# 深度神经网络 FPGA 设计进展、实现与展望

焦李成 孙其功 杨育婷 冯雨歆 李秀芳

（西安电子科技大学智能感知与图像理解教育部重点实验室 西安 710071）

（智能感知与计算国际联合研究中心 西安 710071）

（智能感知与计算国际合作联合实验室 西安 710071）

**摘 要** 近年来，随着人工智能与大数据技术的发展，深度神经网络在语音识别、自然语言处理、图像理解、视频分析等应用领域取得了突破性进展. 深度神经网络的模型层数多、参数量大且计算复杂，对硬件的计算能力、内存带宽及数据存储等有较高的要求. FPGA 作为一种可编程逻辑器件，具有可编程、高性能、低能耗、高稳定、可并行和安全性的特点. FPGA 与深度神经网络的结合成为推动人工智能产业应用的研究热点. 本文首先简述了人工神经网络坎坷的七十年发展历程与目前主流的深度神经网络模型，并介绍了支持深度神经网络发展与应用的主流硬件；接下来，在介绍 FPGA 的发展历程、开发方式、开发流程及型号选取的基础上，从六个方向分析了 FPGA 与深度神经网络结合的产业应用研究热点；然后，基于 FPGA 的硬件结构与深度神经网络的模型特点，总结了基于 FPGA 的深度神经网络的设计思路、优化方向和学习策略；接下来，归纳了 FPGA 型号选择以及相关研究的评价指标与度量分析原则；最后，我们总结了影响 FPGA 应用于深度神经网络的五个主要因素并进行了概要分析.

**关键词** 深度神经网络；FPGA；产业应用；硬件结构；设计思路；度量分析

**中图法分类号** TP18 **DOI 号** 10.11897/SP.J.1016.2022.00441

# Development, Implementation and Prospect of FPGA-Based Deep Neural Networks

JIAO Li-Cheng SUN Qi-Gong YANG Yu-Ting FENG Yu-Xin LI Xiu-Fang

(*Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xi'an* 710071)

(*International Research Center for Intelligent Perception and Computation, Xi'an* 710071)

(*Joint International Research Laboratory of Intelligent Perception and Computation, Xidian University, Xi'an* 710071)

**Abstract** In recent years, with the development of artificial intelligence and big data technology, the deep neural network has made a breakthrough in many fields, such as speech recognition, natural language processing, image understanding, video analysis and so on. However, along with the increasing of neural network layers, a large number of parameters and complex calculations aggravate the requirements of hardware in computing power, memory bandwidth and data storage. FPGA, a programmable logic device, is of programmability, high performance, low energy consumption, high stability, parallelizability and security. The combination of FPGA and the deep neural network becomes a research hotspot to promote the industrial application of artificial intelligence. This paper introduces the development of the deep neural network in the past 70 years, the mainstream deep learning model and the fundamental hardware that

收稿日期：2020-5-18；在线发布日期：2021-01-11. 本课题得到国家自然科学基金重点项目（61836009）、国家自然科学基金创新研究群体科学基金（61621005）、国家自然科学基金（U1701267，61871310，61977052）、高等学校学科创新引智计划（111计划）（B07048）、重大研究计划（91438201）、陕西省2021年重点研发计划（2021ZDLGY02-08），西安市科技产业化计划"人工智能"产业创新链推进工程（XA2020-RGZNTJ-0097）以及教育部"长江学者和创新团队发展计划"（IRT_15R53）资助. **焦李成**，博士，教授，中国计算机学会（CCF）会员，主要研究领域为智能感知、图像理解等. Email: lchjiao@mail.xidian.edu.cn. **孙其功**，博士研究生，主要研究领域为机器学习、计算机视觉与并行计算. **杨育婷**，博士研究生，主要研究领域为机器学习、计算机视觉与模式识别. **冯雨歆**，硕士研究生，主要研究领域为机器学习与计算机视觉. **李秀芳**，博士研究生，主要研究领域为机器学习与计算机视觉.

support the development and application of deep neural network. Secondly, the research hotspots of FPGA combined with the deep neural network on the industrial applications are analyzed in six respects on the basis of introducing the development process, development mode, development process and type specification of FPGA. And then, the design idea, optimization direction and learning strategy of deep neural network based on FPGA are summarized according to the hardware structure of FPGA and the model characteristics of deep neural network. In addition, the model selection of FPGA together with the evaluation index and measurement analysis principle of related works are listed. Finally, we summarized the five main factors that affect the application of FPGA to deep neural networks and conducted a summary analysis.

**Keywords** deep neural network; FPGA; industrial application; hardware structure; design idea; measurement analysis

# 1 引 言

随着智能化时代的到来，人工智能的应用已经深入到社会的各行各业. 作为人工智能的主要研究分支，神经网络的研究和发展成为主导当前智能化程度的主要力量. 简单来讲，神经网络是通过模拟人脑中神经元的连接方式来实现类脑的信息处理过程. 在过去的七十年发展历史中，神经网络的发展也经历了质疑和低谷，得幸于研究者的坚持探索才使它被普遍认可并有机会更好的造福人类. 为让机器更好地模拟人脑来认识世界，神经网络模型不断革新发展，经历了从浅层神经网络到深度神经网络的重要变革. 目前，深度神经网络可以利用深层的结构很好地提取和拟合数据特征，并在语音识别、自然语言处理、图像理解、视频分析等应用领域取得了突破性进展. 研究者在追求更好精度的同时，深度神经网络模型层数和参数数量也在不断增加，从而对硬件的计算能力、内存带宽及数据存储等的要求也越来越高. 因此，计算能力强、可并行加速、数据吞吐高的高性能硬件平台对于模型训练和产业应用来说显得尤为重要. 本节将概述神经网络的发展史和当前流行的深度神经网络模型，并分析推动深度神经网络产业应用的主流硬件平台.

## 1.1 深度神经网络的发展历程

相比今天神经网络的发展速度，其基础理论研究初期却经历重重波折. 最早的神经网络数学模型是由心理学家 McCulloch 教授和数学家 Pitts 教授于 1943 年提出的模拟人类大脑神经元的 McCulloch-Pitt 神经元模型，并被称为 M-P 模型[1]. 该模型是通过简单的线性加权来实现对人类神经元处理信号的模拟，该工作被称为人工神经网络（ANN）的起点，随后出现的神经网络模型均是以该模型为基础的. 然而，其性能的好坏完全由分配的权重决定，这就使该模型很难达到最优的效果. 随后，为了改善该模型并让计算机自动合理的设置权重，心理学家 Hebb 于 1949 年提出 Hebb 学习规则[2]并得到诺贝尔医学奖得主 Kandel 的认可. 康奈尔大学的实验心理学家 Rosenblatt 于 1958 年提出感知机模型[3]，该模型是第一个真正意义上的人工神经网络，标志着神经网络研究进入了第一次高潮期. Minsky 和 Papert 等学者对感知机模型进行了分析，结论为该模型无法求解简单的异或等线性不可分问题[4]，从此神经网络的发展进入低潮甚至几乎处于停滞状态. 随后，并行分布处理[5]、反向传播算法[6]及 1982 年连续和离散 Hopfield 神经网络模型的提出为研究者重新打开了思路，开启了神经网络发展的又一个春天，此后的神经网络模型研究开始向问题导向发展.

1985 年 Sejnowski 和 Hinton 受 Hopfield 神经网络模型的启示提出的玻尔兹曼机模型[7]. 该模型通过学习数据的固有内在表示来解决困难学习的问题，随后又针对模型局限性进一步提出受限玻尔兹曼机模型[8]和深度玻尔兹曼机模型[9]. 反向传播算法于 1986 年得到进一步发展，成为后续神经网络模型发展的基石[10]，1990 年用于解决数据结构关系的递归神经网络出现[11]. 经过半个世纪的研究，加拿大多伦多大学的教授 Hinton 等人在 2006 年提出了深度置信网络模型[12]，不但提出了多隐层的神经网络，而且提出了深层神经网络在训练问题上的解决方法，该模型开启了深度神经网络的研究热潮. 自此，针对特定研究问题的深度神经网络模型大量涌现.

深度卷积神经网络是一种受启发于人类大脑对

眼睛里接收到的信号的理解过程而提出的神经网络模型. 该网络作为人工神经网络的典型模型之一被提出并出色地应用于计算机视觉领域. LeCun 等人提出的 LeNet 模型作为卷积神经网络的雏形起初被应用于手写体识别. 2012 年 Hinton 等人提出 AlexNet 模型,并应用 ImageNet 图像识别大赛中[13],其精确度颠覆了图像识别领域，使卷积神经网络进入大众视野. 随后出现了大量经典卷积神经网络模型如在网络层次上进行加深的 NIN[14], GoogLeNet[15], VGGNet[16]等，通过拆分卷积核来提升效率的 Inception V2/V3[17]，在深层网络中引入连跳结构来缓解梯度消失的 ResNet[18]和 DenseNet[19]等. 除此之外，还有建模特征通道间相互依赖关系的 SENet[20]、基于 ResNet 进行改进的 ResNext[21]及 ResNeSt[22]等. 在不同的研究领域也出现大量经典的卷积神经网络模型，如致力于全景分割的 UPSNet[23]、FPSNet[24]和 OANet[25]等，致力于目标检测的 Faster-RCNN[26]、YOLO v1/v2/v3[27–29]、SSD[30]、EfficientDet[31]、LRF-Net[32]等，致力于目标跟踪的 SimeseNet[33]、MDNet[34]. 目前，随着社会的不断进步，卷积神经网络的各种变型模型已经被应用于无人驾驶、智能监控和机器人等领域. 胶囊网络是 Hinton 团队于 2017 年为弥补卷积神经网络在物体空间关系上认知的不足而提出的一种新的网络体系结构. 其与卷积神经网络的区别在于，该网络是一种由含有一小群神经元的胶囊组成新型的神经网络[35], 这些胶囊之间通过动态路由来传递特征. 胶囊网络独特的数据表示方式使其考虑了目标的位置、方向、形变等特征，并能对提取的特征进行理解. 随后，为提升胶囊网络的性能，对胶囊进行优化[36–38]和对动态路由进行优化[39, 40]的方法被提出. 目前，胶囊网络的成就主要有抵御对抗性攻击、结合图卷积神经网络进行图像分类、结合注意力机制进行零样本意图识别等.

深度强化学习是一种集感知能力和决策能力为一体的神经网络模型，其应用成果真正进入大众视野是在 Alpha Go 出现后. Google DeepMind 公司提出的深度强化学习模型 Deep Q-Network[41]让这一更接近人类思维方式的模型得到更多学者的青睐. 随后，针对 Deep Q-Network 计算方法、网络结构和数据结构进行改进出现了 Double DQN、Dueling Network 和 Prioritized Replay 三种强化学习模型. 另外，Deep Q-Network 加入了递归思想生成了 Deep Recurrent Q-Network. 田春伟等人将强化学习思想用于目标跟踪领域并提出了 ADNet 模型[42]. 除此之外，继 Alpha Go 之后，DeepMind 又推出基于强化学习的 AlphaZero[43]和 MuZero[44]，提高了深度神经网络的智能化水平. 生成对抗网络（GAN）[45]是由 Goodfellow 等人在 2014 年提出的,是采用博弈对抗理论的一种新型神经网络模型. 该模型打破了已存在的神经网络对标签的依赖性，一出现就受到业界的欢迎并衍生出许多广泛流行的构架模型,主要有：第一次将 GAN 和卷积神经网络相结合的 DCGAN 模型[46]、利用 GAN 刷新人脸生成任务的 StyleGAN 模型[47]、探索文本和图像合成的 StackGAN 模型[48]、进行图像风格转化的 CycleGAN[49]、Pix2Pix[50]和 StyleGAN[47]，首次可生成具有高保真度低品种差距图像的 BigGAN[51]，用于解决视频跟踪问题中样本不均衡问题的 VITAL 网络模型[52]. 图神经网络是针对图结构数据发展而来的一种神经网络模型，该模型可以对可转化为图结构的数据之间的关系进行处理分析，它克服了已有的神经网络模型在处理不规则数据时的不足. 图神经网络模型最早起源于 2005 年[53]，随后由 Franco 博士在 2009 年首次定义了该模型的理论基础[54]，提出之初，该模型并没有引起很大波澜,直到 2013 年图神经网络才得到广泛关注. 近年来图神经网络得到广泛应用，同时结合已有网络模型. 图神经网络的不同拓展模型被不断提出，如图卷积网络（Graph Convolutional Networks）[55]、图注意力网络（Graph Attention Networks）[56]、图自编码器（Graph Auto-encoder）[57]、图时空网络（Graph Spatial-Temporal Networks）[58]、图强化学习[59–61]、图对抗网络模型[62, 63]等. 目前，图神经网络模型应用比较广泛，不仅被应用于计算机视觉、推荐系统、社交网络、智能交通等领域，还被应用于物理、化学、生物和知识图谱等领域.

轻量级神经网络是在保证模型的精度下对神经网络结构进行压缩、量化、剪枝、低秩分解、教师-学生网络、轻量化设计后的小体积网络模型. 2015 年之前，随着神经网络模型性能的不断提升，不断增大的网络体积和复杂度对计算资源也有较高的需求，这就限制了当前高性能的网络模型在移动设备上的灵活应用. 为了解决这一问题，在保证精确度的基础上，一些轻量级网络应运而生. 从 2016 年开始，SqueezeNet[64]、ShuffleNet[65]、NasNet[66]以及 MobileNet[67]、MobileNetV2[68]、MobileNetV3[69]等轻量级网络模型相继出现，这些轻量级网络的出现使一些嵌入式设备和移动终端运行神经网络成为可能，也使神经网络得到更广泛的应用.

自动机器学习（Automatic Machine Learning，

AutoML）是针对机器学习领域对机器学习从业者和所需经费的需求不断增长而提出的一种真正意义上的自动化机器学习系统. AutoML 代替人工进行自动的网络模型选取、目标特征选择、网络参数优化和模型评价. 也就是说，AutoML 可以自动构建具有有限计算预算的机器学习模型结构. AutoML 通过 2017 年 5 月的 Google I/O 大会进入业界视野并得到广泛关注. 随着神经网络深度和模型数量的不断增加，大部分的 AutoML 研究将重点关注在了神经网络搜索算法（Neural Architecture Search algorithm, NAS），NAS 的开创性工作是 GoogleBrain 于 2016 年同时提出的[70]. 随后 MIT 和 GoogleBrain 又在其基础上做了一系列的改进工作，加入了强化学习、基于序列模型的优化、迁移学习等更多合理的逻辑思路，随之依次出现了 NasNet[66,71]、基于正则化进化的 NasNet[72]、PNAS[73]和 ENAS[74]等. 贺鑫等将目前神经网络搜索算法的研究进展进行了详细总结[75]. Google 推出了 Cloud AutoML 平台，只需上传你的数据，Google 的 NAS 算法就会为你找到一个快速简便的架构. AutoML 的出现降低了部分行业对机器学习尤其是神经网络的使用者的在数量和知识储备上的要求，进一步拓宽了机器学习和神经网络的适用范围.

### 1.2 深度神经网络的主流硬件平台

随着硬件技术和深度神经网络的发展，目前形成了以“CPU+GPU”的异构模式服务器为主的深度神经网络的研究平台，如英伟达的 DGX-2. 其具有 16 块 Tesla V100 GPU，可以提供最高达 2 PFLOPs 的计算能力. 面对复杂的实际应用需求和不断加深的神经网络结构，多样化的深度神经网络硬件平台也不断发展起来，形成了以通用性芯片（CPU、GPU）、半制定化芯片（FPGA）、全制定化芯片（ASIC）、集成电路芯片（SoC）和类脑芯片等为主的硬件平台市场. 计算性能、灵活性、易用性、成本和功耗等成为评价深度神经网络硬件平台的因素和标准.

#### 1.2.1 GPU

GPU（Graphic Processing Unit）起初专门用于处理图形任务，主要由控制器、寄存器和逻辑单元构成. GPU 包含几千个流处理器，可将运算并行化执行，大幅缩短模型的运算时间. 由于其强大的计算能力，目前主要被用于处理大规模的计算任务. 英伟达在 2006 年推出了统一计算设备构架 CUDA 及对应的 G80 平台，第一次让 GPU 具有可编程性，使得 GPU 的流式处理器除了处理图形也具备处理单精度浮点数的能力. 在深度神经网络中，大多数计算都是矩阵的线性运算，它涉及大量数据运算，但控制逻辑简单. 对于这些庞大的计算任务，GPU 的并行处理器表现出极大的优势. 自从 AlexNet[13] 在 2012 年的 ImageNet 比赛中取得优异成绩以来，GPU 被广泛应用于深层神经网络的训练和推理. 大量依赖 GPU 运算的深度神经网络软件框架（如：TensorFlow、PyTorch、Caffe、Theano 和 Paddle-Paddle 等）的出现极大地降低了 GPU 的使用难度. 因此它也成为人工智能硬件首选，在云端和终端各种场景均被率先应用，也是目前应用范围最广、灵活度最高的 AI 硬件.

#### 1.2.2 FPGA

FPGA（Field Programmable Gate Array）是现场可编程门阵列，它允许无限次的编程，并利用小型查找表来实现组合逻辑. FPGA 可以定制化硬件流水线，可以同时处理多个应用或在不同时刻处理不同应用，具有可编程、高性能、低能耗、高稳定、可并行和安全性的特点，在通信、航空航天、汽车电子、工业控制、测试测量等领域取得了很大应用市场. 人工智能产品中往往是针对一些特定应用场景而定制的，定制化芯片的适用性明显比通用芯片的适用性高. FPGA 成本低并且具有较强的可重构性，可进行无限编程. 因此，在芯片需求量不大或者算法不稳定的时候，往往使用 FPGA 去实现半定制的人工智能芯片，这样可以大大降低从算法到芯片电路的成本. 随着人工智能技术的发展，FPGA 在加速数据处理、神经网络推理、并行计算等方面表现突出，并在人脸识别、自然语言处理、网络安全等领域取得了很好的应用.

#### 1.2.3 ASIC

ASIC（Application Specific Integrated Circuit）是专用集成电路，是指根据特定用户要求和特定电子系统的需要而设计、制造的集成电路. 相比于同样工艺 FPGA 实现，ASIC 可以实现 5~10 倍的计算加速，且量产后 ASIC 的成本会大大降低. 不同于可编程的 GPU 和 FPGA，ASIC 一旦制造完成将不能更改，因此具有开发成本高、周期长、门槛高等问题. 例如近些年类似谷歌的 TPU、寒武纪的 NPU、地平线的 BPU、英特尔的 Nervana、微软的 DPU、亚马逊的 Inderentia、百度的 XPU 等芯片，本质上都属于基于特定应用的人工智能算法的 ASIC 定制. 与通用集成电路相比，由于 ASIC 是专为特定目的而设计，GoogleBrain 具有体积更小、功耗更低、性

能提高、保密性增强等优点，具有很高的商业价值，特别适合移动终端的消费电子领域的产业应用.

1.2.4　SoC

SoC（System on Chip）是系统级芯片，一般是将中央处理器、储存器、控制器、软件系统等集成在单一芯片上，通常是面向特殊用途的指定产品，如手机 SoC、电视 SoC、汽车 SoC 等. 系统级芯片能降低开发和生产成本，相比于 ASIC 芯片的开发周期短，因此更加适合量产商用. 目前，高通、AMD、ARM、英特尔、英伟达、阿里巴巴等都在致力于 SoC 硬件的研发，产品中集成了人工智能加速引擎，从而满足市场对人工智能应用的需求. 英特尔旗下子公司 Movidius 在 2017 年推出了全球第一个配备专用神经网络计算引擎的 SoC（Myriad X），芯片上集成了专为高速、低功耗的神经网络而设计的硬件模块，主要用于加速设备端的深度神经网络推理计算. 赛灵思推出的可编程片上系统（Zynq 系列）是基于 ARM 处理器的 SoC，具有高性能、低功耗、多核和开发灵活的优势. 华为推出的昇腾 310 是面向计算场景的人工智能 SoC 芯片.

