

引用格式:彭鑫磊,余乐. 基于高层次综合的卷积神经网络设计与优化方法研究[J]. 微电子学与计算机,2019,36(8):63-67.
[PENG X L, YU L. HLS-based design and optimization methodology for convolutional neural network [J]. Microelectronics & Computer,2019,36(8): 63-67.]

基于高层次综合的卷积神经网络设计与优化方法研究

彭鑫磊¹,余 乐^{1,2}

(1 北京工商大学 计算机与信息工程学院,北京 100048;

2 食品安全大数据技术北京市重点实验室,北京 100048)

摘 要: 本文基于 FPGA 高层次综合的设计方法学,在 ZYNQ-7020 上实现了一个卷积神经网络加速器.采用循环展开和并行流水的设计方法对卷积核运算进行优化,均衡了所占用逻辑资源及运算效率,从而实现加速器的最优性能.通过 MINST 数据集在 100MHz 的工作频率下对加速器进行性能测试,结果表明:对单张图片,该加速器相对于通用平台 ARM A9 可实现 3.77 倍加速,而对 1000 张图片的流式处理可实现高达 6.14 倍加速.

关键词: 卷积神经网络;嵌入式;ZYNQ-7020;加速器

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2019)08-0063-05

HLS-based design and optimization methodology for convolutional neural network

PENG Xin-Lei¹, YU Le^{1,2}

(1 School of Computer and Information Engineering, Beijing Technology and Business University, Beijing 100048, China;2 Beijing Key Laboratory of Big Data Technology for Food Safety, Beijing 100048, China)

Abstract: Based on High Level Synthesis (HLS) design methodology of FPGA, this paper implements a convolutional neural network accelerator on ZYNQ-7020. The design method of cyclic unroll and pipelining is used to optimize the convolution kernel operation, and the occupied logic resources and operation efficiency are balanced to achieve the optimal performance of the accelerator. The performance of the accelerator is tested by the MINST dataset at 100MHz working frequency. The results show that: the accelerator can achieve 3.77 times acceleration compared to the general platform ARM A9 for a single picture, and the streaming processing of thousands of pictures can achieve up to 6.14 times acceleration.

Key words: convolutional neural networks;embedded;ZYNQ-7020;accelerator

1 引言

卷积神经网络由于自身网络结构的特点^[1],运算过程需要大量的计算资源. ARM 处理器计算资源有限无法满足实时性要求,专用处理器 GPU、ASIC 和 FPGA 为 CNN 提供了高效可行的加速平台,其中, GPU 能耗大的问题一直难以解决, ASIC

流片成本昂贵而且不可重配置使得它难以大规模应用, FPGA 具有高性能、低功耗、可重配置的特点,已有的基于 FPGA 的 CNN 加速方案^[2]表明, FPGA 比 CPU 和 GPU 不仅具有更好的加速效果,而且能耗方面也远远低于 GPU. 但由于 FPGA 灵活性差而且 Verilog HDL 开发门槛较高,开发周期长,一定程度上影响了 FPGA 在人工智能领域应用的

普及.

本文研究基于高层次综合的卷积神经网络设计与优化方法,在 ZYNQ-7020 上实现了一个卷积神经网络加速器.采用循环展开和并行流水的设计方法对卷积核运算进行优化.

2 ZYNQ 和卷积神经网络

Xilinx 推出的 ZYNQ 芯片采用“ARM+FP-GA”体系架构^[3],其中 ARM 部分也称处理系统(Processing System, PS),FPGA 部分称做可编程逻辑(Programmable Logic, PL). PS 部分包含双核 Cortex-A9 处理器,PL 部分包含了传统的 FPGA 逻辑单元. ZYNQ-7000 平台将 ARM 的软件可编程能力和 FPGA 的硬件可编程能力结合在一起,使系统拥有灵活性、资源丰富和强大的计算能力的同时又具备低功耗和低成本的优点. 这些优点使得 ZYNQ 成为卷积神经网络理想的嵌入式运行平台.

卷积神经网络是一种前馈神经网络,它的人工神经元可以响应一部分覆盖范围内的周围单元,是一种高效识别图像的方法,对于图像处理有出色表现. 它通常包括输入层、卷积层、下采样层和全连接层^[4-6]构成.

