

三维点云深度模型压缩算法*

赵志¹, 许可¹, 马燕新², 万建伟¹

(1. 国防科技大学 电子科学学院, 湖南 长沙 410073; 2. 国防科技大学 气象海洋学院, 湖南 长沙 410073)

摘要:随着计算机三维视觉的广泛应用,近几年基于深度学习的点云处理算法得到了大量研究,而耗时耗存储的缺陷较大程度限制了其在移动端的部署应用。基于改进损失函数的总体思路,提出了一种新的点云深度模型压缩框架,将知识蒸馏方法引入二值量化模型中,同时考虑点云聚合操作的特殊性引入了辅助损失项,改进的损失函数共包括预测损失项、蒸馏损失项和辅助损失项三部分。实验结果表明,和已有算法相比,所提算法可以获取更高的精度,同时对当前点云主流深度网络模型也具有较好的扩展性。

关键词:点云;知识蒸馏;二值量化;损失函数

中图分类号:TN911.7 **文献标志码:**A **文章编号:**1001-2486(2023)05-193-09

Compression method for three-dimensional point cloud deep model

ZHAO Zhi¹, XU Ke¹, MA Yanxin², WAN Jianwei¹

(1. College of Electronic Science and Technology, National University of Defense Technology, Changsha 410073, China;

2. College of Meteorology and Oceanography, National University of Defense Technology, Changsha 410073, China)

Abstract: With the widespread application of computer three-dimensional vision, point cloud processing algorithms based on deep learning have attracted a lot of research in recent years, however, the time and storage consuming characteristics have greatly restricted its deployment and application on the mobile terminal devices. Based on the general idea of improving the loss function, a new point cloud deep model compression framework was proposed, and the knowledge distillation method was introduced into the binary quantization model. At the same time, considering the speciality of the point cloud aggregation operation, an auxiliary loss item was introduced. The improved loss function includes three parts: prediction loss, distillation loss and auxiliary loss. The experimental results show that, compared with the existing algorithms, the proposed algorithm can obtain higher accuracy, meanwhile, the application to current mainstream point cloud deep network models can also achieve good scalability.

Keywords: point cloud; knowledge distillation; binary quantization; loss function

随着点云在自动驾驶、机器人、虚拟现实和增强现实等领域的应用需求不断增加,基于深度学习的点云网络模型得到了广泛的研究^[1-9]。这些成果虽然取得了较为满意的结果,但耗时耗能耗资源的缺陷极大地限制了其在资源有限的移动端设备上的应用。所以如何使现有的深度模型真正落地应用是值得进一步探索的。虽然当前一些研究成果^[7-9]在推理速度和存储上的性能有一定提升,但其仍然是基于昂贵的浮点数运算,占用了较多的计算资源。

网络量化作为模型压缩的一种重要方式,是解决该问题的有效途径,即把高位表示的权值或者激活值用较低位来近似表示,将连续的数值离散化,主要包括8位、4位、2位和1位量化。尤其

对于1位,即二值量化,只存在两个数值-1(0)或+1,烦琐的矩阵乘法可以用简便的按位XNOR操作和Bitcount操作替代,可以实现最大的压缩比,进而最大限度地解决网络模型耗时耗存储的问题^[10-18]。但二值量化的同时伴随的是精度的大幅度下降,如何提升二值量化网络模型的精度一直是国内外研究的热点,也是模型量化领域最具挑战性的问题。

当前关于模型二值量化的研究在图像领域研究较多,而在点云领域研究很少。包括BNN^[16]和XNOR^[17]在内的典型二值量化方法,在二维图像领域已得到了广泛的应用,表明二值量化具有重要的应用价值。而由于图像和点云之间存在的根本性差异,图像网络适用的二值量化方法并

* 收稿日期:2021-08-11

基金项目:国家自然科学基金资助项目(61871386)

作者简介:赵志(1986—),男,山西朔州人,博士研究生,E-mail:zhaozhi@nudt.edu.cn;

许可(通信作者),男,四川乐山人,副教授,博士,硕士生导师,E-mail:xuke@nudt.edu.cn

不能直接移植到点云网络上。点云网络通过设计对称函数(如池化操作)来聚合点的特征以更好地处理点云的无序性。而不同于全精度网络,对于二值量化来说,池化操作较大程度地改变了隐藏层特征的统计特性,导致特征的可区分性下降较为明显。北京航空航天大学研究团队^[19]提出的 BiPointnet 是近年来针对点云网络模型量化压缩研究的重要成果,通过设计最大熵聚合(entropy maximizing aggregation, EMA)模块和分层尺度恢复(layer-wise scale recovery, LSR)模块,尤其是最大熵聚合改善了池化层操作造成的特征同质退化问题,有效提升了全局特征的表示能力,较大程度地降低了量化导致的较大尺度变形。

当前,点云网络模型压缩的研究刚刚起步,仍存在较多的问题亟待解决。二维视觉模型量化压缩主要通过减小量化误差、减小梯度误差、改进损失函数以及改变网络结构等来提升性能^[20]。目前对改进损失函数的研究主要包括给原损失函数添加蒸馏损失项^[21-24]或非蒸馏损失项^[25-28],其中,蒸馏损失是利用从全精度模型提取的信息来指导训练量化网络,可对中间层或 softmax 输出层进行蒸馏,而非蒸馏损失如激活分布损失^[25]、通道相互作用损失^[26]、增量量化损失^[27]、损失感知量化^[28]等,其中损失感知量化采用拟牛顿算法将与权重量化相关的总损失最小化。本文针对点云网络模型二值量化的问题,聚焦改善损失函数,首

