑战。

1) 目标尺度跨度大:受飞行高度与俯冲动作影响,待检测目标的尺度变化剧烈,要求加速器能够高效支撑多尺度特征提取算子,如图12(a)所示。

2) 目标遮挡严重:在城市密林环境下,目标常被建筑物或植被遮挡,加速器需配合深度模型完成对残缺特征的快速检索与分类判断,如图12(b)所示。

3) 拍摄视角多变:无人机在执行转向或盘旋动作时,观察视角的变化会导致目标特征呈现显著差异,如图12(c)所示。

本文设计的多模可重构加速器通过支持可变位宽运算与算子融合技术,有效降低了在处理上述复杂场景模型时的搬运开销,确保了无人机在复杂动态环境下的识别准确率稳定在90%以上。

(a) 目标尺度存在较大变化

(b) 目标可能存在被遮挡的情况

(c) 不同拍摄视角下的目标在图像中呈现的特征可能不同

图12 航拍图像存在的复杂情况

Fig. 12 Complex situations in aerial images

3 结 论

1）在硬件架构方面，提出的多粒度可变位宽运算与剪枝协同设计，能在利用极低额外开销的情况下，完美兼容可见光、红外及雷达等异构数据的混合精度推理；

2）在软件编译方面，基于TVM定制的编译器通过乒乓机制改进和算子融合技术，有效解决了并行度低的问题，显著降低数据搬运开销；

3）在系统效能方面，加速器对三种模态的平均识别时间优于Jetson Nano基准平台4～5倍，不仅准确率保持在90%以上，更在整体能效上展现出显著优势，为复杂环境下的无人机侦察提供了高能效的端侧AI算力支撑；

参 考 文 献

[1] 方芳，彭玉婷. 2021年外军信息系统领域发展综述[J]. 中国电子科学研究院学报，2022，17(4)：311-316.
Fang Fang，Peng Yuting. A comprehensive analysis of the developments of foreign military information systems in 2021 [J]. Journal of China Academy of Electronics and Information Technology，2022，17(4)：311-316.(in Chinese)

[2] 庞召力. 外军重要目标夺控作战中无人作战力量运用探要[J]. 军事文摘，2024(8)：35-40.
Pang Zhaoli. On the application of unmanned combat forces in the capture and control of important foreign targets[J]. Military Digest，2024(8)：35-40.(in Chinese)

[3] 周宇，赵玲，杨航，等. 多信息智能传感器融合技术在联合作战中的应用[J]. 国防科技，2024，45(1)：22-29.
Zhou Yu，Zhao Ling，Yang Hang，et al. Application of multi-information intelligent sensor fusion technology in joint operations [J]. National Defense Technology，2024，45(1)：22-29.(in Chinese)

[4] 杨超明. 卷积神经网络硬件加速器及其量化算法研究[D]. 广州：广东工业大学，2025.
Yang Chaoming. Research on hardware accelerators and quantization algorithms for convolutional neural networks [D]. Guangzhou：Guangdong University of Technology，2025.(in Chinese)

[5] 李霞，孙濛，杨铖. 我国算力产业的链式架构及其效率评估[J]. 社会科学动态，2025(6)：14-29.
Li Xia，Sun Meng，Yang Cheng. A chained architecture of China's computing power industry and its efficiency evaluation[J]. Dynamics of Social Sciences，2025(6)：14-29.(in Chinese)

[6] 申剑蓉. 人工智能技术在计算机网络安全中的应用[J]. 电子技术，2025，54(6)：350-351.
Shen Jianrong. Application of artificial intelligence technology in computer network security[J]. Electronic Technology，2025，54(6)：350-351.(in Chinese)

[7] 路唯佳，贺仁龙. 新一代智算网络技术及其发展趋势研究[J]. 上海信息化，2025(6)：12-17.
Lu Weijia，He Renlong. Research on new generation intelligent computing network technology and its development trend [J]. Shanghai Informatization，2025(6)：12-17.(in Chinese)

[8] 罗晓羽. 现场可编程门阵列标准体系现状及建设思路探析[J]. 信息技术与标准化，2025(8)：63-66.
Luo Xiaoyu. Analysis of the current situation and construction idea of standard system for field-programmable gate array[J]. Information Technology & Standardization，2025(8)：63-66.(in Chinese)

[9] 唐东洋，田科源，韩庆，等. 微小型无人机室内目标自主搜索方案研究[J]. 航空工程进展，2025，16(5)：91-102.
Tang Dongyang，Tian Keyuan，Han Qing，et al. Research on autonomous indoor target search solution for MAVs[J]. Advances in Aeronautical Science and Engineering，2025，16(5)：91-102.(in Chinese)

[10] 石添介，刘飞阳，张晓. 机载超轻量化卷积神经网络加速器设计[J]. 航空工程进展，2024，15(2)：188-194.
Shi Tianjie，Liu Feiyang，Zhang Xiao. Design of airborne ultra-lightweight convolutional neural network accelerator [J]. Advances in Aeronautical Science and Engineering，2024，15(2)：188-194.(in Chinese)

[11] 高远，闫妍，周磊，等. 基于卷积神经网络和多项式混沌方法的翼型鲁棒性优化[J]. 航空工程进展，2021，12(2)：80-87.
Gao Yuan，Yan Yan，Zhou Lei，et al. Airfoil robust optimization based on convolutional neural network and polynomial chaos method[J]. Advances in Aeronautical Science and Engineering，2021，12(2)：80-87.(in Chinese)

[12] Lee J，Kim Y，Lee J，et al. Stitch-x：a sparse CNN accelerator with flexible weight stitching for FPGAs[C]// 2020 IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). Piscataway，NJ，USA：IEEE，2020：1-9.

[13] Jouppi N P，Young C，Patil N，et al. In-datacenter performance analysis of a tensor processing unit[C]// Proceedings of the 44th Annual International Symposium on Computer Architecture. Toronto，ON，Canada. ACM，2017：1-12.

[14] Aimar M，Rusci M，Conti F，et al. NullHop：a low-power CNN accelerator exploiting feature map sparsity[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems，2021，29(3)：511-524.

[15] Hu X，Zhang Y，Liu S，et al. A resource-efficient FPGA accelerator for convolutional neural networks with quad-level PE array[J]. IEEE Access，2020，8：156243-156254.

[16] Zhang C，Li P，Sun G，et al. DNN builder：an automated tool for building high-performance DNN accelerators on FPGAs[C]// 2019 IEEE 27th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). Piscataway，NJ，USA：IEEE，2019：1-9.

[17] Chen T，Moreau T，Jiang Z，et al. TVM：an automated end-to-end optimizing compiler for deep learning[C]// 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18). Berkeley，CA，USA：USENIX Association，2018：578-594.

[18] Geiger A，Lenz P，Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite[C]// 2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence，RI，USA. IEEE，2012：3354-3361.

（编辑：丛艳娟）