基于 DNN 卷积核分割的边缘协作推理性能分析

邸佳琳，滕颖蕾，张新阳，牛涛，宋梅
（北京邮电大学电子工程学院，北京 100876）

摘 要：随着智能芯片在边缘终端设备的普及，未来大量的 AI 应用将部署在更靠近数据源的网络边缘。基于 DNN 的分割方法可以实现深度学习模型在资源受限的终端设备上训练和部署，解决边缘 AI 算力瓶颈问题。在传统基于工作负载的分割方案（WPM, workload based partition method）的基础上，提出基于卷积核的分割方案（KPM, kernel based partition method），分别从计算量、内存占用、通信开销 3 个方面进行推理性能的定量分析，并从推理过程灵活性、鲁棒性、隐私性角度进行定性分析。最后搭建软硬件实验平台，使用 PyTorch 实现 AlexNet 和 VGG11 网络进一步验证所提方案在时延和能耗方面的性能优势，相比于传统工作负载分割方案，所提卷积核分割方案在大规模计算场景下有更好的 DNN 推理加速效果，且具有更低的内存占用和能量消耗。

关键词：边缘智能；深度神经网络分割；协作计算；并行推理
中图分类号：TN911. 22
文献标志码：A
doi: 10.11959/j.issn.2096−3750.2022.00308

Cooperative inference analysis based on DNN convolutional kernel partitioning

ZHI Jialin，TENG Yinglei，ZHANG Xinyang, NIU Tao，SONG Mei
School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: With the popularity of intelligent chip in the application of edge terminal devices, a large number of AI applications will be deployed on the edge of networks closer to data sources in the future. The method based on DNN partition can realize deep learning model training and deployment on resource-constrained terminal devices, and solve the bottleneck problem of edge AI computing ability. The kernel based partition method (KPM) was proposed as a new scheme on the basis of traditional workload based partition method (WPM). The quantitative analysis of inference performance was carried out from three aspects of computation FLOPS, memory consumption and communication cost respectively, and the qualitative analysis of the above two schemes was carried out from the perspective of flexibility, robustness and privacy of inference process. Finally, a software and hardware experimental platform was built, and AlexNet and VGG11 networks were implemented using PyTorch to further verify the performance advantages of the proposed scheme in terms of delay and energy consumption. It was concluded that, compared with the WPM scheme, the KPM scheme had better DNN reasoning acceleration effect in large-scale computing scenarios. And it has lower memory usage and energy consumption.

Key words: edge intelligence, deep neural network partition, cooperative computation, parallel partition

---

收稿日期：2022−06−15；**修回日期：**2022−11−07
通信作者：滕颖蕾，lilytengtt@bupt.edu. cn
基金项目：国家重点研发计划（No.2021YFB3300100），国家自然科学基金资助项目（No.62171062）
Foundation Items: The National Key Research and Development Program of China (No.2021YFB3300100), The National Natural Science Foundation of China (No.62171062)

0　引言

随着深度学习算法的突破和边缘智能[1]的普及，越来越多高服务质量、高服务精度和数据隐私安全的物联网设备被投入各种智能家居、智能工厂、智能城市等应用场景[2-6]，基于深度神经网络（DNN, deep neural network）模型的智能应用在边缘设备中逐渐普及[7]，将 DNN 类应用下沉到边缘设备的智能服务需求日益强烈[8]。然而基于 DNN 类的智能应用通常是计算密集型的，对设备 CPU、GPU、内存、网络等资源有高要求。基于 DNN 分割的协作推理机制[9]可以通过对 DNN 模型合理分割及卸载实现计算压力的分担，这是在资源有限的边缘设备上部署并执行 DNN 推理任务的方法之一。为此，Parthasarathy 等[10]提出一种基于模型分割的分布式边缘推理框架 DEFER，有效降低资源受限设备的能耗。Tian 等[9]提出了一种移动性层粒度的 DNN 分割卸载方案，实现了移动用户和边缘云的协同执行。Ren 等[11]提出一种能够进行细粒度分割的 DNN 弹性计算协作方案，将模型部署在多个终端设备以及云端，最大化协作推理。Xin 等[12]研究了多用户资源约束条件下的 DNN 分割卸载优化问题，选择合适的分割点在端边协同框架下执行 DNN 推理任务。Eshratifar 等[13]将 DNN 分割方案应用于移动设备与云服务器的协作，提出了一种计算卸载框架，在加快 DNN 任务执行的同时减小移动设备能耗。考虑到 DNN 任务不会只考虑一个卸载点，Eshratifar 等[14]提出了一个自适应的 JointDNN 框架，通过端云协作采用多分割点的方式为分布式 DNN 细粒度弹性划分提供了基础。然而传统研究都以卷积层为单位将 DNN 任务分割卸载到云端或边缘服务器端进行处理，每一层存在较大计算量。对此 Mohammed 等[15]提出了一种细粒度的自适应分割方案，通过将 DNN 卷积过程的特征图按行分割为不同子层，分配到雾计算[5]节点实现并行处理。Zhao 等[16]提出了面向 DNN 推理的自适应分布式计算框架 DeepThings，通过特征图分割卸载与融合的方式实现分布式并行推理。

已有对 DNN 细粒度分割方案的研究主要基于数据并行的方式，但该方案不能减少模型执行过程加载的内存参数量，并未减轻运行模型的设备内存压力。基于此，本文从模型并行的角度出发，提出新的基于卷积核分割方案，通过定量、定性分析与现有细粒度分割方案进行对比，并进一步搭建软硬件实验平台，验证所提方案对 DNN 推理的加速效果。

1　卷积核分割方案构建

1.1　卷积计算量和参数量建模

AlexNet 结构如图 1 所示，以该经典 CNN 模型 AlexNet[17]推理过程为例，AlexNet 模型由 5 个卷积层和 3 个全连接层构成，其中，卷积层负责数据特征的提取，全连接层负责结果的聚合和分类并输出结果。在资源受限的移动平台上，卷积层是深度神经网络所有层（卷积层、池化层、激活层、归一化层）的执行瓶颈[16]，因此本文仅讨论对卷积层的分割和量化建模。