次将知识蒸馏方法^[29-34]引入点云二值量化模型,设计点云深度模型压缩框架,考虑特征聚合问题引入了辅助损失项,改进的损失函数包括预测损失、蒸馏损失和辅助损失项。针对设计的损失项开展消融实验,对点云分类、部件分割以及语义分割不同任务进行对比实验。当前,用于点云全局特征聚合的最大池化也是大多数点云网络架构采取的流行设计方法^[35],所以考虑算法的扩展性,在 PointNet ++^[3]、PointCNN^[4]和 DGCNN^[5]其他主流点云网络模型上进行拓展实验,以验证算法的有效性。

1 压缩模型

首先提出新的点云二值量化模型,进而分两个模块具体介绍蒸馏损失和辅助损失。

1.1 模型整体架构

如图 1 所示,模型主要包括教师网络、学生网络和知识蒸馏部分,其中教师网络即点云全精度网络,学生网络即点云二值量化网络,图中聚合(Aggregation)表示最大池化或最大熵聚合方法。通过引入全精度网络相关的软目标(Soft Targets)来诱导量化网络的训练,实现知识迁移,使得量化网络的性能更加接近全精度网络的性能。除蒸馏损失外,量化网络预测值与真实标签存在预测损失;设计量化网络与全精度网络对应池化层输出误差为辅助损失项,起正则化作用。

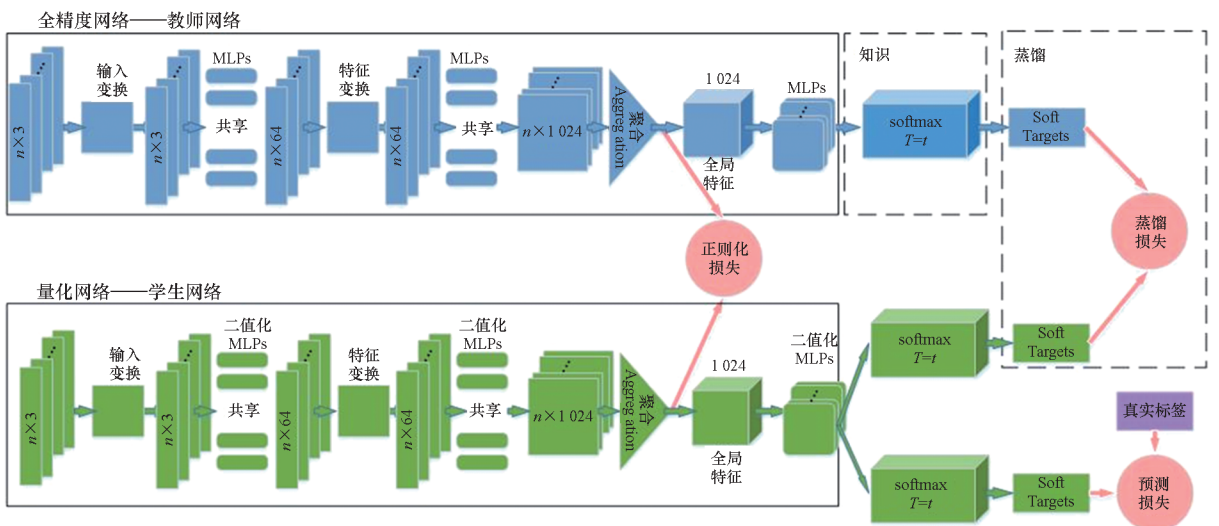


图 1 PointNet 模型压缩总体流程

Fig. 1 Overall flowchart of proposed PointNet model compression

问题定义:全精度网络模型用 $f(x; \alpha)$ 表示,其中 x 是网络输入, α 是全精度模型参数;二值量化网络模型用 $q(x; \beta)$ 表示,其中 β 是量化模型参

数。通过知识蒸馏,最小化目标函数,为二值量化网络学习优化的参数 β 使得 $q(x; \beta)$ 接近 $f(x; \alpha)$ 的性能。学习过程中蒸馏损失函数定义为 L_{KD} ,

其中蒸馏软目标损失项为 L_{DS} , 预测损失项为 L_{CE} (表示量化预测损失项 L_{CES} 或全精度预测损失项 L_{CET}), 辅助损失项为 L_{reg} 。通过最小化损失函数, 使量化模型达到收敛, 减少量化模型预测误差。

1.2 蒸馏损失

假设 $\{x_i, l_i\}_{i=1}^N$ 为训练样本集, 其中 x_i 为输入样本, l_i 为对应真实标签。模型隐藏层输出 $\mathbf{H}_i = \Phi(x_i)$ 经过 softmax 层的输出为 $l'_i = p(l_i | x_i) = \text{softmax}(\mathbf{W}\mathbf{H}_i)$, 其中 \mathbf{W} 为学习的权值矩阵。

训练全精度网络学习得到的优化参数为:

$$\alpha^* = \arg \min_{\theta} \sum_{i \in \{1, 2, \dots, N\}} L_{CET}(x_i, y_i; [\alpha, \mathbf{W}]) \quad (1)$$