本文采用 CNN LeNet5 模型对系统进行测试,模型结构如图 1 所示,一个输入层,两个卷积层,两个降采样层,一个输出层. 采用 MINST 数据集对模型进行训练,每张图片的大小为 28×28 像素,共 70 000 张图片,其中 60 000 张做训练集,10 000 张做测试集.

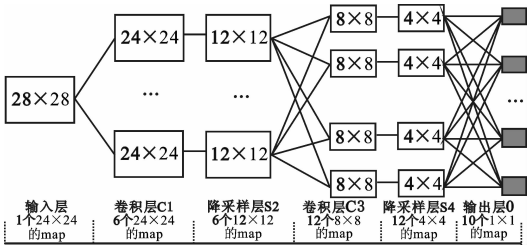


图 1 CNN LeNet5 模型结构

上述模型中,各层卷积核、特征图以及偏置量需要变量个数如表 1 所示,整个网络共需要 9236 个变量,除了池化层的卷积核权重不需要占用存储空间外,其余 9 188 个变量都需要消耗存储资源,若采用 double(4 个 byte)存储这些变量,将需要大约 72KB 存储资源,本文采用的 ZYNQ-7020 有足够的存储空间用于存放这些变量.

表 1 各层卷积核、特征图、偏置量个数和所需变量个数

层	权重 个数	权重 变量	特征图 大小	特征图 个数	特征图 变量	偏置量 个数	偏置量 变量	层变 量数
C1	$5 \times 5 \times 6$	150	24×24	6	3 456	6	6	3 612
S1	$2 \times 2 \times 6$	24	12×12	6	864	0	0	888
C2	$5 \times 5 \times 12 \times 6$	1 800	8×8	12	768	12	12	2 580
S2	$2 \times 2 \times 6$	24	4×4	12	192	0	0	216
O	$16 \times 10 \times 12$	1 920	1×1	10	10	10	10	1 940

3 系统设计与实现

本文设计并实现基于 ZYNQ 的 CNN 加速器,采用 ZYNQ 的 PS 部分做逻辑控制,PL 部分的 CNN IP 核执行卷积神经网络运算,DMA 作为 PS 和 CNN IP 核的数据传输通道,采用 AXI Stream 作为数据传输方式,采用 Counter 记录程序运行所经历的时钟周期数. 系统硬件原理图如图 2 所示,PS Reset 在 PS 的命令下复位各个模块,PS 通过 AXI Interconnet_1 对 DMA 进行配置和与 Counter 进行通信,通过 AXI Interconnet_2 与 CNN IP 核进行数据传输,PS 将手写体图片的像素通过 DMA 送入 CNN IP 核中进行计算.CNN IP 核再将计算的结果

通过 DMA 返回给 PS,从而完成单张图片的 CNN 计算过程.

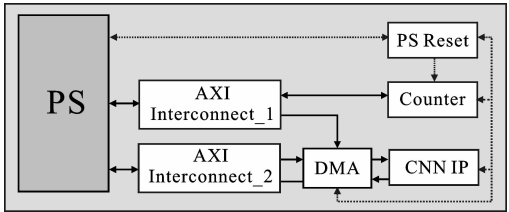


图 2 系统硬件原理图

3.1 CNN LeNet5 模型训练

本文在 Matlab 中搭建 CNN LeNet5 网络模型的并进行训练,训练过程如图 3 所示,其中 X 轴代表训练次数,Y 轴表示每次训练的误差. 设置学习速

率为 1,训练 12000 次,随着训练次数的不断增加,误差越来越小,最终的模型错误率为 2.57%。

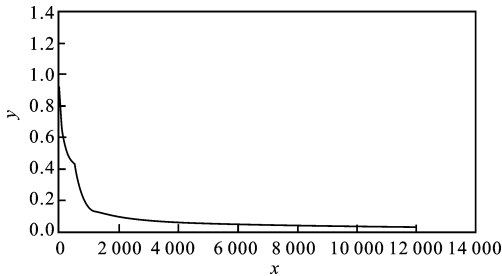


图 3 CNN LeNet5 误差与训练次数的关系

3.2 CNN IP 核设计与实现

Xilinx 推出高层次综合 (High Level Synthesis, HLS) 工具 (Vivado HLS) 可以把行为描述转换成电路结构描述,反映在设计语言上可以表述为把 C、C++、System C 等高级语言转化成硬件描述语言^[8],大大提高了 FPGA 的开发效率。

