

KCPNet: 张量分解的轻量卷积模块设计、部署与应用

王鼎衡¹, 赵广社¹, 姚满¹, 李国齐²

(1. 西安交通大学自动化科学与工程学院, 710049, 西安; 2. 清华大学精密仪器系, 100084, 北京)

摘要: 为解决现有卷积模块在实际应用中内存消耗高、计算效率低的问题,在 Kronecker CANDECOMP/PARAFAC(KCP)张量分解的基础上,提出一种轻量、高效、瓶颈结构的卷积模块(KCPNet)。对普通卷积作 2 阶 KCP 分解,生成的因子张量分别映射为两层负责输入输出通道变化的 1×1 卷积和两层负责特征提取的变通道可分离卷积,再将这 4 层卷积组成含有瓶颈结构的 KCPNet 卷积模块。基于 OpenCL 并行编程框架将 KCPNet 部署于嵌入式 GPU,并围绕 pico-flexx 深度相机开发了动态手势识别应用。实验结果表明:在 ImageNet 大规模标准数据集上,相比 ResNet、ResNeXt 等已有的张量分解卷积模块,KCPNet 在准确率相近的情况下能够兼顾空间和计算复杂度的效率;在中等规模标准数据集 CIFAR-10 上,KCPNet 能够在无明显精度损失的前提下将传统的 VGG 模型压缩至原先的 16.1%并节约 75.5%的计算量;在面向嵌入式 GPU 时,并行部署的 KCPNet 可使 CIFAR-10 的识别速度达到 100 帧/s。以 KCPNet 为核心开发的手势识别应用程序可达到 99.5%的准确率和 100 帧/s 以上的运行速度,内存开销为 22 MB。

关键词: 张量分解;Kronecker CANDECOMP/PARAFAC 张量分解;轻量卷积模块;并行部署;手势识别

中图分类号: TP274+.5 文献标志码: A

DOI: 10.7652/xjtuxb202203014 文章编号: 0253-987X(2022)03-0135-12



OSID 码

KCPNet: Design, Deployment, and Application of Tensor-Decomposed Lightweight Convolutional Module

WANG Dingheng¹, ZHAO Guangshe¹, YAO Man¹, LI Guoqi²

(1. School of Automation Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China;
2. Department of Precision Instrumentation, Tsinghua University, Beijing 100084, China)

Abstract: To deal with the issues of high memory consumption and low computation efficiency of the existing convolutional modules in practical application, following the Kronecker CANDECOMP/PARAFAC (KCP) tensor decomposition, a lightweight, efficient, bottleneck-structured convolutional module (KCPNet) is proposed. The normal convolution is firstly decomposed as the 2nd-order KCP, the generated factor tensors are mapped into two 1×1 convolutions to carry out the transformation of input and output channels, and into two depthwise convolutions with changeable channels to perform the feature extraction, respectively. Then, these 4 convolutions are assembled into the KCPNet module with a bottleneck structure. The KCPNet is deployed in the embedded GPU based on the OpenCL parallel programming

framework, and the dynamic gesture recognition application is developed around a pico-flexx depth camera. The experimental results show that compared with the existing convolutional modules of tensor decomposition such as ResNet, ResNeXt, etc., the KCPNet can consider the efficiency of both space and computation complexity with a similar accuracy on the large-scale benchmark dataset ImageNet; the KCPNet can also compress the traditional VGG model to its original 16.1% and save 75.5% of computation without obvious accuracy loss on the medium-scale benchmark dataset CIFAR-10; the parallelly deployed KCPNet can achieve the recognition rate of 100 frames/s for CIFAR-10 when orienting to the embedded GPU. The developed gesture recognition application with the KCPNet as its core can achieve 99.5% accuracy and the running rate beyond 100 frames/s, and the memory cost is 22 MB.

Keywords: tensor decomposition; Kronecker CANDECOMP/PARAFAC tensor decomposition; lightweight convolutional module; parallel deployment; gesture recognition

卷积神经网络(CNN)是近年来重要的深度学习模型之一^[1],卷积模块出众的特征提取能力使CNN得到了广泛的研究和应用。为了获取更好的数据识别与处理能力,CNN一直在向着深度更深、规模更大的方向发展,但在面向实际应用环境时,却不得不解决资源受限条件下的CNN模型部署问题。因此,对深度神经网络模型,尤其是深度CNN模型的压缩,成为了目前深度学习领域最为热点的研究方向之一^[2-3]。在多种模型压缩方法中,模型裁剪、知识迁移、精细模块设计等方法^[3]一般需要复杂的训练-压缩-再训练过程^[4-5]或高训练成本的网络结构搜索^[6-7]。张量分解凭借其坚实的数学理论基础^[8],具有方便实现、压缩率可观等独特优势^[9-10],还可以根据张量分解的具体形式直接设计轻量化的CNN模型并训练,完成后即可部署而无需进一步调整或压缩^[11-12]。

张量分解往往利用其低秩情况下的信息保留特性,以小规模的多个因子张量来估计原权重矩阵的表达能力,一般称为“估计方式”^[13]。以图 1a 的权重矩阵 W 为例,若其可分解为 4 个因子张量相乘的形式,即 $W = G_1 G_2 G_3 G_4$,则这些因子张量即可替代原矩阵作为压缩模型的权重,如图 1b 所示。然而,估计方式对卷积模块的压缩效果十分有限,虽然能取得可观的压缩率,但精度损失难以避免^[10,14-17],目前只有对 3 维卷积的压缩效果相对较好^[16-17]。

实际上,张量分解在神经网络压缩中还存在另一种所谓“映射方式”^[13],其只借鉴张量分解的数据结构,并将输入按顺序与各个因子张量相计算的过程视为普通神经网络之中的输入数据依次通过不同权重层次的过程^[13]。考虑到神经网络的每层权重之后都存在非线性激活函数,以图 1c 为例,映射方

式可简单表示为 $f(f(f(f(\underline{XG}_1)\underline{G}_2)\underline{G}_3)\underline{G}_4))$,其中, $\underline{X}$ 为神经网络的输入数据, $f(\cdot)$ 指非线性激活函数。由于引入了更多的非线性激活函数,张量分解对神经网络模型带来的总体精度损失在一定程度上被更强的表达能力弥补。此外,映射方式能够从张量数据结构的角解释并发展一些神经网络模型中的精细模块设计。Zhang 等最早从张量分解角度将卷积重构为两个更小的连续卷积^[18];Lebedev 等和 Astrid 等利用 CANDECOMP/PARAFAC (CP) 分解将卷积重构成以可分离卷积为核心的瓶颈卷积模块^[19-20];Kim 等基于 Tucker-2 分解将卷积重新设计为普通的瓶颈卷积模块^[21-22];Chen 等将 BT 张量分解解释为分组的瓶颈卷积模块^[23-24]。

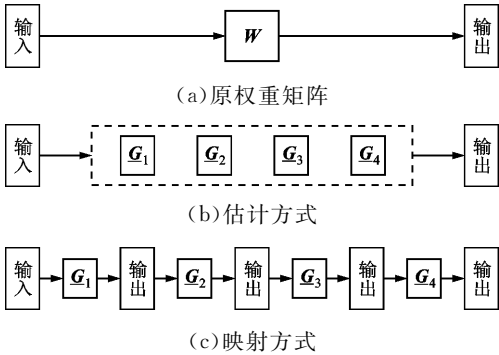


图 1 张量分解压缩神经网络的两种形式
Fig. 1 Compressing neural networks by tensor decomposition in two formats

更重要的是,这些以映射方式得到的卷积模块帮助研究者们在应用方面取得了大量有实际意义的进展,尤其是以 MobileNet 模型^[25]为代表的可分离卷积和以 IGCV 模型^[26]为代表的分组卷积这两种典型卷积模块,几乎已经成为 CNN 应用领域的标准件。例如,基于可分离卷积,文献[4]结合裁剪方

法实现了实时目标跟踪,文献[13]实现了高效的情感识别,文献[27]综合多级特征完成了高效的实例分割任务;围绕分组卷积,周云成等和杨贤志等分别开发了农作物实时识别和轻量级目标检测应用^[28-29]。