在 DNN 推理过程中，推理的时延及能耗主要受计算量影响，而网络参数量直接决定模型运行时的内存占用。其中，卷积层通过一个和输入数据通道数相同的卷积核进行逐个通道卷积并求和，最后得出一个结果数值，在考虑偏置的情况下，一个卷积层处理输入数据时的计算量即每秒浮点操作数（FLOPS, floating-point operations per second）为[18]

$$\mathrm{FLOPS}=(C_{\mathrm{in}}\times 2\times K\times K-1)\times H_{\mathrm{out}}\times W_{\mathrm{out}}\times C_{\mathrm{out}} \quad (1)$$

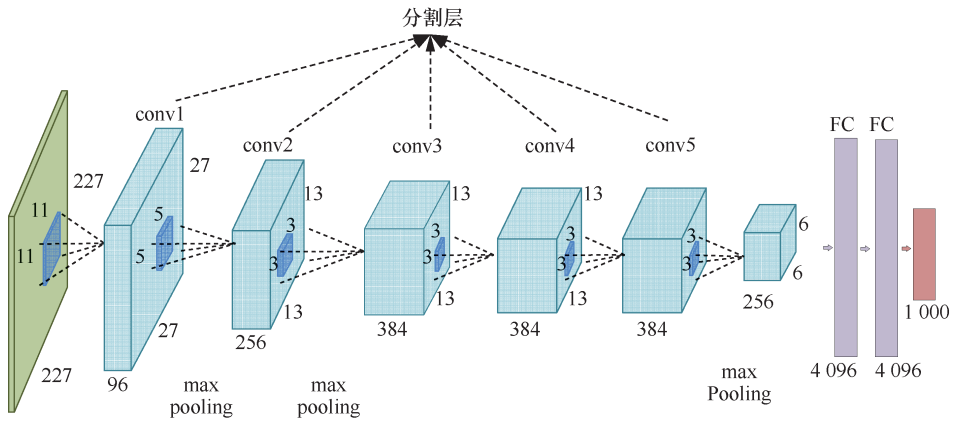


图 1　AlexNet 结构

根据 CNN 卷积属性可以得到特定卷积层的输入输出数据关系为

$$H_{\text{out}}=\frac{H_{\text{in}}+2P-K}{S}+1 \tag{2}$$

$$W_{\text{out}}=\frac{W_{\text{in}}+2P-K}{S}+1 \tag{3}$$

其中，$P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 和 $P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 分别为输入和输出特征图的通道数，$P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 为卷积核尺寸，$P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 和 $P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 分别为输入特征图的尺寸大小，$P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 和 $P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}$ 分别为输出特征图的尺寸大小，P 和 S 则分别表示卷积填充值和卷积步长大小。

卷积操作的参数量与输入通道数、卷积核个数以及卷积核尺寸有关，计算式如下。

$$P^{\text{para}}=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}} \tag{4}$$

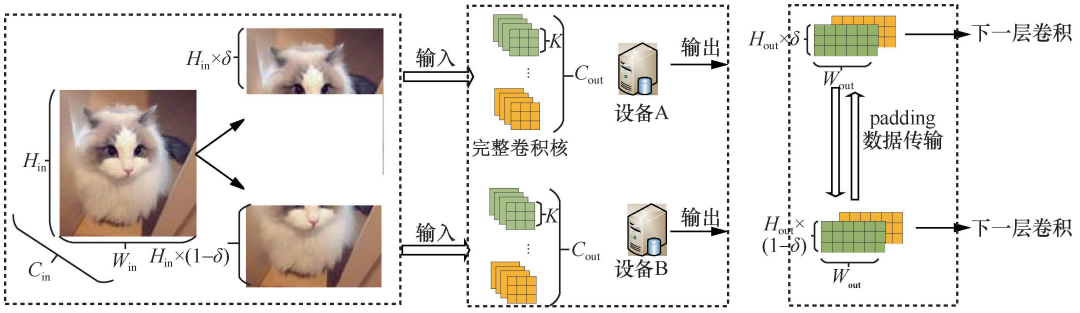
由式(1)和式(4)可知，卷积计算量与参数量均与卷积核个数成正比。由此可知，减少单设备计算的卷积核个数可以成比例减少设备计算量，基于此提出基于卷积核的分割卸载方案，下面对该方案进行详细介绍及性能分析。

1.2　卷积核分割卸载方案构建

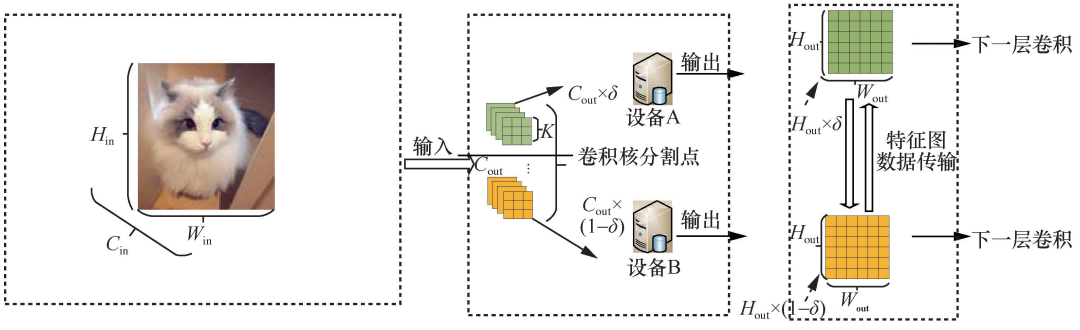
为了更好地体现卷积核分割方案对 DNN 推理的优化效果，本文与传统方案进行对比，分割方案如图 2 所示。

基于工作负载的分割方案（WPM, workload based partition method）[19]：为分担复杂 DNN 层推理的计算压力，将输入数据（工作负载）按行分割成多个子层，并卸载到多个移动设备上并行计算来最小化推理时延。为保证分割后卷积层输出的特征图都能进行完整的卷积操作，前一层卷积结束后，需要将卷积操作输出数据的边界行作为下一层卷积的 padding 发送到其邻近的节点。如图 2(a)所示，将原始工作负载按分割比例 $F_{\text{l}}=(C_{\text{in}}\times 2\times K\times K)\times\delta\times H_{\text{out}}\times W_{\text{out}}\times C_{\text{out}}$（$0\leqslant\delta\leqslant 1$）分割后，卸载到 A、B 两个协作设备并行执行，并在每层卷积结束通信 padding 数据。