1.2.5　类脑芯片

类脑芯片（brain-inspired chip）是仿照人类大脑的信息处理方式，打破了存储和计算分离的架构，实现数据并行传送、分布式处理的低功耗芯片. 在基于冯诺依曼结构的计算芯片中，计算模块和存储模块分离处理从而引入了延时及功耗浪费. 类脑芯片侧重于仿照人类大脑神经元模型及其信息处理的机制，利用扁平化的设计结构，从而在降低能耗的前提下高效地完成计算任务. 在人工智能火热的时代，各国政府、大学、公司纷纷投入到类脑芯片的研究当中，其中典型的有 IBM 的 TrueNorth、英特尔的 Loihi、高通的 Zeroth、清华大学的天机芯等.

目前，深度神经网络芯片正在不断研究开发中，每种芯片均是针对一定的问题而设计的. 因此，不同的芯片有其独特的优势和不足. 通过上述对不同芯片的描述，我们可以了解到相比 GPU，FPGA 具有更强的计算能力和较低的功耗. 相比 ASIC 和 SoC，FPGA 具有更低的设计成本和灵活的可编程性. 相比类脑芯片，FPGA 的开发设计更简单. 综合当前深度神经网络芯片的特性可知，FPGA 的设计性能更适合应用于深度神经网络在普通领域的开发和应用.

随着 FPGA 在深度神经网络领域的应用，相关学者对其进行了分析和整理. 文献[76]对基于 FPGA 的卷积神经网络加速过程进行分析总结. 文献[77]汇总了目前 FPGA 用于卷积神经网络加速的发展研究现状. 文献[78]对目前 FPGA 用于神经网络的发展现状进行总结，并提出所面临的问题和挑战. 文献[79]总结了 FPGA 的设计理论及其用于神经网络加速的技术原理和实现方法. 文献[80]分别介绍了人工神经网络和 FPGA 进行介绍的发展，同时总结了 FPGA 用于人工神经网络的发展和挑战. 本文从 FPGA 应用于深度神经网络的设计原理、型号选择、应用领域、加速器及具体加速原理、实验评估指标到最后的 FPGA 应用与深度神经网络的影响因素等方面进行归纳总结，对 FPGA 用于神经网络加速进行全面的介绍，为读者提供理论和实践指导.

## 2　FPGA 的基本介绍

近年来，随着人工智能的快速发展，FPGA 由于其独有的硬件特点成为深度神经网络产业应用的宠儿. 经过近六十年的发展，FPGA 的制作工艺、封装密度、硬件结构以及开发方式发生了巨大的变化. 本章节将围绕 FPGA 的发展历程以及开发方式展开论述.

### 2.1　深度神经网络的主流硬件平台

FPGA 是基于可编程逻辑器件发展的一种半定制电路，它可以使用硬件描述语言（Verilog 或 VHDL）或 C/C++/OpenCL 编程，利用小型查找表来实现组合逻辑，并对 FPGA 上的门电路以及存储器之间的连线进行调整，从而实现程序功能. 早在 20 世纪 60 年代，Gerald Estrin 就提出了可重构计算的概念. 直到 1985 年，Xilinx 推出全球第一款 FPGA 产品 XC2064，该产品采用 2μm 制作工艺，包含了 64 个逻辑单元、85K 个晶体管和数量不超过 1K 个的门. 1992 年，GANGLION 成为神经网络首次在 FPGA 上实现运行的项目[81]. 1996 年卷积神经网络首次在 Altera 的 EPF81500 上实现运行[82]. 随着神经网络的迅速发展，FPGA 做了一系列针对其需求的开发设计，如 Xilinx 推出的 Versal AI Core 系列和 xDNN 处理引擎为深度神经网络推断加速带来突破性的改善. 另外，为了促进深度神经网络的发展，不少公司设计提出神经网络编译及框架，如 ALAMO 编译器和 Lattice 公司设计的 sensAI 编译器、FP-DNN 框架和 FPGAConvNet 框架. 经过 30 多年的发展，FPGA 的制作工艺、逻辑单元和晶体管的封装密度均得到飞速发展，其发展历程线路如图 1 所示.

### 2.2　FPGA的开发方式及流程

结合 FPGA 自身硬件架构特点，目前其开发主

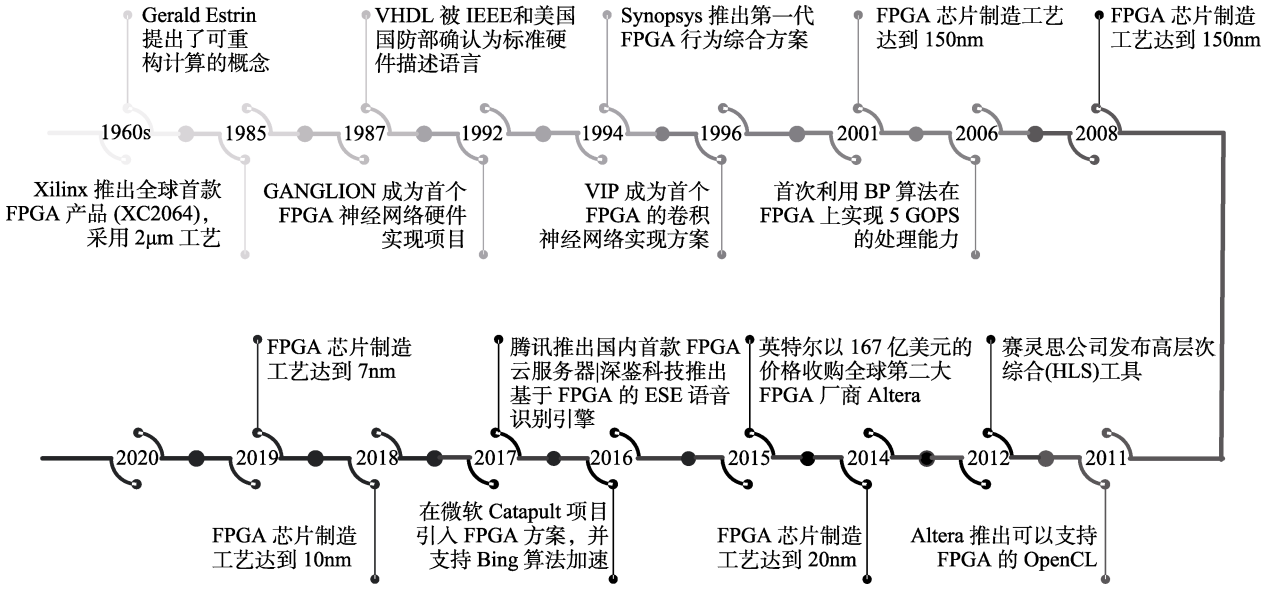


图 1 FPGA 的发展历程线路图

要有两种方式：寄存器传输级（RTL）描述和高层次综合（HLS）描述. 寄存器传输级描述，又称 RTL 级描述，是指用寄存器这一级别的描述方式来描述电路的数据流. 开发人员利用硬件描述语言（Verilog 和 VHDL）或者 IP 核对硬件结构进行描述. RTL 级开发的主要优势是高稳定性、高资源利用率、高性能等. 其劣势也很突出，主要有开发难度大、开发效率低、周期长、成本高等. 基于此，利用高级语言实现算法的开发方式应运而生，即 HLS 级. 开发人员只需要利用高级语言（C、C++）实现算法，而算法程序到 FPGA 硬件结构的映射由编译器自动完成. 广义上讲，Xilinx 公司推出了高层次综合 HLS 工具和 Altera 公司主推的 OpenCL SDK 都属于 HLS 级. HLS 级开发的主要优势是门槛低、开发效率高、周期短，其缺点主要是资源利用率低、性能低、不透明等. 我们对现有文献中的开发方式做了统计，如表 1 所示.

FPGA 的设计流程通常包括设计需求理解、方案评估、芯片理解、详细方案设计、仿真、综合、布局布线、时序优化、芯片编程与调试. 神经网络加速设计开始之前，需要先确定神经网络的网络结构和各层级参数，避免因修改而造成的资源浪费和工程延期. 在神经网络算法效果符合预期后，需要对神经网络算法进行拆解和分析. 通过分析算法所需的计算带宽、存储带宽、存储容量、关键计算瓶颈、计算数据流等，结合 FPGA 芯片的资源特性进行加速效果推算，把适合 FPGA 做加速的算法交给 FPGA 来加速. 其中需要特别注意的是，FPGA 加速计算推演的数据精度需要和软件验证的数据精度保持一致，否则可能在 FPGA 加速后会有精度损失而达不到预期的算法效果.

**表 1 常用的开发方式**

| 开发级别 | 开发语言 | 文献 |
|---|---|---|
| RTL 级[83–87] | Verilog | [88–92] |
| | VHDL | [93–99] |
| HLS 级[100–104] | C | [105, 105–107] |
| | C++ | [108–112] |
| | OpenCL | [112–117] |

## 3 FPGA 深度神经网络相关应用

随着深度神经网络的不断发展，衍生出的智能化产品也越来越多. FPGA 的应用领域已经从原来的通信扩展到消费电子、汽车电子、工业控制、测试测量等更广泛的领域. 在学术界，FPGA 与深度神经网络结合的应用也得到越来越多的关注，成为研究热点. 我们以图像检测与识别、目标跟踪、语音识别、文本处理、网络安全、智能控制 6 个方向来介绍目前 FPGA 与深度神经网络结合的产业应用现状.

### 3.1 图像检测与识别

不管是用于身份核验的人脸识别系统，或是用于植物拍照识别的手机应用系统，这都依赖于图像检测与识别的过程，是深度神经网络的典型应用场景. 而对于商业识别的设备，为了降低产品价格和功耗、提高产品稳定性和速度，大多数企业会选择使用 FPGA 作为图像识别算法的硬件载体. 近年来，

关于 FPGA 上面部署图像识别的应用越来越多，如人脸识别[118−123]、人手姿态识别[124]、字符识别[125, 126]、车牌识别[127]、交通标志识别[128]、自然场景识别[91]等等. 可以看出，基于图像识别的 FPGA 产品应用场景越来越细化，场景越来越具体化. 在以上的场景中，目标检测与识别任务起到了重要的作用. 计算机视觉中的图像目标检测是重要研究方向，关于在 FPGA 上部署深度神经网络检测模型也是研究者比较关注的，如文献[129]就提出了一种稀疏的 YOLO 检测模型，并在 Intel Arria-10 GX1150 FPGA 上达到了 2.13 TOPS（72.5 fps）的吞吐量，并在 PASCAL VOC2007 数据集上的检测精度达到了 74.45%. 还有关于道路交通的目标检测工作，如行人检测[130−132]、车辆检测[133]与障碍检测[134]，用作车道偏离警告系统的多用途道路路径提取[135]等研究工作也在不断进行. 检测路上是否有行人、交通标志以及视野中是否存在汽车或其他车辆，这些均属于自动驾驶实现的基础工作. 在 FPGA 上部署人脸识别系统[118, 119, 136]的研究已经很普遍了，并且场景也越来越多，如应用于移动视频会议、微型无人机和其他小型机器人的低功耗、轻量化嵌入式视觉系统[119]. 为了适应不同场景的需求并且获得更好的用户体验，人们对于识别系统的速度也有了越来越高的要求. 因此，一部分研究工作是在 FPGA 上部署图像识别算法，验证所提出的加速方案和架构[91,137]，加快算法计算速度，提高识别系统的识别速度. 图像检测与识别将成为自动驾驶场景的研究重点之一，同时也就对于 FPGA 这种便携、低功耗设备有着一定的需求. 图像检测与识别不单单只是工业用途，在医疗辅助上面也有具有一定的前景. 医院中的心电图、X 光胸片等医学图像识别的医疗辅助系统，抑或是可穿戴的视觉障碍患者移动辅助系统[134]都是图像检测与识别的具体应用. 我们可以看到，这些技术与设备正在改变着我们的生活方式，为人们带来便利.

### 3.2　目标跟踪

目标跟踪最近几年发展迅速，不少研究者在研究如何在 FPGA 上实现目标跟踪系统，从而推动产业应用. 目标跟踪系统在军事侦察、安防监控等诸多方面均有广泛的应用前景. 目前，较多研究主要是将 FPGA 作为协处理器的目标跟踪系统，用于实时视觉跟踪[138−140]. 不同的实时视觉跟踪系统设计中使用的方法也不尽相同，如：mean shift 跟踪算法、hausdorff 距离算法、光流法等[138, 139, 141]，计算边缘/角点检测、静止背景和噪声滤波等优化操作也常常在实际中进行应用. 近年来，随着深度神经网络模型的不断发展，其跟踪网络性能明显优于传统方法. 如：文献[142]中提出的 MiniTracker，使用的是全卷积的 Siamese 网络，并对其进行了剪枝和量化，使得其在 ZedBoard 上实现并且达到了 18.6 帧每秒的跟踪率.

跟踪系统设计中方法的选择主要是根据应用中使用者对于 FPGA 设备要求的侧重点不同. 当然，多数侧重于实现低功耗和低成本的实时目标跟踪[139, 141, 143]. 设计者会在跟踪精确度与成本之间做一种均衡，在满足精度需求的基础上，尽可能降低功耗与成本. 因此，我们常常需要对部署的深度网络模型进行简化操作，如以上提到的剪枝和量化操作.

### 3.3　语音识别

目前，深度神经网络除了在图像和视频领域应用越来越广泛以外，基于 FPGA 的语音识别系统也成为研究热点. 由于其庞大的市场需求，语音识别发展速度异常迅猛. 在智能语音识别产品中，为保证一定的灵活性和移动性，往往在 FPGA 上部署语音识别模型，以满足智能与生产落地的需求. 在其相关研究中，语音识别模型主要有连续隐马尔可夫模型、液体状态机以及递归神经网络[144] 等等. 其中，文献[145,146]主要在 FPGA 上实现了马尔科夫模型的语音识别效果. 文献[147]在 FPGA 上实现了液体状态机，并利用语音识别进行了评估. 与 AMD Opteron TM 处理器的运行速度相比，该方法在 FPGA 上实现了 88x 运行加速. 文献[144]设计了一种基于神经网络的实时语音识别系统，其包含将用于声学建模的语音特征，以及用于字符级语言建模的两个递归神经网络. Yong Zheng 等人在文献[148]中通过剪枝、量化等操作对 LSTM 模型进行简化，在 FPGA 上实现了一种高性能、高效的 LSTM 接口. 文献[149]提出了一种基于 FPGA 主板的 LSTM 神经网络硬件加速器. 作者也对 LSTM 进行了稀疏剪枝操作并采用流水线方法，在实验性能上，其运行速度高于 ARM Cortex-A9 处理器. 文献[150]主要提出使用一维的通用背景模型（1D CNN）对说话人进行识别. 与 CPU 平台相比，它减少了 ResNet20 的计算复杂度以及参数量使得其在 FPGA 上的运行速度在 3S 与 5S 数据集上分别达到了 5.1 和 6.8 倍的加速. 文献[151]中提出的 AIX 是针对于 DNN 的商业语音识别应用设计的 FPGA 加速接口. 文献[152]提出了基于梯度计算的 LSTM，其与二进制的 LSTM 相比，在几乎不损失精度的情况下减少 73.24%的能

耗. 文献[153]中，作者将 Toeplitz 结构应用于 DNN 模型上，在不计 loss 的情况下实现了较高的压缩率. LSTM 在应用该方法后，模型尺寸减少了 28.7 倍，在 FPGA 上实现了 130000 帧/秒的吞吐量.

在调查过程中，可以发现 LSTM 模型已成为目前 FPGA 部署的典型的语音识别模型，目前已经被成功并广泛地应用于人工智能应用中[154, 155]. 深鉴科技的 ESE 语音识别引擎因其深度压缩技术引起了一时轰动. 在其为代表的相关研究表明，研究者主要希望将语音模型简化并部署在 FPGA 上，从而实现高性能并且高效的语音识别效果. 我们相信，语音识别的应用领域将不断扩大，除了电子产品与通信领域，也终将进入医疗、工业等各个领域.

### 3.4 文本处理

自然语言处理主要是研究人与计算机之间通过自然语言的方式进行有效通信的理论和方法. 作为人工智能领域的一个重要应用方向，自然语言处理已经得到了广泛的关注. 自然语言处理可以分为文本处理以及语音处理，我们将语音识别应用在上一小节进行了归纳总结. 因此，本小节将主要介绍并归纳文本处理的相关应用.

文本处理典型代表有百度 NLP 语义计算整体框架，其中的核心部分就有包含 FPGA 在内的高性能计算模块，以及基于深度神经网络和概率图模型的语义计算引擎. 除此以外，还有很多基于 FPGA 上的常用的自然语言框架，如 RNNLM 框架[156]、DNN 的设计框架[157]、SimNet 语义匹配框架[158]、基于随机计算的深信度字符识别网络框架[159]等等. 文献[160]中主要将有效的 CNNs 模型压缩，进而用于情绪分析. 其压缩过程包含了剪枝、量化，最后将压缩后的模型映射到 FPGA 上进行实验，在准确性不下降的条件下，内存带宽占用比原始模型降低了 85%~93%. 2018 年，谷歌提出的自然语言处理模型即 BERT 模型[161]，一度成为 NLP 的研究热点. 因为其可以在 11 种不同 NLP 测试中创造出最佳成绩. 随后，2019 年 Facebook 提出的具有强大优化能力的 BERT 方法 RoBERTa 模型在 GLUE、SQuAD 和 RACE 三个排行榜上均实现了最优的结果. 文献[162]主要针对基于 Transformer 的大规模语言表示提出了一种有效的加速框架 Ftrans. 该框架可以将 RoBERTa 模型压缩至原来尺寸的 1/16，实现 27.07~81 倍的性能改进. 在能效方面，其比 CPU 的能效高出 8.8 倍.

在文本语言处理方面，FPGA 与深度学习结合的应用成为一大热点，其将被广泛地应用于各种机器翻译，用户情感分析等产品中去. 同时，研究表明人们对于文本处理速度有着一定的要求，因此大量的研究将以简化语义模型以及提升 FPGA 计算速度为目的进行展开.

### 3.5 网络安全

网络安全与入侵检测也是 FPGA 与深度神经网络结合的一个重要应用，主要是对于网络系统中收集的信息进行分析，然后通过某种模型判断是否存在异常的行为. 基于 FPGA 的网络安全与入侵检测系统就是为了对于网络进行实时监控，并在网络系统异常时或者对外来攻击进行及时的反应，以保证网络系统的安全性. 关于该方面的研究也越来越多[163–166]，有降低 FPGA 的计算要求的深度神经网络算法实现在线异常入侵检测系统，也有利用可重构硬件辅助网络入侵检测系统，以及利用 FPGA 搭建了网络传输异常检测体系结构等. 这些系统往往都可以被集成在可重构系统中，作为辅助系统使用.

### 3.6 智能控制

除了以上几种典型的应用，基于 FPGA 的深度神经网络系统还在智能控制领域得到了广泛的应用，如文献[167–171]等等. 文献[167]提出了一种基于人工神经网络的步进电机低速阻尼控制器，该控制器设计用于消除低速时的非线性干扰. 文献[168]介绍了在可编程自动化控制器上嵌入 FPGA 的多层神经网络的实现，在安装在控制器内部的 FPGA 中实现基于神经网络的状态估计，可以将开发的结构直接转移到实际应用中. 文献[169]提出了一种利用人工神经网络( ANN )实现强化学习方法 Q-Learning 的 FPGA 实现方法，并将其用于行星探测器和航天器的智能控制系统. 该方法将神经网络自身的并行性与 FPGA 的硬件的并行性进行匹配，大大提高了处理速度. 与传统的 Intel i5 2.3 GHz CPU 相比，Virtex 7 fpga 的速度提高了 43 倍. 文献[170]提出了一种最大功率点动态跟踪控制器，该动态控制器主要采用了基于级联神经网络( CNN )的 MPPT 算法，从变速条件下的 WPCS 中提取最大功率. 通过实验得出，与传统的 MPPT 算法相比，基于级联神经网络的 MPPT 算法的控制器的控制效率更高，能对风速改变提供更好的响应. 将基于 FPGA 的深度神经网络用于实际控制，打破了传统逻辑控制模式，实现了控制系统的自动化和智能化.

