# 面向卷积神经网络的统一 INT8 训练

Feng Zhu1 Ruihao Gong1,2 Fengwei Yu1 Xianglong Liu2 Yanfei Wang1
Zhelong Li1 Xiuqi Yang1 Junjie Yan1 1SenseTime Group
Limited 2State Key Laboratory of Software
Development Environment, Beihang University {zhufeng1, yufengwei,
wangyanfei, lizhelong, yangxiuqi, yanjunjie}@sensetime.com {gongruihao, xlliu}
@nlsde.buaa.edu.cn

## 抽象的

最近低位（例如,8 位)网络量化已被广泛研究以加速推理。除了推理之外,具有量化梯度的低位训练可以进一步带来更可观的加速,因为反向过程通常是计算密集型的。不幸的是,反向传播的不适当量化通常会使训练不稳定甚至崩溃。

缺乏一个成功的统一低位训练框架,可以支持各种任务的不同网络。在本文中,我们尝试从准确性和速度两个方面为常见的卷积神经网络构建一个统一的 8 位（INT8)训练框架。首先,我们凭经验发现了梯度的四个显着特征,这为我们提供了梯度量化的有见地的线索。然后,我们从理论上对收敛界限进行了深入分析,并推导出稳定 INT8 训练的两个原则。最后,我们提出了两种通用技术,包括减少梯度方向偏差的Direction Sensitive Gradient Clipping和避免沿错误方向更新ille gal梯度的Deviation Counteractive Learning Rate Scaling。实验表明,我们的统一解决方案有望为各种网络和任务提供准确高效的 INT8 训练,包括 MobileNetV2、InceptionV3 和先前研究从未成功过的目标检测。此外,它具有在现成硬件上运行的强大灵活性,无需过多的优化工作即可在 Pascal GPU 上将训练时间减少 22%。我们相信这项开创性研究将有助于引领社区走向完全统一的 INT8 卷积神经网络训练。

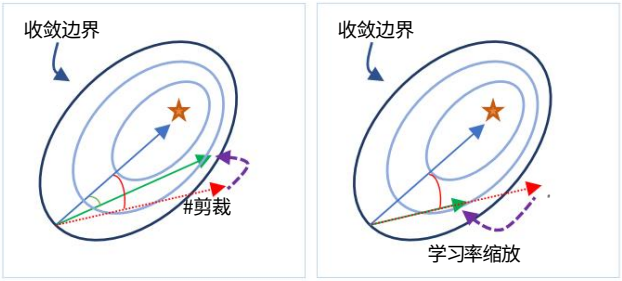



图 1. 我们统一 INT8 训练的基本思想。 gx和^gx分别表示原始浮点梯度和量化梯度。 α和β表示量化带来的不同方向偏差。当方向偏差较大时,红线表示碰撞情况。左子图表示在收敛边界内适当裁剪梯度以减少方向偏差可以避免崩溃。右边的子图指出,控制学习率（步长)可以通过抵消偏差的负面影响来保证稳定的参数更新。

## 一、简介

深度卷积神经网络（DCNNs)在计算机视觉、自然语言处理等许多领域取得了显着的成功。然而,训练和部署 DCNNs 通常需要大量的时间成本和功耗,这对人工智能提出了极大的挑战。工业上的广泛应用。

因此,最近的许多研究都集中在如何通过对权重或激活的定点量化来加速神经网络的推理[6, 3, 24, 29, 27, 64, 42, 52, 63, 54, 21 , 44, 57],并利用高效整数算法设计专用硬件[17, 5, 26, 23]。成功的进展令人惊讶地表明,位宽可以减少到极低的水平,例如 4 位,同时对推理的准确性几乎没有影响[15、59、13 ] 。

除了推理之外,低位训练还可以保证相当大的加速,进一步量化梯度并利用低位高效计算内核进行前向和反向传播。正如[25]中分析的那样,

*通讯作者

反向传播的计算比正向传播占用更多的时间。因此,在考虑反向过程时,利用低位量化加速训练具有更大的潜力。已经存在16位浮点（FP16)训练,证明了低位训练的可行性[41,9,33 ] 。但它仅限于基于 Turing 或 Volta 架构的有限高级 GPU。

与FP16相比,8位整数（INT8)运算得到了基于Turing、Volta甚至低端Pascal架构的通用GPU的广泛支持。此外,8 位整数运算在理论上和实践上比 FP16 快 2 倍,比 FP32 快 4 倍。因此,INT8 训练在现成的硬件上具有更高的效率、更低的功耗和更好的通用性。

尽管有诱人的好处,但当将梯度量化为 8 位时,正常的训练往往会变得不稳定,因为梯度的失真很容易误导训练的方向并导致优化崩溃。这无疑使 INT8 训练变得非常困难,尤其是对于深度网络。目前只有少数研究试图解决这个问题[64, 56, 60, 2, 53, 48]。不幸的是,他们都只是测试了具有高冗余度的有限量化友好网络,并且通常需要复杂的结构调整或引入额外的操作来减少量化误差,同时显着增加了计算复杂度。此外,这些作品大多缺乏对 ad-hoc 技巧的理论分析,更糟糕的是,它们都没有报告真实案例中的实际加速。所有这些原因使得现有的INT8训练方法在没有通用设计的情况下与实用性相去甚远。

为了构建一个强大而统一的 INT8 训练框架,我们对梯度量化的挑战进行了更深入的探索。我们凭经验发现梯度的分布具有四个特点:尖而宽、进化、深度特定和结构特定。这些独特的特性使得梯度量化与权重或激活的朴素量化有很大不同,并且 INT8 训练更难以稳定。重要的是要了解量化梯度在训练收敛中的行为和影响。因此,我们在理论上建立了关于梯度量化误差和学习率的收敛界限。

基于特殊特性和理论分析,我们提出了两种通用技术:Direction Sensitive Gradient Clipping 和 Deviation Counteractive Learning Rate Scaling 来稳定 INT8 训练。 Direction Sensitive Gradient Clipping 通过在训练过程中进行适当的裁剪来最小化方向偏差。有时,即使裁剪有助于减少量化误差,它仍可能会受到深层累积梯度偏差的影响。为了消除这种影响,偏差反作用学习进一步设计速率缩放以保证稳定的参数更新。我们方法的基本思想如图 1 所示。对各种网络结构和任务的大量实验证明了我们方法的优越性和通用性。