然而,这些卷积模块虽然比原始卷积在理论上有更低的复杂度,但在面向部署应用时仍然存在一些实际效率方面的工程问题。例如: CUDA 的官方库 cuDNN 直到第7版才对可分离卷积作出了有限的优化,其实际效率仍较差,文献[4]辅以裁剪方法才取得了较好的运行效率;文献[24]提到分组的瓶颈卷积在理论上虽有着明显的并行计算优势,但在 cuDNN 等通用计算框架上的实际效率较差,同时更多的特征图处理使其内存消耗反高于普通卷积。并且,除少数张量分解估计方式的应用^[11-12]之外,其他应用方面的研究^[4,13,25-29]对如何将张量分解映射方式得到的轻量卷积模块与具体硬件结合部署着墨甚少。

为了解决这些卷积模块在实际部署应用中内存消耗高、计算效率低的问题,本文在深入研究 Kronecker CP(KCP)张量分解^[30]的基础上,将 KCP 映射为一种以变通道可分离卷积为核心的瓶颈卷积模块(KCPNet)。分析了 KCPNet 的多种表现形式可对应于一些其他现存的张量分解卷积模块。在 ImageNet 大规模数据集上的实验表明,KCPNet 相比其他瓶颈卷积模块能够同时达到较高效的空间和计算复杂度。基于 CIFAR-10 标准图像分类数据集的实验验证了 KCPNet 对普通卷积具有良好的压缩效果。结合 GPU 内存架构的特点给出了将 KCPNet 部署于嵌入式 GPU 的具体方法。结合 KCPNet 与深度相机开发了动态手势识别应用,测试结果表明,在消耗 22 MB 内存的前提下,可实现超过 100 帧/s 的识别效率。

1 KCPNet 轻量卷积模块设计

卷积窗口尺寸为 $w \times h$ 的普通卷积一般可表示为 4 阶张量 $\underline{C} \in \mathbb{R}^{w \times h \times S \times T}$, S 和 T 分别对应输入和输出特征图的通道数。相应的 KCPNet 轻量卷积模块设计思路为:首先,借助 KCP 对 $\underline{C}$ 进行分解;然后,将得到的因子张量重新组合为 4 个新的 4 阶张量;最后,令这 4 个张量映射为 1×1 卷积与可分离卷积组合而成的 KCPNet 轻量卷积模块,并给出 KCPNet 的计算过程。

1.1 KCP 分解

对于任意 d 阶张量 $\underline{W} \in \mathbb{R}^{l_1 \times l_2 \times \dots \times l_d}$, 如果存在

正整数因数分解, $l_i = m_i n_i (i \in \{1, 2, \dots, d\})$, 那么同样存在 K 对因子张量 $\underline{A} \in \mathbb{R}^{m_1 \times m_2 \times \dots \times m_d}$ 和 $\underline{B} \in \mathbb{R}^{n_1 \times n_2 \times \dots \times n_d} (k \in \{1, 2, \dots, K\})$, 使得 $\underline{W}$ 可写为^[30]

$$\underline{W} = \sum_{k=1}^K \underline{A}_k \otimes \underline{B}_k \quad (1)$$

式中 $\otimes$ 表示 Kronecker 积。式(1)即为张量的 Kronecker 分解。进而,张量 $\underline{A}_k$ 和 $\underline{B}_k$ 一般均存在 CP 分解,即

$$\left. \begin{aligned} \underline{A}_k &= \underline{I}_k^{(A)} \times_1^2 \underline{A}_k^{(1)} \times_2^2 \underline{A}_k^{(2)} \times_3^2 \dots \times_d^2 \underline{A}_k^{(d)} \\ \underline{B}_k &= \underline{I}_k^{(B)} \times_1^2 \underline{B}_k^{(1)} \times_2^2 \underline{B}_k^{(2)} \times_3^2 \dots \times_d^2 \underline{B}_k^{(d)} \end{aligned} \right\} \quad (2)$$

式中: $\underline{A}_k^{(i)} \in \mathbb{R}^{m_i \times R_k^{(A)}}$; $\underline{B}_k^{(i)} \in \mathbb{R}^{n_i \times R_k^{(B)}}$; $\underline{I}_k^{(A)}$ 和 $\underline{I}_k^{(B)}$ 是模长分别为 $R_k^{(A)}$ 与 $R_k^{(B)}$ 的 d 阶超对角单位张量, $R_k^{(A)}$ 与 $R_k^{(B)}$ 分别为 $\underline{A}_k$ 和 $\underline{B}_k$ 的 CP 秩; $\times_\alpha^\beta$ 为张量乘法符号,该运算为矩阵乘法的拓展,表示前一项的第 α 阶与后一项的第 β 阶模长相等且可相乘^[31]。将式(2)代入式(1),可得 $\underline{W}$ 的 KCP 分解为^[30]

$$\underline{W} = \underline{I}^{(W)} \times_1^2 \underline{W}^{(1)} \times_2^2 \underline{W}^{(2)} \times_3^2 \dots \times_d^2 \underline{W}^{(d)} \quad (3)$$

式中: $\underline{W}^{(i)}$ 是分块矩阵,并且满足 $\underline{W}^{(i)} = [\underline{A}_1^{(i)} \otimes \underline{B}_1^{(i)} \quad \underline{A}_2^{(i)} \otimes \underline{B}_2^{(i)} \quad \dots \quad \underline{A}_K^{(i)} \otimes \underline{B}_K^{(i)}]$ 和 $\underline{W}^{(i)} \in \mathbb{R}^{m_i n_i \times R^{(W)}}$; $\underline{I}^{(W)}$ 亦为超对角张量。明显地,式(3)实际上也是 CP 分解形式, $\underline{W}$ 的 CP 秩为 $R^{(W)} = \sum_{k=1}^K R_k^{(A)} R_k^{(B)}$ 。

简而言之,KCP 可以将原始张量转化为 CP 分解形式,且其因子矩阵呈现分块稀疏性。KCP 的最小构成单位 $\underline{A}_k^{(i)}$ 和 $\underline{B}_k^{(i)}$ 兼有 Kronecker 分解(角标 k)和 CP 分解(肩标 i)的贡献,故其尺寸很小,使得 KCP 具有细粒度的特点。

1.2 KCPNet 轻量卷积模块

将普通卷积 $\underline{C} \in \mathbb{R}^{w \times h \times S \times T}$ 按照式(3)进行 2 阶 KCP 分解,可得

$$\left. \begin{aligned} \underline{C} &= \underline{I} \times_1^2 \underline{C}^{(1)} \times_2^2 \underline{C}^{(2)} \\ \underline{C}^{(1)} &= [\underline{A}_1^{(1)} \otimes \underline{B}_1^{(1)} \quad \dots \quad \underline{A}_K^{(1)} \otimes \underline{B}_K^{(1)}] \\ \underline{C}^{(2)} &= [\underline{A}_1^{(2)} \otimes \underline{B}_1^{(2)} \quad \dots \quad \underline{A}_K^{(2)} \otimes \underline{B}_K^{(2)}] \end{aligned} \right\} \quad (4)$$

式中: $\underline{A}_k^{(1)} \in \mathbb{R}^{w \times R_k^{(A)}}$; $\underline{A}_k^{(2)} \in \mathbb{R}^{S \times R_k^{(A)}}$; $\underline{B}_k^{(1)} \in \mathbb{R}^{h \times R_k^{(B)}}$, $\underline{B}_k^{(2)} \in \mathbb{R}^{T \times R_k^{(B)}}$; $\underline{I}$ 是模长为 $\sum_{k=1}^K R_k^{(A)} R_k^{(B)}$ 的单位张量。