本文采用 C++ 重新搭建已经训练好的 CNN 的前向过程,通过 HLS 的方式综合生成 CNNIP 核,系统采用 AXI Stream 通信方式实现外部与 CNN IP 核的数据传输,IP 核的输入是图片的像素 (28×28),输出是图片像素所代表的数字,图 4 是 CNN IP 原理图。

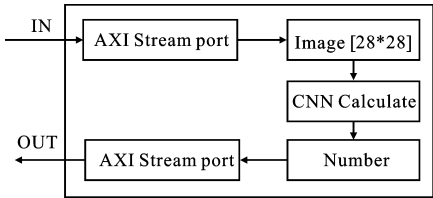


图 4 CNN IP 原理图

3.3 CNN IP 核优化

CNN 的卷积运算由多层 for 循环嵌套组成,以输入层到卷积层 C1 的卷积运算为例,C1 层卷积代码如图 5 所示,本文设定最内层为第一层,依次往外层数递增,说明不同的循环处理方式与所生成 IP 核性能的关系,下面将详细介绍优化的具体过程。

```
for(j=0;j<24;j++) //输出特征图的宽:24 (第四层)
for(k=0;k<24;k++) //输出特征图的高:24 (第三层)
for(l=0;l<24;l++) //卷积核宽:5 (第二层)
for(m=0;m<24;m++) //卷积核高:5 (第一层)
c1_map[i][j*24+k] += c1_kernel[i][l*5+m]*input_double[j*28+k+1*28+m]
```

图 5 C1 层卷积代码

在 FPGA 开发中,通常采用流水线 (Pipelining) 或者展开 (Unroll) 的方法来提高循环的执行速度。

相对于默认执行方式,Pipelining 可以在增加少量逻辑资源的情况下,降低 Interval 增强处理能力,Unroll 是把循环完全展开,意味着多个单元同时并行计算,所以资源消耗比 Pipelining 更大,当然处理能力也更强。

平衡 CNN IP 核资源消耗和计算性能最优的方案是对最内层循环做流水线处理,当嵌套循环的某层被流水线操作时,HLS 对这层循环下的子循环都将执行展开操作。只对最内层循环做流水线处理,将会产生最小的资源消耗,可以为大多数应用程序提供一般可接受的吞吐量。流水线处理越外层的循环,将会消耗越多的计算和存储资源,但这样也会在吞吐量和延时上带来越高的性能设计^[4]。

本文将卷积循环从最内层向最外层依次展开,通过 HLS 生成的综合报告获取当前处理方式下的 CNN IP 各方面性能,不同展开方式综合后对应的计算速度与资源消耗如表 2 所示,本文采用 ZYNQ-7020 处理器,其中 DSP48E 表示 PL 的数字信号处理逻辑单元,共 220 个,LUT 代表 PL 中的一种用于实现组合逻辑和时序逻辑的存储资源,共 53 200 个。由表 2 可得,当流水线处理第三层循环时,综合的 IP 具有最低的计算延时时间 (约 4 ms),同时 LUT 几乎被完全消耗。

表 2 不同 Pipelining 生成的 CNN IP 核计算速度与资源消耗

Pipelining	Latency	DSP48E	LUT
默认	3 288 021	149(67%)	36 279(68%)
第一层	1 537 170	150(68%)	37 114(69%)
第二层	1 201 467	164(70%)	39 693(74%)
第三层	415 330	204(92%)	53 207(100%)

表 3 不同方式计算图片所需要的时间(周期:10ns)

运行方式	优化情况	消耗的时钟周期数	实际消耗时间(ms)
PS	默认	3 314 942	33
CNN IP	默认	3 073 518	30
CNN IP	流水线第一层	1 597 955	16
CNN IP	流水线第二层	1 262 038	13
CNN IP	流水线第三层	911 366	9

4 实验与结果

4.1 实验测试平台

针对嵌入式处理器 ARM 计算性能和基于软件的算法执行方式难以满足卷积神经网络计算需求的缺点,本文第二章提出采用可编程逻辑器件设计出

CNN IP 并优化,旨在提高平台运算卷积神经网络的效率.