我们的贡献可以总结如下：

- 我们观察到梯度分布的四个特点:尖而宽、进化、深度特定和结构特定,这导致梯度的量化误差较大。
- 我们在理论上提供了INT8 训练的收敛界限,并分别设计了两种可以稳定 INT8 训练的通用技术。
- 我们率先实现了对MobileNetV2/InceptionV3 等各种网络和目标检测等各种任务的稳定INT8 训练,其精度可与全精度训练相媲美。
- 我们使用各种网络为各种任务构建灵活统一的INT8 训练框架,可以轻松替代原来的全精度训练。
- 我们率先在具有Pascal 架构的低端GPU（即NVIDIA GeForce GTX 1080Ti)上完成了INT8 训练的实际加速,在没有过多优化的情况下实现了约22% 的加速。

## 二、相关工作

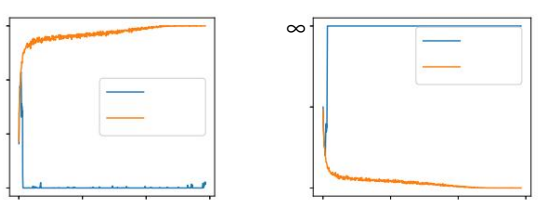
与大量通过模型量化加速推理的研究相比[45、62、7、52、11、40 ] ,很少有工作全面探索包括反向传播在内的量化训练。 DoReFa-Net [64]将梯度量化为 4 位和 6 位,但只用低精度梯度对 AlexNet 进行实验。 WAGE [56]和 WAGEUBN [60]将梯度量化为 8 位整数,但它们都会导致相当大的精度损失（大于5%）。

RangeBN [2]和 FP8 training [53]达到了与全精度模型相当的精度,但它们都在梯度中使用浮点数,这不利于硬件优化来提高速度。除了量化训练外,大多数低精度训练研究都将梯度精度保持在 16 位浮点数。 Flexpoint [33]、 MPT [41]和 DFP [9]都使用 16 位浮点来训练 DNN,其精度与全精度模型相当。

为了更有效地训练神经网络,INT8 训练比 FP16 训练更具优势。

## 3.统一INT8训练

在本文中,我们旨在构建一个统一的 INT8 训练框架,该框架利用 8 位整数运算来加速深度神经网络的昂贵训练过程,包括前向和反向传播。



(a) 精度曲线　　(b) 损失曲线

图 2. 将梯度量化为 8 位后,MobileNetV2 在 CIFAR-10 上的训练崩溃。

### 3.1.预赛

对称均匀量化[30]是现有量化方法中最有效的方案,因为它对硬件计算友好。因此,为保证加速性能,我们构建了基于它的INT8训练框架。给定范围(l, u)内的数据x （即权重、激活和梯度)和裁剪值c ∈ (0, max(|l|, |u|)],对称均匀量化可以表述为：

$$q = \text{round}\left(\frac{\text{clip}(x, c)}{s}\right), \quad (1)$$

其中clip(x, c) = min(max(x, −c), c), s = in表示将浮点数投影到定点8位整数的比例因子， q表示quan定点数。随后,相应的反量化数据x^可以通过以下方式计算：

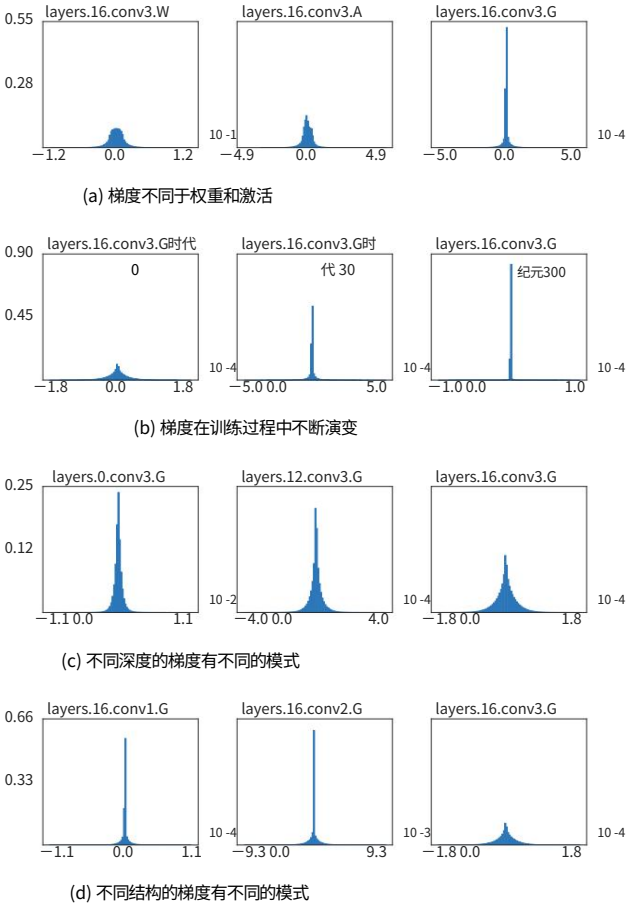
$$\hat{x} = q \cdot s。 \quad (2)$$

与大多数主要关注加速推理（即前向传播)的先前研究不同,我们的INT8 训练框架试图通过对梯度应用量化来进一步加速训练阶段的反向传播。也就是说,我们以适当的方式从全精度梯度g中追求量化-反量化梯度g^g 。

为了确保量化梯度与原始梯度相比保持无偏期望,我们采用了[16] 之后的随机舍入：

$$\text{回合}(x) = \begin{cases} \lfloor x \rfloor, & \text{wp } 1 - (x - \lfloor x \rfloor) \\ \lfloor x \rfloor + 1, & \text{wp } x - \lfloor x \rfloor \end{cases} \cdot \quad (3)$$

不幸的是,虽然随机舍入技术从统计角度在一定程度上限制了量化误差,但每次训练迭代的扰动仍然是不可避免的并且不利于收敛,其原因将在下一节中讨论。




(a) 梯度不同于权重和激活

(b) 梯度在训练过程中不断演变

(c) 不同深度的梯度有不同的模式

(d) 不同结构的梯度有不同的模式

图 3. MobileNetV2 不同层和训练迭代的激活分布、权重和梯度。

### 3.2.梯度量化的挑战

梯度决定了优化的方向和参数更新的幅度,因此在追求高精度模型中起着至关重要的作用。在 INT8 训练中,在我们对梯度应用量化之后,in的扰动会导致优化方向的偏差。一旦偏差累积到不可接受的程度,训练过程可能会不稳定甚至崩溃,从而导致严重的性能下降。图2显示了我们的经验观察,对于一些特殊的网络架构,如 MobileNetV2,直接量化梯度会导致训练快速崩溃。