基于卷积核的分割方案（KPM, kernel based partition method）：工作负载分割方案要求每个协作设备模型参数完整，造成在小设备部署时内存占用过大，基于神经网络中卷积运算独立性的原则和模型并行角度，提出 KPM，该方案将卷积层以卷积核为单位划分为多个子层，卸载到协作设备并行处理，每层卷积结束后，协作设备将各自推理运算得到的相应输出特征图通过多播的方式发送给其他



(a) 基于工作负载的分割方案



(b) 基于卷积核的分割方案

图 2　分割方案

协作设备进行数据整合，以保证在下一次卷积之前每个设备拥有完整的输入数据/特征图。图 2(b)给出了基于卷积核分割的两设备协同推理实例，首先将完整图片作为输入特征图分别发送到协作设备 A 和 B，协作设备并行处理部分卷积核运算，并生成对应颜色的特征图，卷积结束后进行数据共享和同步。

在 WPM 和 KPM 两种方案中，每层卷积都需要互相通信来支持并行计算，且分割和卸载过程可以按照每个设备的 CPU、内存能力以及网络通信环境有选择地进行。为进一步评估所提方案的优劣，接下来对比已有方案，分别从定量和定性两方面进行分析。

2　定量建模分析

为了综合考量两种分割方案在 DNN 模型分布式部署和高效执行方面的性能，下面首先从 DNN 模型计算复杂度、内存占用以及协作推理过程中的通信开销 3 个方面对两种细粒度分割方案进行定量性能分析对比。以经典 AlexNet 为例，设输入尺寸为 $3\text{ px}\times 227\text{ px}\times 227\text{ px}$ 的 RGB 图片，进行如下分析[1]。

2.1　计算量建模分析

针对图 2(a)中的 WPM，以分割比例 $F_1=(C_{\text{in}}\times 2\times K\times K)\times\delta\times H_{\text{out}}\times W_{\text{out}}\times C_{\text{out}}$ 按行分割，得到输入数据尺寸高度为 $\delta\times H_{\text{in}}$。为保证卷积操作的完整性，每层卷积结束每个设备会互相通信交界 padding，使得该设备输出特征图大小保持为分割前的 $F_1=(C_{\text{in}}\times 2\times K\times K)\times\delta\times H_{\text{out}}\times W_{\text{out}}\times C_{\text{out}}$ 比例关系。此时设备 A 输出特征图高度为 $\delta\times H_{\text{out}}$，根据式(1)，工作负载分割方案中设备 A 上卷积计算量（FLOPS）为

$$F_1=(C_{\text{in}}\times 2\times K\times K)\times\delta\times H_{\text{out}}\times W_{\text{out}}\times C_{\text{out}} \tag{5}$$

针对图 2(b)中的 KPM，同样以设备 A 为例，卷积核分割比例为 $F_2=(C_{\text{in}}\times 2\times K\times K)\times H_{\text{out}}\times W_{\text{out}}\times\delta\times C_{\text{out}}$，即在设备 A 上只执行 $\delta\times C_{\text{out}}$ 个卷积核，对应的计算量为

$$F_2=(C_{\text{in}}\times 2\times K\times K)\times H_{\text{out}}\times W_{\text{out}}\times\delta\times C_{\text{out}}=F_1 \tag{6}$$

由式(5)和式(6)可知，两种分割方案下单设备计算量是一致的，这是因为 WPM 减小了卷积输入数据尺寸，KPM 减少了卷积核个数，由于乘性关系，两者对卷积计算量的降低程度一致。

AlexNet 模型中两种分割方案和不进行分割时的计算量比较如图 3 所示，当两种细粒度分割方案使用 4 个设备协作时，相比较于无其他设备协作的不分割方案，两种方案通过分割卸载均可以成比例地降低单设备计算量。

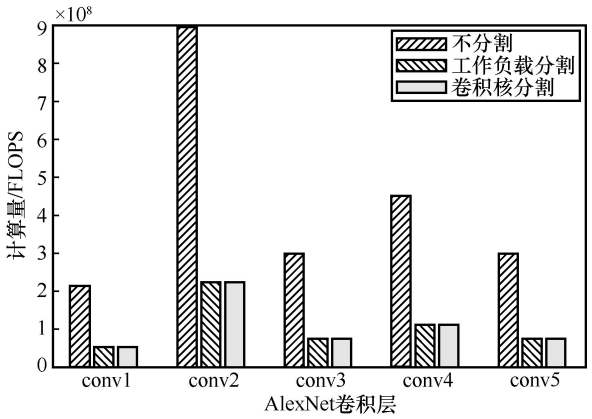


图 3　AlexNet 模型中两种分割方案及不分割方案的计算量比较

2.2　内存占用建模分析

在 DNN 推理过程中，需要先加载模型参数到内存再执行任务推理。期间模型参数量以及卷积过程中的输入输出数据会带来巨大的内存占用，使得复杂 DNN 模型难以直接部署到资源受限的物联网边缘设备上[6]。

由式(4)可知，两种分割方案下单设备参数量分别为

$$P_1=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}} \tag{7}$$

$$P_2=(C_{\text{in}}\times(K\times K)+1)\times C_{\text{out}}\times\delta \tag{8}$$

此外输入输出数据分别为

$$D_{\text{in}}=C_{\text{in}}\times H_{\text{in}}\times W_{\text{in}} \tag{9}$$

$$D_{\text{out}}=C_{\text{out}}\times H_{\text{out}}\times W_{\text{out}} \tag{10}$$