实验平台采用 Zedboard 开发板(主芯片是 XCZ7020),其 PS 部分代表基于 ARM 的通用处理器,时钟频率为 667 MHz,PL 部分的时钟频率为 100 MHz,分别进行单张和多张并且记录计算时间来测试加速效果.

4.2 单张图片测试

本实验采用 PS 和 CNN IP 核分别对同一张手写体数字图片(28×28)进行 CNN 运算,并记录各自识别图片所需要的时间.实验中 ZYNQ-7020 的 PS 部分的时钟频率为 667 MHz,CNNIP 核的时钟频率为 100 MHz,采用 Counter(每个周期自动加一)来记录时间,实验得到两种方式所需的计算时间如表 3 所示.

由表 3 可得,与时钟频率为 667 MHz 的 PS 执行 CNN 运算相比,时钟频率为 100 MHz 的 CNN IP 计算速度比其快 3 ms;结合表 2,当对各个卷积层、池化层的最内层循环做流水线处理的时,DSP48 和 LUT 的消耗量仅需要增加 1%,就可将 CNN 运算速度提高将近 1 倍;在 ZYNQ-7020 中,当 CNN IP 对 PL 部分资源充分利用时,可将 CNN 的运算速度提高 3 倍以上.

4.3 图片流测试

在上述实验的基础上,设计将 MINIST 数据集中多张图片按照图片流形式进行测试,即从数据集中直接读取多张图片分为 10 组,第一组 100 张,后继每组依次累加 100 张,将每组图片分别放在

Linux 应用层和硬件层的 CNN IP(流水线第三层)中做 CNN 运算,并且记录各自的计算时间,从而测试 CNN IP 对图片流的加速效果.实验结果如图 6 所示.

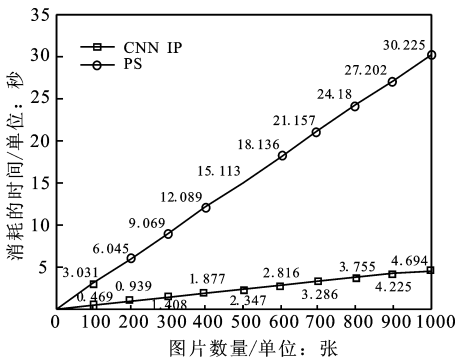


图 6 图片流测试结果

两种平台运算的图片数量与消耗的时间均成线性关系($R^2=0.999\ 7$),表明每张图片的运算时间保持不变,其中 CNN IP 处理单张图片的平均时间约为 4.7 ms.当运算相同数量图片时,相较于 PS,CNN IP 可将运算速度提高 6.4 倍左右,远大于单张图片的加速效果.当处理 1 000 张图片时,CNN IP 比 PS 快了 25s 左右,并且随着图片数量的继续增多,CNN IP 的加速效果将更加显著.

4.4 实验结果比较

本文介绍了以高层次综合实现的 CNN IP 与采用 FPGA 实现 LeNet5 网络^[5]相比较,比较结果显示在表 4 和表 5 中.

表 4 文献[5]与本文实验平台对比

文献[5]		本文	
FPGA	—	virtex-5 XC5VLX110T	Zynq XC7Z020
通用 CPU	Core i5 2500K	—	Cortex-A9
实现方式	—	Verilog HDL	—
时钟频率	3.3GHz	75MHz	667MHz
			100MHz

表 5 文献[5]与本文在不同实验平台下对不同张图像处理的运算速度比较

图像数量	文献[5]		本文	
	CPU	FPGA	CPU	FPGA
	耗时	加速比	耗时	加速比
1 张	1.13ms	0.272ms	4.17	33.149ms
1 000 张	—	—	—	30.225s
10 000 张	12.103s	3.522s	3.43	—

由表 5 可以看出,文献[5]中提出的采用 FPGA 实现的卷积神经网络加速器与传统 CPU 相比有较大的加速效果,处理单张图像的加速比为4.17,处理 10 000张图像的加速比为 3.43.本文提出的采用基 ZYNQ 构造的 CNN IP 是 ZynqXC7Z020,采用高层次综合方式,时钟频率为 100 MHz,在处理单张图像所耗时间为 9 ms,与通用 CPU 处理单张图像所耗时 33.149 ms 相比具有明显的加速效果,加速比为 3.67.当处理 1 000 张图像时,CNN IP 耗时 4.69

s,通用 CPU 耗时 30.225 s,此时的加速比为 6.44,而且处理的图像越多,加速效果越明显,具有一定的意义.