为了进一步研究这种现象背后的本质原因,我们对没有梯度量化的训练过程中的梯度分布进行了详细分析,如图 3 所示。我们惊奇地观察到梯度具有以下独特的特征：

C1:锐利而宽阔。如图 3(a) 所示,与权重和激活相比,梯度遵循一种不寻常的分布,即更多的值集中在零附近,同时一定数量的极值也来自

主义者。因此,分布曲线非常尖锐,小值占据了大部分梯度,但范围相对非常宽。这使得许多梯度量化为零,并且在使用均匀量化时量化误差非常大。

C2:进化的。图3(b) 描述了同一层的梯度分布如何随着训练迭代而演变。我们可以发现,随着训练的进行,梯度分布的形状变得越来越尖锐和越来越窄,这意味着不可能像我们通常对权重和激活所做的那样,在整个训练过程中固定量化设置,例如就像在整个训练中假设相同的剪裁范围一样。

C3:深度特定。图3(c) 比较了同一时期不同层的梯度分布。很明显,浅层的分布比深层更尖锐,极值更大。这意味着神经网络的前几层往往面临更严重的量化损失。

C4:结构特异性。从图 3(d) 可以看出,不同结构层在同一时期的梯度呈现出明显不同的模式。对于 MobileNetV2,每个块中的第二个卷积层都是深度结构。它的梯度即使在更深的块中也具有更大的范围和更尖锐的形状,这使得 MobileNetV2 从梯度方面更难量化。

基于以上观察,我们可以得出结论,当简单地对权重和激活采用常见的量化技术时,梯度与权重和激活有很大的不同,这不可避免地导致训练不稳定。这意味着我们需要一定的技术来处理梯度量化中的独特性,这给实践中真正统一的 INT8 训练带来了巨大挑战。

在考虑梯度的特性来设计所需的技术之前,我们首先尝试通过从理论上揭示训练收敛和梯度量化之间的联系来理解梯度对训练稳定性的影响。这将为我们构建稳健统一的 INT8 训练框架提供可靠线索。

## 3.3.稳定训练:理论视角

正如深度学习优化器分析中常用的[12,28,46,39 ] ,收敛能力通常由遗憾R(T)来评估。

$$R(T) = \sum_{t=1}^{吨} (英尺(重量) - 英尺(w^* )), \quad (4)$$

其中T表示迭代次数。 wt ∈ S是凸紧集S中时间t的参数， ft(wt)表示对应的损失函数。最优的 pa R(T)参数用w* T表示,随着T 的增加,T迅速趋近于零,可以保证收敛的速度和能力。

· 如果平均后悔 ——

由于 DCNN 的复杂性,很难直接分析其行为。正如之前的研究[1, 34, 22, 61]所做的那样,我们首先做出以下假设:

假设 1. ft是凸的;

假设 2.∀wi , wj ∈ S , wi − wj∞ ⩽ D∞。

尽管凸性假设可能不适用于深度网络,但基于此的分析可以为我们提供合理且有价值的见解,这已在先前的研究中得到证明[12,39,22,61 ] 。

考虑到标准的随机梯度下降算法,基于量化梯度^gt和学习率ηt的优化可以表示为:

$$wt+1 = wt - ηt\hat{}gt。 \quad (5)$$

然后我们有以下理论发现（详细证明见补充材料）：

定理 1.如果将量化梯度的误差定义为 t = gt − ^gt并将权重的维度定义为 d,则根据假设1和2,我们有:

$$\frac{R(T)}{吨} \leqslant \underbrace{\frac{dD_\infty^2}{2T\,\eta T}}_{(1)} \underbrace{\frac{D_\infty}{吨}\sum_{t=1}^{吨}}_{(2)} \underbrace{\sum_{t=1}^{吨}\left(1+\frac{\eta t}{吨}\right)^{2}\,{}^{>GT}}_{(3)}. \quad (6)$$

我们可以发现,平均后悔的界限由三项决定。随着T 的增加,项 (1) 趋近于零,因此在梯度量化中可以忽略。项（2)表示梯度的量化误差极大地影响收敛能力,并且通常很大,如3.2节中分析的那样。对于第（3)项,其大小主要受学习率和量化梯度的 l2-范数的影响。基于理论分析,为了稳定 INT8 训练,我们有两个基本原则来设计更好的量化技术：(1)减少梯度的量化误差； (2) 缩小学习率。它们也非常直观,因为一方面,较低的量化误差意味着优化方向的小偏差,从而避免了训练崩溃,另一方面,逐渐降低学习率是一种常识,可以提供更好的解决方案优化。

现在有了设计原则,问题是如何设计 INT8 训练的通用技术,同时考虑梯度的特性。我们分别介绍了两种新技术:Direc

Table 1. KS-statistics of gradient and weight respect to different layers conv3 in MobiletNetV2, 最后一列表示可以接受显着性水平为0.05的假设的最大值。

| 数据 | 分配 | | | 临界值 |
|---|---|---|---|---|
| | 高斯拉普拉斯学生g 0.1934 w 0.0391 | | | |
| 层0 | | | 0.0790 0.2005<br>0.0721 0.1011 | 0.0012<br>0.0765 |
| 第八层 | g 0.2061 w<br>0.0294 | | 0.1091 0.2303<br>0.0569 0.1084 | 0.0024<br>0.0110 |

敏感梯度裁剪和偏差反作用学习率缩放,它们共同降低了平均后悔界限并保证了稳定的 INT8 训练。

## 3.4.方向敏感梯度裁剪

考虑到深度神经网络中的基本操作z = W a ,权重gW的梯度实际上可以通过gz a来计算。从这个方面来看,式(6)中gW的量化误差主要源于激活梯度gz的量化误差。因此,在我们的 INT8 训练中,我们可以主要关注 gz 的量化,这将有助于控制（6)中量化梯度的误差。为了符号的简单,在下面的讨论中我们直接使用g来表示gz。

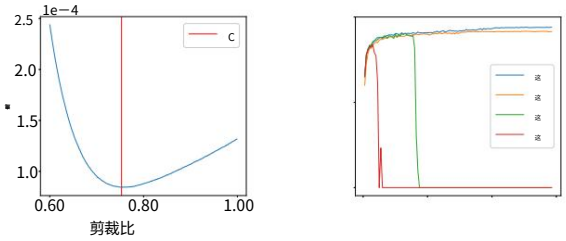
为了最小化量化误差,以前的工作主要通过假设一定的数据分布,例如高斯分布[3,4,20,2,22,11,55,58 ]来寻找（1)中的最佳裁剪值c 。然而,根据我们发现的梯度特征 C1 和 C2,对进化的和不寻常的梯度分布做出共同假设是不切实际的。为了进一步证明这一点,我们进行了 Kolmogorov–Smirnov 检验[50],其中分布参数通过最大似然估计求解,并报告了 KS 统计量,该统计量始终拒绝梯度服从表 1 中的任何公共分布的假设。

