文章编号:1007－6735(2024)03－0320－11　DOI: 10.13255/j.cnki.jusst.20230115002

基于点云图卷积神经网络的 3D 目标检测

刘振威，　黄影平，　梁振明，　杨静怡
（上海理工大学 光电信息与计算机工程学院，上海 200093）

摘要：随着激光雷达传感器的快速发展，目标检测算法从传统的 2D 检测快速转向 3D 检测。然而，激光雷达产生的点云是不规则和非结构化的数据，传统的卷积神经网络无法对其进行处理。基于此提出了一种新颖的图卷积神经网络，能够更好地利用数据的几何关系和拓扑结构直接从点云中学习特征以进行 3D 目标检测。首先将原始激光雷达点云数据进行下采样，再进行固定半径邻域图的构建，随后设计了一个新型的图卷积神经网络对点云进行编码来预测图中每个顶点所属对象的类别和形状。为提升检测准确度，网络中加入了一种校准机制来减少特征在不同维度变化时引入的平移误差，此外还引入了注意力机制，以使用权重来进一步强化输出的顶点特征。在 KITTI 数据集上进行实验，实验结果表明，此方法能够有效对 3D 目标进行检测。对比其他多种检测算法，此方法在检测准确度上具有一定的优势。

关键词：图；图卷积神经网络；激光雷达点云；3D 目标检测
中图分类号：TP 391.4　　**文献标志码：**A

3D object detection using graph convolutional neural network with point clouds

LIU Zhenwei,　HUANG Yingping,　LIANG Zhenming,　YANG Jingyi
(*School of Optical−Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China*)

Abstract: With the rapid development of light detection and ranging (LiDAR) sensors, object detection algorithm moved quickly from traditional 2D detection to 3D detection. However, the point clouds generated by LiDAR were irregular and unstructured data, which cannot be adopted by conventional Convolutional Neural Networks. Thus, a novel Graph Convolutional Neural Network was proposed,

收稿日期: 2023−01−15
基金项目: 国家自然科学基金资助项目(62276167)；上海市自然科学基金资助项目(20ZR1437900)
第一作者: 刘振威(1998−)，男，硕士研究生．研究方向：计算机视觉．E-mail：1065439410@qq.com
通信作者: 黄影平(1966−)，男，教授．研究方向：汽车电子、计算机视觉．E-mail：huangyingping@usst.edu.cn

引文格式: 刘振威，黄影平，梁振明，等．基于点云图卷积神经网络的 3D 目标检测[J]．上海理工大学学报，2024，46(3)：320-330．DOI：10.13255/j.cnki.jusst.20230115002
Citation: LIU Zhenwei, HUANG Yingping, LIANG Zhenming, et al．3D object detection using graph convolutional neural network with point clouds[J]．Journal of University of Shanghai for Science and Technology, 2024, 46(3)：320-330.

which could better utilize the geometric relationship and topology of the data to learn features directly from point clouds for 3D object detection. First, the original LiDAR point cloud was randomly down-sampled and voxel down-sampled, and then constructed a fixed-radius neighborhood graph. Then, a novel graph convolutional neural network was designed to encode the point cloud to predict the class and shape of the object to which each vertex in the graph belongs. In order to improve the detection accuracy, we added a calibration mechanism to the network to reduce the translation error introduced by features changing in different dimensions, and also introduced an attention module to use weights to further strengthen the output vertex features. Finally, we conducted experiments on the KITTI dataset, and the experimental results showed that the method in this work could effectively detect 3D objects. Compared with many other detection algorithms, the method in this work had certain advantages in detection accuracy.

Keywords: *graph*; *graph convolutional neural network*; *LiDAR point clouds*; *3D object detection*

目标检测是自动驾驶汽车识别其行驶环境的基本要求。随着能够提供场景 3D 几何信息的激光雷达 (LiDAR) 传感器的快速发展，目标检测方案从传统的 2D 检测迅速转变为 3D 检测。早期的 3D 目标检测主要依赖于双目立体匹配技术[1]。近年来，随着基于卷积神经网络 (CNN) 的深度学习技术的兴起，现有的 3D 目标检测主要通过激光雷达点云数据结合深度学习方法[2] 来完成。

激光雷达可以提供场景的全景 3D 几何信息。然而，激光雷达产生的点云通常是稀疏的、不规则的和非结构化的，被称为非欧几里得结构数据，传统卷积神经网络无法直接对其进行处理。图卷积网络（GCN）[3] 充分利用了数据的几何和拓扑结构，已经被证明是一种处理非欧几里德结构数据的理想方法。对于输入图G，图卷积运算F首先聚合k个最近邻域节点特征$N^k(v)$到中心节点v的特征，然后通过非线性激活函数更新这些聚合特征。对于不规则的点云数据，图卷积网络有一个天然的优势，就是可以将点云中的每个点都看成图的一个节点，将中心节点和它的邻居的关系看成边。因此，图卷积网络可以直接作用于不规则的点云，无需任何预处理，并保留所有原始特征。借助图卷积网络的这些特性，本文使用图卷积网络从激光雷达点云数据中学习特征，将图卷积神经网络应用于 3D 目标检测。