# 4　FPGA 深度神经网络的加速与优化

深度神经网络往往是在大内存、较强计算力的 GPU 上进行训练学习的. 但在相关模型进行产品化落地应用时，必须考虑设备资源的尺寸、内存、能耗、带宽和成本等因素. 神经网络模型压缩和加速的提出，让复杂的深度神经网络在小型设备( FPGA ) 上的实现成为了可能. 这些实现使得神经网络得以搭载在 FPGA 芯片上，进一步应用到自动驾驶、航天航空以及手机等设备中. 本文将从 FPGA 神经网络加速器、神经网络压缩与加速技术、计算加速与优化、基于带宽的神经网络加速以及基于 FPGA 的神经网络编译器及框架等五个方面进行总结与归纳.

## 4.1　FPGA神经网络加速器

随着深度学习的不断发展，神经网络在图像、视频、语音处理等各个领域取得了巨大的成功. VGGNet、GoogleNet、ResNet 的出现，让我们清楚的看到神经网络正往更深、更复杂的网络设计方向发展. 那么，如何将更复杂的神经网络部署到FPGA上，并能够满足一定的速度要求，成为研究者们关注的焦点. 在现有的研究中，涌现出大量的 FPGA 深度学习加速器，例如：DLAU[172]、Deep-Burning[173]、DeepX[174]、BISMO[175]、Bit Fusion[176]与为 SGEMV 设计的 FPGA 加速器[177]等等.

DLAU[172]是一种可扩展的加速器架构，主要使用三级流水线处理单元与分片技术去提高吞吐量并对深度学习应用程序的局部性特征进行探索. 其中，三级流水线处理单元主要包含分片矩阵乘法单元，部分和累加单元和活化函数加速单元，可以重复使用以实现超大规模的神经网络. 分片技术是来分割大规模的输入数据，将输入节点数据转换成更小的集合并重复计算，这样使得 DLAU 加速器可以将处理不同尺寸的分片数据进行灵活配置. 经实验验证，与 Intel Core2 处理器相比，DLAU 加速器可以实现 36.1 倍的加速效果.

DeepBurning[173]是一种自动化神经网络加速器的实现方法. 用户只需提供网络拓扑上层描述和硬件资源约束，框架中的神经网络集成器就可以自动分析网络特征，结合硬件约束在由内积单元、累加单元、池化单元等组成的组件库中选出合适组件组成硬件网络，以及对应的控制流、数据流和数据布局方案. 该方法的出现，方便了设计者使用 FPGA 加速神经网络计算，同时提高了 FPGA 的领域适应性. 实验表明，与最先进的基于 FPGA 的解决方案相比，DeepBurning 设计的加速器具有更高的功耗效率. DeepX[174]是一个用于深度学习执行的软件加速器，可以降低深度学习应用时所需的内存，计算量与功耗. 它通过利用移动片上系统的异构处理器（如 GPU，LPU）的混合，降低资源开销. 在执行深度模型的不同推理阶段时，每个计算单元提供不同的资源效率. DeepX 在深度学习推理阶段时，通过扩展模型压缩原理，可以扩展单个模型层的复杂度，并以此控制推理过程中层的内存计算和能量消耗. 同时，它还可以分解深层模型结构，将各块分配给本地和远程处理器，有效并最大限度地提高资源利用率. 实验表明，DeepX 可以让大规模的深度学习模型在移动处理器上高效地执行，其性能显著优于现有的解决方案.

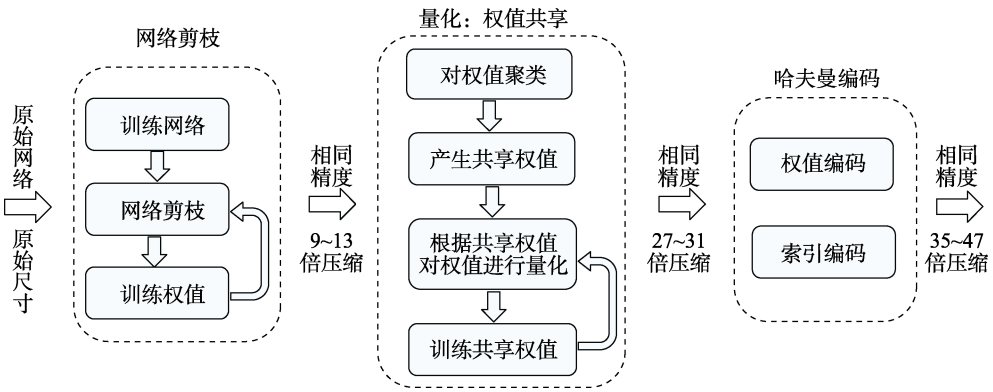


图 2　神经网络深度压缩方法的主要流程

BISMO[175]是一个可扩展的位串行矩阵乘法覆盖，可以有效地在 FPGA 上实例化. 它的核心是一个软件可编程加权二进制矩阵乘法引擎和用于获取数据并存储结果的相关硬件. 软件的可编程性使得它可以在任何矩阵大小和任何定点或整数精度上进行操作. 文献[177]中主要提出了二值神经网络

（BNN）硬件加速器的设计方案，它支持处理任意 BNN 所需的所有操作. 网络参数（例如，二进制权值、标准化常数）被保存在这些片上 RAM 中，并提供给并行执行计算的许多 PEs. 片上 RAM 为 PEs 提供了足够的带宽，提高了吞吐量. 与 CPU 和 GPU 上经过良好优化的软件相比，所提出的二值神经网络加速器在性能和能效方面都有一定数量级的提高. Bit Fusion[176]是一个支持不同位计算的加速器，动态融合不同位以匹配各个单独 DNN 层的位宽，这使得我们可以将动态 bit-level 融合/分解引入到 DNN 加速器的设计中. 与目前两种先进的 DNN 加速器 Eyeriss 和 Stripes 相比，Bit Fusion 实现了 2~3 倍的加速以及降低了 3.9~5.1 倍的能耗.

### 4.2 神经网络压缩与加速技术

众所周知，深度神经网络在多个领域上表现出优于传统算法的效果. 但在应用过程中，其计算量巨大，占据内存较大. 因此，若想真正将神经网络应用到嵌入式系统中去，就必须对神经网络自身进行处理，以实现神经网络的压缩与加速. FPGA 加速设计中涉及的几种常见的神经网络压缩与加速方法：包含网络剪枝在内的深度压缩、低秩估计、模型量化以及知识蒸馏方法[178, 179].

#### 4.2.1 神经网络深度压缩

2015 年，Song Han 等人在文献[180]中提出了在不损失精度的前提下，深度压缩对于 AlexNet 可以减少 35 倍的内存占用，对于 VGG-16 的内存占用可以减少 49 倍，其中网络剪枝就可以实现的模型压缩率就能达到 10 倍以上. 2017 年，深鉴科技的语音识别技术成为了在 FPGA 上实现深度压缩应用的成功典型. 当然，依然存在不少在 FPGA 上实现的剪枝技术的研究. 2017 年，在文献[181]中，Fujii 等人提出了一种神经元修剪技术，将 VGG-11 网络层中神经元数量减少了 89.3%，但是保持了 99%的准确性. 在权重参数大大减少的情况下，由 FPGA 上的片上存储器实现的权重存储器，就可以对于存储器进行高速访问. 文献[182]提出了一种结构化的修剪方法，不仅可以减小 LSTM 模型的大小，而且不会损失预测精度，同时可以消除不平衡计算和不规则的内存访问. 神经网络的深度压缩的过程中包含：模型剪枝、权值量化与共享和霍夫曼编码，它们一起工作大大减少了神经网络的存储需求，在精度几乎无损的情况下，将模型压缩几十倍以上，具体的深度压缩方法的主要流程如图 2 所示.

网络剪枝的提出可以溯源到 1989 年，由图灵奖获得者 Yann LeCun 在文献[183]中提出来的. 其核心思想主要是通过估计每个参数的重要程度，删除不重要的参数，以达到模型压缩的目的，效果示意如图 3 所示. 1993 年，发表在 NIPS 上的一篇文献[184]中，利用了二阶泰勒展开对网络参数进行选取并进行剪枝操作，提出了将剪枝看作正则项来改善网络的训练和泛化能力. 根据裁剪方式不同，现有的剪枝方法分为阈值裁剪方法、动态补救方法、Filters 裁剪方法和重要性裁剪方法. 在剪枝过程中，可以通过计算损失函数对参数的二阶导数的大小或计算参数的绝对值的大小来衡量网络参数重要程度，数值越小参数越不重要，可以进行删除. 设计者在进行网络裁枝时，需要考虑两个问题：对输出的一个节点进行裁剪是否影响到其他输出节点和如何对于删掉的参数进行彻底清除. 针对于以上两个问题，目前提出了 Filter-level、Group-level 与稀疏卷积方法[185]. 不管使用哪一种剪枝方法，剪枝后的网络需要进行再次参数调优. 2017 年，深鉴科技在文献[186]中提出了负载平衡感知剪枝，考虑到最终多核并行加速的时候不同核心之间的负载均衡问题. 当然，在剪枝的过程中，剪枝的粒度大小也会影响网络的精度. 剪枝按照粒度大小可分为细粒度剪枝和粗粒度剪枝，其中细粒度剪枝主要是对于权重进行裁剪，主要是进行局部的调整，可以保留模型的精度，粗粒度的剪枝则是剪去滤波器或通道，模型在速度上有较大的提升. 2017 年，斯坦福大学的 Huizi Mao 等人发表在 CVPR 上的文献[187]就探索了不同修剪粒度对于模型精度的影响，并证明粗粒修剪也能够达到与细粒修剪接近的甚至更好的精度.

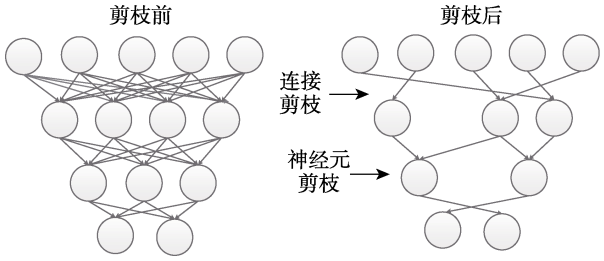


图 3 连接与神经元剪枝前后

神经网络深度压缩过程的第二阶段是模型量化与参数共享，在这里只介绍参数共享，模型量化将在下一小节进行系统介绍. 文献[180, 188]将参数共享归为量化操作中的一小步. 不同的研究采用的共享规则也是存在差异的. 文献[189]使用了一个低成本的哈希函数将连接权重随机地分配到哈希桶中，然后所有在同一个哈希桶中的连接共享一个参数值；而文献[190]是对所有权重使用 *k*-means 聚类操

作，聚类完成之后属于同一类的权重将共享一个参数值. 它们都是通过共享参数的方法来压缩神经网络模型. 其中典型的聚类量化方法将训练的每一层的权值参数进行 *k*-means 聚类，聚类完成之后属于同一类的参数都使用该类聚类中心的数值作为它们的权值参数数值，然后通过索引矩阵将共享权值一一对应到权值参数的确定位置. 通过权值共享，原来的权值矩阵变成了一个查找表与共享权值矩阵，也就是说原来权值矩阵被一个与权值矩阵相同大小的查找表矩阵代替，查找表矩阵上面的索引可以准确找到对应的共享权值，共享权值则只需要存储在一个大小为共享权值总个数的矩阵中即可，如图 4 所述.

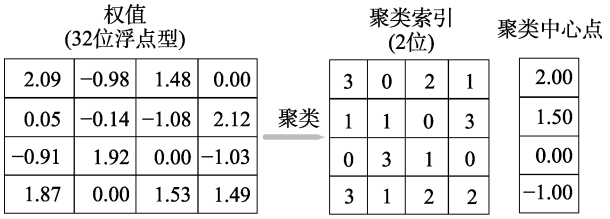


图 4　神经网络深度压缩方法中的权值共享

神经网络深度压缩过程的第三阶段是哈夫曼编码. 哈夫曼编码是 1952 年 David A.Huffman 在文献[191]中提出来的. 它是一种可变字长编码的编码方式. 哈夫曼编码可以用短的码值来表示更多的数字，提高编码效率，以达到深度压缩的效果. 因此，在深度压缩过程中最后保存网络的时候常常使用哈夫曼编码进行进一步的压缩.

4.2.2　低秩估计

低秩估计是利用矩阵分解或者张量分解以及矩阵乘法或者卷积运算这种线性运算的结合律，将原本的参数张量分解成若干个小张量，或将原本的卷积用几个小卷积代替. 高维矩阵的运算会采用 Tensor 分解方法对神经网络进行加速和压缩. 目前的 Tensor 分解方法主要包含了 CP 分解、Tucker 分解、Tensor Train 分解[192]和 Block Term 分解方法等等. 典型的神经网络卷积核是一个四维张量，全连接层可以看作是一个二维矩阵，这些张量中往往存在大量的冗余. 我们可以利用低秩估计的方法对于参数矩阵进行分解，减少参数量，达到模型压缩和加速效果. 低秩估计是对神经网络的每一层利用低秩滤波器进行逐层逼近，每一层的参数就会被固定后，而对于每一层之前的层会根据重建误差准则进行微调[193−195]. 压缩 2D 卷积层的典型低秩方法，如图 5 所示.

在文献[196]中，Jaderberg 等人提出的低秩近似

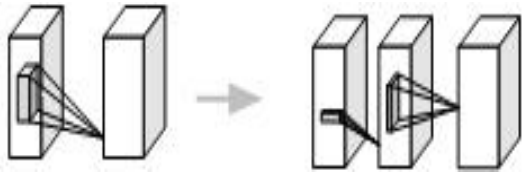

图 5　2D 卷积层的典型低秩方法

方法主要是使用 $n\times1+1\times n$ 的线性组合卷积核替代 $n\times n$ 的卷积核. 该方法在文字识别实验中，在精度无损失的情况下，速度达到了原来 2.5 倍。同时，论文中也列出了使用低秩估计的方法来做模型压缩与加速的优劣势，具体如下：

**优势**　低秩估计没有改变基础运算的结构，也不需要进行额外定义新的操作. 低秩估计的分解方法有很多种，任何的矩阵或者张量分解方法都可以用于低秩分解. 在神经网络中应用低秩估计的方法进行网络的分解，分解后的网络参数量将大大减少.

**劣势**　为保证分解后网络模型的准确率，一般都需要对于分解后的网络进行参数调优. 同时，在低秩估计时，对于秩的保留数量没有明确的规定. 保留的秩太多，可以保证一定的准确率，但加速压缩效果不好. 保留的秩太少，加速压缩的效果较好，但准确率无法保证.

关于 FPGA 利用低秩估计进行网络的加速设计较多，如文献[197, 198]利用低秩矩阵或者低秩扩展的方法进行神经网络加速也是一大研究热点. 文献[199]中，作者基于信噪子空间理论，针对正交频分复用提出了一种低秩信道估计方法并在 FPGA 中进行实现. 2014 年，Jaderberg 等人在文献[196]中提出了通过交叉通道和滤波器冗余的方法来构建秩为 1 的低秩滤波器，实现加速效果. 同年，Denton 等人在 NIPS 上发文[200]，提出了构建秩为 k 的滤波器，对于卷积使用一个中间层进行降秩操作，这种方法在损失 1%的精度前提下，对于每个卷积层可以提速 1 倍. 2017 年，文献[198]中提出了一种新型的 FPGA 加速器，可以加速基于矩阵梯度分解的低秩矩阵实现算法. 文献[197]中在低秩矩阵的理论基础上提出了一种新颖的自动计算框架，可用于基于 FPGA 的非稀疏相关矩阵的大数据在线分析.

4.2.3　模型量化

模型量化可以通过量化函数将全精度的数（激化量，参数，甚至是梯度值）映射到有限的整数空间. 模型量化可以减少每个参数的存储空间，降低计算复杂度，因此可以实现神经网络加速. 与前面介绍的方法不同，网络裁枝方法与低秩估计方法都是从矩阵乘法角度出发，着眼于减少参数量和计算

量. 而模型量化则着眼于参数本身，直接减少每个参数的存储空间，提升计算速度. BWN[201], TWN[202]、TNN[203] 将模型的参数量化到 $\{-1,+1\}$ 或者 $\{-1,0,1\}$，这样就能将网络计算里的乘加运算转化为加减运算，从而实现模型压缩和计算加速的目的. 除了参数外，BNN[204]、XNOR-Net[205]将模型的激活值也量化到 $\{-1,+1\}$，这样就能将乘加运算转化为比特级别的位运算. XNOR-Net[205]在 CPU 上实现了 58 倍的加速比，同时内存节省了 32 倍. 由于位运算在 FPGA 上可以实现更高的加速比，文献[206]实现的二值神经网络，在峰值时比 CPU 加速约 705 倍，比 GPU 快 70 倍. 极低比特的量化带来的精度损失也是非常明显的，为此出现了多比特的模型量化方法. 文献 MBN[207]、Dorefa-Net[208]、ABC-Net[209]将多比特的数值运算分解为多个位运算，可以在保证模型精度的同时实现模型压缩和计算加速. 目前，现有的量化方式主要有均匀量化[205, 207, 208, 210]、对数量化[211–213]和自适应量化[209, 214–216]. 以上文献都是将模型所有层的参数或者激活值量化到相同的精度，而混合精度量化可以根据模型不同层次的重要性和灵敏度来匹配更好的解，从而获得更好的性能. HAQ[217]将硬件模拟器评估的加速度信息反馈加入训练过程，并利用强化学习自动确定量化策略. 文献[218,219]将量化任务转换为神经网络结构搜索问题，使用 Gumbel-softmax 采样选择量化支路，采用反向传播方法优化网络权值和结构参数. 通过量化对神经网络模型进行不同位数的编码，应用少量的乘法与位运算加速了网络模型的运算速度. 或者将神经网络中卷积运算中的矩阵乘法变为计算简单的加减法运算，简化了计算量，进而实现神经网络加速.

基于 FPGA 的深度神经网络实现也将量化作为一种常用技术手段. 2011 年，文献[220]中完成了一种基于 H.264 压缩的整数离散余弦变换和模型量化的 FPGA 实现. 深鉴科技在 FPGA 上部署的尺寸压缩 20 倍的 LSTM 模型实现了语音识别引擎，压缩的 20 倍中，其中 10 倍来自剪枝，2 倍来自量化[186]. 文献[144]将权值量化到 6 位，然后将所有权值存储在 FPGA 的片上存储器中，大大提高了神经网络的计算速度，最终的系统运行速度远高于实时性. 2018 年，文献[206]提出了一种用于 FPGA 的二值化神经网络 FP-BNN，通过数据量化和优化片上存储消除了参数访问的瓶颈.

#### 4.2.4 知识蒸馏

基于 FPGA 的深度神经网络模型的设计一般都趋向于计算复杂度低的小模型，对模型性能有了更高的要求，既要求计算速度快、资源少，又要具有复杂模型的精度性能. 因此，对小模型的训练成为一个焦点问题. 知识蒸馏[221]是 Hinton 等人在 2015 年提出的一种模型训练方式，它是将训练好的复杂网络模型具备的推广能力“知识”迁移到一个结构简单的网络中. 以上提到的网络裁枝、低秩估计以及模型量化，都是对特定网络模型进行压缩和加速. 然而知识蒸馏方法直接设计了一个简单结构的小网络，将难点转移成对小网络的训练上. 整个思想中最大的难题在于如何有效地表达“知识”，并有效地指导小网络的训练. 知识蒸馏的结构主要由三大部分构成：指导网络、指导损失函数以及学习网络，如图 6 所示.