根据文献[19-20],将 CP 分解映射为卷积模块时,单位张量对计算过程无影响,故可忽略。包含卷积窗口尺寸 w 和 h 的因子 $\underline{A}_k^{(1)}$ 与 $\underline{B}_k^{(1)}$ 能够映射为可分离卷积,其余因子 $\underline{A}_k^{(2)}$ 与 $\underline{B}_k^{(2)}$ 可作为 1×1 普通卷积。因此,式(4)意味着普通卷积在 KCP 分解后存在 K 组可分离卷积。对可分离卷积进行分组实质上是无意义的,因为可分离卷积本身是一种以通

道数为基准的一对一计算,如图2所示。

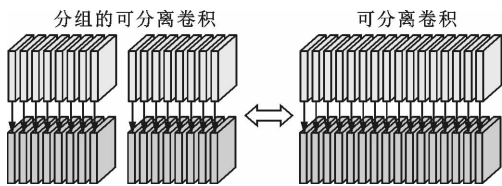


图2 一对一计算的可分离卷积

Fig. 2 Depthwise convolutions with one-to-one calculations

一般而言,可分离卷积相对于传统卷积能够极大地降低参数量和计算量^[4]。例如:图3a的普通卷积与输入特征图对应计算后需要相加来得到一张输出特征图;图3b的可分离卷积在与输入特征图进行一对一的计算后无需相加。显然,可分离卷积缺少特征图通道融合的相关操作,因此经常与独立负责通道融合的 1×1 普通卷积搭配使用^[13]。从张量分解的角度,可分离卷积与CP分解关系密切^[19-20],这是因为CP的因子张量中只存在矩阵和超对角张量(非主对角线元素皆为0),而不存在其他张量分解中的高阶因子。例如,Tucker分解必然存在与原张量阶数相同的核,也就无法得到能够表示可分离卷积的因子。KCP的所有因子实际上皆为矩阵,故可自然映射为可分离卷积与 1×1 普通卷积。

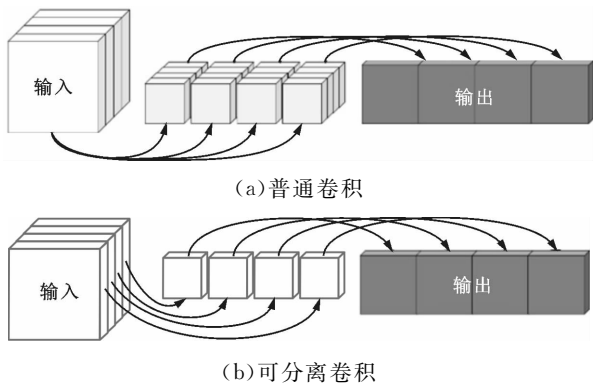


图3 普通卷积和可分离卷积

Fig. 3 Normal convolutions and depthwise convolutions

于是,由式(4)可定义分块矩阵

$$\left. \begin{aligned} \mathbf{A}^{(1)} &= [\mathbf{A}_1^{(1)} \quad \mathbf{A}_2^{(1)} \quad \cdots \quad \mathbf{A}_K^{(1)}] \in \mathbb{R}^{w \times R^{(A)}} \\ \mathbf{A}^{(2)} &= [\mathbf{A}_1^{(2)} \quad \mathbf{A}_2^{(2)} \quad \cdots \quad \mathbf{A}_K^{(2)}] \in \mathbb{R}^{S \times R^{(A)}} \\ \mathbf{B}^{(1)} &= [\mathbf{B}_1^{(1)} \quad \mathbf{B}_2^{(1)} \quad \cdots \quad \mathbf{B}_K^{(1)}] \in \mathbb{R}^{h \times R^{(B)}} \\ \mathbf{B}^{(2)} &= [\mathbf{B}_1^{(2)} \quad \mathbf{B}_2^{(2)} \quad \cdots \quad \mathbf{B}_K^{(2)}] \in \mathbb{R}^{T \times R^{(B)}} \end{aligned} \right\} \quad (5)$$

式中: $R^{(A)} = \sum_{k=1}^K R_k^{(A)}$; $R^{(B)} = \sum_{k=1}^K R_k^{(B)}$ 。为了向卷积计算映射,式(5)的矩阵可进一步通过升阶的方式分别重构为: $\mathbf{A}^{(1)} \in \mathbb{R}^{w \times I \times R^{(A)}}$, $\mathbf{A}^{(2)} \in \mathbb{R}^{I \times I \times S \times R^{(A)}}$, $\mathbf{B}^{(1)}$

$\in \mathbb{R}^{I \times h \times R^{(B)}}$, $\mathbf{B}^{(2)} \in \mathbb{R}^{I \times I \times R^{(B)} \times T}$ 。若 S 个通道特征图的输入记作 $\mathbf{U} \in \mathbb{R}^{w \times h \times S}$, 则其首先通过 $\mathbf{A}^{(2)}$ 所表示的 1×1 普通卷积, 将通道数从 S 变为 $R^{(A)}$; 然后, 依次经过 $w \times 1$ 和 $1 \times h$ 的非对称可分离卷积 $\mathbf{A}^{(1)}$ 和 $\mathbf{B}^{(1)}$, 这里需要注意的是, CP 秩 $R_k^{(A)}$ 与 $R_k^{(B)}$ 必须有整除关系, 才能满足可分离卷积的通道数从 $R^{(A)}$ 到 $R^{(B)}$ 的变化要求; 最后, 特征图再经过 $\mathbf{B}^{(2)}$ 所映射的 1×1 普通卷积, 将通道数由 $R^{(B)}$ 转化为最终的输出通道数 T 。如此, 这一过程在微观的张量元素层面可表示为

$$\left. \begin{aligned} \underline{U}_A^{(2)}(w_i, h_j, r^{(A)}) &= \\ \sum_{s=1}^S \underline{U}(w_i, h_j, s) \underline{A}^{(2)}(1, 1, s, r^{(A)}) \\ \underline{U}_A^{(1)}(w', h_j, r^{(A)}) &= \\ \sum_{i=1}^w \underline{U}_A^{(2)}(w_i, h_j, r^{(A)}) \underline{A}^{(1)}(i, 1, r^{(A)}) \\ \underline{U}_B^{(1)}(w', h_j, r^{(B)}) &= \\ \sum_{j=1}^h \underline{U}_A^{(1)}(w', h_j, r^{(A)}) \underline{B}^{(1)}(1, j, r^{(B)}) \\ \underline{V}(w', h', t) = \underline{U}_B^{(2)}(w', h', t) &= \\ \sum_{t=1}^T \underline{U}_B^{(1)}(w', h', r^{(B)}) \underline{B}^{(2)}(1, 1, r^{(B)}, t) \end{aligned} \right\} \quad (6)$$

式中: 最后一个子式得到的 $\underline{V} \in \mathbb{R}^{w \times h \times T}$ 是最终输出特征图; $w_i = (w' - 1) + i - p$ 和 $h_j = (h' - 1) + j - p$ 是步长为 1 且补 0 尺寸为 p 的卷积坐标变换; $r^{(B)} = r^{(A)} + \delta R^{(A)}$, δ 为自然数。式(6)将输入特征图 $\underline{U}$ 转换为输出特征图 $\underline{V}$, 这一过程即为本文所提出的 KCPNet 轻量卷积模块。

2 KCPNet 性能分析

2.1 KCPNet 的可变表现形式

根据式(6), KCPNet 在宏观张量层面的表现形式可描述为

$$\left. \begin{aligned} \underline{U}_A^{(2)} &= f(\underline{U} \otimes \underline{A}^{(2)}) \\ \underline{U}_A^{(1)} &= f(\underline{U}_A^{(2)} \otimes \underline{A}^{(1)}) \\ \underline{U}_B^{(1)} &= f(\underline{U}_A^{(1)} \otimes \underline{B}^{(1)}) \\ \underline{V} = \underline{U}_B^{(2)} &= f(\underline{U}_B^{(1)} \otimes \underline{B}^{(2)}) \end{aligned} \right\} \quad (7)$$

式中 $\otimes$ 与 $\otimes$ 分别表示普通卷积与可分离卷积。式(7)更易于和图4a所展示的 KCPNet 网络结构相对照, 图中可见 KCPNet 呈现明显的瓶颈结构。