为了在没有任何假设的情况下找到最佳裁剪值c ,一个直接的想法是通过梯度下降算法使量化梯度与原始梯度保持一致。通常,可以使用流行的均方误差 (MSE) 对一致性进行建模。不幸的是,由于梯度的特征 C2 和 C3 具有巨大的差异和幅度波动,MSE 使得优化变得脆弱并且无法在跨不同层的相同简单设置下工作。

因此,为了追求保证稳定训练的不同层的期望裁剪值,我们选择余弦距离来指导裁剪值的学习,这不仅避免了不同梯度幅度的负面影响,而且保持了网络优化方向一致:

$$\hat{g}\,dc = 1 - \cos(\langle g, \hat{g} \rangle) = 1 - \frac{g \cdot \hat{g}}{|g| \cdot |\hat{g}|} \qquad (7)$$

其中g和^ g表示原始浮点梯度及其量化-反量化对应物。




(a) 剪裁效果 (b) 扩展策略的效果

图 4. 削波和学习率对 INT8 训练的影响。 (a) 中的 γ 表示最佳裁剪值。在 (b) 中,η1 将初始学习率设置为 0.1,具有 ϕ(dc)缩放,η2、η3 和 η4 分别选择 0.01、0.05、0.1 作为初始学习率,没有缩放。

余弦距离测量的方向偏差

量化梯度,以及余弦距离和训练稳定性之间的强相关性将在补充材料中得到验证。通过最小化余弦距离,我们随后减少了(6) 中的项 (2)。图4(a) 显示了使用不同限幅值的量化误差,其中存在一个显着减小余弦距离的最佳限幅值。

## 3.5.偏差反作用学习率缩放

对量化训练收敛能力的理论分析表明有必要按比例缩小学习率,因为梯度的量化误差不能完全消失。为了验证这一点,我们降低了3.2节中提到的 MobileNetV2 原始崩溃训练的学习率,发现它以极低的学习率推迟甚至消除了崩溃,尽管面临性能下降（见红色,图 4(b) 中的绿色和橙色线条）。

由于梯度是逐层反向传播的,微小的梯度偏差在经过大量的乘法和加法计算后会呈指数级累积。为了解决这个问题,我们进一步提出了 Deviation Counteractive Learning Rate Scaling,通过根据方向偏差dc的程度指数衰减学习率来平衡误差,缩放函数公式为:

$$\phi(dc) = \text{最大值}(e^{-\alpha dc}, \beta) \qquad (8)$$

其中α控制衰减程度， β限制缩放的下限。

这个缩放函数生成一个因子来缩小原始的全精度学习率。我们凭经验发现自适应缩放函数在分层方式中表现良好,根据不同层的方向偏差自适应调整学习率。这抵消了跨层梯度偏差的不良影响,并恰好解决了

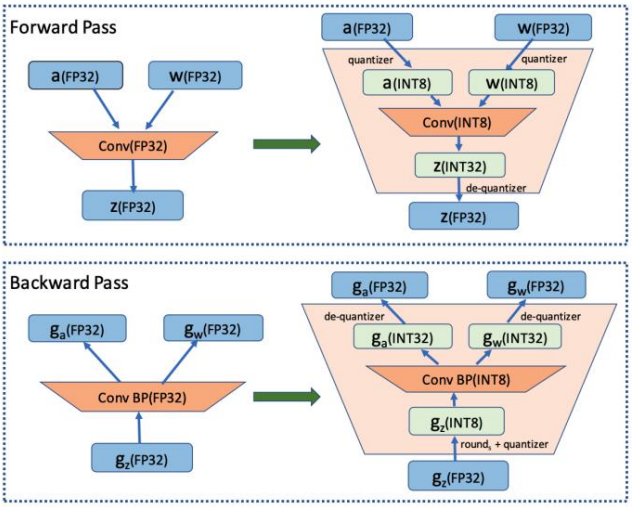



图 5. 灵活的 INT8 卷积层替换。

表 2. 通过定期更新减少开销（在 ResNet-50 上）。

| 第 100 期 | | 10 | | 1000 |
|---|---|---|---|---|
| 平均时间(s/iter) 1.006 0.364 0.301 0.297 | | | | |

如第 3.2 节中的特征 C3 和 C4 中观察到的深度特定和结构特定模式。图 4(b) 中的蓝线表明配备ϕ(dc)缩放的训练比手动调整的训练具有更高的精度（在 CIFAR-10 上使用 MobileNetV2 进行测试）。

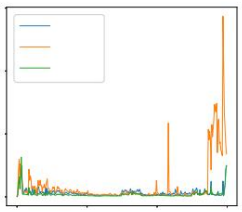
## 3.6.通用培训框架

除了确保稳定和准确的收敛外，在实践中我们的统一 INT8 训练框架还应满足以下三个特点：(1)易于插入任何 DCNN 架构。为了实现这一点，我们在 PyTorch [43]中实现了一种自动匹配和替换机制，相应地用 8 位对应物替换卷积层和全连接层。包括前向和后向传递的整个工作流程如图 5 所示。

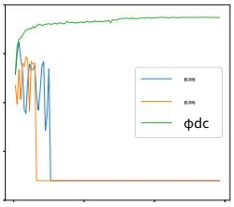
(2)没有过多的额外计算开销。为了避免计算裁剪值的额外时间成本，我们设计了一种 Periodic Update 方法来定期优化裁剪值。正如我们在表 2 中看到的，定期更新方法显着降低了优化限幅值的计算开销。

(3)易于在现成的硬件上实现。为了验证其潜力，我们利用低端 NVIDIA Pascal GPU 上的 DP4A 指令（8 位整数 4 元素向量点积）来实现高效的 8 位内核来计算梯度。据我们所知，我们是第一个实现 INT8 训练实际加速的人，包括反向传播。详细的加速将在第 4.4 节中报告和讨论。

对于特征 F3，INT8 列车的实际加速 -



(a) 余弦距离




(b) 准确率曲线图6.余弦距

离与学习率缩放函数分析。

ing依赖于硬件是否支持8位高效指令。由于 8 位定点运算已被广泛用于推理，大多数现有硬件都具有加速训练的能力，包括反向传播，例如华为的 Ascend310 [23]、寒武纪的 MLU100 [5]、谷歌的TPU [26]以及几乎所有来自 NVIDIA [18] 的 GPU。为了验证加速反向传播的潜力，我们在具有 Pascal 架构的低端 GPU 上使用 DP4A 指令来实现高效的 8 位内核来计算梯度。

## 4. 实验

我们进行了大量的实验，以证明我们提出的框架在流行的图像分类和目标检测任务上以最先进的精度统一于各种网络结构，同时它可以很容易地部署在主流设备上（NVIDIA Pascal GPU ）与全精度训练相比，具有令人满意的加速。