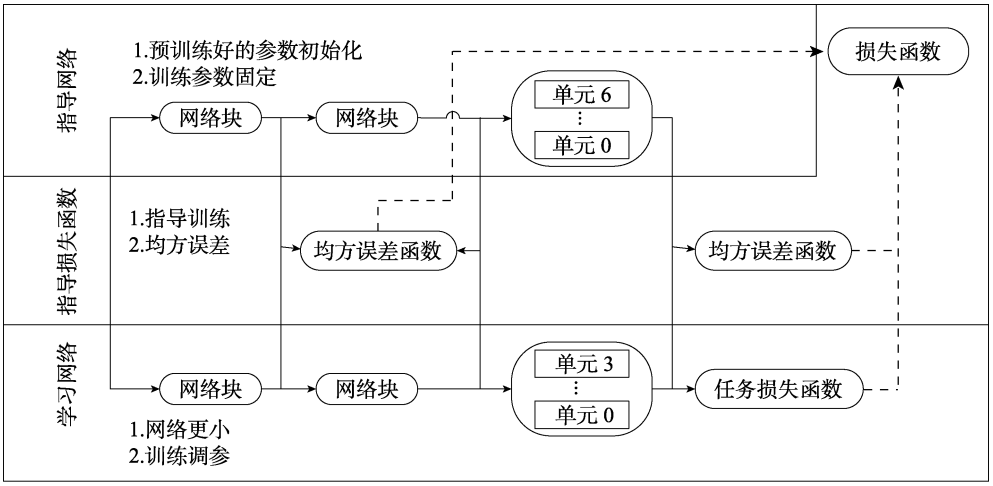


图 6 知识蒸馏的主要训练过程

指导网络加载了预训练模型对于参数初始化，并且在训练过程中参数固定；学习网络的网络结构比较简单，主要用来训练学习参数；对于中间的指导损失函数部分，知识蒸馏一般选取均方误差损失

函数作为用于指导网络与学习网络的损失函数. 在以模型压缩为目的的知识蒸馏任务中，指导网络与学习网络由相同个数、相似结构的网络块组成. 每个网络块中有若干个单元，但学习网络中的每个网络块中含有的单元个数比指导网络的单元个数少. 指导网络又常常被称为“教师网络”，学习网络被常常称为“学生网络”. 知识蒸馏的核心思想中使用了软目标辅助硬目标一起训练. 其中，软目标指的是教师网络模型输出的预测结果，硬目标指的是样本原始标签，即真实标签. 2015 年，Hinton 在他的论文[221]中推荐软目标与硬目标经验权重为 9:1.

在现有的神经网络加速的研究中，有一部分工作是使用蒸馏进行模型优化训练的. 相关研究[222–225]中设计了或者改进了学生模型，在几乎不损失精度的前提下，使用更少的参数、更浅的网络结构提高了网络计算速度. 同时，模型蒸馏往往会与模型量化同时进行神经网络的压缩与加速. 文献[226, 227]中利用知识蒸馏技术，弥补因量化导致的性能下降，提高低精度网络的性能. 文献[228]则是使用了在线蒸馏的方法，突破了分布式 SGD 存在的瓶颈，提高了集群上的模型计算效率与精度.

### 4.3　计算加速与优化

#### 4.3.1　矩阵乘法优化

矩阵运算在深度神经网络的训练与前向计算中占据主导地位，因此加速矩阵运算具有重大意义. 矩阵乘法的优化可以通过减少乘法计算次数和提高计算速度来实现. 矩阵分块技术与 Winograd 转换方法常常作为神经网络中的优化算法. 下面我们将针对这两种方法进行叙述.

（1）矩阵分块技术

在深度神经网络中存在大量的矩阵运算，常常会因为矩阵太大而出现高速缓存缺失的情况. 循环分块可以将矩阵循环拆分成更小的模块进行计算，这样可以使片上存储数据重复利用，减少访问内存的次数，提高计算的效率. 矩阵分块时前后的乘法计算总数恒定不变，将 $n\times n$ 矩阵按 $m\times m$ 进行分块，整个矩阵被分成 $n^2/m^2$ 个子矩阵，乘法计算总数仍然是 $(n^2/m^2)\times n\times m^2$，即 $n^3$. 每次都在相邻位置上进行读写，提高了访问性能. 卷积神经网络算法大部分是利用循环来实现的，循环过程中往往采用循环分块和循环展开进行优化. 循环分块大小会影响计算的并行度，在一定程度上决定了单位时间内进行的计算操作. 所以在卷积神经网络运算时，我们常常需要考虑输出特征图行和列分块大小，并进行自身约束，如行分块大小必须小于总行数等，以及考虑片上资源的限制给出相应的约束条件.

（2）矩阵 Winograd 转换

Winograd 转换方法的核心思想就是使用更多的加法来代替乘法[229]. 因此，我们可以在滤波器维度较小的情况下使用 Winograd 方法做卷积运算. 2016 年，文献[230]是 CVPR 首个将 Winograd 算法引入的研究工作，文中利用 Winograd 算法与傅里叶变换算法分别对卷积神经网络进行剪枝操作，得出 Winograd 算法的剪枝性能优于傅里叶变换算法的结论. 现有的部分研究工作中引入了 Winograd 滤波，通过使用循环展开和平铺策略优化数据路径以支持 Winograd 卷积，在一定程度上使得所设计的卷积神经网络模型加速器性能提升，功耗降低[231, 232]. 部分研究工作中也明确介绍了 Winograd 算法在 FPGA 上的具体实现[233, 234]. 2017 年，商汤科技在文献[235]中提出了 Winograd 算法的 FPGA 实现，采用 Winograd 算法，使得卷积计算循环层数减少. 2018 年，文献[236]中在 FPGA 上实现的稀疏 Winograd 卷积加速器，与最新技术相比，其在 VGG-16 和 YOLO 网络上的实验结果在速度提高了 2.9 倍至 3.1 倍.

#### 4.3.2　卷积优化

神经网络结构往往是由卷积层，池化层以及全连接层构成的. 其中的卷积层和全连接层在硬件结构上，往往是针对于矩阵运算进行设计的. 将卷积层和全连接层可以转化为矩阵，然后利用 FPGA 进行计算并加速. 如图 7 就是将二维卷积转化为矩阵的示意图.

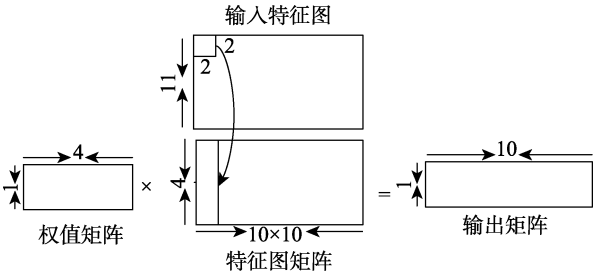


图 7　二维卷积转化为矩阵操作

卷积就是卷积核与图像矩阵的运算. 卷积操作的实现方法主要有滑窗法、傅里叶变换法和 im2col. 滑窗法：卷积核是一个小窗口，在输入图像上按步长滑动，每次操作卷积核对应区域的输入图像，将卷积核中的权值和对应的输入图像的值相乘再相加，得到的最终值赋给特征图对应于卷积核中心位置变量. 以 4×4 的矩阵、3×3 的卷积核为例，从第一个像素开始滑动，逐个计算，如图 8 所示. im2col 操作是用来优化卷积运算，它的核心是将卷积核感

受野转化成一行（列）来存储，优化运算速度，减少内存访问时间. 其计算过程如图 9 所示.

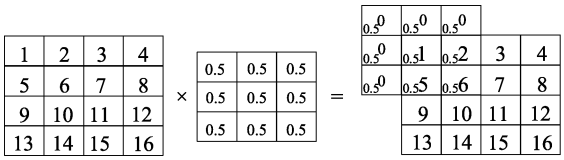

图 8 滑窗法示意图

卷积操作主要涉及输入特征图和卷积核权重的三维乘法和累加操作. 优化计算里的循环方式也是在 FPGA 上部署神经网络的常用方法，主要包括循环展开[91]、循环平铺[237]、循环交换[238]等. 在现有 FPGA 与深度神经网络的相关研究工作中，也均有使用以上三种技术定制具有三级内存层次结构的加速器的计算和通信模式[83,90,232,239,240]，可以有效地映射与执行卷积循环.

4.3.3 频率优化

理论峰值工作频率也是 FPGA 性能指标之一，提高 FPGA 的峰值性能也是目前 FPGA 加速设计中的一个研究方向[85,91,241]. FPGA 能够达到多少工作频率不仅需要考虑 FPGA 芯片自身支持的频率是多少，同时需要考虑如何内部提高时钟频率，即如何对程序优化，来提高多用寄存器工作频率. 28nm 的 Altera FPGA 上实现傅里叶变换等简单算法可以达到数百 GFLOPS，QR 与 Cholesky 分解等复杂算法则达到 100 GFLOPS 以上. 比较新的 FPGA 能够支持 700-900MHz 的 DSP 理论峰值工作频率，但现有的设计通常在 100~400MHz 下工作[109, 116]. 因此，提高 FPGA 工作频率，也是神经网络在 FPGA 上进行部署的一个重点研究方向.

### 4.4 基于带宽的神经网络加速

在基于 FPGA 的神经网络加速中，内存带宽也常常是影响计算速度的瓶颈. 当模型的计算强度小于计算平台的计算强度上限时，此时模型理论性能的大小完全由计算平台的带宽上限以及模型自身的计算强度所决定. 由此可见，在模型处于带宽瓶颈区间的前提下，计算平台的带宽越大，模型的理论性能可呈线性增长. 因此，通过提高带宽上限或者减小带宽需求，可以实现对模型的加速. 本节将从三个方面对基于带宽的神经网络加速进行阐述：首先介绍了衡量神经网络模型性能的 Roof-line 模型，其次分别总结了针对提高带宽上限和减小带宽需求的现有方法.

4.4.1 Roof-line 模型

Roof-line 模型就是用来衡量模型在一个计算平台的限制下，所达到的最大浮点计算速度. 通常计算平台使用算力π与带宽β这两个指标进行性能衡量. 算力也称为计算平台的性能上限，指的是一个计算平台倾尽全力每秒钟所能完成的浮点运算数. 单位是 FLOP/s. 带宽也即计算平台的带宽上限，指的是一个计算平台倾尽全力每秒所能完成的内存交换量. 单位是 Byte/s. 与此对应的，计算强度上限 Imax 就是算力和带宽相除的结果. Roof-line 模型计算力公式为

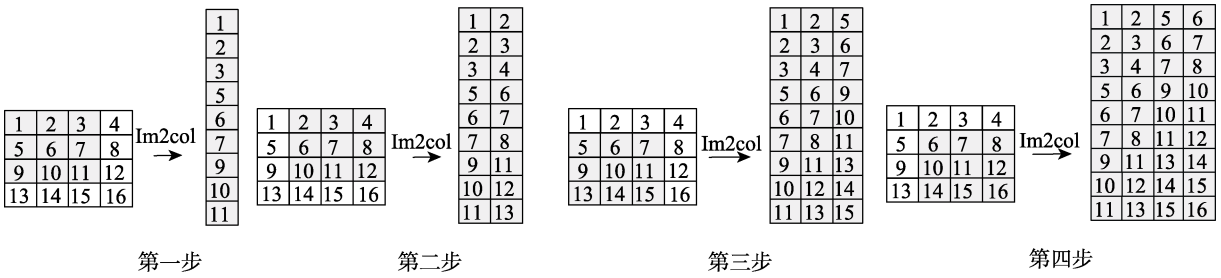


图 9 im2col 法示意图

$$P=\begin{cases}\beta\times I, I<I_{max}\\ \pi, \quad x\geqslant I_{max}\end{cases} \tag{1}$$

有了 Roof-line 模型，我们就可以知道神经网络模型在相关计算平台上的性能. 由计算平台的算力和带宽上限所决定的 Roof-line 模型通常划分为两个区域，计算瓶颈区域和带宽瓶颈区域. 一些基于 FPGA 的加速文献[79, 186, 242]将 Roof-line 模型中的计算强度定义为 CTC 率，即用来表示每次内存访问的操作数. 当模型处于带宽瓶颈状态时，模型理论性能的大小完全由计算平台的带宽上限以及模型自身的计算强度所决定. 由此可见，提高带宽上限或者减小系统的带宽需求，都可以提高系统的性能，从而进行加速. 具体的，通过数据复用、规范数据访问的模式可以提高带宽的上限；而通过数据量化、矩阵计算和存储优化，可以减小系统的带宽需求. 因此，本节接下来将从上述几个方面具体阐述如何

从带宽的角度提高系统的性能，从而进行加速.

4.4.2　提高带宽上限

在基于 FPGA 的深度神经网络加速中，内存带宽常常是提高速度的瓶颈. 对于全连接层部分，由于包含大量的权重，因此会产生大量的内存访问；而对于卷积层部分，大量的乘加操作也会导致大量的内存访问. 由于片外内存的限制，理论带宽有限，实际带宽上限一般低于理论带宽上限，而实际的带宽是由数据访问模式所决定的. 因此，一方面，复用数据可以变相地提高带宽上限；另一方面，对数据访问模式进行优化和改进也可以提高带宽上限，从而达到加速的目的. 因此，提高带宽可以从以下两个方面进行考虑：数据复用和对数据访问模式的规范. 下面本节将对这两个方法的原理和研究现状进行介绍.

（1）数据复用

从 Roof-line 模型可以看出，在数据复用率较低的时候，带宽就成为影响性能的瓶颈，FPGA 运算资源没有被全部利用. 因此，通过数据复用，就可以使内存的实际应用带宽大于理论带宽，从而增加了带宽上限. 数据复用分为输入复用、输出复用和权重复用三种[243]. 输入复用对输入缓冲区具有最小访问权限，输出复用则是复用输出寄存器中的数据，而权重复用则是指一定情况下对 DRAM 进行重复权重访问的情况. 总的来说，这三种数据复用模式就是在不同的阶段对数据进行重复使用，以达到增加性能的目的. 在基于 FPGA 的深度神经网络加速实验中，有很大一部分是基于数据复用从而提高带宽上限进行加速的. 文献[172]所提出的技术利用数据复用来减少冗余通信操作以最大化数据共享与复用. 文献[243]采取数据复用和任务并行化进行数据复用从而在内存带宽不变的情况下提升 CTC 率，变相提高了实际带宽上限. 文献[109]提出了一种数据分配方案，该方案最大化每个事务的突发长度到外部存储器，以加速运算卷积层和全连接层，以及避免不必要的访问延迟，从而有效提高了 DRAM 的传输带宽.

（2）规范数据访问模式

在基于 FPGA 的神经网络加速中，FPGA 作为协处理器. 更具体地说，CPU 把指令写入内存，FPGA 从内存读取指令执行，并把计算结果写入内存. 因此，如果规范了数据访问模式，那么数据的读取效率也会提高，从而增加了实际带宽上限. 文献中常常将特征图切割成存储在不连续地址中的小数据块进行特征映射来规范数据访问模式. 具体的，文献[79]指出，外部存储器中常见的特征映射格式包括 *NCHW* 或 *CHWN*，其中 *N* 表示批次维度，*C* 表示通道维度，*H* 和 *W* 表示特征映射维度. 使用这些格式中的任何一种，可以将特征图切割成存储在不连续地址中的小数据块. 因此，通过规范数据访问模式，增加了实际带宽上限. 文献[85]提出了一种特征映射存储格式，将 $H\times W$ 特征映射排列成 $r\times c$ 大小的（$HW/rc$）块，从而将写突发大小从 $c/2$ 增加到 $rc/2$，从而实现通过数据规范进行加速的目的.

4.4.3　减小带宽需求

在深度神经网络模型计算过程中，数据量化与矩阵计算优化会降低神经网络处理系统的带宽和存储需求，从而减小带宽的需求. 在与数据量化有关的神经网络加速文献中，研究者们常使用二值神经网络进行优化加速. 文献[110]通过利用一组新的优化，能够有效地将二值化神经网络映射到硬件，并且实现了全连接、卷积和池化层，每层计算资源都根据用户提供的吞吐量需求进行定制从而实现了性能的提升. 文献[201]使用二值权值对模型进行量化，并在 ImageNet 数据集上带宽要求降低了 32%.

在深度神经网络的优化与加速中，需要针对不同的网络层操作进行考虑，如：卷积操作和全连接操作所面临的问题是有所差异的. 卷积操作主要以计算优化为主，参数少，带宽需求不高. 全连接操作除了计算优化外，权重参数较大，需要大量内存访问，因此带宽的优化也尤为重要. 在基于矩阵优化减小带宽需求的研究中，研究者们常用的方法有基于 Winograd 的方法、循环展开和矩阵稀疏性分析等的方法. 它们可以有效减少内存访问的总数，从而减少带宽需求并提高计算性能. 具体的，文献[116]利用三维卷积的矩阵转换来实现基于 FPGA 神经网络加速功能. 文献[244]为了减少全连接层的内存占用，将奇异值分解（SVD）应用于全连接层的权值矩阵. 他们还提出了一种动态的数据量化流组件，进一步减少卷积神经网络的内存占用和带宽需求. 文献[109]为了提高带宽利用率，设计了一个统一的矩阵乘法内核. 文献[115]提出了一种基于 OpenCL 的深度神经网络体系结构，该方法利用 Winograd 变换来减少乘法累加运算，可以减少约 50%的运算量. 他们提出的体系结构将卷积层和全连接层的外部存储器带宽需求降低了一个数量级.

### 4.5　基于FPGA的神经网络编译器及框架

FPGA 的价值所在就是高度灵活、快速部署，在这方面涌现出很多可在 FPGA 上部署神经网络的编

译器和框架. 例如：sensAI 和 ALAMO 编译器[239]、FP-DNN 框架[112]、FPGAConvNet 框架[101]、Caffeinated FPGAs 框架[231]以及 FINN 框架[110]等等. 在文献[87]中，Yufei Ma 等人提出了一个基于库的 RTL 级 CNN 编译器，该编译器可以自动生成用于各种 CNN 推理任务的定制 FPGA 硬件，以便实现 CNN 从软件到 FPGA 的高级快速原型设计. 针对 FPGA 部署神经网络，研究者又提出了编译器 ALAMO[239]，它使用模块化 RTL 编译器加速深度学习算法，目标是提供一种自动方式，将 CNN 推理过程映射到可以使用的高效 RTL 代码. 该文献表明，自动编译器解决方案有望实现深度学习的模块化和可扩展硬件加速.

随着神经网络的发展，在 FPGA 上部署神经网络的框架也成为研究热点. 为了将神经网络部署在 FPGA 上，研究者提出各种不同的框架. 例如：Yijin Guan 等人提出了 FP-DNN 框架[112]，该框架可以自动将 DNN 映射到 FPGA 上以加速模型推理. Stylianos I 等人提出了 FPGAConvNet 框架[101]，该框架可以将卷积神经网络 CNN 映射到 FPGA 上. Roberto 等人提出了 Caffeinated FPGAs 框架[231]，该框架是 CNN 框架 Caffe 的修改版，并带有 FPGA 支持，可实现 CNN 模型和专用的 FPGA 实现，并在必要时灵活地对设备进行重新编程. 文献[110]中提出了 FINN 框架，主要使用灵活的异构流架构的 FPGA 加速器，通过利用一组新颖的优化可以有效映射二值化神经网络.

大量计算和频繁的内存访问是神经网络在便携式系统上部署的挑战性问题. 现有高度适用于 FPGA 的综合工具（例如 HLS，OpenCL）大大减少了设计时间，硬件级设计（即 RTL）可以提高效率并实现更大的加速. 随着神经网络的不断发展，其在 FPGA 上的部署问题也将成研究者关注的重点. 这将进一步促使更多的针对 FPGA 上的神经网络部署的编译器以及框架的出现.

# 5 FPGA 深度神经网络的型号选择和性能度量分析

对于基于 FPGA 深度神经网络的研究，器件的选择和实验结果的分析评估是实验过程中不可忽视的两个部分. 其中，FPGA 的型号选择是实验顺利进行的先决条件，在此基础上对实验结果进行度量和分析，便可以从多个维度对实验进行分析和把控，从而得到较为全面的实验结果，有助于推进 FPGA 深度神经网络的研究进展.