5 结束语

本系统在 ZYNQ-7020 处理器上采用 HLS 的方式设计 CNN IP 用于执行卷积神经网络算,并采用流水线、展开的技术对 CNN IP 核进行优化,旨在提高嵌入式处理器计算 CNN 的速度.通过实验证明,本方案对嵌入式处理器针对 CNN 的计算速度有显著的提升.在 ZYNQ-7020 中当对第三层循环进行流水线处理时,就几乎消耗了全部 PL 的存储资源,若采用资源更多 ZYNQ-7030、ZYNQ-7040 处理器可将更高层的循环进行流水线处理,从而可以将 CNN IP 的计算速度进一步提升.

参考文献:

[1] ZHANG C, LI P, SUN G, et al. Optimizing fpga-based accelerator design for deep convolutional neural networks[C]// Acm/sigda International Symposium on Field-programmable Gate Arrays. ACM, 2015.

[2] LI H, FAN X, JIAO L, et al. A high performance

FPGA-based accelerator for large-scale convolutional neural networks[C]// 2016 26th International Conference on Field Programmable Logic and Applications (FPL). IEEE, 2016.

[3] Xilinx. ds187-XC7Z010-XC7Z020-Data-Sheet [EB/OL]. (2018-07-02)[2018-07-02]<https://china.xilinx.com/support/>[8]范秋香. FPGA 高级开发技术的研究[D]. 北京:北京理工大学,2015.

[4] Xilinx. UG902-vivado-high-level-synthesis[EB/OL]. https://china.xilinx.com/support/documentation/sw_manuals/xilinx2015_4/ug902-vivado-high-level-synthesis.pdf.

[5] 余子健, 马德, 严晓浪, 等. 基于 FPGA 的卷积神经网络加速器[J]. 计算机工程, 2017, 43(1):109-114.

[6] 陈岩, 李洋洋, 余乐, 等. 基于卷积神经网络的手写体数字识别系统[J]. 微电子学与计算机, 2018, 35(2): 71-74.

作者简介:

彭鑫磊 男,(1993-),硕士研究生.研究方向为数字集成电路设计、嵌入式开发.

余乐(通讯作者) 男,(1983-),博士,讲师.研究方向为计算视觉与类脑芯片. E-mail:ladd_u@163.com

(上接第 62 页)

[2] DABOV K, FOI A, KATKOVNIK V, et al. Image denoising by sparse 3-D transform-domain collaborative filtering [J]. IEEE Transactions on Image Processing, 2007, 16(8):2080-2095.

[3] ZHANG K, ZUO W, CHEN Y, et al. Beyond a Gaussian denoiser: Residual learning of Deep CNN for image denoising [J]. IEEE Transactions on Image Processing, 2017, 26(7): 3142- 3155.

[4] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [C]. Proceedings - 29th IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE Computer Society, 2016:770-778.

[5] PORTILLA J, STRELA V, WAINWRIGHT M J, et al. Image denoising using scale mixtures of Gaussians in the wavelet domain [J]. IEEE Transactions on Image Processing, 2003, 12(11): 1338- 1351.

[6] IOFFE S, SZEGEDY C. Batch normalization: accelerating deep network training by reducing internal covariate shift [C]. 32nd International Conference on Machine Learning. Lile: International Machine Learning Society, 2015: 448-456.

[7] KINGMA D P, BA J. Adam: A Method for Stochastic Optimization [C]. 3rd International Conference on Learning Representations. San Diego, 2015; arXiv: 1412.6980.

作者简介:

魏江 男,(1968-),硕士,副教授.研究方向为智能信息处理、嵌入式系统应用. E-mail:joeway@nwpu.edu.cn.

刘潇 女,(1996-),硕士研究生.研究方向为智能信息处理、人工智能与机器学习.

梅少辉 男,(1984-),博士,副教授.研究方向为智能信息处理、人工智能与机器学习.