由于 KCP 是一种很灵活且细粒度的张量分解方法, 故 KCPNet 也可通过特定的变换而获得多种表现形式, 这些不同的形式即可指向其他具有瓶颈

结构的张量分解卷积模块。例如:若 $R^{(A)} = R^{(B)}$, 则 KCPNet 等价于文献[19]的非对称可分离卷积瓶颈结构,如图 4b 所示;若将 $A_k^{(1)} \otimes B_k^{(1)} = D_k^{(1)}$ 看作整体,且重构为张量 $D_k^{(1)} \in \mathbb{R}^{w \times h \times R_k^{(A)} \times R_k^{(B)}}$, 则 KCPNet 将转化为文献[23]所描述的共包含 K 组的分组卷积瓶颈结构,如图 4c 所示;若进一步有 $K=1$, 则分组卷积瓶颈结构退化为普通卷积瓶颈结构[21],如图 4d 所示。

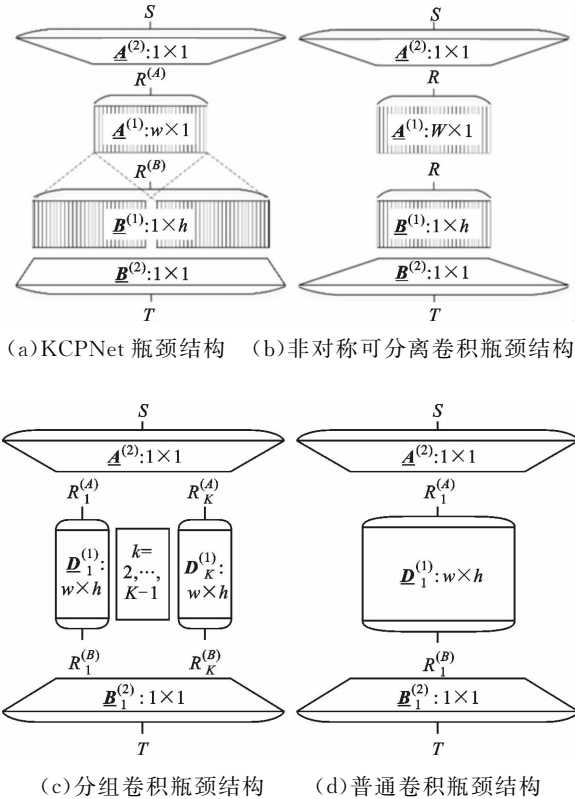


图4 KCPNet 及可转换的瓶颈结构

Fig. 4 KCPNet and its transformable bottleneck structures

由于可分离卷积相比于普通卷积会节省可观的空间和计算复杂度(如图 3 所示),所以 KCPNet 将 $D_k^{(1)} = A_k^{(1)} \otimes B_k^{(1)}$ 中的 $A_k^{(1)}$ 与 $B_k^{(1)}$ 分开,即将瓶颈结构中的普通卷积分割为两段可分离卷积,极大地降低了卷积模块的空间和计算复杂度。更重要的是, KCPNet 所包含的两层可分离卷积之间具备一定的通道变化,相较于源自纯粹 CP 分解的可分离卷积[13,19-20]更具灵活性。

2.2 ImageNet 图像分类对比实验

考虑到具备瓶颈结构的卷积模块发展历程,本小节基于 ImageNet 标准大规模图像分类数据集[32],以 Tucker 分解映射[21]的 ResNet-50[22]和 BT 分解映射[23]的 ResNeXt-50[24]为主要参考对象,在整体卷积神经网络总参数量相近的前提下,对比

普通可分离卷积[20]、非对称可分离卷积[19]以及 KCPNet 的变通道可分离卷积之间的性能差异。所有网络的结构如表 1 所示。表中,对网络结构的描述遵循文献[22-24]的标准:“ $w \times h, T$, 步长 z ”表示窗口尺寸为 $w \times h$ 、输出通道为 T 、步长为 z ($z=1$ 则忽略)的卷积;多个卷积组成的 block 之后的乘数 y 表示该 block 重复堆叠 y 个;池化参数“ $w \times h$, 步长 z ”指池化的窗口尺寸为 $w \times h$, 池化步长为 z ($z=1$ 则忽略);全连接的参数“ $M \times N$ ”为该全连接权重矩阵的尺寸。

由表 1 可知:ResNet 的计算量虽小但其参数量却最大;ResNeXt 和另外两个基于 CP 分解的模型虽然参数量稍少,计算量却超过了 5×10^9 ;只有 KCPNet 能够在参数量较低时仍然保持较少的计算量,相对更能兼顾空间与计算复杂度;Tensor power 网络和 Speeding-up 网络的通道数相同,虽然理论上后者的非对称卷积参数量更少,但由于绝大部分参数都由 1×1 卷积所贡献,故总参数量并无明显区别。

需要注意的是, KCPNet 的 $R^{(A)}$ 和 $R^{(B)}$ 之间的倍数关系 δ 决定了图 4a 中 $1 \times h$ 卷积的数量倍于 $w \times 1$ 卷积的数量,造成了 KCPNet 这种变通道可分离卷积在不同方向上的特征提取程度有异,从而影响整个 CNN 模型的识别精度。为了优化该特征采集不平衡的问题,需要对图 4a 的可分离卷积部分略作改进。具体地,可将第 1 层 $R^{(A)}$ 通道的 $w \times 1$ 卷积改为 $R^{(A)}/\delta$ 通道的 $w \times 1$ 与 $R^{(A)}/\delta$ 通道的 $1 \times h$ 卷积,并对第 2 层 $R^{(B)}$ 通道的卷积作出类似处理,最后将这两层可分离卷积按 $w \times 1$ 的通道与 $1 \times h$ 的通道对应即可,如图 5 所示(图中 $\delta=2$)。改进之前,图 5 的 KCPNet 可分离卷积所输出的特征图更多地包含了 $1 \times h$ 提取的高度方向特征,改进之后的输出特征图在高度和宽度方面的特征提取程度将变得均衡。这样的改进虽然使其不能严格符合式(7)的 KCP 分解数学形式,但在神经网络的实际应用中,

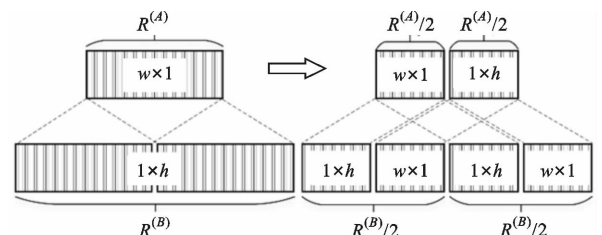


图5 KCPNet 非对称可分离卷积改进

Fig. 5 Amelioration of the asymmetrical depthwise convolutions in KCPNet

表 1 ImageNet 数据集图像分类对比实验的不同网络结构

Table 1 Different network structures of contrast experiments for image classification on ImageNet dataset