## 4.1.消融研究

设置。我们首先使用 MobileNetV2 [49]对 CIFAR-10 数据集进行消融研究，以验证所提出技术的有效性。对于所有实验，我们使用初始学习率设置为 0.1 的余弦调度程序[1] 。在 Periodic Update 实验中，学习率缩放中的α和β分别设置为 20 和 0.1。

方向敏感梯度裁剪。图6(a) 显示了关于训练步骤的余弦距离。我们可以观察到每个块的 conv2（第二个卷积层）在大多数情况下拥有比其他层大得多的余弦距离。这与 C4 一致，即 conv2 的梯度具有更尖锐的形状，表明余弦距离可以很好地反映梯度特征。

此外，如表3所列，我们提出的方向敏感梯度裁剪技术确实可以防止 INT8 训练崩溃，这证明优化梯度裁剪值以最小化方向偏差dc确实可以确保稳定的 INT8 训练。

偏差反作用学习率缩放。我们

表 3. INT8 训练裁剪方法的消融研究。

| 裁剪方式 不裁剪 | | 方向敏感梯度裁剪 |
|---|---|---|
| 准确性 （%） | 钠盐 | 93.02 |

评估三种形式的学习率缩放策略,无需剪裁以控制变量以进行合理比较。图 6(b) 所示的结果表明,线性和二次形式太弱,无法控制收敛边界内的优化方向,并且模型在训练过程中崩溃。与线性和二次形式相比,指数形式的缩放更能抵消方向偏差并防止优化超出收敛边界。我们在表 4 中进一步探讨了它对超参数选择的敏感性,我们可以看到α和β的不同设置达到了相似的精度,这表明我们的偏差反主动学习率缩放的稳定性。

表 4. 学习率缩放的不同超参数比较。

| A | 10 | 10 | 20 | 20 |
|---|---|---|---|---|
| b | 0.1 | 0.2 | 0.1 | 0.2 |
| 准确度 （%） 92.82 | 93.28 | 93.38 | 93.27 | |

裁剪值的定期更新。为了减少额外的计算开销,我们增加了更新裁剪值的周期,发现它对准确性几乎没有影响,如表 5 所示。这一经验结论为INT8 训练的实际加速提供了可能性。

此外,这里我们同时应用了梯度裁剪和学习率缩放,并获得了比表3和表 4 更好的性能（参见周期 1）。这进一步验证了这两种通用技术的积极效果。

表 5. 更新周期的消融研究。

| 期间 1000 | | 10 | 100 | |
|---|---|---|---|---|
| 准确度 （%） 93.66 | 93.07 | 93.38 | 92.75 | |

## 4.2.图片分类

现在我们考虑大多数先前研究选择评估量化性能的流行图像分类任务。我们在 CIFAR 10 [31 ]和 ImageNet (ILSVRC2012) [10]上对 AlexNet [32]、 ResNet [19]、 MobileNetV2 [49]和 InceptionV3 [51]进行了实验。 CIFAR-10 数据集包含 50K 图像的训练集和 10k 图像的测试集。每张图片大小为 32×32,有 10 个类别。 ImageNet (ILSVRC2012) 包含 120 万张训练图像和 5 万张测试图像,共 1000 个类别。

设置。至于 ResNet 的超参数,我们使用[19] 中描述的相同设置。对于其他神经网络,我们使用初始学习率设置为 0.1 的余弦调度程序[1] 。学习率缩放中的 α和β设置为 20 和 0.1

分别。所有实验的裁剪值每 100 次迭代更新一次。

CIFAR-10。如表6所示,我们的方法在 ResNet-20 和 FP8 训练上实现了相当的精度,但由于定点操作,占用的内存和计算消耗要少得多。此外,我们的方法在 MobileNetV2（精度下降 1.01%)和 InceptionV3（甚至优于全精度模型)上表现出奇的好。

图片网。表7列出了现有的最先进的量化训练方法,包括 WAGE [56]、WAGEUBN [60]和 FP8 训练[53]。对于 AlexNet INT8 训练,我们的方法比 DoReFa-Net [64] 提高了 5.84% 。

没有像 tanh 这样的额外开销,我们的方法比 DoReFa-Net 具有更高的效率。至于 2 位权重和 8 位激活/梯度情况,我们以大约 3% 的精度增益显着优于 WAGE。更重要的是,采用我们的方法,ResNet 架构的 INT8 训练几乎没有性能下降,而以前的研究都没有做到这一点。与 FP8 训练方法相比,我们的方法提高了近 3% 的准确率。需要注意的是,我们可以直接在流行的现成设备上获得真正的加速,而像 FP8 训练这样的方法需要专门设计的硬件,这意味着我们的框架对于统一训练加速更通用。

正如[36]中分析的那样,卷积层占据了大部分训练时间,而 BatchNorm 和 ReLU 等其他层则不是计算密集型的。因此,我们目前主要关注量化卷积层,而不量化像 RangeBN [2]和 WAGEUBN [60] 这样的 BatchNorm 层。即便如此,INT8 训练仍然有显着的加速。此外,我们可以获得与全精度训练相当的精度,远高于 RangeBN 和 WAGEUBN。

首次使用 INT8 训练的网络。据我们所知,我们是第一个量化 MobileNetV2 梯度的人,这在这个社区中是众所周知的困难。我们的方法在使用 MobileNetV2 的 CIFAR-10 和 ImageNet 数据集上都获得了非常好的性能,只有大约1% 的精度损失。我们还首次尝试在 InceptionV3 上进行 INT8 训练,并达到了与全精度模型相当的精度。请注意,对于 CIFAR-10 上的 InveptionV3,我们的 INT8 训练方法甚至可以获得比全精度模型更好的性能。

## 4.3.物体检测

为了证明我们方法的多功能性,我们进一步在两个广泛使用的数据集： PASCAL VOC [ 14 ]和可可[38]。 PASCAL VOC 数据集包含 20 个类别的 11k 图像。 COCO 数据集包含超过 20k 个图像和 80 个对象类别。注意

表 6. CIFAR-10 数据集的结果。

| 模型 | 方法 | 位宽（摇摆） | 准确性 （%） |
|---|---|---|---|
| ResNet-20 | 计划生育 | 32/32/32 | 92.32 |
| | FP8 训练[53] | 8/8/8 | 92.21 |
| | 我们的 | 8/8/8 | 91.95 |
| MobileNetV2 | 计划生育 | 32/32/32 | 94.39 |
| | 我们的 | 8/8/8 | 93.38 |
| 盗梦空间V3 | 计划生育 | 32/32/32 | 94.89 |
| | 我们的 | 8/8/8 | 95.00 |