卷积过程内存占用主要受输入输出数据和加载的网络模型参数量影响，根据式(4)可得工作负载分割和卷积核分割方案下单设备的内存占用，分别为

$$M_1=P_1+(D_{\text{in}}+D_{\text{out}})\times\delta \tag{11}$$

$$M_2=P_2+D_{\text{in}}+D_{\text{out}}\times\delta \tag{12}$$

AlexNet 模型中两种分割方案内存占用情况如图 4 所示，可以看出，与工作负载分割方案相比，卷积核分割的方案可以有效降低内存占用，且

1 为简化问题，协作设备之间均采用等比分割以及向上取整的方式。

最低只有不分割方案的 26%。这是因为 WPM 单个设备模型参数量不随协作设备数改变，而 KPM 可以通过卷积核分割卸载成比例减少参数量。

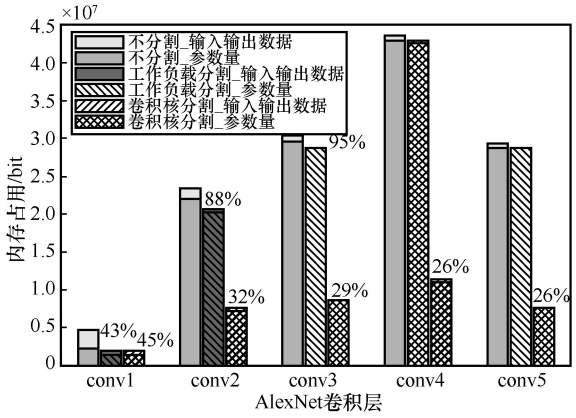


图 4 AlexNet 模型中两种分割方案内存占用情况

2.3 通信量建模分析

在每层卷积操作结束后，两种细粒度分割方案均需要进行数据同步后再进入下一层卷积。不同的是，WPM 每层卷积结束需要交换输出数据的边界行作为下一层卷积的 padding，KPM 则需要将完整的输出特征图通过多播的方式传输给其他设备进行数据整合。

WPM 中，当协作设备数大于 2 时，设备的输入特征图数据会分为边界数据和中间数据，两者对应的 padding 行数不同，工作负载分割输入数据示意图如图 5 所示，边界数据需要传输顶部或底部的边界 padding，而中间数据则需要同时传输顶部和底部的边界 padding，因此不失一般性，以中间数据为例求解 padding 的行数。

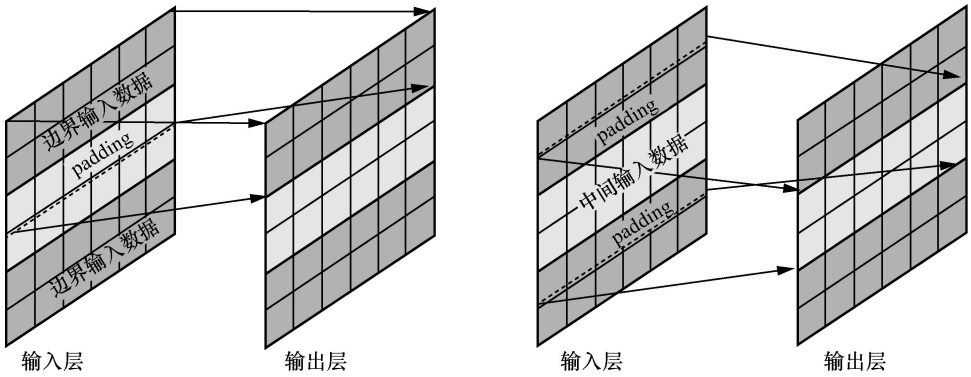


图 5 工作负载分割输入数据示意图

为保证特征图完整性，设备协作卷积的特征图应与单点卷积效果一致，即分割前卷积输出的特征图尺寸等于分割后各设备卷积特征图尺寸之和，故对于中间数据（中间特征图切片），考虑上下 padding 行数一致的情况有

$$\delta\left[\frac{H_{\text{in}}+2P-K}{S}+1\right]=\frac{\delta H_{\text{in}}+2P'-K}{S}+1 \quad (13)$$

$$p'=\left\lceil\frac{2\delta P+(1-\delta)(K-S)}{2}\right\rceil \quad (14)$$

其中，p'即为需要传输的 padding 行数。由于边界输入数据只需要对头部或尾部加 padding，故 p'相对中间输入数据要小。KPM 每层卷积结束需要通信的特征图数据量与设备所承担的计算卷积核个数成正比，即 $H_{\text{out}}\times W_{\text{out}}\times C_{\text{out}}\times\delta$；在此对比等比分割的情况下 KPM 方案的单设备通信量数据：在协作设备不超过 2 时，单设备通信量最大为 $p'\times W_{\text{out}}\times C_{\text{out}}$；协作设备大于 2 时，单设备通信量最大为 $2p'\times W_{\text{out}}\times C_{\text{out}}$。

两种分割方案下，系统单设备通信量随协作设备数变化情况如图 6 所示，其中 WPM 每个协作设备需要通信的 padding 数据随着协作设备数目变化保持不变。而 KPM 中单设备通信量随协作设备数增多成比例减少。在设备数大于 8 时，卷积核分割的通信量小于工作负载分割的方案，由此可知，基于卷积核的分割方案在大规模设备协作计算时具有更小的通信开销[2]。