网络结构	参数				
	Tucker:ResNet 普通卷积 ^[21-22]	BT:ResNeXt 分组卷积 ^[23-24]	CP:Tensor power 普通可分离卷积 ^[20]	CP:Speeding-up 非对称可分离卷积 ^[19]	KCP:KCPNet 普通通道可分离卷积
卷积	7×7,64,步长 2				
最大池化	3×3,步长 2				
block-1	$\begin{bmatrix} 1\times 1,64 \\ 3\times 3,64 \\ 1\times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,128 \\ 3\times 3,4\times 32 \\ 1\times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,160 \\ 3\times 3,160 \\ 1\times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,160 \\ 3\times 1,160 \\ 1\times 3,160 \\ 1\times 1,256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,64 \\ 3\times 1,64 \\ 1\times 3,192 \\ 1\times 1,256 \end{bmatrix} \times 3$
block-2	$\begin{bmatrix} 1\times 1,128 \\ 3\times 3,128 \\ 1\times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1,256 \\ 3\times 3,8\times 32 \\ 1\times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1,288 \\ 3\times 3,288 \\ 1\times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1,288 \\ 3\times 1,288 \\ 1\times 3,288 \\ 1\times 1,512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times 1,128 \\ 3\times 1,128 \\ 1\times 3,384 \\ 1\times 1,512 \end{bmatrix} \times 4$
block-3	$\begin{bmatrix} 1\times 1,256 \\ 3\times 3,256 \\ 1\times 1,1\ 024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1,512 \\ 3\times 3,16\times 32 \\ 1\times 1,1\ 024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1,544 \\ 3\times 3,544 \\ 1\times 1,1\ 024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1,544 \\ 3\times 1,544 \\ 1\times 3,544 \\ 1\times 1,1\ 024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times 1,384 \\ 3\times 1,384 \\ 1\times 3,768 \\ 1\times 1,1\ 024 \end{bmatrix} \times 6$
block-4	$\begin{bmatrix} 1\times 1,512 \\ 3\times 3,512 \\ 1\times 1,2\ 048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,1\ 024 \\ 3\times 3,32\times 32 \\ 1\times 1,2\ 048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,1\ 088 \\ 3\times 3,1\ 088 \\ 1\times 1,2\ 048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,1\ 088 \\ 3\times 1,1\ 088 \\ 1\times 3,1\ 088 \\ 1\times 1,2\ 048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times 1,512 \\ 3\times 1,512 \\ 1\times 3,1\ 536 \\ 1\times 1,2\ 048 \end{bmatrix} \times 3$
平均池化	7×7				
全连接	2\ 048×1\ 000				
总参数量	25.5×10 ⁶	25.0×10 ⁶	24.9×10 ⁶	24.9×10 ⁶	25.0×10 ⁶
总计算量	4.9×10 ⁹	5.1×10 ⁹	5.2×10 ⁹	5.2×10 ⁹	4.9×10 ⁹

能提供 1%~2%的性能提升。

ImageNet 数据集图像分类对比实验结果如表 2 所示。可以看出,KCPNet 与其他基于 CP 分解的高效卷积模块相比具有相当的精度,且相对于 Res-Net 的精度损失十分有限(在 0.5%以内)。虽然以 BT 分解映射的 ResNeXt 准确率最高,但其特殊的分组卷积架构在实际训练中运行效率最低,使其与

表 2 ImageNet 数据集图像分类对比实验结果

Table 2 Results of contrast experiments for image classification on ImageNet dataset

网络模型	分类准确率/%
Tucker:ResNet ^[21-22]	75.4±0.08
BT:ResNeXt ^[23-24]	76.5±0.07
CP:Tensor power ^[20]	75.2±0.06
CP:Speeding-up ^[19]	75.0±0.03
KCP:KCPNet	75.1±0.04

实际应用部署之间仍有一定的距离,这一点与文献[24]结论相一致。

3 KCPNet 卷积模块压缩与部署

3.1 CIFAR-10 图像分类网络压缩实验

作为一种源自 KCP 张量分解的方法,KCPNet 除了体现从张量结构到卷积模块的映射设计方式之外,同样能在传统卷积网络压缩的方面起到一定作用。本小节基于 CIFAR-10 标准图像分类数据集,以一个自设计的传统 VGG 卷积网络为参考,将其以 KCPNet 形式压缩,并与相关的 Kronecker 张量分解压缩方法 KConv^[14]和 GKPD^[15]进行对比,结果如表 3 所示。

简单澄清 Kronecker 张量分解压缩方法的关键细节,以便解释表 3 的相关参数。文献[14]和[15]皆以式(1)为理论基础,将卷积模块 $\underline{C} \in \mathbb{R}^{w \times h \times S \times T}$ 分解为 K 组 $\underline{A}_k \in \mathbb{R}^{w \times 1 \times S_1 \times T_1}$ 和 $\underline{B}_k \in \mathbb{R}^{1 \times h \times S_2 \times T_2}$ 的双层

表 3 CIFAR-10 数据集图像分类对比实验的不同网络结构

Table 3 Different network structures of the contrast experiments for image classification on CIFAR-10 dataset

网络结构	参数		
	VGG	KConv ^[14] & GKPD ^[15]	KCPNet
卷积	3×3,64		
block-1	$\begin{bmatrix} 3\times 3,128 \\ 3\times 3,128 \end{bmatrix}$	$\left[\begin{bmatrix} 3\times 1,8 \\ 1\times 3,16 \end{bmatrix}\times 2\right],K=26$	$\begin{bmatrix} 1\times 1,48 \\ 3\times 1,48 \\ 1\times 3,96 \\ 1\times 1,128 \end{bmatrix}\times 2$
block-2	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ 3\times 3,256 \\ 3\times 3,256 \\ 3\times 3,256 \end{bmatrix}$	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ \begin{bmatrix} 3\times 1,16 \\ 1\times 3,16 \end{bmatrix}\times 3 \end{bmatrix},K=50$	$\begin{bmatrix} 1\times 1,96 \\ 3\times 1,96 \\ 1\times 3,192 \\ 1\times 1,256 \end{bmatrix}\times 3$
block-3	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ 3\times 3,384 \\ 3\times 3,384 \\ 3\times 3,384 \end{bmatrix}$	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ \begin{bmatrix} 3\times 1,16 \\ 1\times 3,24 \end{bmatrix}\times 3 \end{bmatrix},K=66$	$\begin{bmatrix} 1\times 1,128 \\ 3\times 1,128 \\ 1\times 3,256 \\ 1\times 1,384 \end{bmatrix}\times 3$
block-4	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ 3\times 3,512 \\ 3\times 3,512 \\ 3\times 3,512 \end{bmatrix}$	$\begin{bmatrix} 3\times 3,\text{最大池化} \\ \begin{bmatrix} 3\times 1,16 \\ 1\times 3,32 \end{bmatrix}\times 3 \end{bmatrix},K=98$	$\begin{bmatrix} 1\times 1,192 \\ 3\times 1,192 \\ 1\times 3,384 \\ 1\times 1,512 \end{bmatrix}\times 3$
平均池化	4×4		
全连接	512×10		
总参数量	11.7×10 ⁶	1.9×10 ⁶	1.9×10 ⁶
总计算量	0.94×10 ⁹	1.9×10 ⁹	0.23×10 ⁹

普通卷积,其中 $S_1S_2=S$ 且 $T_1T_2=T$ 。如此带来的输入特征图通道数与卷积模块不匹配的问题有两种处理方式:一是将卷积 $\underline{\mathbf{A}}_k$ 的输入特征图的通道数由 S 改为 S_1 ,余下的 S_2 通道则看作批处理参数^[14];二是将卷积 $\underline{\mathbf{A}}_k$ 看作输入通道为 1、输出通道为 T_1 、窗口尺寸为 $w\times 1\times S_1$ 的 3D 卷积模块^[15]。实际上,这两种处理方式的效果是等价的,都是将特征图的卷积分配至两层依次处理,文献[14]和[15]中的复杂度计算也是相同的,故表 3 将它们视为同一种方法。

相比之下,本文所提的 KCPNet 应参考式(4)和式(7),即将 VGG 中的卷积模块 $\underline{\mathbf{C}}\in\mathbb{R}^{w\times h\times S\times T}$ 分解为 $\underline{\mathbf{A}}^{(1)}、\underline{\mathbf{A}}^{(2)}、\underline{\mathbf{B}}^{(1)}$ 和 $\underline{\mathbf{B}}^{(2)}$ 共 4 个因子,并保证 $R^{(A)}$ 和 $R^{(B)}$ 之间的倍数关系,例如,block-1 中, $R^{(A)}=48$, $R^{(B)}=96$ 。VGG 中存在的 3×3 最大池化,在 KCPNet 的对应 block 中应令第一个 3×1 可分离卷积的步长为 2 以替代,实现同样的下采样效果。为了对比公平,将 Kconv 方法^[14-15] 和 KCPNet 方法得到的网络模型总参数量调整至相近,并且对 Kconv 的两层卷积模块也添加了与 KCPNet 中的瓶颈结构相同的额外非线性函数和残差连接。