全精度网络输出预测概率为:

$$l'_i = p_i(l_i | x_i) = \text{softmax}\left(\frac{\mathbf{W}\Phi(x_i; \alpha^*)}{T}\right) \quad (2)$$

其中: 下标 f 表示全精度网络; T 为温度超参数, 该参数控制对软目标的依赖程度, 随着参数值的增大, 软目标的分布更趋均匀。

蒸馏软目标损失为:

$$L_{DS} = - \sum_{i \in \Delta} \sum_{z \in Z} [p_f(l_i = z | x_i; \alpha^*) \cdot \log_2 p_q(l_i = z | x_i; \beta^*)] \quad (3)$$

其中, Δ 表示训练样本序号标识集, Z 表示标签集, z 为类别标签, 下标 q 表示量化网络, β^* 为量化网络学习的参数。

一方面, 考虑量化预测与全精度预测的差异; 另一方面, 同时考虑量化预测与真实标签的差异:

$$L_{DS} = - \sum_{i \in \Delta} \sum_{z \in Z} [\mathbf{1}(l_i = z) \cdot \log_2 p_q(l_i = z | x_i; \beta^*)] \quad (4)$$

则蒸馏损失为:

$$L_{KD} = \lambda L_{DS} + (1 - \lambda) L_{CE} \quad (5)$$

其中, λ 是平衡二者重要性的超参数。

1.3 辅助损失

在分析二值量化网络中最大池化对其性能影响的基础上, 提出正则化辅助损失项。

1.3.1 最大池化对性能影响分析

从信息熵^[19, 36-38]角度考虑, 信息熵反映了所含信息量的多少或者复杂程度, 经过池化操作后得到的分布的熵越小, 特征的可区分性越低, 相反熵越大, 所含信息越“多样化”, 特征可鉴别能力越强。

对全精度进行二值量化, 考虑条件熵, 即在已知全精度变量 F 下的二值量化变量 Q 的不确

定性:

$$H(Q|F) = \sum_f p_F(f) \left(\sum_q p_{Q|F=f}(q) \log_2 p_{Q|F=f}(q) \right) \quad (6)$$

使用量化函数 sign 进行二值量化, 有 $p_{Q|F=f}(q) = \{0, 1\}$, 则下式成立:

$$H(Q|F) = \sum_f p_F(f) \cdot 0 = 0 \quad (7)$$

可知二值量化后的熵必然小于全精度的熵, 量化必然导致特征表达能力下降, 远不如全精度特征表达性能。信息熵的取值范围为 $[0, \log_2 m]$, 其中 m 为类别数。对于二值量化而言, m 取值为 2, 熵最大值远小于全精度的最大熵值, 决定了其特征表征的能力范围十分有限。

记最大池化操作为 $\phi(\cdot)$, 池化层输入为 I_ϕ , 输出为 R , 经最大池化输出与输入概率质量函数关系为:

$$\sum_r p_R(r) = \left(\sum_{i_\phi} p_{I_\phi}(i_\phi) \right)^n \quad (8)$$

其中, n 为最大池化元素个数, $\sum_{i_\phi} p_{I_\phi}(i_\phi) \leq 1$ 。

对于二值量化, 最大池化后特征熵为:

$$H_Q(I_\phi) = - \left(\sum_{i_\phi \geq 0} p_{I_\phi}(i_\phi) \right)^n \log_2 \left(\sum_{i_\phi \geq 0} p_{I_\phi}(i_\phi) \right)^n - \left[1 - \left(\sum_{i_\phi \geq 0} p_{I_\phi}(i_\phi) \right)^n \right] \log_2 \left[1 - \left(\sum_{i_\phi \geq 0} p_{I_\phi}(i_\phi) \right)^n \right] \quad (9)$$

则当 n 增大且 $n \rightarrow \infty$ 时, 经计算可得 $\lim_{n \rightarrow \infty} H_Q(I_\phi) = 0$ 。令 $w_n = \left(\sum_{i_\phi \geq 0} p_{I_\phi}(i_\phi) \right)^n$, 对 $H_Q(w_n)$ 求导得:

$$\frac{\partial H_Q(w_n)}{\partial w_n} = \log_2 \left(\frac{1 - w_n}{w_n} \right) \quad (10)$$

故当 $w_n = 0.5$ 时取最大值, 设此时 n 取值为 n_{opt} 。当 $w_n < 0.5$ 时递增, 正如之前分析的, 即使增到最大值, 二值量化熵值依然较小, 特征表征能力很有限; 当 $w_n \geq 0.5$ 时递减, 随着 n 的不断增大, 熵值递减直至趋于 0, 最大池化后的特征表征能力也大幅下降。从二维图像到三维点云, 网络池化聚合核的尺寸也急剧增大, 导致特征可鉴别能力严重下降。

1.3.2 正则损失项

通过以上分析, 为了进一步减小最大池化对二值量化性能的影响, 通过让量化网络的池化层输出去学习全精度网络的池化层输出, 进而引入误差正则化项来优化目标损失函数, 提高量化模型泛化能力。对于隐藏层, 仅学习有较大影响的池化层输出知识对计算代价要求较低, 是可取的。