## 5.1 FPGA型号选择

作为应用程序的硬件执行载体，FPGA 器件的选型至关重要. 合理的选型不仅可以避免设计问题，而且可以提高系统的性能、延长产品的生命周期、降低成本等. 进行 FPGA 选择时，需要根据应用的方向结合 FPGA 器件的特点进行考虑. 例如，从器件特色、规模大小、速度需求、功耗、成本、稳定性、安全性等方面进行选择. FPGA 芯片内部资源主要包括 IO 资源、时钟资源、逻辑资源、RAM 资源、DSP 资源、高速接口资源、硬核 IP 等，型号选择通常需要平衡各种资源需求并优先考虑关键资源的瓶颈. 例如，在深度神经网络计算加速设计中，通常需要优先考虑 DSP 数量、BRAM 数量以及所支持的外部存储最大带宽等. 在选择具体的芯片型号以及封装时，要根据实验的需求与具体情况做出选择. 一般来说，可以从芯片特点、规模大小、速度需求、功耗等几个方面做综合的考量. 具体来说，芯片特点应该关注所选器件的高速接口、通道及各个通道需要的最高收发速度等信息；规模大小应该考虑所选器件系列和 IP 的大致规模估计以及调试过程的资源消耗；速度需求应该根据所需功能进行选择；功耗则需要根据之前的设计、FPGA 供应商提供的功耗评估软件等估算将要消耗的功耗，从而确定所需的器件. 总的来说，在以器件特色为基准进行芯片选择时，需要综合考虑实验研究与 FPGA 芯片中的各种因素，以求达到最好的实验效果. 较为常用的 FPGA 厂家 Xilinx 与 Altera 均推出了不同系列的产品以供选择. 总的来说，企业开发选择时，Xilinx 占得比重相对较大，但 Altera 价格相较 Xilinx 更低. 而对于二者的特点，Xilinx 的短线资源非常丰富，因此在实现时布线成功率高. 而 Altera 的 FPGA 的短线资源经常不够用，需要经常占用逻辑单元来充当布线资源. 具体来说，Xilinx FPGA 产品主要分为 Virtex 系列、Spartan 系列、 Kintex 系列、Artix 系列、Zynq 系列. Altera FPGA 产品主要分为 Stratix 系列、Arria 系列、Cyclone 系列、FLEX-10K 系列等. 在 Altera FPGA 产品中 Cyclone 系列偏向于低成本应用，可以满足一般的逻辑设计要求. 而 Startix 系列更侧重于高性能应用，其性能可满足各类高端应用需求. 与此对应，在 Xilinx FPGA 产品中，Spartan 系列偏向低成本应用，Virtex 系列则侧重于高端应用. 因此，研究人员们可以根据实际需求，灵活的进行选择. 而在具体选择 FPGA 型号时，除了考虑 FPGA 生产厂家的特点之外，还可以从网络模型和应用方向两个角度来进行选择. 我们首先

总结列举了文献中常用的 FPGA 型号，然后就网络模型和应用方向两个角度对不同研究所应用的 FPGA 特点进行列举总结，供读者们参考. 具体如表 2 所示.

**表 2　FPGA 常用型号**

| 系列 | 型号 | 文献 |
|---|---|---|
| Virtex-2 | – | [96, 166, 245–247] |
| | XC2V8000 | [80] |
| | XC2V1000 | [165, 248, 249] |
| | XC2VP50 | [125] |
| | XC2VP30 | [136, 141] |
| Virtex-4 | – | [127] |
| | XCV400BG560-6 | [93] |
| | XC4VLX25-FF668-10 | [238, 250] |
| | XC4VLX160/200-FF1513 | [97, 125, 251] |
| | XC4VFX100 | [252] |
| | XCV400hq240 | [253] |
| Virtex-5 | – | [119, 254–256] |
| | XC5VLX110T | [257] |
| | XC5VFX200T | [258] |
| Virtex-6 | SX50T | [259] |
| | – | [156, 255] |
| Virtex-7 | XC7VX690T-2 | [231] |
| | VX490T | [260] |
| | VC709 | [159, 261] |
| | 485T | [114, 238, 262–265] |
| Spartan-2 | XC2S200 | [94, 266] |
| Spartan-3 | XC3S3000 | [132] |
| | XC3090s | [267] |
| Spartan-6 | – | [84, 242, 268] |
| Kintex-7 | XC7K325T | [269] |
| | 350-410t | [158] |
| Kintex-UltraScale | XCKU060 | [186] |
| Zynq | Zynq-7000 XC7Z045 | [137, 144, 260, 270] |
| | Zynq-7000 XC7Z020 | [88, 101, 157, 246] |
| | Zynq ZC702 | [271] |
| | Zynq ZC706 | [85, 110] |
| Stratix III | EP3S50F4842 | [95] |
| Stratix V | 5SGSD5 | [106, 271] |
| | 5SGXA7 | [83, 113, 114] |
| | 5SGSD8 | [113, 177, 206, 260] |
| | GSMD5 | [112] |
| Stratix 10 | – | [238, 272–276] |
| Arria 10 | GX1150 | [91, 106, 115, 116, 238, 271, 277, 278] |
| Cyclone II | Cyclone II 2C35 | [140, 279] |
| Cyclone V | AT6005 | [280, 281] |
| FLEX 10K | EPF1010LC84 | [282] |
| | EPF10K70RC240–4 | [165] |
| | FLEX10K20 | [283] |
| | EPF10K200SRC240-1 | [89] |

5.1.1　基于网络的 FPGA 型号选择

不同的应用及网络模型常常适合不同的 FPGA 型号. 目前基于 FPGA 的深度学习实验中，常用的网络模型有 AlexNet、VGG、GoogleNet、ResNet、RNN 等. 其中以 AlexNet 为实验模型时常选用 Kintex-7、Virtex-7、Stratix-V 系列；以 VGG 为实验模型时常选用 Virtex-7、Stratix-V、Kintex-7、Arria-10 系列；以 GoogleNet 为实验模型时常选用 Kintex-7、Arria-10 系列；以 ResNet 为实验模型时常选用 Arria-10、Kintex-7 系列；以 RNN 为实验模型时常选用 Kintex-7、Airtex-7、Arria-10、Virtex-6 系列. 具体系列选择如表 3 所示.

5.1.2　基于应用的 FPGA 型号选择

对于不同应用领域的深度学习实验，研究者们所选择的 FPGA 型号常常是不同的. 深度学习应用中常见的图像识别、目标跟踪、目标检测、语音识别等应用中常常使用 Virtex-5、Virtex-2、Virtex-4 系列的 FPGA 芯片，除此之外，在自然语言处理中，研究者们还使用了 Airtex-7 系列；在网络安全与入侵检测和电力应用中，Virtex-2 系列被用的最多. 每一种应用所常用的 FPGA 芯片系列具体如表 4 所示.

### 5.2　评估指标和度量分析

除了 FPGA 选型之外，如何衡量基于 FPGA 的深度神经网络的实验效果是另一个需要着重考虑的问题. 以实验常用的衡量指标为基础，基于 FPGA 的深度神经网络的性能衡量指标具体可从速率、能效、资源利用率、深度神经网络的性能和特定应用五个维度进行具体衡量. 其中，速率、能效、资源利用率更侧重于对 FPGA 的性能评估，而后两者则偏向于深度神经网络的性能的衡量. 可以看出，对基于 FPGA 的深度神经网络的实验效果进行衡量要多维度、全方面的考虑，这样才能更加准确的进行性能评估与度量分析. 下面将对其进行详细介绍.

### 5.3　基于速度的实验评估指标

在实验的结果评估中，速度是判断实验效果的重要标准之一. 论文中常用的速度评估方法有吞吐量和响应时间、CPU 执行时间、MIPS、MFLOPS、GFLOPS 和 TFLOPS 等. 例如，文献[106]采用了吞吐量作为图像分类硬件加速平台的评估指标；文献[238]将卷积吞吐量与总吞吐量作为平铺变量的设计空间探索、性能设计空间和性能模型验证实验的评估指标. 文献[113]将响应时间作为不同模型在 FPGA 和 CPU 上进行图像分类速度的评判标准之一. 文献[114]在验证所提网络的速度优势时，将不同网络模

**表 3 基于网络的 FPGA 型号选择**

| 系列 | VGG | AlexNet | GoogleNet | ResNet | SCNN | PCNN | RNN | SqueezeNet | SNN | MobileNet |
|---|---|---|---|---|---|---|---|---|---|---|
| Virtex-2 | | | | | [245] | [245] | | | [125] | |
| Virtex-7 | [231, 260] | [231,260, 284] | [231] | | | | | | | |
| Virtex-6 | | | | | | | [156] | | | |
| Stratix-V | [113, 271] | [113, 271] | | | | | | | | |
| Kintex-7 | [85, 137, 260] | [260] | [137] | | | | [285] | | | |
| Airtex-7 | | | | | | [101] | [286] | | | |
| Arria-10 | [116, 238, 277, 287] | | [238] | [238, 277] | | | [177] | [277] | | [277] |
| Cyclone V | | | | | | | | | [281] | |
| Virtex-4 | | | | | | | | | [125,250] | |
| Spartan-6 | | | | | | | | [84] | | |

**表 4 基于应用的 FPGA 型号选择**

| 应用分类 | Virtex-2 | Virtex-4 | Virtex-5 | Virtex-6 | Virtex-7 | Arria-10 | VC709 | Spartan 6 | DE2 Altera |
|---|---|---|---|---|---|---|---|---|---|
| 图像检测与识别 | | [125,127] | [119] | [130] | [121] | [91] | | | |
| 目标跟踪 | [189] | [288] | [289] | | | | | | |
| 文本处理 | | | | [156] | [160] | | [159] | | |
| 网络安全与人入侵检测 | [166,246] | | | | | | | | |
| 智能控制 | | | | | | [169] | | | [171] |
| 语音识别 | | | | [147] | [152] | [129,148] | [150] | [145] | |

型的 OpenCL 核的运行时间作为评价指标. 文献[156]在比较不同网络大小在不同反向传播时间下的性能时,将基于 RNN 的语言模型的运行时间作为评估指标. 文献[85,206]在对比其他 FPGA 加速器的实验中，将时钟周期 CLOCK 作为评估指标.

### 5.4 基于能效的实验评估指标

FPGA 具有高性能、低能耗、高并行、较强的灵活性等优点. 因此，能源效率是深度 FPGA 实验性能的另一个重要评估指标. 论文中常常采用 GOPS/W 的比值来评价在 1W 功耗的情况下，处理器的运算能力. 功率、带宽、能耗等常常作为具体的评估指标. 在不同 FPGA 加速平台进行比较时，将功率为比较标准之一，如文献[85,88,114,159]. 文献[238]在验证所设计 CNN 加速器性能时，采用了能耗比来对比软件实现与 FPGA 实现的性能. 文献[267]在比较三种 RRANN 实现结构的网络的性能时，以权重更新速度（WUPS）作为度量标准，在比较三种 RRANN 的实现结构的操作模式性能时以连接速度（CPS）作为度量标准. 文献[172]在比较 FPGA 与 GPU 的性能时采用功率和能量进行度量. 文献[156, 260]分别使用 GOPS 与 GOPS/J 对 FPGA 和 CPU 实现的计算能效进行度量.

### 5.5 基于资源利用率的实验评估指标

在 FPGA 的设计中，必须要充分了解各个芯片的内部资源利用情况，包括 FF、DSP、LUT、BRAM 等单元，在后续的实验设计中才能使得以上各个单元的利用达到平衡，并最大限度地发挥作用. 因此，FPGA 每一个单元的资源使用情况同样可以对实验效果进行评价. 论文中常用某一元器件已用资源与可用资源的比值计算资源利用率. 因此,FPGA 每一个单元的资源使用情况同样可以成为对实验效果的评估指标. 论文中常用某一元器件用的资源与可用的资源的比值作为资源利用率的计算方法. 文献 [88,91,101,114,206,231,238,253,255,260,271,278]等列举出 FPGA 的资源利用情况. 其中文献[238]列举不同数据类型下的资源占用比；文献[253]在有无 LUT 神经元的条件下对比了该论文所提出结构与传统神经网络架构的资源利用情况. 文献[135]列出了视频解码、颜色转换、分割和区域标记、计算位置和纵横比的 FPGA 资源利用率. 文献[101]列举了不同网络的资源利用情况；文献[271]列举出了每个基准 DNN 在不同 FPGA 平台上的资源利用率.

### 5.6 基于网络结构的实验评价指标

网络结构和数据集是深度学习必不可少的一部分. 对于深度 FPGA 实验来说，实验中常用网络结构有 VGG、AlexNet、GoogleNet、ResNet、NiN、LeNet、PCNN、RNN、SqueezeNet、SNN、MobileNet 等；实验常用数据集有 MINST、CIFAR-10、

ImageNet、Kaldi、GTSDB、Penn Treebank、TSUKUBA、Venus、Cones、Teddy、TIMIT、Speaker dependent TI46、INRIA、PPI、Reddit、Yelp、STL，LIDAR、PTSB、Sports-1M、Oxford Flowers 102、Birds-200 等. 深度神经网络由多层神经网络结构组成，网络层有卷积层、池化层、全连接层等. 不同层对实验结果都起着不同的作用. 因此，对每一层进行性能分析，可以得到其对实验性能的影响. 文献[85]分析了不同平台下的 VGG-16-SVD 网络各个层的性能，得出在带宽限制下，即使采用数据排列方式，系统在全连接层的性能也远低于卷积层的结论. 文献[91]分解了 VGG 不同层的时延，得出卷积层的计算时间占总延迟的 70%. 可以看到，不同层对实验结果有着不同的影响，因此对每一层进行实验评估对于深度 FPGA 实验非常必要.

### 5.7　基于应用的实验评价指标

基于 FPGA 的深度神经网络在各个方面均得到了应用. 对于不同的应用，其相应的评估指标也应该有所不同. 本节列举总结了三个应用方向的常用的实验评估指标.

（1）图像识别

图像识别的评估指标主要侧重于图像的处理效率与速度. 其中，文献[262]采用 images/sec 作为图像分类吞吐量的评估指标. 文献[89]分析了 VGG-16 处理一幅输入图像的所需时间. 文献[167]利用速度来评估基于 FPGA 的人脸检测系统性能. 文献[119]采用识别正确率作为评估指标. 文献[119]分析了姿态识别系统的响应速度.

（2）目标跟踪

目标跟踪的评估指标主要有分析跟踪过程中每一帧的处理时间. 文献[266]采用帧率（FPS）来评估自适应颜色直方图的视觉跟踪系统的性能. 文献[290]展示出了不同尺寸的图像分割区域数目与处理时间的关系图. 文献[291]对不同参数条件下，图像所检测到的特征点的大小和数量进行了展示，并计算所提出算法的执行时间.

（3）语音识别

语音识别的评估指标主要侧重于语音识别的正确率和识别时间. 其中，文献[165]采用测量的时间、观测数（观测序列大小）和时钟脉冲数来评估 FPGA-viterbi 实现和经典 viterbi 算法实现的单词识别性能；文献[292]采用字错率与句错率的比值（WER/CER）来评估系统的性能. 文献[249]在分析语音识别速度时，采用语音识别归一化的时间（ms）来进行衡量，并采用词向量 $\boldsymbol{W}$ 与声学观测特征序列 $O$ 的比值（$\boldsymbol{W}/O$）和 $\boldsymbol{W}$ 来评估两种语音识别系统的 FPGA 综合结果. 可以看到，从速率、能效、资源利用率、深度神经网络的性能和特定应用五个方面来评估深度 FPGA 实验效果，能够较为全面的对语音识别结果进行评价，得出准确的实验结论.

## 6　影响 FPGA 应用于深度神经网络的主要因素

（1）生态环境不完善

FPGA 生态系统主要包含芯片设计与生产、开发工具、IP 核、软件系统、销售、技术支持等，为其市场应用提供支持的一整套的产业链. 目前，FPGA 生态系统相对封闭，主要产业链集中把控在少数的几家公司中，且没有行业标准和通用规范，这使得 FPGA 学习和开发门槛高、投入周期长、开发工具使用受限、开源的优质 IP 比较缺乏、芯片价格昂贵等. 深度神经网络已被广泛应用于各大领域，如何基于 FPGA 来加速深度神经网络，提高深度神经网络性能显得尤为重要. 由于 FPGA 生态系统限制，其落地应用的速度无法跟上软件算法的开发速度，同时也限制了深度神经网络的进一步应用，成为限制 FPGA 应用于深度神经网络，乃至其自身发展的最大难题.

（2）编程语言门槛高

随着 FPGA 的不断发展，其复杂程度也不断增加，如何方便简单的对这些硬件资源进行编程成为影响 FPGA 能否更好的应用于深度神经网络模型的一个重要因素，传统的硬件描述语言包括 VHDL、verilog HDL、System Verilog 以及相对成熟的硬件 C 语言 System C、Handle-C 等，但对于芯片复杂程度不断增加的 FPGA 来说，其对编程人员的要求也越来越高，这极大的影响了 FPGA 应用的设计效率，也限制了其在深度神经网络上的应用. 在这种情况下，FPGA 的高层次综合（High-Level Synthesis）应运而生. 其可将 C++、C 等高层语言，通过特定编译工具直接转化成 FPGA 上可以运行的硬件代码. 虽然现在已经出现了包括 AutoPilot、OpenCL SDK 等成功的 FPGA HLS 工具，但对于完全取代人工硬件编程来说，HLS 仍有很长的路要走. 如何进行 HLS 的仿真调试、如何利用 HLS 提高系统的性能等等都是 HLS 所面临的问题. 在对 HLS 进行探索的过程中，文献[293]在基于 Python 的 FPGA 编程建模方面进行了探索，提出的 HeteroCL 可重构计算编程语言框架对算法设计师十分友好. 除此之外，文献

[294]在针对网络数据包领域方面，使用了高层语言“P4”构建网络算法和应用. 虽然 FPGA 的应用范围不断扩大，但目前其编程模型还是以硬件工程师进行 RTL 开发为主. 因此，将高层次语言描述的逻辑结构映射到硬件语言的 HLS，必将成为助力 FPGA 广泛应用的得力工具，将拥有十分广阔的发展前景.

（3）数据和模型的复杂化

随着智能化成为各行各业的发展需求，深度神经网络所处理的数据不断增加，模型不断复杂化，这使得以 FPGA 为载体的深度神经网络对硬件性能有了更高的要求. 这就需要我们对于 FPGA 的带宽、计算速度以及功耗等进行新的探究. FPGA 性能常常受到内存系统的吞吐量的限制，即当应用程序受到内存带宽限制时容易崩溃[295,296]，为了使 FPGA 能够搭载大数据以及复杂网络计算，我们必须增加 FPGA 带宽或者提高带宽利用率. 但是提高带宽往往成本较高，因此，如何以较低的成本扩展 FPGA 带宽或提高带宽利用率将成为 FPGA 的一大发展趋势. 除此之外，人工智能产品往往需要迅速的反应能力，这也就要求 FPGA 具有较快的计算速度. 目前，关于 FPGA 的计算加速研究中也出现了较多的硬件加速方法以及加速的网络框架. 但是人工智能不断发展对于速度的要求会越来越高，因此，如何加快大数据或者神经网络在 FPGA 上的计算速度依旧会成为 FPGA 的研究热点. 可以看到，人工智能产品的落地应用一定会驱使 FPGA 的设计会朝着低成本、高带宽、高速度的方向进行.

（4）自身结构设计瓶颈

随着科技的发展，现代电子设备从通信、娱乐等多个方面改善了人类生活质量. 在现代微电子技术中，通过不断缩小芯片上器件特征尺寸来提高计算能力，这才使得电子产品具有高速度、低成本以及低功耗等优势. 电子设备的性能增益不仅仅与搭载的新技术新算法有关，其与自身结构器件特征尺寸的缩小有关. 就人工智能系统而言，除了研究算法优化可以加速 FPGA 计算，FPGA 自身硬件结构设计相关的逻辑单元、内存、DSP 块、以及 I/O 的最大值的研究将成为其硬件加速需要突破瓶颈. 为保证具有计算引擎高密度、高速度、宽频带特性，FPGA 在一定的大小的片上需要更多的片上资源. 逻辑单元越多越好，集成的 DSP 块数越多，以及可用 I/O 口的数量越多，才可能使得 FPGA 具有更高的性能. 如今，硅 CMOS 技术正在接近其尺寸的基本物理极限，摩尔定律的延续性已经变得越来越具有挑战性. 那么，当算法优化达到一定程度时，FPGA 自身的这些设计指标将会成为其加速的最后的瓶颈.