由表 3 可以看出,KCPNet 在计算效率上的优势十分明显,原因在于 KCP 张量分解本就是 Kronecker 张量分解的进阶方法,其结合了 CP 分解的特性从而使普通卷积可以转变为计算高效的可分离卷积。

VGG 及两个网络压缩模型在 CIFAR-10 上的实验结果如表 4 所示。可以看出,KCPNet 的精度损失在 0.3%左右,KConv 的精度损失超过 1%,文献[14-15]的实验记录也表明 KConv 难以在 CIFAR-10 这类中等规模数据集上弥补精度损失。

表 4 CIFAR-10 数据集图像分类对比实验结果

Table 4 Results of the contrast experiments for image classification on CIFAR-10 dataset

网络模型	分类准确率/%
VGG	93.3±0.05
KConv ^[14] & GKPD ^[15]	92.0±0.03
KCPNet	93.0±0.01

结合本小节实验可知,KCPNet 可将 VGG 压缩至原先的 16.1%,且无明显精度损失,并可节省

75.5%的计算量,相对基于 Kronecker 张量分解的 KConv 方法^[14-15]有较明显综合优势。

3.2 可分离卷积的部署

KCPNet 卷积模块的部署目标主要考虑为嵌入式 GPU 等通用并行计算设备。虽然目前的绝大多数深度学习平台都已经有了基于 CUDA 的接口,但对可分离卷积的实现并不十分高效,因为这些平台一般采取将整个模型的所有卷积模块转化为矩阵乘法来计算,以便于使用成熟的并行矩阵乘法。如此,将对 GPU 造成很大的存储压力^[33],这也是目前深度神经网络训练往往需要大显存 GPU 的原因。另一方面,大多平台皆是基于 CUDA 开发,这就迫使相关具体应用只能选择 NVIDIA 的产品。

本小节在图 6 所示的 GPU 的 3 层内存架构下讨论可分离卷积的部署,采用跨硬件平台的 OpenCL 并行编程框架,具体方式与 CUDA 中 cuDNN 将卷积简单转化为矩阵乘法的方式相异。部署不考虑神经网络的训练过程,仅与前向推理计算相关。

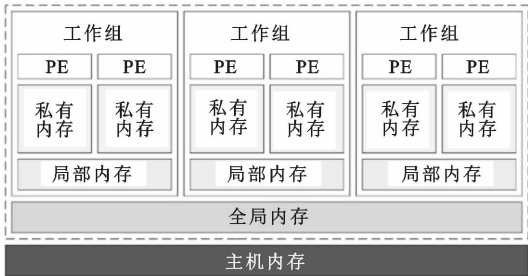


图 6 GPU 内存架构
Fig. 6 Memory architecture of GPU

由图 6 可知,GPU 内存架构具有多层次分组的特点,部署的原则为尽量平衡地将可并行的计算内容分配至各个工作组内。因此,如何对可分离卷积的计算过程进行分组分块即为部署的关键。由于

GPU 一般允许至多两个不同维度方向的分组,再考虑实际的卷积模块通道数往往远大于单个特征图的边长,故这里一方面设为按特征图通道进行分组,另一方面则将单个特征图分割为若干组。

以图 7 所示的 3×1 卷积为例,特征图通道数 $D=9$,每张特征图有 $W=8,H=3$ 。对计算进行分组:第一个方向的分组将每张特征图分为 gp_1(1)、gp_1(2)和 gp_1(3)共 3 个部分;第二个方向则在通道上进行,每 3 个通道为一组,记为 gp_2(1)、gp_2(2)和 gp_2(3);总工作组数为这两个分组方向交叉对应所得,即 $3\times 3=9$ 组。如此,GPU 的每个工作组内将被分配 3 块局部内存,每块局部内存负责 3 个通道的一部分特征图计算。从另一个角度来看,每张特征图的计算被分配至 3 个不同工作组内。由于绘图空间限制,故图 7 并未画出所有工作组,仅描绘了 gp_2(3)分别与 gp_1(1)、gp_1(2)和 gp_1(3)交叉对应所得的 3 个工作组。每块局部内存之内的卷积计算可展开并行处理,每个独立的 3×1 卷积由一个执行单元负责计算。

图 7 为绘制方便起见,特征图按每组 8 个像素分组,通道方面则按每组 3 个通道分组。在实际部署中,每组包含的数据量一般设为 16×16 最为合适,即整个可分离卷积应分为 $\frac{W\times H}{16}\times \frac{D}{16}$ 组。此种部署设计之下,表 3 的 KCPNet 处理单张图片的实测速度在最大功耗 40 W 的 Quadro P1000 型 GPU 上可达 10 ms 左右,即运行速度约 100 帧/s。

4 手势识别应用的开发

4.1 时间飞行深度相机

时间飞行(ToF)是一种较新的深度相机,其利用发射激光及其反射光之间的相位差对被摄物体的

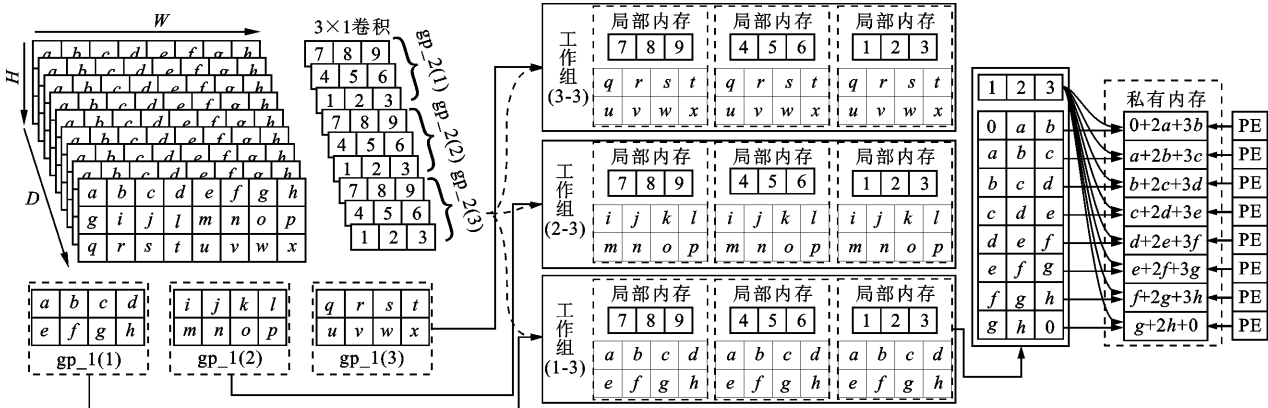


图 7 可分离卷积在 GPU 内存架构上的部署
Fig. 7 Deployment of depthwise convolution on memory architecture of GPU

距离进行定位,并能得到被摄物体表面点云^[34]。ToF 技术目前已在一些特定场景中实际应用,如 Melexis 与 BMW 公司合作在汽车中控台部位安装 ToF 相关应用,以替代部分传统的接触式人机交互^[35]。本文选用 PMD 公司生产的型号为 pico-flexx 的 ToF 深度相机^[34],该相机具有体积小、功耗低的特点^[36]。

4.2 数据集构建

ToF 相机是带有光源的主动拍摄设备,会生成红外和点云两路不同的视觉数据。对于手势识别,手轮廓外的其余信息是多余甚至是有干扰的,而 ToF 相机可通过红外的光强阈值和点云的距离阈值互相滤波来得到较好的手势轮廓图像。本文实验中令同时满足距离大于 500 mm 的点云和灰度值大于 110 的红外数据作为手势轮廓中的有效像素,如图 8 所示。