辅助损失项定义为:

$$L_{reg} = g(h_q^p, h_f^p) \quad (11)$$

其中, $g(\cdot)$ 表示正则化函数, h 表示池化层输出, 下标 q 和 f 分别表示量化网络 and 全精度网络, 上标 p 标识对应的池化层。

这里, $g(\cdot)$ 选取常见正则化项^[39-40], 作为辅助损失项便于对比分析, 如表 1 所示。各范数表达式依据上标区分, 如上标 F 标识取 F -范数 (通常也称 L_2 范数), 其余符号标识类同。

表 1 正则化项
Tab.1 Regularization terms

正则化项	具体表达式
F-范数	$L_{reg}^F = \ h_q^p - h_f^p\ _F = \sqrt{\sum_{i=1}^M \sum_{j=1}^N h'_{ij} ^2}$
1-范数	$L_{reg}^1 = \ h_q^p - h_f^p\ _1 = \max_j \sum_{i=1}^M h'_{ij} $
2-范数	$L_{reg}^2 = \ h_q^p - h_f^p\ _2 = \sqrt{\lambda_{\max}(H^T H)}$
L1 范数	$L_{reg}^{L1} = \ h_q^p - h_f^p\ _1 = \sum_{i=1}^M \sum_{j=1}^N h'_{ij} $
核范数	$L_{reg}^* = \ h_q^p - h_f^p\ _* = \text{tr}(\sqrt{H^T H})$
无穷范数	$L_{reg}^\infty = \ h_q^p - h_f^p\ _\infty = \max_i \sum_{j=1}^N h'_{ij} $

1.4 训练策略

1.4.1 训练损失函数

当采用预先训练好的全精度模型作为教师模型时, 将第 1.2 节和 1.3 节设计的损失函数相加作为二值量化网络蒸馏训练总的损失函数:

$$L_{total} = L_{KD} + \eta \cdot L_{reg} = \lambda \cdot L_{DS} + (1 - \lambda) \cdot L_{CE} + \eta \cdot L_{reg} \quad (12)$$

其中, η 为衡量蒸馏池化层输出特征重要性的超参数。

1.4.2 训练方案

全精度网络模型已预先训练好, 网络训练采用离线训练的方式, 知识从预训练的全精度模型转移到二值量化模型, 具体见算法 1。

2 实验

实验配置: Intel® Core™ i5 - 9400, 内存 16 GB; 显卡 NVIDIA GeForce RTX 2080Ti。本文实验采用点云公开数据集 ModelNet40, 包含 40 个类别的 12 311 个三维形状。实验选取 4 种典型的二值量化算法——BNN、XNOR、IRNET 和 BiPointnet, 算法对比如表 2 所示。实验中各算法标识含义: BNN + MAX 为点云网络量化采用 BNN 算法、最大池化; BNN + MAX + KDR 为在 BNN + MAX 的基础上进行知识蒸馏和辅助正则化(用 KDR 标

算法 1 离线蒸馏训练

Alg.1 Offline training with knowledge distillation

输入: 预训练好的全精度网络模型 W_f , 输入 $a \in \mathbb{R}^d$; 随机初始化二值量化模型 W_q , 温度参数 T ; 超参数 λ, η , 学习率 lr
输出: 二值量化网络模型 W_q

1. 前向传播
2. 全精度网络 W_f :
3. 计算池化层输出矩阵 H_f 和预测输出 l_f
4. 量化网络:
5. 计算二值权重 $B_w = Quant(W)$
6. 计算二值输入 $B_a = Quant(a)$
7. 计算输出 $B_a \odot B_w$
8. 保存池化层输出特征矩阵 H_q
9. 计算 L_{DS} 和 L_{CE} , 蒸馏损失 L_{KD}
10. 根据 H_f 和 H_q 计算正则化损失项 L_{reg}
11. 求总损失 $L = L_{total}(W_q)$
12. 反向传播
13. 量化网络:
14. 计算关于 a 的梯度 $\frac{\partial L}{\partial a}$
15. 计算关于 W 的梯度 $\frac{\partial L}{\partial W_q}$
16. 参数更新
17. 量化模型参数更新 W :
18. $W_q = W_q - lr \cdot \frac{\partial L}{\partial W_q}$
19. 返回量化模型 W_q

表 2 所选量化算法对比

Tab.2 Selected quantization algorithms comparison

方法	权重/激活 (位数)	尺度或 位移因子	浮点 计算量	按位 计算量
BNN	1/1	无	0	$O_1 \times O_2$
IRNET	1/1	位移	0	$O_1 \times O_2 + O_1$
XNOR	1/1	尺度	O_1	$O_1 \times O_2$
BiPointnet	1/1	尺度 池化位移	O_1 S	$O_1 \times O_2$ 0

注: 按位计算包括异或运算、位计数和位移量; $O_1 = C_{out} \times W_{out} \times H_{out}$ (C_{out} 为输出通道数, $W_{out} \times H_{out}$ 为输出特征图尺寸); $O_2 = C_{in} \times W_k \times H_k$ (C_{in} 为输入通道数, $W_k \times H_k$ 为核尺寸); $S = C_p \times W_{p,in} \times H_{p,in}$ (C_p 为池化输入通道数, $W_{p,in} \times H_{p,in}$ 为池化输入特征图尺寸)。