综合以上分析，相比 WPM 方案，KPM 在同比降低设备计算量的同时，具有更少的内存占用，且在多设备协作时具有更小的通信开销。

2 对于 AlexNet 而言，conv3 和 conv4 参数设置相同，故 conv3 和 conv4 的通信量数据是重合的。

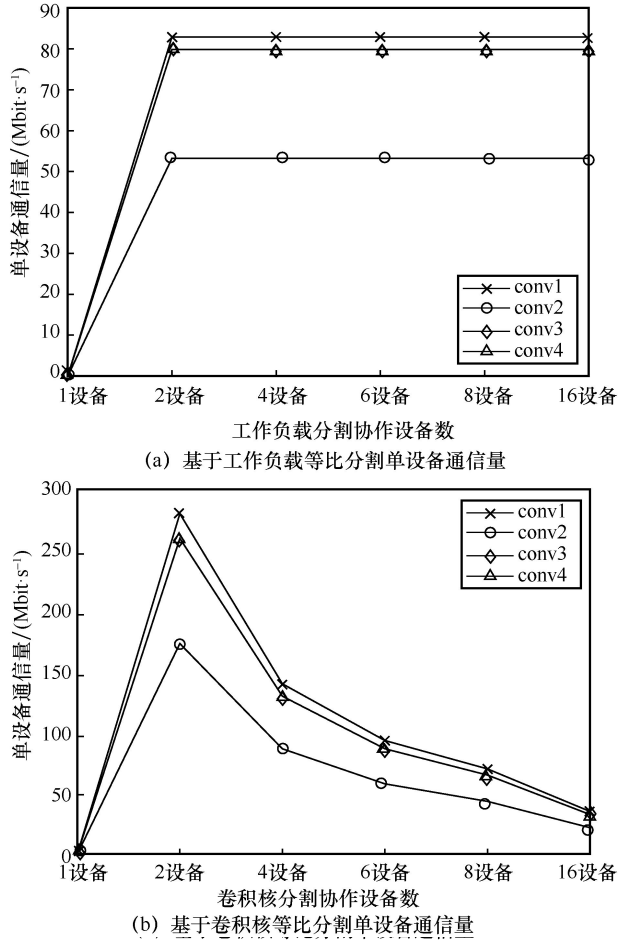


图 6　两种分割方案下，系统单设备通信量随协作设备数变化情况

3　定性理论分析

在实际推理过程中，计算框架灵活性、系统鲁棒性及数据隐私保护[20]是 DNN 分割卸载研究中需要重点考虑的因素。接下来对 WPM 和 KPM 两种方案相关方面性能进行对比分析。

灵活可变的计算框架可以更好地适应网络环境。基于工作负载分割的方案，其计算开销取决于最初的输入数据分割情况。但实际推理过程中，可用计算资源以及网络通信环境是时变的[21]，不同卷积层的计算量也不同，只在最初将输入数据分割一次会导致后续推理过程缺乏任务卸载分配的灵活性[22]；而 KPM 由于每层输入的都是完整数据，可以在计算过程中根据系统计算资源、网络条件以及任务计算量大小自适应调整分割方案，最大限度地利用网络资源，因此具有更大的卸载灵活性。此外，在 WPM 中，一般卷积过程的最小特征图不应小于卷积核尺寸[19]，只适合小数量设备的协作。而针对 KPM，每个卷积核之间的卷积运算是独立的，协作设备数量上限理论上为卷积核个数（如 AlexNet 的 conv2 有 256 个卷积核），可分割上限更高。故 KPM 相比于 WPM，具有更高的卸载和分割弹性。

系统鲁棒性直接影响任务响应效率。在协作推理过程中，节点移动性、电量限制等原因均会造成计算中断、数据传输丢包等问题[21]，尤其是卷积前期的数据丢失将直接影响推理结果的准确性。以图像识别任务为例，WPM 会将猫咪图片分割为上下两部分，如果具有猫咪头部特征的子负载在卷积前期发生数据丢失，则缺失了执行猫咪图像识别任务所需的关键信息（如脸部特征眼睛、耳朵等），会对后期的特征提取和全连接层的特征融合带来很大的误差。而在卷积核分割方案下，卷积前期主要提取基础特征，VGG11 网络第一层卷积后可视化的特征图如图 7 所示，第一层卷积结束生成 9 个可视化特征图，其中若发生特征图丢失其他特征图还是保留了关键信息（如眼睛、耳朵等特征），故 KPM 针对数据缺失情况具有较高的容错性，推理过程相对稳定。

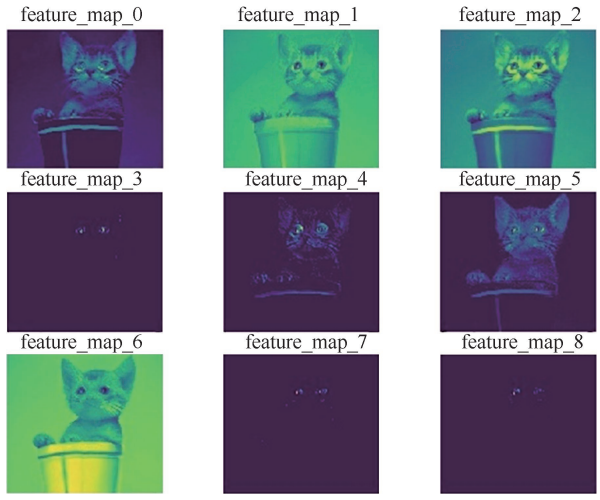


图 7　VGG11 网络第一层卷积后可视化的特征图

物联网设备产生、交换并处理大量安全和隐私敏感数据，易成为各种攻击的目标。在 DNN 协作推理时，DNN 模型需要通过无线网络将私有数据发送到协作设备或远程服务器，并将其公开给服务提供者。即使提供者是可信的，在提供者的通信通道上收到旁通道攻击时[23]，数据隐私机制仍然可能是脆弱的。如果对推理设备共享 DNN 模型，则参数可能由不可信方重构[24]。两种细粒度分割方案推理过程既需要共享数据，又需要共享 DNN 模型，均

可能产生隐私问题。对此，最近的一些研究[25-26]提出了一种密码协议，通过干扰电路或同态加密确保推理中的隐私性，这些加密协议依赖昂贵的密码原语，可能会导致相当高的计算和通信开销。与此同时在人群计算模型下通过选择可靠节点也可以降低隐私泄露风险[27]。

综合以上定量、定性分析，卷积核分割的方案可以通过增加协作设备有效降低单设备的内存占用和通信开销，且具有更高的计算灵活性和鲁棒性。接下来，通过搭建软硬件实验平台实测推理过程时延和能耗数据，进一步验证该方案对 DNN 推理的加速效果。

4 卷积核分割软硬件实验

WPM 已经被验证可以较好地加速 DNN 推理，为进一步验证 KPM 的效果，本文使用了一个软硬件测试平台来模拟该分割方案。在此选择 Raspberry Pi 4、AI 人工智能开发板 NVIDIA Jetson Xavier NX 以及桌面移动 PC 分别代表计算能力由弱到强的边缘物联网异构设备，共同协作推理 DNN 模型。其中，设备置于同一局域网下通过 Wi-Fi 互相连接。协作设备硬件参数见表 1。