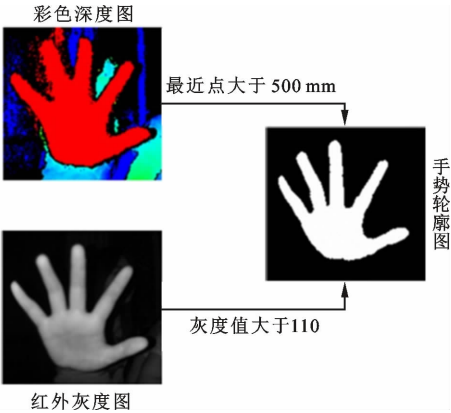


图 8 手势数据的互滤波获取方法

Fig. 8 Method for obtaining gesture data by cross filtering

为保证应用的健壮性和泛化性,对数据集进行扩充。在数据集容量方面,相对文献[37]进行扩充,训练集和验证集分别扩大至 10 000 和 4 000 幅手势轮廓。在手势轮廓的形状表达上,尽力做到多样化,如图 9 的第 2 行中,不仅以食指和中指表示数字 2,也要考虑其他任何两根手指的组合。

4.3 应用软件开发

应用软件开发采取界面框架与驱动引擎相结合的两层架构,如图 10 所示。其中,上层界面以微软的 WPF 为开发框架,下层引擎以 C++ 的 DLL 动态链接库形式进行开发,二者之间以 DLL 提供的 C 语言 API 为接口。界面框架的逻辑相对简单,以主界面框架为总体,下辖若干子功能界面框架,这些子功能包括了手势识别的界面和一些 ToF 镜头设置相关界面。ToF 引擎稍显复杂,内部主要包含 4 部



图 9 手势轮廓的形状多样化

Fig. 9 Variant shapes of gesture profiles

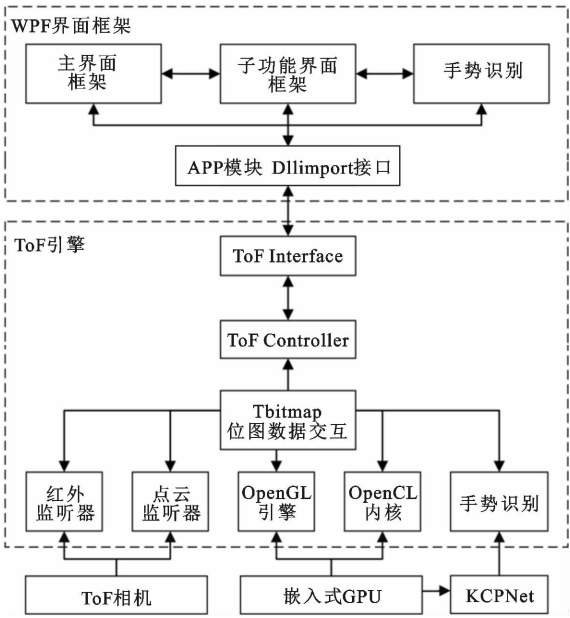


图 10 手势识别应用软件架构

Fig. 10 Architecture of gesture recognition application software

分内容:①与 ToF 相机相关的传感器监听模块,分别实时地接收相机端的红外与点云数据;②OpenGL 引擎和 OpenCL 引擎联合实现镜头数据的显示,前者主要负责点云的显示渲染,后者则承担位图的一些快速计算;③以表 3 的 KCPNet 为基础的手势识别模块,负责将实时的手势轮廓位图数据送入 KCPNet 并将结果送于接口;④接口模块,由负责管理任务的控制器(ToF Controller)和纯 C 语言所包装的 API 接口(ToF Interface)组成。

该软件执行手势识别的流程可描述为:①红外监听器和点云监听器实时获取 ToF 相机的数据,二

者都通过 OpenGL 引擎帮助生成各自的位图;② OpenCL 以图 8 所示的方法快速生成手势轮廓图,并送于手势识别模块;③手势识别模块调用 KCP-Net 对手势轮廓作出判断,这里 KCPNet 应以图 7 所示的方式部署于 GPU 上;④手势轮廓和手势识别结果通过接口模块同时送给上层的界面框架。本文完成的手势识别应用软件视觉效果如图 11 所示。



图 11 手势识别应用软件截图

Fig. 11 Screenshot of gesture recognition application software

4.4 手势识别应用的效果

为了凸显 KCPNet 的优势,手势识别应用同时部署了表 3 的 VGG 和 KCPNet 两个模型,综合的识别效果对比如表 5 所示。可以看出,KCPNet 相对于传统的 VGG 网络节省了大量的运行时内存占用。虽然 KCPNet 的理论计算复杂度低于 VGG,但如图 1 所示,KCPNet 实际上是将一个传统卷积模块分解映射为多层卷积的形式,从而使得网络模型的深度加深,这就必然带来更多层次的特征图处理过程。即便 KCPNet 在内存使用上可以通过内存复用或及时清理使其总运行时内存占用处于很低的水平,但更多的特征图处理过程却难以令 KCPNet 在真实运行时间上相比 VGG 产生优势。

表 5 VGG 和 KCPNet 的手势识别效果对比
Table 5 Comparison of gesture recognition effect between VGG and KCPNet

网络模型	识别准确率 /%	内存占用 /MB	单次识别用时 /ms
VGG	99.5±0.01	137	约 9
KCPNet	99.5±0.01	22	约 9

从表 5 还可看出,KCPNet 在面向实际应用时依旧有着可观的优势。首先,一些实际应用的数据集往往并不如 ImageNet、CIFAR-10 等有挑战性的标准数据集复杂,这使得 KCPNet 在网络模型压缩

的角度更容易取得无损效果。在该手势识别应用中,KCPNet 与 VGG 同样达到 99.5%的验证集分辨率,而内存占用仅为后者的 16.1%。KCPNet 的网络模型压缩效果很好,在节省大量内存的基础上能够保证实际执行时间不受影响,手势识别应用的单次识别用时在 9 ms 左右,即运行速度超过 100 帧/s,能够达到动态即时识别的实用效果。

5 结 论

本文在 KCP 张量分解的基础上,将 KCP 的因子张量结构映射为变通道可分离卷积瓶颈结构,从而提出了一种轻量化卷积模块 KCPNet。性能分析表明,所提的 KCPNet 能够转换为其他已有的多种张量分解卷积模块,在 ImageNet 上的实验验证了 KCPNet 能够兼顾空间和计算复杂度的高效性。在 CIFAR-10 上的 CNN 压缩实验表明,KCPNet 可将其对应的 VGG 压缩至 16.1%而无明显精度损失,同时可节省 75.5%的计算量。针对 KCPNet 中的可分离卷积,提出了在嵌入式 GPU 上部署的方法,使 KCPNet 对 CIFAR-10 的识别效率达到 100 帧/s。围绕 ToF 深度相机开发了手势识别应用,该应用识别率可达 99.5%,内存占用 22 MB,识别效率超过 100 帧/s。总之,KCPNet 相比已有卷积模块,具有最优的复杂度和可观的压缩能力,所提的部署方法在实际内存占用和计算耗时方面皆有较高效率,体现了在实际应用中的高效性能优势。

参考文献:

[1] LECUN Y, BENGIO Y, HINTON G. Deep learning [J]. Nature, 2015, 521(7553): 436-444.

[2] DENG Lei, LI Guoqi, HAN Song, et al. Model compression and hardware acceleration for neural networks: a comprehensive survey [J]. Proceedings of the IEEE, 2020, 108(4): 485-532.

[3] 林景栋, 吴欣怡, 柴毅, 等. 卷积神经网络结构优化综述 [J]. 自动化学报, 2020, 46(1): 24-37.

LIN Jingdong, WU Xinyi, CHAI Yi, et al. Structure optimization of convolutional neural networks: a survey [J]. Acta Automatica Sinica, 2020, 46(1): 24-37.

[4] 毛远宏, 贺占庄, 刘露露. 目标跟踪中基于深度可分离卷积的剪枝方法 [J]. 西安交通大学学报, 2021, 55(1): 52-59.