识); BNN + EMA 表示点云网络量化采用 BNN 算法和最大熵聚合 (EMA) 方法, 其中, EMA 利用信

息熵减少池化特征信息损失;BNN + EMA + KDR 表示在 BNN + EMA 的基础上进行知识蒸馏和辅助正则化;对于其他二值量化算法 XNOR、IRNET 和 BiPointnet,其标识含义类似,其中 LSR + EMA 即为 BiPointnet。本实验所有原算法运行结果与对应改进算法基于相同的实验配置得出。

2.1 不同参数 T 的影响

实验主要验证不同 T 取值对蒸馏结果的影响,其取值为 1 到 10。任选一种正则化项进行验证,本实验选取 F - 范数,针对 8 种算法的具体实验结果如图 2 所示。

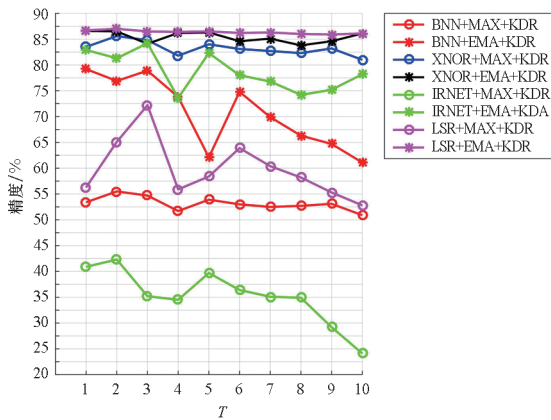


图 2 不同 T 值的结果变化曲线

Fig. 2 Variable curves for various T

综合分析可以看出,随着 T 值的增大,精度整体呈现下降趋势,较高的精度以大概率集中在 $[1,3]$ 的 T 值区间内。当 T 值取 1 到 10 时,8 种方法各有 10 个精度值,有 2 种算法在 $T=1$ 时取到各自最大值,4 种算法在 $T=2$ 时取到各自最大值,2 种算法在 $T=3$ 时取到各自最大值,故本实验统一取 T 值为 2。此实验结果符合 Hinton 在文献[29]中指出的结论,当学生网络比教师网络小得多时取值较小的 T 比较大的 T 更有效。而二值量化网络与全精度网络相比,位数极限压缩至 1 位,与全浮点数的网络相比小很多,因此选取小的 T 值效果更好,本实验选取 T 的值为 2 是合理的。

2.2 不同正则化项的比较

本实验主要对比不同正则化项在不同量化方法中的性能,进而验证后续实验选取何种正则化项。实验选取 $T=2$,针对 4 种方法 BNN、XNOR、IRNET 和 BiPointnet 进行对比实验,其中前三种池化聚合为 MAX,最后一种聚合方法采取 EMA。为了验证正则化项的有效性,在原损失函数的基础上,设计总损失函数为:

$$L_{total} = L_{origin} + \eta \cdot L_{reg} \quad (13)$$

实验结果如表 3 所示,加粗数字分别为同种量化方法不同正则化项中的最高精度。可以看出,1 - 范数在三种方法中取得了最高精度,且具有明显比较优势,在对 BiPointnet 正则化后精度甚至高达 86.9%,对 XNOR 正则化后精度也突破 86%,在 BNN 方法中精度 55.8% 仅次于无穷范数的 56.4%。而 L1 范数性能最差,正则化提升幅度小,甚至存在低于原精度情况,如对 BiPointnet 正则化精度仅为 82.6%,低于原精度 3.5%,没有起到正则化作用。从精度来看,1 - 范数总体上性能优于其他类型。图 3 为 XNOR 量化方法添加不同正则化项得到的误差随训练次数变化曲线。可以看出,1 - 范数正则化方法具有较好的收敛性能,随着训练次数的增加,尤其在训练 160 次后,相比较于其他范数其误差收敛到最小值,获得最高精度,而 L1 范数的正则化性能明显低于其他范数。综上分析,后续实验统一选取 1 - 范数作为损失正则化项。

表 3 采取不同正则化项的实验结果

Tab. 3 Experiment results of various regularization terms

正则化项	BNN	IRNET	XNOR	BiPointnet
L_{origin}	26.8	18.5	71.8	86.1
F - 范数	32.6	28.2	85.0	86.4
1 - 范数	55.8	48.1	86.1	86.9
2 - 范数	51.5	29.5	85.6	86.8
L1 范数	27.6	24.4	74.6	82.6
核范数	33.6	47.9	82.2	86.7
无穷范数	56.4	41.1	85.0	86.7