（5）缺乏专业人才

随着大数据与人工智能技术的发展，FPGA 领域专业人才需求越来越大，人才培养的速度明显跟不上 FPGA 的发展需求的增多. 因此，FPGA 专业人才的培养成为一个亟待解决的问题，这也是产业生态建设及发展的需求. FPGA 专业人才培养现状为：人才储备少、后备力量不足，学习门槛高、内容复杂、学习周期长，培养体系不完备（师资力量匮乏、实验平台少、课程体系不完善、教学手段的单一、认证评估体系不足等），复合型创新人才培养难度大，人才培养是一个长期的积累的过程，不可一蹴而就. 完善培养体系、加大人才培养力度、加强专业技能培养、构建良好人才体系需要政府、企业、高校联动，共建 FPGA 产业生态.

## 参 考 文 献


[1] Warren S McCulloch, Walter Pitts. A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 1943, 5(4):115-133

[2] Hebb D O. The organization of behavior: A neuropsycholocigal theory. A Wiley Book in Clinical Psychology, 1949, 62: 78

[3] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 1958, 65(6): 386-408

[4] Minsky M, Papert S A. Perceptrons: An introduction to computational geometry. Cambridge, UK: MIT press, 2017

[5] McClelland J L, Rumelhart D E, PDP Research Group. Parallel distributed processing. Cambridge, UK: MIT press, 1986.

[6] Paul Werbos. Beyond regression: New tools for prediction and analysis in the behavioral sciences [Ph. D. dissertation]. Harvard University, USA, 1974

[7] Hinton G E. Boltzmann machine. Scholarpedia, 2007, 2(5): 1668-1675

[8] Fischer A, Igel C. An introduction to restricted Boltzmann machines//Iberoamerican Congress on Pattern Recognition. Berlin, Germany, 2012: 14-36

[9] Salakhutdinov R, Hinton G. Deep Boltzmann machines//Artificial Intelligence and Statistics. Beijing, China, 2009: 448-455

[10] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors. Nature, 1986, 323(6088): 533-536

[11] Pollack J B. Recursive distributed representations. Artificial Intelligence, 1990, 46(1-2): 77-105

[12] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. Science, 2006, 313(5786): 504-507

[13] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 2012, 25: 1097-1105

[14] Lin M, Chen Q, Yan S. Network in network. arXiv preprint

arXiv: 1312.4400, 2013

[15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. Going deeper with convolutions//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 1-9

[16] Karen Simonyan, Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556, 2014

[17] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna. Rethinking the inception architecture for computer vision//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 2818-2826

[18] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition//Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 770-778

[19] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 4700-4708

[20] Hu J, Shen L, Sun G. Squeeze-and-excitation networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA. 2018: 7132-7141

[21] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 1492-1500

[22] Zhang H, Wu C, Zhang Z, et al. Resnest: Split-attention networks. arXiv preprint arXiv:2004.08955, 2020

[23] Xiong Y, Liao R, Zhao H, et al. Upsnet: A unified panoptic segmentation network//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 8818-8826

[24] de Geus D, Meletis P, Dubbelman G. Fast panoptic segmentation network. IEEE Robotics and Automation Letters, 2020, 5(2): 1742-1749

[25] Liu H, Peng C, Yu C, et al. An end-to-end network for panoptic segmentation//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 6172-6181

[26] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv preprint arXiv:1506.01497, 2015

[27] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 779-788

[28] Joseph Redmon, Ali Farhadi. Yolo9000: better, faster, stronger//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017:7263-7271

[29] Joseph Redmon, Ali Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv: 1804.02767, 2018

[30] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector//European Conference on Computer Vision. Amsterdam, Netherlands, 2016: 21-37

[31] Mingxing Tan, Ruoming Pang, Quoc V Le. Efficientdet: Scalable and efficient object detection//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, USA, 2020: 10781-10790

[32] Tiancai Wang, Rao Muhammad Anwer, Hisham Cholakkal, Fahad Shahbaz Khan, Yanwei Pang, Ling Shao. Learning rich features at high-speed for single-shot object detection//Proceedings of the IEEE International Conference on Computer Vision. Long Beach, USA, 2019: 1971-1980

[33] Anfeng He, Chong Luo, Xinmei Tian, Wenjun Zeng. A twofold siamese network for real-time object tracking//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018:4834-4843

[34] Zizhao Zhang, Yuanpu Xie, Fuyong Xing, Mason McGough, Lin Yang. Mdnet: A semantically and visually interpretable medical image diagnosis network//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 6428-6436

[35] Hinton G E, Krizhevsky A, Wang S D. Transforming auto-encoders//International Conference on Artificial Neural Networks. Berlin, Germany, 2011: 44-51

[36] Zhang L, Edraki M, Qi G J. Cappronet: Deep feature learning via orthogonal projections onto capsule subspaces. arXiv preprint arXiv:1805.07621, 2018

[37] Sai Samarth R Phaye, Apoorva Sikka, Abhinav Dhall, Deepti Bathula. Dense and diverse capsule networks: Making the capsules learn better. arXiv preprint arXiv:1805.04001, 2018

[38] Shahroudnejad A, Afshar P, Plataniotis K N, et al. Improved explainability of capsule networks: Relevance path by agreement//2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP). California, USA, 2018: 549-553

[39] Sahu S K, Kumar P, Singh A P. Dynamic routing using inter capsule routing protocol between capsules//2018 UKSim-AMSS 20th International Conference on Computer Modelling and Simulation (UKSim). Kuantan, Malaysia, 2018: 1-5

[40] Sabour S, Frosst N, Hinton G E. Dynamic routing between capsules. arXiv preprint arXiv:1710.09829, 2017

[41] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller. Playing atari with deep reinforcement learning. arXiv preprint arXiv: 1312.5602, 2013

[42] Sangdoo Yun, Jongwon Choi, Youngjoon Yoo, Kimin Yun, Jin Young Choi. Action-decision networks for visual tracking with deep reinforcement learning//Proceedings of the IEEE conference on computer vision and pattern recognition. Hawaii, USA, 2017: 2711-2720

[43] Silver D, Schrittwieser J, Simonyan K, et al. Mastering the game of go without human knowledge. Nature, 2017, 550(7676): 354-359

[44] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. arXiv preprint arXiv:1911.08265, 2019

[45] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks. arXiv preprint arXiv:1406.2661, 2014

[46] Alec Radford, Luke Metz, Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015

[47] Karras T, Laine S, Aila T. A style-based generator architecture for generative adversarial networks//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 4401-4410

[48] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks//Proceedings of the IEEE International Conference on Computer Vision. Hawaii, USA, 2017: 5907-5915

[49] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks//Proceedings of the IEEE International Conference on Computer Vision. Hawaii, USA, 2017: 2223-2232

[50] Isola P, Zhu J Y, Zhou T, et al. Image-to-image translation with conditional adversarial networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA, 2017: 1125-1134

[51] Andrew Brock, Jeff Donahue, Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv: 1809.11096, 2018

[52] Yibing Song, Chao Ma, Xiaohe Wu, Lijun Gong, Linchao Bao, Wangmeng Zuo, Chunhua Shen, Rynson WH Lau, Ming-Hsuan Yang. Vital: Visual tracking via adversarial learning//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 8990-8999

[53] Gori M, Monfardini G, Scarselli F. A new model for learning in graph domains//Proceedings. 2005 IEEE International Joint Conference on Neural Networks. Montreal, Canada, 2005, 2: 729-734

[54] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model. IEEE Transactions on Neural Networks, 2008, 20(1): 61-80

[55] Thomas N Kipf, Max Welling. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv: 1609.02907, 2016.

[56] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio. Graph attention networks. arXiv preprint arXiv: 1710.10903, 2017

[57] Thomas N Kipf, Max Welling. Variational graph autoencoders. arXiv preprint arXiv:1611.07308, 2016

[58] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition//Proceedings of the AAAI Conference on Artificial Intelligence. Louisiana, USA, 2018, 32(1)

[59] You J, Liu B, Ying R, et al. Graph convolutional policy network for goal-directed molecular graph generation. arXiv preprint arXiv:1806.02473, 2018

[60] Nicola De Cao, Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. arXiv preprint arXiv: 1805.11973, 2018

[61] Do K, Tran T, Venkatesh S. Graph transformation policy network for chemical reaction prediction//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Nice, France. 2019: 750-760

[62] Wang H, Wang J, Wang J, et al. Graphgan: Graph representation learning with generative adversarial nets//Proceedings of the AAAI Conference on Artificial Intelligence. Louisiana, USA, 2018, 32(1): 2508-2515

[63] Ming Ding, Jie Tang, Jie Zhang. Semi-supervised learning on graphs with generative adversarial nets//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. Gold Coast, Australia, 2018: 913-922

[64] Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv: 1602.07360, 2016

[65] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 6848- 6856

[66] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, Quoc V Le. Learning transferable architectures for scalable image recognition//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA. 2018: 8697-8710

[67] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017

[68] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 4510-4520

[69] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3//Proceedings of the IEEE International Conference on Computer Vision. Long Beach, USA, 2019: 1314-1324

[70] Zoph B, Le Q V. Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578, 2016

[71] Baker B, Gupta O, Naik N, et al. Designing neural network architectures using reinforcement learning. arXiv preprint arXiv: 1611.02167, 2016

[72] Real E, Aggarwal A, Huang Y, et al. Regularized evolution for image classifier architecture search//Proceedings of the AAAI Conference on Artificial Intelligence. Hawaii, USA, 2019, 33(01): 4780-4789

[73] Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, Kevin Murphy. Progressive neural architecture search//Proceedings of the European Conference on Computer Vision (ECCV). Munich, Germany, 2018: 19-34

[74] Hieu Pham, Melody Y Guan, Barret Zoph, Quoc V Le, Jeff Dean. Efficient neural architecture search via parameter sharing. arXiv

preprint arXiv: 1802.03268, 2018

[75] He X, Zhao K, Chu X. AutoML: A Survey of the State-of-the-Art. Knowledge-Based Systems, 2019, 212: 106622-106659

[76] Wu R, Guo X, Du J, et al. Accelerating neural network inference on FPGA-based platforms—A survey. Electronics, 2021, 10(9): 1025-1050

[77] Mittal S. A survey of FPGA-based accelerators for convolutional neural networks. Neural Computing and Applications, 2020, 32(4): 1109-1139

[78] Teng Wang, Chao Wang, Xuehai Zhou, Huaping Chen. A survey of FPGA based deep learning accelerators: Challenges and opportunities. arXiv preprint arXiv:1901.04988, 2018

[79] Guo K, Zeng S, Yu J, et al. A survey of FPGA-based neural network inference accelerators. ACM Transactions on Reconfigurable Technology and Systems (TRETS), 2019, 12(1): 1-26

[80] Liu J, Liang D. A survey of FPGA-based hardware implementation of ANNs//2005 International Conference on Neural Networks and Brain. Beijing, China, 2005, 2: 915-918

[81] Cox C E, Blanz W E. GANGLION-a fast field-programmable gate array implementation of a connectionist classifier. IEEE Journal of Solid-State Circuits, 1992, 27(3): 288-299

[82] Jocelyn Cloutier, Eric Cosatto, Steven Pigeon, Francois R Boyer, Patrice Y Simard. Vip: An FPGA-based processor for image processing and neural networks//Proceedings of Fifth International Conference on Microelectronics for Neural Networks. Lausanne, Switzerland, 1996: 330-336

[83] Ma Y, Suda N, Cao Y, et al. ALAMO: FPGA acceleration of deep learning algorithms with a modularized RTL compiler. Integration, 2018, 62: 14-23

[84] Shi S. FusionAccel: A General Re-configurable Deep Learning Inference Accelerator on FPGA for Convolutional Neural Networks. arXiv preprint arXiv:1907.02217, 2019

[85] Jiantao Qiu, Jie Wang, Song Yao, Kaiyuan Guo, Boxun Li, Erjin Zhou, Jincheng Yu, Tianqi Tang, Ningyi Xu, Sen Song, et al. Going deeper with embedded fpga platform for convolutional neural network//Proceedings of the 2016 ACM/SIGDA International Symposium on FieldProgrammable Gate Arrays. Monterey, USA, 2016: 26-35

[86] Sangjun Yang, Heejun Shim, Wooseung Yang, ChongMin Kyung. A new RTL debugging methodology in FPGA-based verification platform//Proceedings of the 2004 IEEE Asia-Pacific Conference on Advanced System Integrated Circuits. Fukuoka, Japan, 2004: 180-183

[87] Yufei Ma, Yu Cao, Sarma Vrudhula, Jae-sun Seo. An automatic RTL compiler for high-throughput FPGA implementation of diverse deep convolutional neural networks//2017 27th International Conference on Field Programmable Logic and Applications (FPL). Ghent, Belgium, 2017: 1-8

[88] Ritchie Zhao, Weinan Song, Wentao Zhang, Tianwei Xing, Jeng-Hau Lin, Mani Srivastava, Rajesh Gupta, Zhiru Zhang. Accelerating binarized convolutional neural networks with software-programmable FPGAs//Proceedings of the 2017 ACM/SIGDA International Symposium on Field Programmable Gate Arrays. Monterey, USA, 2017: 15-24

[89] Yen C T, Lin Y T. FPGA realization of a neural-network-based nonlinear channel equalizer. IEEE Transactions on Industrial Electronics, 2004, 51(2): 472-479

[90] Ma Y, Kim M, Cao Y, et al. End-to-end scalable FPGA accelerator for deep residual networks//2017 IEEE International Symposium on Circuits and Systems (ISCAS). Baltimore, USA, 2017: 1-4

[91] Yufei Ma, Yu Cao, Sarma Vrudhula, Jae-sun Seo. Optimizing loop operation and dataflow in FPGA acceleration of deep convolutional neural networks//Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA, 2017: 45-54

[92] Chhedaiya N, Moyal V. Implementation of back propagation algorithm in Verilog. International Journal of Computer Technology Application, 2012, 3(1): 340-343

[93] Gadea R, Cerdá J, Ballester F, et al. Artificial neural network implementation on a single FPGA of a pipelined on-line backpropagation//Proceedings of the 13th International Symposium on System Synthesis. Madrid, Spain, 2000: 225-230

[94] Suhap Sahin, Yasar Becerikli, Suleyman Yazici. Neural network implementation in hardware using FPGAs//International Conference on Neural Information Processing. Bali, Indonesia, 2006: 1105-1112

[95] S Hariprasath, TN Prabakar. FPGA implementation of multilayer feed forward neural network architecture using VHDL//2012 International Conference on Computing, Communication and Applications. Dindigul, Tamilnadu, 2012: 1-6

[96] Messai A, Mellit A, Guessoum A, et al. Maximum power point tracking using a GA optimized fuzzy logic controller and its FPGA implementation. Solar Energy, 2011, 85(2): 265-277

[97] Jung Uk Cho, Seung Hun Jin, Xuan Dai Pham, Dongkyun Kim, Jae Wook Jeon. FPGA-based real-time visual tracking system using adaptive color histograms//2007 IEEE International Conference on Robotics and Biomimetics (ROBIO). Sanya, China, 2007: 172-177

[98] Taright Y, Hubin M. FPGA implementation of a multilayer perceptron neural network using VHDL//1998 Fourth International Conference on Signal Processing. Beijing, China, 1998, 2: 1311-1314

[99] Sameep Singh, Kuldip S Rattan. Implementation of a fuzzy logic controller on an FPGA using VHDL//22nd International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2003. Chicago, USA, 2003: 110-115

[100] Venieris S I, Bouganis C S. Latency-driven design for FPGA-based convolutional neural networks//2017 27th International Conference on Field Programmable Logic and Applications (FPL). Ghent, Belgium, 2017: 1-8

[101] Stylianos I Venieris, Christos-Savvas Bouganis. FpgaConvNet: A framework for mapping convolutional neural networks on FPGAs//2016 IEEE 24th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). Washington, USA, 2016: 40-47

[102] Nazanin Calagar, Stephen D Brown, Jason H Anderson. Source-level debugging for FPGA high-level synthesis//2014 24th International Conference on Field Programmable Logic and Applications (FPL). Munich, Germany, 2014: 1-8

[103] Muslim F B, Ma L, Roozmeh M, et al. Efficient FPGA implementation of OpenCL high-performance computing applications via high-level synthesis. IEEE Access, 2017, 5: 2747-2762.

[104] Guanwen Zhong, Alok Prakash, Yun Liang, Tulika Mitra, Smail Niar. Lin-analyzer: a high-level performance analysis tool for FPGA-based accelerators// 2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC). Austin, USA, 2016:1–6

[105] Murugan Sankaradas, Venkata Jakkula, Srihari Cadambi, Srimat Chakradhar, Igor Durdanovic, Eric Cosatto, Hans Peter Graf. A massively parallel coprocessor for convolutional neural networks//2009 20th IEEE International Conference on Application-specific Systems, Architectures and Processors. Boston, USA, 2009, 53-60

[106] Ovtcharov K, Ruwase O, Kim J Y, et al. Accelerating deep convolutional neural networks using specialized hardware. Microsoft Research Whitepaper, 2015, 2(11): 1-4

[107] Maurice Peemen, Arnaud AA Setio, Bart Mesman, Henk Corporaal. Memory-centric accelerator design for convolutional neural networks//2013 IEEE 31st International Conference on Computer Design (ICCD). Asheville, USA, 2013: 13-19

[108] Srihari Cadambi, Abhinandan Majumdar, Michela Becchi, Srimat Chakradhar, Hans Peter Graf. A programmable parallel accelerator for learning and classification//2010 19th International Conference on Parallel Architectures and Compilation Techniques (PACT). Vienna, Austria, 2010: 273-283

[109] Cheng J, Wang P, Li G, et al. Recent advances in efficient computation of deep convolutional neural networks. Frontiers of Information Technology & Electronic Engineering, 2018, 19(1): 64-77

[110] Yaman Umuroglu, Nicholas J Fraser, Giulio Gambardella, Michaela Blott, Philip Leong, Magnus Jahre, Kees Vissers. Finn: A framework for fast, scalable binarized neural network inference// Proceedings of the 2017 ACM/SIGDA International Symposium on Field Programmable Gate Arrays. Monterey, USA, 2017: 65-74

[111] Yongming Shen, Michael Ferdman, Peter Milder. Maximizing CNN accelerator efficiency through resource partitioning//2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA). Toronto, Canada, 2017: 535-547

[112] Yijin Guan, Hao Liang, Ningyi Xu, Wenqiang Wang, Shaoshuai Shi, Xi Chen, Guangyu Sun, Wei Zhang, Jason Cong. FP-DNN: An automated framework for mapping deep neural networks onto FPGAs with RTL-HLS hybrid templates//2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). Napa, USA, 2017: 152-159

[113] Naveen Suda, Vikas Chandra, Ganesh Dasika, Abinash Mohanty, Yufei Ma, Sarma Vrudhula, Jae-sun Seo, Yu Cao. Throughput-optimized OpenCL-based FPGA accelerator for large-scale convolutional neural networks//Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA, 2016: 16-25

[114] Wang D, An J, Xu K. PipeCNN: an OpenCL-based FPGA accelerator for large-scale convolution neuron networks. arXiv preprint arXiv:1611.02450, 2016

[115] Utku Aydonat, Shane O'Connell, Davor Capalija, Andrew C Ling, Gordon R Chiu. An opencl™ deep learning accelerator on arria 10//Proceedings of the 2017 ACM/SIGDA International Symposium on FieldProgrammable Gate Arrays. Monterey, USA, 2017: 55-64

[116] Jialiang Zhang, Jing Li. Improving the performance of opencl-based FPGA accelerator for convolutional neural network//Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA, 2017: 25-34

[117] Jack Yinger, Eriko Nurvitadhi, Davor Capalija, Andrew Ling, Debbie Marr, Srivatsan Krishnan, Duncan Moss, Suchit Subhaschandra. Customizable FPGA OpenCL matrix multiply design template for deep neural networks//2017 International Conference on Field Programmable Technology (ICFPT). Melbourne, Australia, 2017: 259-262