MAO Yuanhong, HE Zhanzhuang, LIU Lulu. Pruning based on separable convolutions for object tracking

- [J]. Journal of Xi'an Jiaotong University, 2021, 55(1): 52-59.
- [5] LIN Shaohui, JI Rongrong, LI Yuchao, et al. Toward compact ConvNets via structure-sparsity regularized filter pruning [J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(2): 574-588.
- [6] 梁峰, 董名, 田志超, 等. 面向轻量化神经网络的模型压缩与结构搜索 [J]. 西安交通大学学报, 2020, 54(11): 106-112.
- LIANG Feng, DONG Ming, TIAN Zhichao, et al. Model compression and structure search for light-weight neural network [J]. Journal of Xi'an Jiaotong University, 2020, 54(11): 106-112.
- [7] WANG Tianzhe, WANG Kuan, CAI Han, et al. APQ: joint search for network architecture, pruning and quantization policy [C]// Proceeding of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA; IEEE, 2020: 2075-2084.
- [8] 程士卿, 郝问裕, 李晨, 等. 低秩张量分解的多视角谱聚类算法 [J]. 西安交通大学学报, 2020, 54(3): 119-125, 133.
- CHENG Shiqing, HAO Wenyu, LI Chen, et al. Multi-view clustering by low-rank tensor decomposition [J]. Journal of Xi'an Jiaotong University, 2020, 54(3): 119-125, 133.
- [9] AGGARWAL V, WANG Wenlin, ERIKSSON B, et al. Wide compression: tensor ring nets [C]// Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA; IEEE, 2018: 9329-9338.
- [10] WU Bijiao, WANG Dingheng, ZHAO Guangshe, et al. Hybrid tensor decomposition in neural network compression [J]. Neural Networks, 2020, 132: 309-320.
- [11] HUANG Hantao, NI Lebin, WANG Kanwen, et al. A highly parallel and energy efficient three-dimensional multilayer CMOS-RRAM accelerator for tensorized neural network [J]. IEEE Transactions on Nanotechnology, 2018, 17(4): 645-656.
- [12] DENG Chunhua, SUN Fangxuan, QIAN Xuehai, et al. TIE: energy-efficient tensor train-based inference engine for deep neural network [C]// Proceedings of the 46th International Symposium on Computer Architecture. New York, USA; ACM, 2019: 264-278.
- [13] KOSSAIFI J, TOISOUL A, BULAT A, et al. Factorized higher-order CNNs with an application to spatio-temporal emotion estimation [C]// Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA; IEEE, 2020: 6059-6068.
- [14] ZHOU Shuchang, WU Jianan, WU Yuxin, et al. Exploiting local structures with the Kronecker layer in convolutional networks [EB/OL]. (2016-02-04) [2021-07-01]. <https://arxiv.org/abs/1512.09194>.
- [15] HAMEED MGA, TAHAEI MS, MOSLEH A, et al. Convolutional neural network compression through generalized Kronecker product decomposition [EB/OL]. (2021-09-29)[2021-10-13]. <https://arxiv.org/abs/2109.14710>.
- [16] WANG Dingheng, ZHAO Guangshe, LI Guoqi, et al. Compressing 3DCNNs based on tensor train decomposition [J]. Neural Networks, 2020, 131: 215-230.
- [17] LEE D, WANG Dingheng, YANG Yukuan, et al. QTTNet: quantized tensor train neural networks for 3D object and video recognition [J]. Neural Networks, 2021, 141: 420-432.
- [18] ZHANG Xiangyu, ZOU Jianhua, MING Xiang, et al. Efficient and accurate approximations of nonlinear convolutional networks [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA; IEEE, 2015: 1984-1992.
- [19] LEBEDEV V, GANIN Y, RAKHUBA M, et al. Speeding-up convolutional neural networks using fine-tuned CP-decomposition [C]// Proceedings of the 3rd International Conference on Learning Representations. London, UK; ICLR, 2015: 1-11.
- [20] ASTRID M, LEE S I. CP-decomposition with tensor power method for convolutional neural networks compression [C]// Proceedings of the 2017 IEEE International Conference on Big Data and Smart Computing (BigComp). Piscataway, NJ, USA; IEEE, 2017: 115-118.
- [21] KIM Y D, PARK E, YOO S, et al. Compression of deep convolutional neural networks for fast and low power mobile applications [C]// Proceedings of the 4th International Conference on Learning Representations. London, UK; ICLR, 2016: 1-16.
- [22] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA; IEEE, 2016: 770-778.
- [23] CHEN Yunpeng, JIN Xiaojie, KANG Bingyi, et al. Sharing residual units through collective tensor factori-

- zation to improve deep neural networks [C]//Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. Sacramento, CA, USA; IJCAI, 2018: 635-641.
- [24] XIE Saining, GIRSHICK R, DOLLÁR P, et al. Aggregated residual transformations for deep neural networks [C]//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA; IEEE, 2017: 5987-5995.
- [25] SANDLER M, HOWARD A, ZHU Menglong, et al. MobileNetV2: inverted residuals and linear bottlenecks [C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA; IEEE, 2018: 4510-4520.
- [26] SUN Ke, LI Mingjie, LIU Dong, et al. IGCv3: interleaved low-rank group convolutions for efficient deep neural networks [C]//Proceedings of the 2018 British Machine Vision Conference. Guildford, UK; BMVA Press, 2018: 1-13.
- [27] 王子愉, 袁春, 黎健成. 利用可分离卷积和多级特征的实例分割 [J]. 软件学报, 2019, 30(4): 954-961.
WANG Ziyu, YUAN Chun, LI Jiancheng. Instance segmentation with separable convolutions and multi-level features [J]. Journal of Software, 2019, 30(4): 954-961.
- [28] 周云成, 许童羽, 邓寒冰, 等. 基于面向通道分组卷积网络的番茄主要器官实时识别 [J]. 农业工程学报, 2018, 34(10): 153-162.
ZHOU Yuncheng, XU Tongyu, DENG Hanbing, et al. Real-time recognition of main organs in tomato based on channel wise group convolutional network [J]. Transactions of the Chinese Society of Agricultural Engineering, 2018, 34(10): 153-162.
- [29] 杨贤志, 黄国方, 周宁宁. 基于分组卷积和特征图级联的轻量级目标检测 [J]. 计算机应用研究, 2021, 38(5): 1590-1594.
YANG Xianzhi, HUANG Guofang, ZHOU Ningning. Light-weight object detection network based on group convolution and feature maps cascade [J]. Application Research of Computers, 2021, 38(5): 1590-1594.
- [30] PHAN A H, CICHOCKI A, TICHAVSKÝ P, et al. From basis components to complex structural patterns [C]//Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. Piscataway, NJ, USA; IEEE, 2013: 3228-3232.
- [31] LEE N, CICHOCKI A. Regularized computation of approximate pseudoinverse of large matrices using low-rank tensor train decompositions [J]. SIAM Journal on Matrix Analysis and Applications, 2016, 37(2): 598-623.
- [32] RUSSAKOVSKY O, DENG Jia, SU Hao, et al. ImageNet large scale visual recognition challenge [J]. International Journal of Computer Vision, 2015, 115(3): 211-252.
- [33] ITO Y, MATSUMIYA R, ENDO T. ooc_cuDNN: accommodating convolutional neural networks over GPU memory capacity [C]//Proceedings of the 2017 IEEE International Conference on Big Data. Piscataway, NJ, USA; IEEE, 2017: 183-192.
- [34] PRESNOV D, LAMBERS M, KOLB A. Robust range camera pose estimation for mobile online scene reconstruction [J]. IEEE Sensors Journal, 2018, 18(7): 2903-2915.
- [35] Melexis Inspired Engineering. Melexis to participate in the MinTOFKA project to develop a miniaturized 3D camera system for vehicle interiors [EB/OL]. (2019-03-25) [2021-07-01]. <https://www.melexis.com/zh/news/2019/25-mar2019-mintofka-project-kick-off>.
- [36] PMD Technologies. Picofamily website relaunch and new online shop [EB/OL]. (2018-03-13) [2021-07-01]. <https://pmdtec.com/en/company/news/pico-family-website-relaunch-and-new-online-shop/>.
- [37] 王鼎衡, 赵广社, 李国齐, 等. 基于张量链压缩的卷积神经网络及手势识别应用研究 [C]//2018 中国自动化大会论文集. 北京: 中国自动化学会, 2018: 87-92.

(编辑 陶晴)