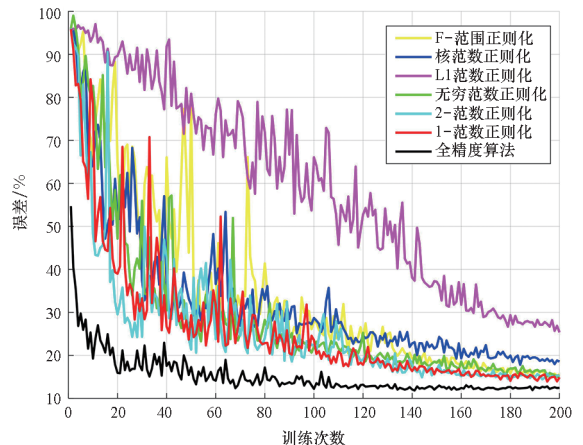


图 3 正则化误差变化曲线

Fig. 3 Regularization error variation curves

2.3 对比实验

本实验依据选取的二值量化算法 BNN、XNOR、IRNET 和 BiPointnet, 选取最大池化或改进的最大熵聚合以及是否进行知识蒸馏和正则化, 对 8 种算法的精度进行对比验证, 结果如表 4 所示。可以看出, 在选取最大池化而未进行蒸馏正则化的四种算法 BNN + MAX、XNOR + MAX、IRNET + MAX 和 LSR + MAX 中, 除 XNOR 外, 其他算法精度均大幅小于全精度算法; 在进行知识蒸馏和池化正则化后, XNOR 算法精度提高了 14.6%, 其余三种算法精度提升幅度均在双倍以上, 由此可以看出加入本文蒸馏正则化后实现了算法性能的大幅提升。采用最大熵聚合的四种方法 BNN + EMA、XNOR + EMA、IRNET + EMA 和 LSR + EMA, 加入蒸馏正则化后精度分别提高了 35.7%、3.6%、10.4% 和 1.0%, 更接近全精度模型精度, 尤其是 LSR + EMA + KDR 将精度提高到了 87.1%。

表 4 对比实验结果

Tab. 4 Comparable experiment results

方法	位宽	精度/%
全精度算法	32	88.2
BNN	BNN + MAX	26.8
	BNN + MAX + KDR	64.2
	BNN + EMA	41.2
	BNN + EMA + KDR	76.9
XNOR	XNOR + MAX	71.8
	XNOR + MAX + KDR	86.4
	XNOR + EMA	82.9
	XNOR + EMA + KDR	86.5
IRNET	IRNET + MAX	18.5
	IRNET + MAX + KDR	53.0
	IRNET + EMA	72.2
	IRNET + EMA + KDR	82.6
BiPointnet	LSR + MAX	4.1
	LSR + MAX + KDR	73.3
	LSR + EMA	86.1
	LSR + EMA + KDR	87.1

2.4 消融实验

本实验主要验证两个模块包括知识蒸馏和辅助正则化的作用效果, 分别用 L_{KD} 和 L_{reg} 标识。实

验结果如表 5 所示, 与原算法相比, 两个模块单独作用时精度均有提升, 两个模块共同作用时的算法精度均高于单个模块作用时的精度。使用最大池化, BNN、IRNET、XNOR 和 LSR 四种算法单模块作用时精度均大幅提升, 知识蒸馏单独作用时精度分别提升了 16.9%、16.8%、12.5% 和 45.8%, 而在正则化单独作用时精度分别提升了 29.0%、29.6%、14.3% 和 50.3%。使用最大熵聚合, 四种算法单模块作用时精度也有不同程度的提升; BNN 算法提升幅度最大, 其中知识蒸馏单独作用时精度提升了 34.3%, 在正则化单独作用时精度提升了 33.9%。综上可以看出, 知识蒸馏和正则化对量化模型精度的提升作用非常明显。

表 5 消融实验结果

Tab. 5 Ablation experiment results

方法	原算法	L_{KD}	L_{reg}	$L_{KD} + L_{reg}$
BNN + MAX	26.8	43.7	55.8	64.2
IRNET + MAX	18.5	35.3	48.1	53.0
XNOR + MAX	71.8	84.3	86.1	86.4
LSR + MAX	4.1	49.9	54.4	73.3
BNN + EMA	41.2	75.5	75.1	76.9
IRNET + EMA	72.2	76.7	73.9	82.6
XNOR + EMA	82.9	83.6	86.3	86.5
LSR + EMA	86.1	86.5	86.9	87.1

2.5 拓展实验

本实验旨在验证本文所提算法的泛化性能。在 PointNet 模型实验基础上, 选取另外典型的 3 种点云主流骨干网络模型 PointNet++、PointCNN 和 DGCNN, 选用 XNOR 和 BiPointnet 两种二值量化方法, 所得实验结果如表 6 所示。经分析得出, 相比于 XNOR + MAX 和 LSR + EMA (由于 PointCNN 采用 X-Conv 取代最大池化, 因此本实验中未添加池化正则化损失项, 采取 XNOR + KD 和 LSR + KD), 各网络的精度均有提升, 验证了算法的泛化性能。图 4 给出 DGCNN 在全精度、LSR + EMA 和 LSR + EMA + KDR 三种算法下测试误差随训练次数变化的曲线, 本文算法误差曲线整体上介于全精度和 LSR + EMA 误差曲线之间, 说明基于知识蒸馏和辅助损失的方法改善了 DGCNN 量化精度性能, 更接近于全精度网络精度。

表6 拓展实验结果

Tab.6 Extended experiment results

方法	位宽	精度/%
PointNet	全精度算法	32 88.2
	XNOR + MAX	71.8
	XNOR + MAX + KDR	86.4
	LSR + EMA	1 86.1
	LSR + EMA + KDR	87.1
PointNet ++	全精度算法	32 90.7
	XNOR + MAX	59.6
	XNOR + MAX + KDR	64.1
	LSR + EMA	1 88.5
	LSR + EMA + KDR	89.0
PointCNN	全精度算法	32 89.7
	XNOR	82.1
	XNOR + KD	83.9
	LSR	1 81.5
	LSR + KD	84.4
DGCNN	全精度算法	32 90.9
	XNOR + MAX	64.0
	XNOR + MAX + KDR	70.4
	LSR + EMA	1 75.0
	LSR + EMA + KDR	86.2