[118] Clément Farabet, Cyril Poulet, Jefferson Y Han, Yann LeCun. Cnp: An fpga-based processor for convolutional networks//2009 International Conference on Field Programmable Logic and Applications. Prague, Czech, 2009: 32-37

[119] Janarbek Matai, Ali Irturk, Ryan Kastner. Design and implementation of an fpga-based real-time face recognition system//2011 IEEE 19th Annual International Symposium on Field-Programmable Custom Computing Machines. Salt Lake City, Utah, 2011: 97-100

[120] Nikolaos Stekas, Dirk van den Heuvel. Face recognition using local binary patterns histograms (LBPH) on an FPGA-based system on chip (SoC)//2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). Chicago, USA, 2016:300–304

[121] M Tousif Ahmed, Sanjay Sinha. Design and development of efficient face recognition architecture using neural network on FPGA//2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS). Wuhan, China, 2018: 905-909

[122] Phan-Xuan H, Le-Tien T, Nguyen-Tan S. FPGA platform applied for facial expression recognition system using convolutional neural networks. Procedia Computer Science, 2019, 151: 651-658

[123] Zeeshan Ahmed Soomro, Tayab Din Memon, Falak Naz, Ahmed Ali. FPGA based real-time face authorization system for electronic voting system//2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET). East Sarajevo, Bosnia and Herzegovina, 2020: 1-6

[124] Al Koutayni M R, Rybalkin V, Malik J, et al. Real-time energy efficient hand pose estimation: A case study. Sensors, 2020, 20(10): 2828-2853

[125] Kenneth L Rice, Mohammad A Bhuiyan, Tarek M Taha, Christopher N Vutsinas, Melissa C Smith. FPGA implementation of izhikevich spiking neural networks for character recognition// 2009 International Conference on Reconfigurable Computing and FPGAs. Quintana Roo, Mexico, 2009: 451-456

[126] Lammie C, Hamilton T, Azghadi M R. Unsupervised character recognition with a simplified FPGA neuromorphic system// IEEE International Symposium on Circuits and Systems (ISCAS). Florence, Italy, 2018: 1-5

[127] Caner H, Gecim H S, Alkar A Z. Efficient embedded neural-net-

work-based license plate recognition system. IEEE Transactions on Vehicular Technology, 2008, 57(5): 2675-2683

[128] Yan Han, Erdal Oruklu. Real-time traffic sign recognition based on Zynq FPGA and ARM SoCs//IEEE International Conference on Electro/Information Technology. Milwaukee, USA, 2014: 373-376

[129] Wang Z, Xu K, Wu S, et al. Sparse-YOLO: Hardware/Software co-design of an FPGA accelerator for YOLOv2. IEEE Access, 2020, 8: 116569-116585

[130] Ma X, Najjar W A, Roy-Chowdhury A K. Evaluation and acceleration of high-throughput fixed-point object detection on FPGAs. IEEE Transactions on Circuits and Systems for Video Technology, 2014, 25(6): 1051-1062

[131] Bauer S, Köhler S, Doll K, et al. FPGA-GPU architecture for kernel SVM pedestrian detection//2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops. San Francisco, USA, 2010: 61-68

[132] Bauer S, Brunsmann U, Schlotterbeck-Macht S. FPGA implementation of a HOG-based pedestrian recognition system//Proceedings of the MPC-Workshop. Baden-Württemberg, Germany, 2009: 49-58

[133] Price A, Pyke J, Ashiri D, et al. Real time object detection for an unmanned aerial vehicle using an FPGA based vision system// Proceedings 2006 IEEE International Conference on Robotics and Automation. Orlando, USA, 2006: 2854-2859

[134] Matteo Poggi, Stefano Mattoccia. A wearable mobility aid for the visually impaired based on embedded 3d vision and deep learning//2016 IEEE Symposium on Computers and Communication (ISCC). Messina, Italy, 2016: 208-213

[135] Marzotto R, Zoratti P, Bagni D, et al. A real-time versatile roadway path extraction and tracking on an FPGA platform. Computer Vision and Image Understanding, 2010, 114(11): 1164-1179

[136] Kyrre Glette, Jim Torresen, Mats Hovin. Intermediate level FPGA reconfiguration for an online EHW pattern recognition system//2009 NASA/ESA Conference on Adaptive Hardware and Systems. San Francisco, USA, 2009: 19-26

[137] Eisaku Ohbuchi. Low power AI hardware platform for deep learning in edge computing//2018 IEEE CPMT Symposium Japan (ICSJ). Kyoto, Japan, 2018: 89-90

[138] Miguel Arias-Estrada, Eduardo Rodríguez-Palacios. An FPGA co-processor for real-time visual tracking//International Conference on Field Programmable Logic and Applications. Montpelier, France, 2002: 710-719

[139] Usman Ali, MB Malik, Khalid Munawar. FPGA/softprocessor based real-time object tracking system//2009 5th Southern Conference on Programmable Logic (SPL). Carlos, Brazil, 2009: 33-37

[140] Yuan-Pao Hsu, Hsiao-Chun Miao, Ching-Chih Tsai. FPGA implementation of a real-time image tracking system//Proceedings of SICE Annual Conference 2010. Taiwan, China, 2010: 2878-2884

[141] Jason Schlessman, Cheng-Yao Chen, Wayne Wolf, Burak Ozer, Kenji Fujino, Kazurou Itoh. Hardware/software co-design of an fpga-based embedded tracking system//2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06). New York, USA, 2006: 123-123

[142] Bingyi Zhang, Xin Li, Jun Han, Xiaoyang Zeng. Minitracker: a lightweight cnn-based system for visual object tracking on embedded device//2018 IEEE 23rd International Conference on Digital Signal Processing (DSP). Shanghai, China, 2018: 1-5

[143] Blanco-Filgueira B, Garcia-Lesta D, Fernández-Sanjurjo M, et al. Deep learning-based multiple object visual tracking on embedded system for iot and mobile edge computing applications. IEEE Internet of Things Journal, 2019, 6(3): 5423-5431

[144] Lee M, Hwang K, Park J, et al. FPGA-based low-power speech recognition with recurrent neural networks//2016 IEEE International Workshop on Signal Processing Systems (SiPS). Dallas, USA, 2016: 230-235

[145] Chen Che Wen, Wei-Xiang Liao, Ta-Wen Kuan, JhingFa Wang. Implementation of ASSR system based on HMM and syllable models on FPGA//2016 International Conference on Orange Technologies (ICOT). Melbourne, Australia, 2016: 68-71

[146] Lazaro Jr J B, Po M C P, Ramones L M, et al. Real-time speech recognition engine for accent correction using Hidden Markov Model//AIP Conference Proceedings. Ouarzazate, Morocco: AIP Publishing LLC, 2018, 2045(1): 020069-020076

[147] Wang Q, Li Y, Li P. Liquid state machine based pattern recognition on FPGA with firing-activity dependent power gating and approximate computing//2016 IEEE International Symposium on Circuits and Systems (ISCAS). Montreal, Canada, 2016: 361-364

[148] Yong Zheng, Haigang Yang, Zhihong Huang, Tianli Li, Yiping Jia. A high energy-efficiency FPGA-based lstm accelerator architecture design by structured pruning and normalized linear quantization//2019 International Conference on Field-Programmable Technology (ICFPT). Tianjin, China, 2019: 271-274

[149] Yiwei Zhang, Chao Wang, Lei Gong, Yuntao Lu, Fan Sun, Chongchong Xu, Xi Li, Xuehai Zhou. A powerefficient accelerator based on FPGAs for lstm network//2017 IEEE International Conference on Cluster Computing (CLUSTER). Honolulu, USA, 2017: 629-630

[150] Xu J, Li S, Jiang J, et al. A simplified speaker recognition system based on FPGA platform. IEEE Access, 2019, 8: 1507-1516

[151] Minwook Ahn, Seok Joong Hwang, Wonsub Kim, Seungrok Jung, Yeonbok Lee, Mookyoung Chung, Woohyung Lim, Youngjoon Kim. Aix: A high performance and energy efficient inference accelerator on FPGA for a DNN-based commercial speech recognition//2019 Design, Automation & Test in Europe Conference & Exhibition (DATE). Florence, Italy, 2019: 1495-1500

[152] Guy Maor, Xiaoming Zeng, Zhendong Wang, Yang Hu. An fpga implementation of stochastic computing-based lstm//2019 IEEE 37th International Conference on Computer Design (ICCD). Abu Dhabi, United Arab Emirates, 2019: 38-46

[153] Liao S, Samiee A, Deng C, et al. Compressing deep neural networks using toeplitz matrix: Algorithm design and fpga implementation//ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Brighton, UK, 2019: 1443-1447

[154] Nakayama M, Shigekawa N, Yokouchi T, et al. Frame-by-frame

speech signal processing and recognition for FPGA devices. Sensors for Everyday Life. Cham, 2017: 87-102

[155] Al-Shamma O, Fadhel M A, Hasan H S. Employing FPGA accelerator in real-time speaker identification systems. Recent Trends in Signal and Image Processing. Singapore, 2019: 125-134

[156] Li S, Wu C, Li H, et al. Fpga acceleration of recurrent neural network based language model//2015 IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines. Vancouver, Canada, 2015: 111-118

[157] Yufeng Hao, Steven Quigley. The implementation of a deep recurrent neural network language model on a Xilinx FPGA. arXiv preprint arXiv:1710.10296, 2017

[158] Kayode Sanni, Guillaume Garreau, Jamal Lottier Molin, Andreas G Andreou. FPGA implementation of a deep belief network architecture for character recognition using stochastic computation//2015 49th Annual Conference on Information Sciences and Systems (CISS). Baltimore, USA, 2015: 1-5

[159] Xiaofan Zhang, Xinheng Liu, Anand Ramachandran, Chuanhao Zhuge, Shibin Tang, Peng Ouyang, Zuofu Cheng, Kyle Rupnow, Deming Chen. High-performance video content recognition with long-term recurrent convolutional network for fpga//2017 27th International Conference on Field Programmable Logic and Applications (FPL). Ghent, Belgium, 2017: 1-4

[160] Wróbel K, Karwatowski M, Wielgosz M, et al. Compression of convolutional neural network for natural language processing. Computer Science, 2020, 21(1)

[161] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv: 1810.04805, 2018

[162] Bingbing Li, Santosh Pandey, Haowen Fang, Yanjun Lyv, Ji Li, Jieyang Chen, Mimi Xie, Lipeng Wan, Hang Liu, Caiwen Ding. Ftrans: Energy-efficient acceleration of transformers using fpga. arXiv preprint arXiv: 2007.08563, 2020

[163] Khaled Alrawashdeh, Carla Purdy. Reducing calculation requirements in FPGA implementation of deep learning algorithms for online anomaly intrusion detection//2017 IEEE National Aerospace and Electronics Conference (NAECON). Dayton, USA, 2017: 57-62

[164] Brad L Hutchings, Rob Franklin, Daniel Carver. Assisting network intrusion detection with reconfigurable hardware, in Proceedings//10th Annual IEEE Symposium on Field-Programmable Custom Computing Machines. Napa, USA, 2002: 111-120

[165] Das A, Nguyen D, Zambreno J, et al. An FPGA-based network intrusion detection architecture. IEEE Transactions on Information Forensics and Security, 2008, 3(1): 118-132

[166] Paulsson K, Hubner M, Jung M, et al. Methods for run-time failure recognition and recovery in dynamic and partial reconfigurable systems based on Xilinx Virtex-II Pro FPGAs//IEEE Computer Society Annual Symposium on Emerging VLSI Technologies and Architectures (ISVLSI'06). Karlsruhe, Germany, 2006: 6-12

[167] Le Q N, Jeon J W. Neural-network-based low-speed-damping controller for stepper motor with an FPGA. IEEE Transactions on Industrial Electronics, 2009, 57(9): 3167-3180

[168] Orlowska-Kowalska T, Kaminski M. FPGA implementation of the multilayer neural network for the speed estimation of the two-mass drive system. IEEE Transactions on Industrial Informatics, 2011, 7(3): 436-445

[169] Duncan JM Moss, Srivatsan Krishnan, Eriko Nurvitadhi, Piotr Ratuszniak, Chris Johnson, Jaewoong Sim, Asit Mishra, Debbie Marr, Suchit Subhaschandra, Philip HW Leong. A customizable matrix multiplication framework for the intel harpv2 xeon+ fpga platform: A deep learning case study//Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA, 2018: 107-116

[170] Chandrasekaran K, Mohanty M, Golla M, et al. Dynamic MPPT controller using cascade neural network for a wind power conversion system with energy management. IETE Journal of Research, 2020: 1-15

[171] Minh Tuan Le, Ngoc Tran Ta Thi, Minh Thanh Vo. Design and implementation of real time robot controlling system using upper human body motion detection on FPGA//2019 19th International Symposium on Communications and Information Technologies (ISCIT). Ho Chi Minh City, Vietnam, 2019: 215-220

[172] Wang C, Gong L, Yu Q, et al. DLAU: A scalable deep learning accelerator unit on FPGA. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 36(3): 513-517

[173] Ying Wang, Jie Xu, Yinhe Han, Huawei Li, Xiaowei Li. DeepBurning: automatic generation of FPGA-based learning accelerators for the neural network family//Proceedings of the 53rd Annual Design Automation Conference. Austin, USA, 2016: 110-116

[174] Kim L W. DeepX: Deep learning accelerator for restricted boltzmann machine artificial neural networks. IEEE Transactions on Neural Networks and Learning Systems, 2017, 29(5): 1441-1453

[175] Yaman Umuroglu, Lahiru Rasnayake, Magnus Själander. Bismo: A scalable bit-serial matrix multiplication overlay for reconfigurable computing//2018 28th International Conference on Field Programmable Logic and Applications (FPL). Dublin, Ireland, 2018: 307-3077

[176] Hardik Sharma, Jongse Park, Naveen Suda, Liangzhen Lai, Benson Chau, Vikas Chandra, Hadi Esmaeilzadeh. Bit fusion: Bit-level dynamically composable architecture for accelerating deep neural network//2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA). California, USA, 2018: 764-775

[177] Eriko Nurvitadhi, Jaewoong Sim, David Sheffield, Asit Mishra, Srivatsan Krishnan, Debbie Marr. Accelerating recurrent neural networks in analytics servers: Comparison of FPGA, CPU, GPU, and ASIC//2016 26th International Conference on Field Programmable Logic and Applications (FPL). Lausanne, Switzerland, 2016: 1-4

[178] Cheng Y, Wang D, Zhou P, et al. A survey of model compression and acceleration for deep neural networks. arXiv preprint arXiv:1710.09282, 2017

[179] Cheng J, Wang P, Li G, et al. Recent advances in efficient computation of deep convolutional neural networks. Frontiers of Information Technology & Electronic Engineering, 2018, 19(1): 64-77

[180] Han S, Mao H, Dally W J. Deep compression: Compressing deep

neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149, 2015

[181] Tomoya Fujii, Simpei Sato, Hiroki Nakahara, Masato Motomura. An FPGA realization of a deep convolutional neural network using a threshold neuron pruning//International Symposium on Applied Reconfigurable Computing. Delft, Netherlands, 2017: 268-280

[182] Wang S, Lin P, Hu R, et al. Acceleration of LSTM with structured pruning method on FPGA. IEEE Access, 2019, 7: 62930-62937

[183] LeCun Y, Denker J S, Solla S A, et al. Optimal brain damage// 3rd Annual Conference on Neural Information Processing Systems1989, NIPS 1989. Denver, USA, 1989, 2: 598-605

[184] Babak Hassibi, David G Stork. Second order derivatives for network pruning: Optimal brain surgeon. Advances in Neural Information Processing Systems. Morgan Kaufmann, 1993: 164-171

[185] Gao Huang, Shichen Liu, Laurens Van der Maaten, Kilian Q Weinberger. Condensenet: An efficient densenet using learned group convolutions//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 2752-2761

[186] Song Han, Junlong Kang, Huizi Mao, Yiming Hu, Xin Li, Yubin Li, Dongliang Xie, Hong Luo, Song Yao, Yu Wang, et al. Ese: Efficient speech recognition engine with sparse lstm on fpga// Proceedings of the 2017 ACM/SIGDA International Symposium on FieldProgrammable Gate Arrays. Monterey, USA, 2017: 75-84

[187] Mao H, Han S, Pool J, et al. Exploring the regularity of sparse structure in convolutional neural networks. arXiv preprint arXiv: 1705.08922, 2017

[188] Gong Y, Liu L, Yang M, et al. Compressing deep convolutional networks using vector quantization. arXiv preprint arXiv: 1412.6115, 2014

[189] Chen W, Wilson J, Tyree S, et al. Compressing neural networks with the hashing trick//International Conference on Machine Learning. Lille, France, 2015: 2285-2294

[190] Choi Y, El-Khamy M, Lee J. Towards the limit of network quantization. arXiv preprint arXiv:1612.01543, 2016

[191] Huffman D A. A method for the construction of minimum-redundancy codes. Proceedings of the IRE, 1952, 40(9): 1098-1101

[192] Oseledets I V. Tensor-train decomposition. SIAM Journal on Scientific Computing, 2011, 33(5): 2295-2317

[193] Nathan Srebro, Tommi Jaakkola, Weighted low-rank approximations//Proceedings of the 20th International Conference on Machine Learning (ICML-03). Oregon, USA, 2003: 720-727

[194] Zhao Z, Wang H, Sun H, et al. L1-norm low-rank linear approximation for accelerating deep neural networks. Neurocomputing, 2020, 400: 216-226

[195] Swaminathan S, Garg D, Kannan R, et al. Sparse low rank factorization for deep neural network compression. Neurocomputing, 2020, 398: 185-196

[196] Jaderberg M, Vedaldi A, Zisserman A. Speeding up convolutional neural networks with low rank expansions. arXiv preprint arXiv: 1405.3866, 2014

[197] Rouhani B D, Songhori E M, Mirhoseini A, et al. Ssketch: An automated framework for streaming sketch-based analysis of big data on fpga//2015 IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines. Vancouver, Canada, 2015: 187-194

[198] Shijie Zhou, Rajgopal Kannan, Viktor K Prasanna. Accelerating low rank matrix completion on fpga//2017 International Conference on ReConFigurable Computing and FPGAs (ReConFig). Cancun, Mexico, 2017: 1-7

[199] Han Q, Zeng L. FPGA Implementation for low-rank channel estimation of OFDM. Journal of Networks, 2012, 7(10): 1631

[200] Denton E, Zaremba W, Bruna J, et al. Exploiting linear structure within convolutional networks for efficient evaluation//28th Annual Conference on Neural Information Processing Systems 2014, NIPS 2014. Montreal, Canada, 2014: 1269-1277

[201] Courbariaux M, Bengio Y, David J P. BinaryConnect: training deep neural networks with binary weights during propagations// Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2. Montreal, Canada, 2015: 3123-3131

[202] Li F, Zhang B, Liu B. Ternary weight networks. arXiv preprint arXiv:1605.04711, 2016

[203] Zhu C, Han S, Mao H, et al. Trained ternary quantization. arXiv preprint arXiv:1612.01064, 2016

[204] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or -1. arXiv preprint arXiv: 1602.02830, 2016

[205] Rastegari M, Ordonez V, Redmon J, et al. Xnor-net: Imagenet classification using binary convolutional neural networks//European Conference on Computer Vision. Amsterdam, the Netherlands, 2016: 525-542

[206] Liang S, Yin S, Liu L, et al. FP-BNN: Binarized neural network on FPGA. Neurocomputing, 2018, 275: 1072-1086

[207] Sun Q, Shang F, Yang K, et al. Multi-precision quantized neural networks via encoding decomposition of {-1, + 1}//Proceedings of the AAAI Conference on Artificial Intelligence. Hawaii, USA, 2019, 33(01): 5024-5032

[208] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. arXiv preprint arXiv: 1606.06160, 2016