表 1 协作设备硬件参数

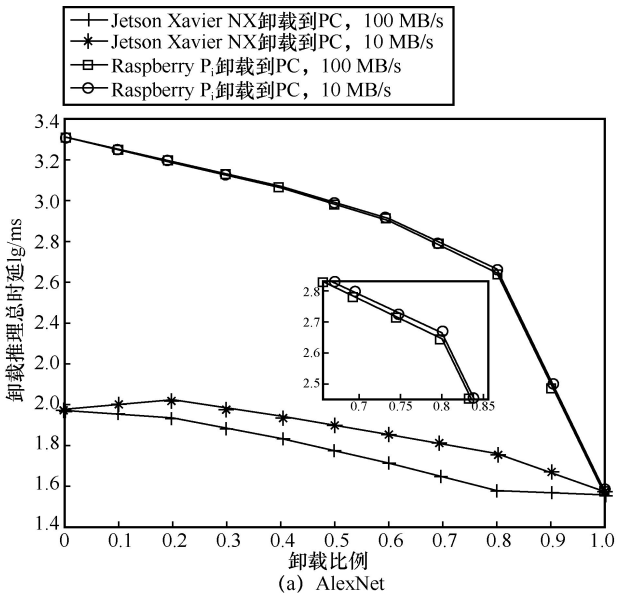
硬件	CPU	内存
移动 PC	i5-8265U CPU@ 1.6 GHz(8 CPUs) ～ 1.8 GHz	8 GB RAM
NVIDIA Jetson Xavier NX	2-core @ 1 900 MHz 4/6-core @ 1 400 MHz	8 GB LPDDR4 1 600 MHz
Raspberry Pi 4	4/6-core @ 600 MHz～1.5 GHz	4 GB LPDDR4 3 200 RAM

用 PyTorch 分别实现 AlexNet 和 VGG11 网络[3]，并用 CIFAR-10 数据集对模型预先训练。分别将 Raspberry Pi 4 和 NVIDIA Jetson Xavier NX 作为计算主体，将移动 PC 作为协作设备，向每个计算设备都输入尺寸大小为 3 px × 227 px × 227 px 的图片，将 DNN 模型以卷积核为单位分割卸载到 PC 端并行推理。此外，将设备之间无线通信速率分别固定为 100 MB/s 和 10 MB/s 以研究通信环境对计算卸载的影响。本文将推理过程中的能耗以及时延作为系统推理优化效果的重要参考依据，基于此测量完整推理过程的时延并计算推理过程中的能耗。不失一般性，推理时延为执行推理任务过程 100 次的平均时延。而在计算设备推理能耗时，考虑了通信能耗和计算能耗，并采用文献[19]中的通信和计算能耗模型理论计算，对于第 i 个设备，通信时延和能耗分别为 $T_i^{\mathrm{t}}=d_i^{\mathrm{t}}/B_i^{\mathrm{t}}$ 、$E_i^{\mathrm{t}}=T_i^{\mathrm{t}}\times p_i^{\mathrm{t}}$。其中，$d_i^{\mathrm{t}}$ 为设备需要传输的数据量，B_i^{t} 表示传输速率，p_i^{t} 为传输功率，在此取 $p_i^{\mathrm{t}}=0.8$ W；计算能耗采用 $E_i^{\mathrm{c}}=kf_i^3\times T_i^{\mathrm{c}}$ 计算，其中 k 为与计算机硬件有关的参数，在此取 $k=1\times10^{-24}$ [29]，f_i 为设备频率，设置为 1 GHz，T_i^{c} 为实测的设备推理时延。

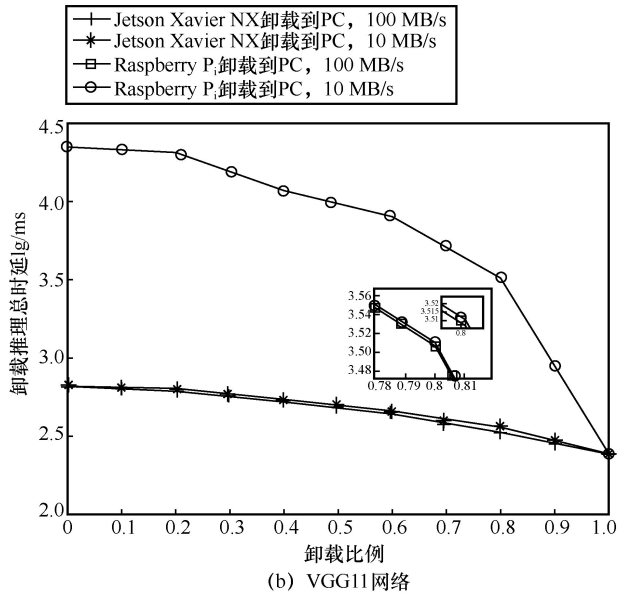
定义卸载率 $\delta=0.2$ 为从计算主体（Raspberry Pi 4 和 NVIDIA Jetson Xavier NX）卸载到 PC 的 DNN 卷积核个数占比。当 $\delta=0.2$ 表示在计算主体本地推理，$\delta=0.2$ 表示将所有工作负载卸载到移动 PC 上执行。不同卸载比例下的推理总时延如图 8 所示。由图 8(a)可知，在网络传输速率为 100 MB/s 时，将推理任务卸载到移动 PC 均可以减小推理时延。当 $\delta=0.2$ 时，AlexNet 在 NVIDIA Jetson Xavier NX 上的卸载推理时延只有不卸载时的 54.25%，在 Raspberry Pi 4 上的卸载推理时延则降到 38.82%。当网络传输速率为 10 MB/s，卸载比例 $\delta=0.2$ 时，将 AlexNet 从 NVIDIA Jetson Xavier NX 卸载到移动 PC 并不能带来推理加速效果，这是由于卸载增加额外的通信开销过大，说明对于计算能力强的设备，在网络状况较差时，适合本地计算。由图 8(b)可知，在不同的通信条件下，计算卸载均能给 VGG11 网络推理带来较好的加速效果。另外两个设备在 100 MB/s 和 10 MB/s 两种网络带宽下的两条曲线相比图 8(a)更紧凑，这是由于相对 AlexNet，VGG11 具有更大的计算复杂度，协作推理的通信开销对总时延影响较弱。