PointNet 模型量化后的性能,在其余主流点云深度模型中的量化性能分析与此类似。表7给出了PointNet 及其各种量化方法的表现性能,包括单样本浮点运算数(FLOPs/sample)、加速比、参数量和压缩比四项。

表7 复杂度分析结果

Tab.7 Complexity analysis results

方法	FLOPs/sample	加速比	参数量	压缩比
全精度算法	443.38×10^6	1	3.48×10^6	1
BNN + MAX + KDR	8.35×10^6	53	0.15×10^6	23
IRNET + MAX + KDR	8.94×10^6	50	0.16×10^6	22
XNOR + MAX + KDR	9.89×10^6	45	0.62×10^6	6
LSR + MAX + KDR	8.46×10^6	52	0.15×10^6	23
BNN + EMA + KDR	10.45×10^6	42	0.15×10^6	23
IRNET + EMA + KDR	11.04×10^6	40	0.16×10^6	22
XNOR + EMA + KDR	11.99×10^6	37	0.62×10^6	6
LSR + EMA + KDR	10.56×10^6	42	0.15×10^6	23

由表7可以分析得出,相比于全精度模型,8种方法平均加速45倍,其中BNN + MAX + KDR、IRNET + MAX + KDR和LSR + MAX + KDR这3种方法加速50倍以上,而加速比最小的XNOR + EMA + KDR也加速了37倍。相比于全精度模型参数量,除XNOR + MAX + KDR和XNOR + EMA + KDR外,其余6种方法参数量压缩均在22倍以上。由于XNOR量化是对各通道进行量化,因而相比于其他方法,参数量要大,精度也相对高于BNN和IRNET。相比较而言,LSR + MAX + KDR和LSR + EMA + KDR分别在最大池化(MAX)和最大熵聚合(EMA)的同类方法中性能最优,保持高精度的同时具有高加速比和高压缩比。综上分析,本文所提点云深度模型加入知识蒸馏和正则化后,大幅度提高精度的同时,具备了较高的加速和压缩性能。

3 结论

针对三维点云深度学习网络模型存在的应用实际问题,结合当前二值量化模型研究现状,将知识蒸馏方法引入点云二值量化模型中,解决了量化精度大幅下降问题。同时考虑相比于二维图像,三维点云深度学习模型有其特殊性,本文在分析池化聚合影响的基础上,提出了正则化辅助损失函数项,与蒸馏损失项共同构成总的损失函数,有效地解决池化对点云二值量化模型预测精度的

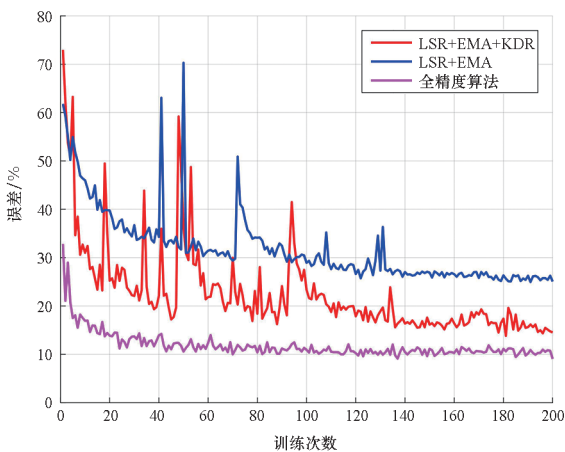


图4 误差随训练次数变化曲线

Fig.4 Variable error curves with training numbers

2.6 复杂度分析

本实验通过时间复杂度浮点运算次数(floating point operations, FLOPs)和空间复杂度参数量两个指标来衡量点云深度模型量化压缩后的性能。本节主要分析采用所提8种量化方法对

不利影响。通过在公共数据集上实验,结果显示,该算法可以使点云二值量化模型取得更接近全精度模型精度的性能,同时取得较高的加速比和压缩比,而且可以较方便地移植到其他主流点云深度学习网络模型上。

参考文献 (References)