[209] Lin X, Zhao C, Pan W. Towards accurate binary convolutional neural network//Proceedings of the 31st International Conference on Neural Information Processing Systems. Vancouver, Canada, 2017: 344-352

[210] Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, Kailash Gopalakrishnan. Pact: Parameterized clipping activation for quantized neural networks. arXiv preprint arXiv: 1805.06085, 2018

[211] Daisuke Miyashita, Edward H Lee, Boris Murmann. Convolutional neural networks using logarithmic data representation. arXiv preprint arXiv: 1603.01025, 2016

[212] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, Yurong Chen. Incremental network quantization: Towards lossless cnns with

low-precision weights. arXiv preprint arXiv:1702.03044, 2017

[213] Li Y, Dong X, Wang W. Additive powers-of-two quantization: An efficient non-uniform discretization for neural networks. arXiv preprint arXiv:1909.13144, 2019

[214] Dongqing Zhang, Jiaolong Yang, Dongqiangzi Ye, Gang Hua. Lq-nets: Learned quantization for highly accurate and compact deep neural networks//Proceedings of the European Conference on Computer Vision (ECCV). Munich, Germany, 2018: 365-382

[215] Jung S, Son C, Lee S, et al. Learning to quantize deep networks by optimizing quantization intervals with task loss//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 4350-4359

[216] Qing Jin, Linjie Yang, Zhenyu Liao. Adabits: Neural network quantization with adaptive bit-widths//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, USA, 2020: 2146-2156

[217] Kuan Wang, Zhijian Liu, Yujun Lin, Ji Lin, Song Han. Haq: Hardware-aware automated quantization with mixed precision// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Angeles, USA, 2019: 8612-8620

[218] Gong C, Jiang Z, Wang D, et al. Mixed precision neural architecture search for energy efficient deep learning//2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). Westminster, UK, 2019: 1-7

[219] Bichen Wu, Yanghan Wang, Peizhao Zhang, Yuandong Tian, Peter Vajda, Kurt Keutzer. Mixed precision quantization of convnets via differentiable neural architecture search. arXiv preprint arXiv:1812.00090, 2018

[220] Jun L, Qijun H, Sheng C, et al. A FPGA implementation of integer discrete consine transform and quantification for H. 264 compression. Journal of Image and Graphics, 2011, 5

[221] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, Distilling the knowledge in a neural network. arXiv preprint arXiv: 1503.02531, 2015

[222] Zhenyang Wang, Zhidong Deng, Shiyao Wang. Accelerating convolutional neural networks with dominant convolutional kernel and knowledge pre-regression//European Conference on Computer Vision. Amsterdam, Netherlands, 2016: 533-548

[223] Liang Lu, Michelle Guo, Steve Renals. Knowledge distillation for small-footprint highway networks//2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). New Orleans, USA, 2017: 4820-4824

[224] Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, Matt Richardson. Do deep convolutional nets really need to be deep and convolutional?. arXiv preprint arXiv:1603.05691, 2016

[225] Crowley E J, Gray G, Storkey A. Moonshine: distilling with cheap convolutions//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montreal, Canada, 2018: 2893-2903

[226] Polino A, Pascanu R, Alistarh D. Model compression via distillation and quantization. arXiv preprint arXiv:1802.05668, 2018

[227] Mishra A, Marr D. Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. arXiv preprint arXiv:1711.05852, 2017

[228] Anil R, Pereyra G, Passos A, et al. Large scale distributed neural network training through online distillation. arXiv preprint arXiv:1804.03235, 2018

[229] Silverman H. An introduction to programming the Winograd Fourier transform algorithm (WFTA). IEEE Transactions on Acoustics, Speech, Signal Processing, 1977, 25(2): 152-165

[230] Andrew Lavin, Scott Gray. Fast algorithms for convolutional neural networks//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 4013-4021

[231] Roberto DiCecco, Griffin Lacey, Jasmina Vasiljevic, Paul Chow, Graham Taylor, Shawki Areibi. Caffeinated FPGAs: Fpga framework for convolutional neural networks//2016 International Conference on Field-Programmable Technology (FPT). Xi'an, China, 2016: 265-268

[232] Liqiang Lu, Yun Liang, Qingcheng Xiao, Shengen Yan. Evaluating fast algorithms for convolutional neural networks on FPGAs// 2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). Napa, USA, 2017: 101-108

[233] Liu X, Pool J, Han S, et al. Efficient sparse-winograd convolutional neural networks. arXiv preprint arXiv:1802.06367, 2018

[234] Junzhong Shen, You Huang, Zelong Wang, Yuran Qiao, Mei Wen, Chunyuan Zhang. Towards a uniform templatebased architecture for accelerating 2D and 3D CNNs on FPGA//Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA,2018: 97-106

[235] Liang Y, Lu L, Xiao Q, et al. Evaluating fast algorithms for convolutional neural networks on FPGAs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2019, 39(4): 857-870

[236] Liqiang Lu, Yun Liang. Spwa: an efficient sparse winograd convolutional neural networks accelerator on fpgas//2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC). Monterey, USA, 2018: 1-6

[237] Bacon D F, Graham S L, Sharp O J. Compiler transformations for high-performance computing. ACM Computing Surveys (CSUR), 1994, 26(4): 345-420

[238] Chen Zhang, Peng Li, Guangyu Sun, Yijin Guan, Bingjun Xiao, Jason Cong. Optimizing fpga-based accelerator design for deep convolutional neural networks//Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Monterey, USA, 2015: 161-170

[239] Yufei Ma, Naveen Suda, Yu Cao, Jae-sun Seo, Sarma Vrudhula. Scalable and modularized RTL compilation of convolutional neural networks onto FPGA//2016 26th International Conference on Field Programmable Logic and Applications (FPL). Lausanne, Switzerland, 2016: 1-8

[240] Chi P, Li S, Xu C, et al. Prime: A novel processing-in-memory architecture for neural network computation in reram-based main memory. ACM SIGARCH Computer Architecture News, 2016, 44(3): 27-39

[241] Guo K, Sui L, Qiu J, et al. Angel-eye: A complete design flow for mapping CNN onto embedded FPGA. IEEE Transactions on

Computer-Aided Design of Integrated Circuits and Systems, 2017, 37(1): 35-47

[242] Gaur N, Gupta A, Sharma A K, et al. HDL implementation of prepaid electricity billing machine on FPGA//2014 5th International Conference-Confluence The Next Generation Information Technology Summit (Confluence). Noida, India, 2014: 972-975

[243] Tu F, Yin S, Ouyang P, et al. Deep convolutional neural network architecture with reconfigurable computation patterns. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2017, 25(8): 2220-2233

[244] Charles F Van Loan, Gene H Golub, Matrix computations. Johns Baltimore, USA: Hopkins University Press, 1983

[245] Martin J Pearson, Chris Melhuish, Anthony G Pipe, Mokhtar Nibouche, L Gilhesphy, K Gurney, Benjamin Mitchinson. Design and FPGA implementation of an embedded real-time biologically plausible spiking neural network processor//International Conference on Field Programmable Logic and Applications. Tampere, Finland, 2005: 582-585

[246] JL Bastos, HP Figueroa, A Monti. FPGA implementation of neural network-based controllers for power electronics applications// Twenty-First Annual IEEE Applied Power Electronics Conference and Exposition, APEC'06. Dallas, USA, 2006: 1443-1448

[247] Ashraf R A, DeMara R F. Scalable FPGA refurbishment using netlist-driven evolutionary algorithms. IEEE Transactions on Computers, 2013, 62(8): 1526-1541

[248] dos Santos M P S, Ferreira J A F. Novel intelligent real-time position tracking system using FPGA and fuzzy logic. ISA Transactions, 2014, 53(2): 402-414

[249] Stefan Oniga, Alin Tisan, Daniel Mic, Attila Buchman, Andrei Vida-Ratiu. Hand postures recognition system using artificial neural networks implemented in FPGA// 2007 30th International Spring Seminar on Electronics Technology (ISSE). Cluj-Napoca, Romania, 2007: 507-512

[250] Schrauwen B, D'Haene M, Verstraeten D, et al. Compact hardware liquid state machines on FPGA for real-time speech recognition. Neural Networks, 2008, 21(2-3): 511-523

[251] Kok J, Gonzalez L F, Kelson N. FPGA implementation of an evolutionary algorithm for autonomous unmanned aerial vehicle on-board path planning. IEEE Transactions on Evolutionary Computation, 2012, 17(2): 272-281

[252] Tomasi M, Vanegas M, Barranco F, et al. Real-time architecture for a robust multi-scale stereo engine on FPGA. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2011, 20(12): 2208-2219

[253] Himavathi S, Anitha D, Muthuramalingam A. Feedforward neural network implementation in FPGA using layer multiplexing for effective resource utilization. IEEE Transactions on Neural Networks, 2007, 18(3): 880-888

[254] Ortega-Zamorano F, Jerez J M, Gómez I, et al. Layer multiplexing FPGA implementation for deep back-propagation learning. Integrated Computer-Aided Engineering, 2017, 24(2): 171-185

[255] Kim L W, Asaad S, Linsker R. A fully pipelined fpga architecture of a factored restricted boltzmann machine artificial neural network. ACM Transactions on Reconfigurable Technology and Systems (TRETS), 2014, 7(1): 1-23

[256] Michael Schaeferling, Gundolf Kiefer. Flex-surf: A flexible architecture for fpga-based robust feature extraction for optical tracking systems//2010 International Conference on Reconfigurable Computing and FPGAs. Cancun, Mexico, 2010: 458-463

[257] Ortega-Zamorano F, Jerez J M, Franco L. FPGA implementation of the c-mantec neural network constructive algorithm. IEEE Transactions on Industrial Informatics, 2013, 10(2): 1154-1161

[258] Michael Hahnle, Frerk Saxen, Matthias Hisung, Ulrich Brunsmann, Konrad Doll. FPGA-based real-time pedestrian detection on high-resolution images//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. Portland, USA, 2013: 629-635

[259] Jungwook Choi, Kisun You, Wonyong Sung. An FPGA implementation of speech recognition with weighted finite state transducers//2010 IEEE International Conference on Acoustics, Speech and Signal Processing. Dallas, USA, 2010: 1602-1605

[260] Chen Zhang, Di Wu, Jiayu Sun, Guangyu Sun, Guojie Luo, Jason Cong. Energy-efficient cnn implementation on a deeply pipelined fpga cluster//Proceedings of the 2016 International Symposium on Low Power Electronics and Design. San Francisco, USA, 2016: 326-331

[261] Huimin Li, Xitian Fan, Li Jiao, Wei Cao, Xuegong Zhou, Lingli Wang. A high performance FPGA-based accelerator for large-scale convolutional neural networks//2016 26th International Conference on Field Programmable Logic and Applications (FPL). Lausanne, Switzerland, 2016: 1-9

[262] Griffin Lacey, Graham W Taylor, Shawki Areibi. Deep learning on fpgas: Past, present, future. arXiv preprint arXiv:1602.04283, 2016

[263] Muthuramalingam A, Himavathi S, Srinivasan E. Neural network implementation using FPGA: issues and application. International Journal of Information Technology, 2008, 4(2): 86-92

[264] Jung-Woo Chang, Keon-Woo Kang, Suk-Ju Kang. SDCNN: An efficient sparse deconvolutional neural network accelerator on FPGA//2019 Design, Automation & Test in Europe Conference & Exhibition (DATE). Florence, Italy, 2019: 968-971

[265] Mário Véstias, Rui Policarpo Duarte, José T de Sousa, Horácio Neto. Parallel dot-products for deep learning on FPGA//2017 27th International Conference on Field Programmable Logic and Applications (FPL). Ghent, Belgium, 2017: 1-4

[266] Christopher T Johnston, Kim T Gribbon, Donald G Bailey. FPGA based remote object tracking for real-time control//International Conference on Sensing Technology. Palmerston North, New Zealand, 2005: 66-72

[267] James G Eldredge, Brad L Hutchings. Density enhancement of a neural network using FPGAs and run-time reconfiguration// Proceedings of IEEE Workshop on FPGA's for Custom Computing Machines. Napa Valley, USA, 1994: 180-188

[268] Neil D, Liu S C. Minitaur an event-driven FPGA-based spiking network accelerator. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2014, 22(12): 2621-2628

[269] Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, Pritish Narayanan. Deep learning with limited numerical precision// International Conference on Machine Learning. Lille, France, 2015: 1737-1746

[270] Jinhwan Park, Wonyong Sung. FPGA based implementation of deep neural networks using on-chip memory only//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Shanghai, China, 2016: 1011-1015

[271] Sharma H, Park J, Mahajan D, et al. From high-level deep neural models to FPGAs//2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO). Taipei, China, 2016: 1-12

[272] Ke He, Bo Liu, Yu Zhang, Andrew Ling, Dian Gu. Fecaffe: Fpga-enabled caffe with opencl for deep learning training and inference on intel stratix 10. arXiv preprint arXiv:1911.08905, 2019

[273] Tian Zhao, Yaqi Zhang, Kunle Olukotun. Serving recurrent neural networks efficiently with a spatial accelerator. arXiv preprint arXiv:1909.13654, 2019

[274] Carl Ebeling, Dana How, David Lewis, Herman Schmit. Stratix 10 high performance routable clock networks//Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Virtual Event, USA, 2016: 64-73

[275] David Lewis, Gordon Chiu, Jeffrey Chromczak, David Galloway, Ben Gamsa, Valavan Manohararajah, Ian Milton, Tim Vanderhoek, John Van Dyken. The stratix 10 highly pipelined fpga architecture//Proceedings of the 2016 ACM/SIGDA International Symposium on Field Programmable Gate Arrays. Virtual Event, USA, 2016: 159-168

[276] Andrew M Keller, Michael J Wirthlin. Singleevent characterization of a stratix 10 FPGA using neutron irradiation//2019 IEEE Radiation Effects Data Workshop. San Antonio, USA, 2019: 1-6

[277] Lin Z, Yih M, Ota J M, et al. Benchmarking deep learning frameworks and investigating FPGA deployment for traffic sign classification and detection. IEEE Transactions on Intelligent Vehicles, 2019, 4(3): 385-395

[278] Andrew Boutros, Sadegh Yazdanshenas, Vaughn Betz. Embracing diversity: Enhanced dsp blocks for low-precision deep learning on fpgas//2018 28th International Conference on Field Programmable Logic and Applications (FPL). Dublin, Ireland, 2018: 35-357

[279] Manikandan J, Venkataramani B, Avanthi V. FPGA implementation of support vector machine based isolated digit recognition system//2009 22nd International Conference on VLSI Design. New Delhi, India, 2009: 347-352

[280] Lysaght P, Stockwood J, Law J, et al. Artificial neural network implementation on a fine-grained FPGA//International Workshop on Field Programmable Logic and Applications. Berlin, Germany, 1994: 421-431

[281] Antara Ganguly, Rajeev Muralidhar, Virendra Singh. Towards energy efficient non-von neumann architectures for deep learning//20th International Symposium on Quality Electronic Design (ISQED). Santa Clara, USA, 2019: 335-342

[282] F Mohd-Yasin, AL Tan, MI Reaz. The FPGA prototyping of iris recognition for biometric identification employing neural network//Proceedings of the 16th International Conference on Microelectronics. Tunis, Tunisia, 2004: 458-461

[283] Fabian Luis Vargas, Rubem Dutra Ribeiro Fagundes, D Barros Júnior. A FPGA-based Viterbi algorithm implementation for speech recognition systems//2001 IEEE International Conference on Acoustics, Speech, Signal Processing. Salt Lake City, USA, 2001: 1217-1220

[284] Li H, Fan X, Jiao L, et al. A high performance FPGA-based accelerator for large-scale convolutional neural networks//2016 26th International Conference on Field Programmable Logic and Applications (FPL). Lausanne, Switzerland, 2016: 1-9

[285] Lee M, Hwang K, Park J, et al. FPGA-based low-power speech recognition with recurrent neural networks//2016 IEEE International Workshop on Signal Processing Systems (SiPS). Dallas, USA, 2016: 230-235

[286] Yufeng Hao, Steven Quigley. The implementation of a deep recurrent neural network language model on a Xilinx FPGA. arXiv preprint arXiv:1710.10296, 2017

[287] Ma Y, Cao Y, Vrudhula S, et al. Performance modeling for CNN inference accelerators on FPGA. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2019, 39(4): 843-856

[288] Cho J U, Jin S H, Dai Pham X, et al. FPGA-based real-time visual tracking system using adaptive color histograms//2007 IEEE International Conference on Robotics and Biomimetics (ROBIO). Sanya, China, 2007: 172-177

[289] K Yamaoka, Takashi Morimoto, Hidekazu Adachi, Tetsushi Koide, Hans Jürgen Mattausch. Image segmentation and pattern matching based FPGA/ASIC implementation architecture of real-time object tracking//Asia and South Pacific Conference on Design Automation. Yokohama, Japan, 2006: 176-181

[290] Schaeferling M, Kiefer G. Flex-SURF: A flexible architecture for FPGA-based robust feature extraction for optical tracking systems//2010 International Conference on Reconfigurable Computing and FPGAs. Cancun, Mexico, 2010: 458-463

[291] Paschalakis S, Bober M. A low cost FPGA system for high speed face detection and tracking//Proceedings 2003 IEEE International Conference on Field-Programmable Technology (FPT). Tokyo, Japan, 2003: 214-221

[292] Yu Shi, Timothy Tsui. An FPGA-based smart camera for gesture recognition in HCI applications//Asian Conference on Computer Vision. Tokyo, Japan, 2007: 718-727

[293] Yi-Hsiang Lai, Yuze Chi, Yuwei Hu, Jie Wang, Cody Hao Yu, Yuan Zhou, Jason Cong, Zhiru Zhang. Heterocl: A multi-paradigm programming infrastructure for softwaredefined reconfigurable computing//Proceedings of the 2019 ACM/SIGDA International Symposium on Field Programmable Gate Arrays. Virtual Event, USA, 2019: 242-251

[294] Stephen Ibanez, Gordon Brebner, Nick McKeown, Noa Zilberman. The p4-> netfpga workflow for line-rate packet processing// Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. Virtual Event, USA, 2019: 1-9

[295] Mikhail Asiatici, Paolo Ienne. Dynaburst: Dynamically assemblying dram bursts over a multitude of random accesses//2019 29th International Conference on Field Programmable Logic and Applications (FPL). Barcelona, Spain, 2019: 254-262

[296] Johan Peltenburg, Jeroen van Straten, Lars Wijtemans, Lars van Leeuwen, Zaid Al-Ars, Peter Hofstee. Fletcher: A framework to efficiently integrate fpga accelerators with apache arrow//2019 29th International Conference on Field Programmable Logic and Applications (FPL). Barcelona, Spain, 2019: 270-277

**JIAO Li-Cheng**, Ph.D., professor, Ph.D. supervisor. His research interests include intelligent perception and image understanding.

**SUN Qi-Gong**, Ph.D. candidate. His research interests include machine learning, computer vision and parallel computing.

**YANG Yu-Ting**, Ph.D. candidate. Her research interests include machine learning, computer vision and pattern recognition.

**FENG Yu-Xin**, master candidate. Her research interests include machine learning and computer vision.

**LI Xiu-Fang**, Ph.D. candidate. Her research interests include machine learning and computer vision.

## Background

After 70 years of development, the neural network has made a breakthrough in speech recognition, natural language processing, image understanding, video analysis and other applications. How to improve the efficiency of model training and forward computing becomes much more important. Because of its unique hardware characteristics, FPGA has become a research hotspot of the industry application. On the basis of previous research work, this paper systematically summarizes the development process, development mode, application direction, optimization strategy, chip selection, hardware circuit design and measurement evaluation of deep neural networks based on FPGA. We hope that readers can systematically understand the research status, implementation and research progress of deep neural network based on FPGA through this paper. This work was partially supported by the State Key Program of National Natural Science of China (No. 61836009), Project supported the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (No. 61621005), the National Natural Science Foundation of China (Nos. U1701267, 61871310), the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) (No. B07048), the Major Research Plan of the National Natural Science Foundation of China (Nos. 91438201), the Program for Cheung Kong Scholars and Innovative Research Team in University (No. IRT_15R53).