AlexNet 在两种分割方案下，随不同协作设备数目和网络传输速率的推理时延和设备平均能耗变化情况如图 9 所示（这里采用性能相近的移动 PC 协作，且采用平均分割的形式）。由于本实验采用的输入图片尺寸为 3 px × 227 px × 227 px，KPM 理论上每层卷积的协作设备数目上限为卷积核个数，AlexNet 模型中最大为 384。由图 9 可得以下几个结论。

3 本文主要讨论卷积层分割并行方案的相关问题，因此，本文优先选择经典网络 Alexnet 和 VGG 进行实验。而复杂网络结构（如 ResNet[28]等）的网络结构可能包含环路等，因此需要针对特定网络采用卷积等效等方法讨论。



(a) AlexNet



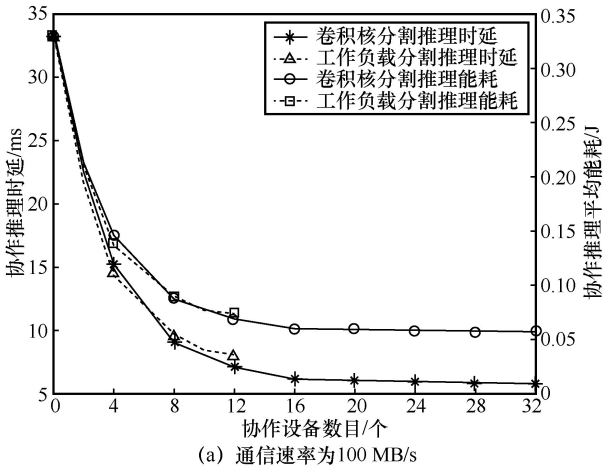
(b) VGG11网络

图 8　不同卸载比例下的推理总时延

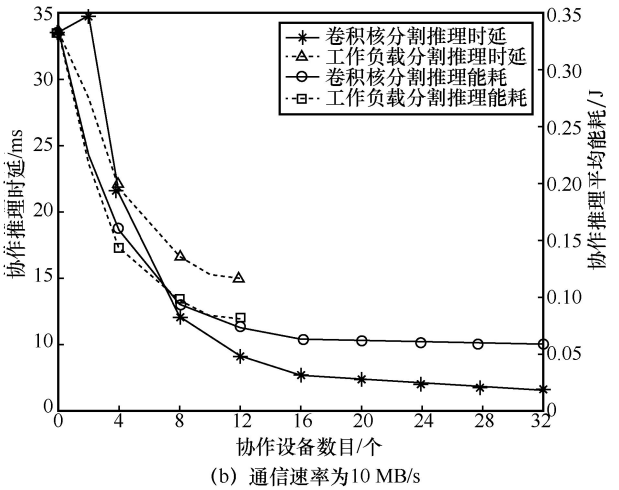
1) 在无线通信状态良好（速率为 100 MB/s）时，当设备数小于 8，两种方案的推理时延和单设备能耗均随着设备增多断崖式降低，且工作负载分割方案更低。设备数大于 8 时，KPM 分割方案对 DNN 推理的优化效果更好。这是因为 WPM 分割方案只需要传输 padding 数据，通信量更小，而随着协作设备增多，KPM 分割方案的单设备通信量成倍减少，在设备数大于 8 时通信开销比前者更小。

2) 当无线通信较差（速率为 10 MB/s），KPM 在两设备协作时，推理时延比不分割的方案大，这是由于此时卸载带来的额外通信开销不能抵消本地计算开销的减少。此外在通信较差时，两种方案对时延的优化效果差距更加明显，这是因为当计算量一致时，时延差距主要来源于通信量。这进一步说明，KPM 适合通信状况良好的大规模计算场景。

3) 当通信速率良好时（速率为 100 MB/s），KPM 和 WPM 的协作推理平均能耗差异较小，这是因为此时计算能耗占据主导。



(a) 通信速率为100 MB/s



(b) 通信速率为10 MB/s

图 9　AlexNet 在两种分割方案下，随不同协作设备数目和网络传输速率的推理时延和设备平均能耗变化情况[4]

5　结束语

在现有 DNN 细粒度工作负载分割的基础上，提出卷积核分割的新方案，为了验证该方案的效果，从定量（计算、内存占用、通信）和定性（灵活性、鲁棒性、隐私性）等方面进行分析，并搭建软硬件实验平台，用 PyTorch 框架实现 AlexNet 和 VGG11 网络，从时延和能耗上验证两种方案对

4 WPM 分割方案下协作设备数不能超过 AlexNet 卷积过程特征图最小尺寸 13 px×13 px，为了保证特征图分割单位（行）的完整性，图 9 中，WPM 方案对应的图像只展示到协作设备数目为 12。

DNN 推理的优化效果。结果表明，卷积核分割的方案在大规模计算场景下，具有更小的通信开销和内存占用，且在计算灵活性和推理鲁棒性上有更大的优势。本文的卷积核分割方案与传统方案的差异在数据分发和部署方案方面，并不会改变对卷积推理的逻辑，故卷积核分割的方案并不会对传统模型的准确率造成影响。后续将会在此基础上，进一步研究分割卸载决策和资源分配策略对推理优化效果的影响。

参考文献：

[1] ZHOU Z, CHEN X, LI E, et al. Edge intelligence: paving the last mile of artificial intelligence with edge computing[J]. Proceedings of the IEEE, 2019, 107(8): 1738-1762.

[2] DENG S G, ZHAO H L, FANG W J, et al. Edge intelligence: the confluence of edge computing and artificial intelligence[J]. IEEE Internet of Things Journal, 2020, 7(8): 7457-7469.

[3] ZHANG Y, MA X, ZHANG J, et al. Edge intelligence in the cognitive Internet of Things: improving sensitivity and interactivity[J]. IEEE Network, 2019, 33(3): 58-64.