表 7.ImageNet 数据集的结果。

| 模型 | 方法 | 位宽（摇摆） | 准确性 （%） |
|---|---|---|---|
| 亚历克斯网 | 计划生育 | 32/32/32 | 59.84 |
| | DoReFa-Net [64] | 8/8/8 | 53.00 |
| | 我们的 | 8/8/8 | 58.84 |
| | 工资[56] | 2/8/8 | 48.40 |
| | 我们的 | 2/8/8 | 51.28 |
| ResNet-18 | 计划生育 | 32/32/32 | 70.30 |
| | WAGEUBN [60] | 8/8/8 | 66.92 |
| | FP8 训练[53] | 8/8/8 | 67.34 |
| | 我们的 | 8/8/8 | 69.67 |
| ResNet-34 | 计划生育 | 32/32/32 | 73.68 |
| | WAGEUBN [60] | 8/8/8 | 68.50 |
| | 我们的 | 8/8/8 | 73.29 |
| ResNet-50 | 计划生育 | 32/32/32 | 76.60 |
| | WAGEUBN [60] | 8/8/8 | 69.07 |
| | 我们的 | 8/8/8 | 76.34 |
| MobileNetV2 | 计划生育 | 32/32/32 | 72.39 |
| | 我们的 | 8/8/8 | 71.20 |
| 盗梦空间V3 | 计划生育 | 32/32/32 | 77.28 |
| | 我们的 | 8/8/8 | 76.59 |

我们是第一个在目标检测任务上成功实现 INT8 训练的人。

设置。至于超参数，我们遵循[35] 中描述的相同规则。学习率缩放的α和β与图像分类任务中使用的相同。

帕斯卡挥发性有机化合物。我们用不同的主干测试了 RFCN 和 Faster R-CNN，发现我们的方法配备的量化训练仅遭受非常轻微的检测精度 (mAP) 下降。 RFCN 的结果表明，即使对于像 ResNet-101 这样更深的主干，我们的 INT8 训练仍然保持与全精度几乎相同的精度。

可可。在大规模 COCO 数据集上，我们使用 RetinaNet（单阶段）和 Faster R-CNN（两阶段）进行实验。
我们的方法在两个网络上执行稳定，精度下降不到 1.8% 。我们发现 RetinaNet 的 mAP 损失比 Faster R-CNN 更高，这与之前研究中的结论不一致[35]。这可能是由于一级检测器中使用的焦点损失对梯度量化更敏感。

表 8. PASCAL VOC 数据集的结果。

| 模型骨干法 | | | 位宽（摇摆） | 地图 （%） |
|---|---|---|---|---|
| 快点 R-CNN | ResNet-50 | 计划生育 | 32/32/32 | 82.0 |
| | ResNet-50 | 我们的 | 8/8/8 | 81.9 |
| RFCN | ResNet-101 | 计划生育 | 32/32/32 | 80.8 |
| | ResNet-101 | 我们的 | 8/8/8 | 79.1 |

表 9. COCO 数据集的结果。

| 模型 | 主干方法 | | 位宽 (W/A/G) | 地图 （%） |
|---|---|---|---|---|
| 快点 R-CNN | ResNet-50 | 计划生育 | 32/32/32 | 36.2 |
| | ResNet-50 | 我们的 | 8/8/8 | 34.95 |
| 视网膜网络 | ResNet-50 | 计划生育 | 32/32/32 | 36.9 |
| | ResNet-50 | 我们的 | 8/8/8 | 35.1 |

表 10. 一轮 INT8 训练的端到端平均时间。 （在 GeForce GTX1080TI 上使用 ResNet-50 进行测试，批量大小为 64。）

| 精确 | 向前 (s) | 向后 (s) | 迭代 (s) |
|---|---|---|---|
| FP32 (cuDNN) | 0.117 | 0.221 | 0.360 |
| INT8（我们的） | 0.101 | 0.171 | 0.293 |

## 4.4. NVIDIA GPU 上的速度结果

现有的库都不能直接支持完整的 INT8 训练。因此，我们使用 DP4A 指令在 NVIDIA Pascal GPU 上自行实现，以验证我们方法的加速能力。表10显示，在使用我们的解决方案的前向过程中，INT8 可以带来平均 1.63 倍的加速，而在后向过程中，它可以实现更高的1.94 倍加速。即使我们只用稍微优化的 INT8 卷积层替换 FP32 卷积层，ResNet-50 的训练时间也可以减少约 22%。有关速度结果的更多详细信息包含在补充材料中。

## 5。结论

在本文中，我们尝试为常见的 DCNN 构建一个 INT8 训练框架。我们发现了梯度的四个显着特征，然后给出了两个理论原则来稳定具有收敛边界的训练。在此基础上，我们提出了Direction Sensitive Gradient Clipping and Deviation Counteractive Learning Rate Scaling。大量实验证明了我们的方法对各种网络和任务的通用性。我们仅通过微不足道的优化就将 Pascal GPU 上的训练时间减少了 22%。如果每一层都得到充分优化，训练将实现更高的加速和更低的内存消耗。我们希望我们的第一次成功尝试能够帮助引导社区实现完全统一的 INT8 培训。

## 承认

这项工作得到了国家自然科学基金 (61872021、61690202)、北京科技新星计划 (z191100001119050) 和商汤集团有限公司的支持。

## 参考


[1] Dan Alistarh、Demjan Grubic、Jerry Li、Ryota Tomioka 和 Milan Vojnovic。 Qsgd:通过梯度量化和编码实现通信高效的 sgd。 In Advances in Neural Information Processing Systems 30,第 1709–1720 页。 2017. 4, 6, 7

[2] Ron Banner、Atay Hubara、Elad Hoffer 和 Daniel Soudry。 神经网络 8 位训练的可扩展方法,2018. 2, 5, 7

[3] Ron Banner、Yury Nahshan、Elad Hoffer 和 Daniel Soudry。 用于快速部署的卷积网络的训练后 4 位量化。 arXiv 预印本 arXiv:1810.05723, 2018. 1, 5

[4] Zhaowei Cai, Xiaodong He, Jian Sun, and Nuno Vasconce los. Deep learning with low precision by half-wave gaussian quantization. In CVPR, July 2017. 5 [5] Yunji Chen, Huiying Lan, Zidong Du, Shaoli Liu, Jinhua Tao, Dong Han, Tao Luo, Qi Guo, Ling Li, Yuan Xie, et al. 机器学习的指令集架构。 ACM 计算机系统交易 (TOCS), 36(3):9, 2019. 1, 6

[6] Jungwook Choi、Zhuo Wang、Swagath Venkataramani、Pierce I-Jen Chuang、Vijayalakshmi Srinivasan 和 Kailash Gopalakrishnan。 Pact:量化神经网络的参数化剪裁激活。 arXiv 预印本 arXiv:1805.06085, 2018. 1