- [1] 龙霄潇,程新景,朱昊,等. 三维视觉前沿进展[J]. 中国图象图形学报, 2021, 26(6): 1389-1428.
LONG X X, CHENG X J, ZHU H, et al. Recent progress in 3D vision [J]. Journal of Image and Graphics, 2021, 26(6): 1389-1428. (in Chinese)
- [2] CHARLES R Q, HAO S, MO K C, et al. PointNet: deep learning on point sets for 3D classification and segmentation[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
- [3] QI C R, YI L, SU H, et al. PointNet++: deep hierarchical feature learning on point sets in a metric space [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017.
- [4] LIY Y, BU R, SUN M C, et al. PointCNN: convolution on χ -transformed points[C]//Proceedings of 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), 2018.
- [5] WANG Y, SUN Y B, LIU Z W, et al. Dynamic graph CNN for learning on point clouds [J]. ACM Transactions on Graphics, 2019, 38(5): 1-12.
- [6] WU W X, QI Z A, LI F X. PointConv: deep convolutional networks on 3D point clouds [C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [7] XU Q G, SUN X D, WU C Y, et al. Grid-GCN for fast and scalable point cloud learning [C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [8] HU Q Y, YANG B, XIE L H, et al. RandLA-net: efficient semantic segmentation of large-scale point clouds [C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [9] LIU Z J, TANG H T, LIN Y J, et al. Point-voxel CNN for efficient 3D deep learning [C]//Proceedings of 33rd Conference on Neural Information Processing System(NeurIPS 2019), 2019.
- [10] COURBARIAUX M, BENGIO Y, DAVID J P. BinaryConnect: training deep neural networks with binary weights during propagations [C]//Proceedings of the 28th International Conference on Neural Information Processing Systems, 2015.
- [11] KIM M, SMARAGDIS P. Bitwise neural networks [C]//Proceedings of the 31st International Conference on Machine Learning(ICML), 2015.
- [12] LI Z F, NI B B, ZHANG W J, et al. Performance guaranteed network acceleration via high-order residual quantization[C]//Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV), 2017.
- [13] LIN X F, ZHAO C, PAN W. Towards accurate binary convolutional neural network [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017.
- [14] WANG P S, HU Q H, ZHANG Y F, et al. Two-step quantization for low-bit neural networks [C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.
- [15] HU Q H, WANG P S, CHENG J. From hashing to CNNs: training binary weight networks via hashing [C]//Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence(AAAI-18), 2018.
- [16] HUBARA I, COURMATTHIEU C, SOUDRY D, et al. Binarized neural networks [C]//Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016.
- [17] RASTEGARI M, ORDONEZ V, REDMON J, et al. XNOR-Net: ImageNet classification using binary convolutional neural networks [C]//Proceedings of European Conference on Computer Vision(ECCV), 2016.
- [18] QIN H T, GONG R H, LIU X L, et al. Forward and backward information retention for accurate binary neural networks [C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [19] QIN H T, CAI Z A, ZHANG M Y, et al. BiPointnet: binary neural network for point clouds [C]//Proceedings of International Conference on Learning Representations (ICLR), 2021.
- [20] SIMONS T, LEE D J. A review of binarized neural networks[J]. Electronics, 2019, 8(6): 661.
- [21] CHEN X Y, LIU G C, SHI J, et al. Distilled binary neural network for monaural speech separation [C]//Proceedings of 2018 International Joint Conference on Neural Networks (IJCNN), 2018.
- [22] POLINO A, PASCANU R, ALISTARH D. Model compression via distillation and quantization [C]//Proceedings of International Conference on Learning Representations(ICLR), 2018.
- [23] MISHRA A, MARR D. Apprentice: using knowledge distillation techniques to improve low-precision network accuracy [C]//Proceedings of International Conference on Learning Representations(ICLR), 2018.
- [24] XU Y H, DONG X, LI Y D, et al. A main/subsidiary network framework for simplifying binary neural networks [C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [25] DING R Z, CHIN T W, LIU Z Y, et al. Regularizing activation distribution for training binarized deep networks [C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [26] WANG Z W, LU J W, TAO C X, et al. Learning channel-wise interactions for binary convolutional neural networks [C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [27] ZHOU A J, YAO A B, GUO Y W, et al. Incremental network quantization: towards lossless CNNs with low-precision weights [C]//Proceedings of International Conference on Learning Representations (ICLR), 2017.
- [28] HOU L, YAO Q M, KWOK J T. Loss-aware binarization of deep networks [C]//Proceedings of International Conference on Learning Representations (ICLR), 2017.
- [29] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network [J]. Computer Science, 2015, 14(7):

- 38 – 39.
- [30] FURLANELLO T, LIPTON Z C, TSCHANNEN M, et al. Born-again neural networks. [C]//Proceedings of the 35th International Conference on Machine Learning (ICML), 2018.
- [31] AGUILAR G, LING Y, ZHANG Y, et al. Knowledge distillation from internal representations [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(5): 7350 – 7357.
- [32] AHN S, HU S X, DAMIANOU A, et al. Variational information distillation for knowledge transfer [C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [33] ARORA S, COHEN N, HAZAN E. On the optimization of deep networks: implicit acceleration by overparameterization [C]//Proceedings of the 35th International Conference on Machine Learning (ICML), 2018.
- [34] ANIL R, PEREYRA G, PASSOS A, et al. Large scale distributed neural network training through online distillation [C]//Proceedings of International Conference on Learning Representations (ICLR), 2018.
- [35] GUO Y L, WANG H Y, HU Q Y, et al. Deep learning for 3D point clouds: a survey [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43 (12): 4338 – 4364.
- [36] 李航. 统计学习方法 [M]. 2 版. 北京: 清华大学出版社, 2019.
- LI H. Statistical learning method [M]. 2nd ed. Beijing: Tsinghua University Press, 2019. (in Chinese)
- [37] GRAY R M. Entropy and information theory [M]. 2nd ed. Berlin: Springer-Verlag Press, 2011.
- [38] 姜丹. 信息论与编码 [M]. 4 版. 合肥: 中国科学技术大学出版社, 2019.
- JIANG D. Information theory and coding [M]. 4th ed. Hefei: University of Science and Technology of China Press, 2019. (in Chinese)
- [39] HORN R A, JOHNSON C R. Matrix analysis [M]. 2nd ed. Cambridge: Cambridge University Press, 2013.
- [40] 王晓华. 深度学习的数学原理与实现 [M]. 北京: 清华大学出版社, 2021.
- WANG X H. Mathematical theory and realization of deep learning [M]. Beijing: Tsinghua University Press, 2021. (in Chinese)