[4] CEVALLOS MORENO J F, SATTLER R, CAULIER CISTERNA R P, et al. Online service function chain deployment for live-streaming in virtualized content delivery networks: a deep reinforcement learning approach[J]. Future Internet, 2021, 13(11): 278.

[5] CHIANG M, ZHANG T. Fog and IoT: an overview of research opportunities[J]. IEEE Internet of Things Journal, 2016, 3(6): 854-864.

[6] HUI H W, ZHOU C C, XU S G, et al. A novel secure data transmission scheme in industrial Internet of things[J]. China Communications, 2020, 17(1): 73-88.

[7] XU Z C, ZHAO L Q, LIANG W F, et al. Energy-aware inference offloading for DNN-driven applications in mobile edge clouds[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(4): 799-814.

[8] SUN Y, CUI Y N, HUANG Y H, et al. SDMP: a secure detector for epidemic disease file based on DNN[J]. Information Fusion, 2021, 68: 1-7.

[9] TIAN X Z, ZHU J, XU T, et al. Mobility-included DNN partition offloading from mobile devices to edge clouds[J]. Sensors (Basel, Switzerland), 2021, 21(1): 229.

[10] PARTHASARATHY A, KRISHNAMACHARI B. DEFER: distributed edge inference for deep neural networks[C]//2022 14th International Conference on COMmunication Systems & NETworkS (COMSNETS). Piscataway: IEEE Press, 2022: 749-753.

[11] REN P, QIAO X Q, HUANG Y K, et al. Fine-grained elastic partitioning for distributed DNN towards mobile web AR services in the 5G era[J]. IEEE Transactions on Services Computing, 2021, PP(99): 1.

[12] TANG X, CHEN X, ZENG L K, et al. Joint multiuser DNN partitioning and computational resource allocation for collaborative edge intelligence[J]. IEEE Internet of Things Journal, 2021, 8(12): 9511-9522.

[13] ESHRATIFAR A E, PEDRAM M. Energy and performance efficient computation offloading for deep neural networks in a mobile cloud computing environment[C]//GLSVLSI '18: Proceedings of the 2018 on Great Lakes Symposium on VLSI. [S.l.:s.n], 2018: 111-116.

[14] ESHRATIFAR A E, ABRISHAMI M S, PEDRAM M. JointDNN: an efficient training and inference engine for intelligent mobile cloud computing services[J]. IEEE Transactions on Mobile Computing, 2021, 20(2): 565-576.

[15] MOHAMMED T, JOE WONG C, BABBAR R, et al. Distributed inference acceleration with adaptive DNN partitioning and offloading[C]//IEEE INFOCOM 2020 - IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2020: 854-863.

[16] ZHAO Z R, BARIJOUGH K M, GERSTLAUER A. DeepThings: distributed adaptive deep learning inference on resource-constrained IoT edge clusters[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2018, 37(11): 2348-2359.

[17] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[18] MOLCHANOV P, TYREE S, KARRAS T, et al. Pruning convolutional neural networks for resource efficient transfer learning[EB]. 2016.

[19] ZENG L K, CHEN X, ZHOU Z, et al. CoEdge: cooperative DNN inference with adaptive workload partitioning over heterogeneous edge devices[J]. IEEE/ACM Transactions on Networking, 2021, 29(2): 595-608.

[20] SADEGHI A R, WACHSMANN C, WAIDNER M. Security and privacy challenges in industrial Internet of things[C]//Proceedings of the 52nd Annual Design Automation Conference. New York: ACM Press, 2015: 1-6.

[21] DIN N, CHEN H P, KHAN D. Mobility-aware resource allocation in multi-access edge computing using deep reinforcement learning[C]//Proceedings of 2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking. Piscataway: IEEE Press, 2019: 202-209.

[22] KILCIOGLU E, MIRGHASEMI H, STUPIA I, et al. An energy-efficient fine-grained deep neural network partitioning scheme for wireless collaborative fog computing[J]. IEEE Access, 2021: 79611-79627.

[23] KOCHER P, HORN J, FOGH A, et al. Spectre attacks: exploiting speculative execution[C]//Proceedings of 2019 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2019: 1-19.

[24] TRAMÈR F, ZHANG F, JUELS A, et al. Stealing machine learning models via prediction {APIs}[C]//25th USENIX Security Symposium (USENIX Security 16). [S.l.:s.n], 2016: 601-618.

[25] JUVEKAR C, VAIKUNTANATHAN V, CHANDRAKASAN A. {GAZELLE}: a low latency framework for secure neural network inference[C]//27th USENIX Security Symposium (USENIX Security 18). [S.l.:s.n], 2018: 1651-1669.

[26] GILAD-BACHRACH R, DOWLIN N, LAINE K, et al. Cryptonets: applying neural networks to encrypted data with high throughput and accuracy[C]//International conference on machine learning. New York: PMLR, 2016: 201-210.

[27] ZHAO L, TAN W A, XIE N, et al. An optimal service selection approach for service-oriented business collaboration using crowd-based cooperative computing[J]. Applied Soft Computing, 2020, 92: 106270.

[28] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for

image recognition[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2016: 770-778.

[29] WU Y H, WANG Y H, ZHOU F H, et al. Computation efficiency maximization in OFDMA-based mobile edge computing networks[J]. IEEE Communications Letters, 2020, 24(1): 159-163.

[作者简介]

郅佳琳（1999− ），女，北京邮电大学硕士生，主要研究方向为边缘计算、深度学习等。

滕颖蕾（1983− ），女，博士，北京邮电大学教授、博士生导师，IEEE 高级会员。主要研究方向为 AI 与无线通信，边缘计算及毫米波技术等。

张新阳（2000− ），男，北京邮电大学硕士生，主要研究方向为边缘智能、资源分配等。

牛涛（1997− ），男，北京邮电大学博士生，主要研究方向为边缘计算、人工智能等。

宋梅（1960− ），女，北京邮电大学电子工程学院教授、博士生导师，中国电子教育学会研究生教育分会常务理事，中国电子学会通信分会委员，中国电子学会物联网专家委员会委员，中国铁道学会信息化委员会委员，主要研究方向为数据与服务、通信与管理等。