[7] Matthieu Courbariaux、Yoshua Bengio 和 Jean-Pierre David。 Binaryconnect:在传播过程中使用二进制权重训练深度神经网络。 arXiv preprint arXiv:1511.00363, 2015. 2 [8] 戴继峰,李毅,何开明,孙健。 R-fcn:通过基于区域的全卷积网络进行目标检测。在第 30 届神经信息处理系统年会论文集中,2016 年 12 月。7

[9] Dipankar Das、Naveen Mellempudi、Dheevatsa Mudigere、Dhiraj Kalamkar、Sasikanth Avancha、Kunal Banerjee、Srinivas Sridharan、Karthik Vaidyanathan、Bharat Kaul、Evange los Georganas、Alexander Heinecke、Pradeep Dubey、Jesus Corbal、Nikita Shustrov、Roma Dubtsov、Evarist福缅科和瓦迪姆·皮罗戈夫。使用整数运算的卷积神经网络的混合精度训练。在 ICLR,2018 年 5 月。2 [10] J. Deng、W. Dong、R. Socher、L.-J。 Li、K. Li 和 L. Fei Fei。 ImageNet:大规模分层图像数据库。

2009 年 IEEE 计算机视觉和模式识别会议 (CVPR),2009 年 7 月。7

[11] 丁瑞舟、秦廷武、刘泽野和戴安娜·马尔·库莱斯库。正则化激活分布以训练二值化深度网络。在 CVPR,2019 年 6月。2、5

[12] John Duchi、Elad Hazan 和 Yoram Singer。用于在线学习和随机优化的自适应次梯度方法。机器学习研究杂志,12（7 月）:2121–2159,2011。4 [ 13 ] Steven K Esser、Jeffrey L McKinstry、Deepika Bablani、Rathinakumar Appuswamy 和 Dharmendra S Modha。

学习了步长量化。 arXiv 预印本 arXiv:1902.08153, 2019.1

[14] Mark Everingham、Luc Gool、Christopher K. Williams、John Winn 和 Andrew Zisserman。 Pascal 视觉对象类 (voc) 挑战。诠释。 J. 计算。愿景,2010 年 6 月。 7

[15] 龚瑞豪, 刘祥龙, 江胜虎, 李天祥, 胡鹏, 林家珍, 于凤伟, 严俊杰.可微分软量化:桥接全精度和低位神经网络。在 ICCV,2019 年 10 月。1 [16] Suyog Gupta、Ankur Agrawal、Kailash Gopalakrishnan 和 Pritish Narayanan。数值精度有限的深度学习。在第 32 届机器学习国际会议记录中,第 1737-1746 页,2015 年 7 月 7-9 日。3

[17] 韩松、刘星宇、毛惠子、朴靖、Ardavan Pe dram、Mark A. Horowitz 和 William J. Dally。诶诶ACM SIGARCH 计算机体系结构新闻,44(3):243–254,2016 年 6 月。1

[18] 马克·哈里斯。 cuda混合精度编程 8. https://devblogs.nvidia.com/mixed precision-programming-cuda-8/. 6 [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 用于图像识别的深度残差学习。 CVPR,2016 年 6 月。7

[20] 何哲志、范德良.使用截断高斯近似同时优化三元神经网络的权重和量化器。 In CVPR, June 2019. 5 [21] Lu Hou and James T. Kwok.深度网络的损失感知权重量化。 In ICLR, May 2018. 1 [22] Lu Hou, Ruiliang Zhang, and James T. Kwok.量化模型分析。在 ICLR 中,2019 年 5月。4、5

[23] 华为技术有限公司 升腾 310。https :// e.huawei.com/se/products/cloud computing-dc/atlas/ascend-310。 1, 6

[24] Benoit Jacob、Skirmantas Kligys、Bo Chen、Menglong Zhu、Matthew Tang、Andrew Howard、Hartwig Adam 和 Dmitry Kalenichenko。神经网络的量化和训练,用于高效的仅整数算术推理。 2018 年 IEEE 计算机视觉和模式识别会议 (CVPR),2018 年 6 月。1 [25] jcjohnson。 cnn-基准。 https://github.com/jcjohnson/cnn-benchmarks,2016.1 [ 26 ] Norman P. Jouppi、Cliff Young、Nishant Patil、David Patter son、Gaurav Agrawal、Raminder Bajwa、Sarah Bates、Suresh Bhatia、Nan Boden， Al Borchers 等人。张量处理单元的数据中心内性能分析。在第 44 届年度国际计算机体系结构研讨会论文集中,第 1-12 页。 ACM, 2017. 1, 6 [27] Sangil Jung、Changyong Son、Seohyung Lee、Jinwoo Son、Jae-Joon Han、Youngjun Kwak、Sung Ju Hwang 和 Changkyu Choi。通过使用任务损失优化量化间隔来学习量化深度网络。在 CVPR,2019 年 6 月。1

[28] Diederik P. Kingma 和 Jimmy Ba。 Adam:一种随机优化方法。在 ICLR,2015 年 5 月。4 [29] Raghuraman Krishnamoorthi。量化深度卷积网络以进行有效推理:白皮书。 arXiv 预印本 arXiv:1806.08342, 2018. 1

[30] Raghuraman Krishnamoorthi。量化深度卷积网络以实现高效推理:白皮书,2018 年。3

[31] Alex Krizhevsky、Vinod Nair 和 Geoffrey Hinton。 cifar-10 数据集。在线:http://www. CS。多伦多。 edu/kriz/cifar。 html,第 4 页,2014 年。7 [32] Alex Krizhevsky、Ilya Sutskever 和 Geoffrey E. Hinton。

使用深度卷积神经网络进行 Imagenet 分类。在第 25 届国际神经信息处理系统会议论文集 - 第 1 卷,NIPS 12,2012 年。7

[33] Urs Koster、Tristan J. Webb、Xin Wang、Marcel Nassar、Arjun K. Bansal、William H. Constable、Oguz H. Elibol、Scott Gray、Stewart Hall、Luke Hornof、Amir Khosrowshahi、Carey Kloss、Ruby J. Pai 和 Naveen Rao。 Flexpoint:一种用于高效训练深度神经网络的自适应数字格式。在第 31 届神经信息处理系统年会论文集中,2017 年 12 月。2 [ 34] Hao Li、Soham De、Zheng Xu、Christoph Studer、Hanan Samet 和 Tom Goldstein。训练量化网络:更深入的理解。神经信息处理系统进展 30,第 5811-5821 页。 2017. 4

[35] 李润东, 王艳, 冯亮, 秦宏伟, 严俊杰, 范睿.用于对象检测的完全量化网络。在 IEEE 计算机视觉和模式识别会议 (CVPR) 上,2019 年 6 月。8 [36] Xiaqing Li、Guangyan Zhang、H Howie Huang、Zhufan Wang 和 Weimin Zheng。基于gpu的卷积神经网络的性能分析。 2016 年第 45 届国际并行处理会议 (ICPP),第 67-76 页。 IEEE,2016. 7 [37] Tsung-Yi Lin、Priya Goyal、Ross Girshick、Kaiming He 和 Piotr Dollar。密集物体检测的焦点损失。在 ICCV,2017 年 10 月。7

[38] Tsung-Yi Lin、Michael Maire、Serge Belongie、James Hays、Pietro Perona、Deva Ramanan、Piotr Dollar 和 C. Lawrence Zitnick。 Microsoft coconut:上下文中的常见对象。 In David Fleet, Thomas Pajdla, Bernt Schiele, and Tinne Tuyte laars, editors, Computer Vision – ECCV 2014, pages 740–755. Springer International Publishing, 2014. 7 [39] Liangchen Luo, Yuanhao Xiong, Yan Liu, and Xu Sun ……

具有学习率动态界限的自适应梯度方法。在 ICLR,2019 年 5 月。4 [40] Jeffrey L. McKinstry、Steven K. Esser、Rathinakumar App puswamy、Deepika Bablani、John V. Arthur、Izzet B. Yildiz 和 Dharmendra S. Modha。发现低精度网络接近全精度网络以实现高效的嵌入式推理。 arXiv 预印本 arXiv:1809.04191, 2018.2 [ 41] Paulius Micikevicius、Sharan Narang、Jonah Alben、Gregory Diamos、Erich Elsen、David Garcia、Boris Ginsburg、Michael Houston、Oleksii Kuchaiev、Ganesh Venkatesh 和 Hao Wuv。混合精度训练。在 ICLR,2018 年 5 月。2 [42] Asit Mishra 和 Debbie Marr。徒弟:使用知识蒸馏技术提高低精度网络准确率。 arXiv 预印本 arXiv:1711.05852, 2017. 1

[43] Adam Paszke、Sam Gross、Soumith Chintala、Gregory Chanan、Edward Yang、Zachary DeVito、Zeming Lin、Al 禁止 Desmaison、Luca Antiga 和 Adam Lerer。 pytorch 中的自动微分。 2017. 6

[44] 秦昊彤, 龚瑞昊, 刘翔龙, 沉明珠, 魏自然, 于凤伟, 宋敬宽. Ir-net:高精度二进制神经网络的前向和后向信息保留。在 IEEE 计算机视觉和模式识别 (CVPR) 会议上,2020 年 6 月。1 [45] Mohammad Rastegari、Vicente Ordonez、Joseph Redmon 和 Ali Farhadi。 Xnor-net:使用二进制卷积神经网络的 Imagenet 分类。计算机科学讲义,第 525-542 页,2016 年。2

[46] Sashank J. Reddi、Satyen Kale 和 Sanjiv Kumar。关于亚当和超越的融合。 In ICLR, May 2018. 4 [47] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun.

Faster r-cnn:使用区域建议网络进行实时对象检测。在第 29 届神经信息处理系统年会论文集中,2015 年 12 月。7

[48] Charbel Sakr 和 Naresh Shanbhag。反向传播算法的每张量定点量化。在 ICLR 中,2019 年 5 月。2

[49] 马克·桑德勒、安德鲁·霍华德、朱梦龙、Andrey Zh moginov 和 Liang-Chieh Chen。 Mobilenetv2:倒置残差和线性瓶颈。在 CVPR,2018 年 6月。6、7

[50] N.斯米尔诺夫。用于估计经验分布的拟合优度的表。数理统计年鉴,19(2):279–281, 1948. 5

[51] Christian Szegedy、Vincent Vanhoucke、Sergey Ioffe、Jonathon Shlens 和 Zbigniew Wojna。重新思考计算机视觉的初始架构。 2016 年 IEEE 计算机视觉和模式识别会议 (CVPR),2016 年 6 月。7

[52] Kuan Wang, Zhijian Liu, Yujun Lin, Ji Lin, and Song Han. Haq: Hardware-aware automated quantization. arXiv preprint arXiv:1811.08886, 2018. 1, 2

[53] Naigang Wang、Jungwook Choi、Daniel Brand、Chia-Yu Chen 和 Kailash Gopalakrishnan。使用 8 位浮点数训练深度神经网络。在第 32 届神经信息处理系统年会论文集中,2018 年 12月。2、7、8

[54] Peisong Wang, Qinghao Hu, Yifan Zhang, Chunjie Zhang, Yang Liu, and Jian Cheng. Two-step quantization for low-bit neural networks. IEEE CVPR, June 2018. 1 [55] Yiru Wang, Weihao Gan, Jie Yang, Wei Wu, and Junjie Yan.

不平衡数据分类的动态课程学习。在 IEEE 计算机视觉国际会议 (ICCV),2019 年 10 月。5

[56] 吴爽,李国启,陈峰,施路平.在深度神经网络中使用整数进行训练和推理。在 ICLR 中,2018 年5月。2、7、8

[57] Yudong Wu, Yichao Wu, Ruihao Gong, Yuanhao Lv, Ken Chen, Ding Liang, Xiaolin Hu, Xianglong Liu, and Junjie Yan. Rotation consistent margin loss for efficient low-bit face recognition. In The IEEE Conference on Computer Vi sion and Pattern Recognition (CVPR), June 2020. 1

[58] Jie Yang, Jiarou Fan, Yiru Wang, Yige Wang, Weihao Gan, Lin Liu, and Wei Wu. Hierarchical feature embedding for attribute recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2020. 5 [59] Jiwei Yang, Xu Shen, Jun Xing, Xinmei Tian, Houqiang Li, Bing Deng, Jianqiang Huang, and Xian-sheng Hua. Quanti zation networks. In CVPR, June 2019. 1 [60] Yukuan Yang, Shuang Wu, Lei Deng, Tianyi Yan, Yuan Xie, and Guoqi Li. Training high-performance and large-scale deep neural networks with full 8-bit integers, 2019. 2, 7, 8 [61] Penghang Yin, Shuai Zhang, Jiancheng Lyu, Stanley Osher, Yingyong Qi, and Jack Xin. Blended coarse gradient descent for full quantization of deep neural networks. Research in the Mathematical Sciences, 6(1):14, 2019. 4

[62] 张冬青,杨蛟龙,叶东强子,华钢． Lq-nets:用于高度准确和紧凑的深度神经网络的学习量化。在 ECCV,2018 年 9 月。2

[63] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. arXiv preprint arXiv:1702.03044, 2017. 1 [64] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convo lutional neural networks with low bitwidth gradients. CoRR, abs/1606.06160, 2016. 1, 2, 7, 8