本文中，主要贡献包括：1）实现了一种基于图卷积神经网络的 3D 目标检测方法，用于检测道路交通场景中的汽车、行人以及骑行者。2）引入注意力机制以及坐标校准机制，在特征传输过程中对顶点特征进行细化以及减少邻域相对坐标的位移误差。3）使用 KITTI 数据集进行实验并分析结果，与现有其他方法进行对比分析，结果表明本文所提出方法能够有效地以 3 维的方式检测出交通场景中的各类目标。

1　相关方法

根据对激光雷达点云处理的方式，现有的基于深度学习的 3D 目标检测方法可分为 3 类：基于欧几里得数据变换的方法、基于非欧几里得数据的方法和基于点云–图像融合的方法。

1.1　基于欧几里得数据变换的方法

由于激光雷达点云的不规则性，传统的卷积神经网络无法直接应用。这类方法首先将不规则点云转换为规则的欧几里得数据，以便传统的神经网络对其进行处理。依据点云数据变换的方式，可分为基于投影的方法和基于网格的方法。

投影的方法是通过投影将不规则的 3D 点云转化为规则的 2D 图像数据，再使用 2D 检测器对其进行检测，主要包括鸟瞰图（BEV）投影、前视图（FV）投影以及距离视图（RV）投影。Meyer 等[4] 提出了基于 RV 投影的 LaserNet，它使用一个全卷积网络进行端到端的训练，以此来预测每个激光雷达点的类概率多模态分布，然后有效地融合这些多模态分布来生成网络对每个对象的预测结果。Barrera 等[5] 提出了基于 BEV 投影的 BirdNet+，它先将点云投影至 BEV 视图，然后进行端到端的目标检测，同时预测对象的置信度和边界框。

Yang等[6]提出了基于BEV投影的PIXOR，该方法将点云进行离散化处理，并以类似的方式对反射率进行编码，得到规则的表示，然后使用全卷积网络进行目标检测与定位。这种基于投影的的方法不可避免地将数据从高维转到低维，造成信息的丢失，使检测准确度无法进一步提升。

基于网格的方法是指将点云划分为一个个等间距的规则体素单元，再使用标准向量表示其中的点，这样就可以从每个体素内的点集中提取局部特征，其优势在于可以显著节省内存资源。Zhou等[7]提出了VoxelNet，首先将点云划分为3D小立方体，再采用VEF(voxel feature encoding)网络对体素中的特征进行编码，最后使用3DRPN网络进行检测。Yan等[8]为解决VoxelNet计算量过大的问题提出了SECOND，该方法使用稀疏3D卷积代替VoxelNet中的3D卷积，提高了检测速度，降低了内存使用。基于网格的检测方法虽然可以取得良好的效果，但是，由于体素网格的立方体性质，点云表面很多特征都没有办法被表述出来，因此，模型效果难以进一步提升，同时该方案的时间和空间复杂度都十分高，计算代价昂贵。

1.2 基于非欧几里得数据的方法

为了避免基于欧几里得数据变换的方法中信息丢失的情况，这类基于非欧几里得数据的方法设计特定的网络，通过对不规则点云进行空间和对称变换，将它们直接作为网络的输入。包括基于点和基于图的方法。

Qi等提出的PointNet[9]、PointNet++[10]是基于点方法的先驱模型，它直接使用原始的不规则点云数据而不将其转换为其他格式，因此近年来提出了一些基于PointNet的3D目标检测方法。PointRCNN[11]利用PointNet++作为骨干网络来提取点特征并生成少量高质量的3D提议，然后使用规范的3D边界框细化模块进行细粒度的3D边界框预测。

近年来，图网络的出现为3D目标检测提供了新方向[12]，图网络已被证明是处理3D点云的有效方法，因此，基于图方法的3D目标检测方法也越来越受到关注。Point-GNN[13]首先将原始激光雷达点云编码，通过迭代更新图顶点提取点云特征，最后使用图顶点分类方法预测物体的类别和形状。PointRGCN[14]也是一种基于图的3D对象检测方法，它提出了2个图卷积分支用于目标提议细化，一个分支用于从每个单独的提议中提取点特征，另一个用于从不同提议中共享上下文信息。

1.3 基于点云–图像融合的方法

相机图像可以提供丰富的二维场景语义信息，但深度信息捕获能力通常较差。激光雷达点云可以提供3D场景的全景表示并包含丰富的深度信息，但点云的分辨率总是比图像差。因此，可以将这两种数据组合在一起以进行更全面的3D目标检测。MV3D[15]将激光雷达点云和相机RGB图像作为输入，第一步生成不同类型的2D对象提议，包括BEV、FV和常规图像提议，然后将这些不同的提议视图组合为3D边界框提案。AVOD[16]与MV3D的思路十分相似，可以说是MV3D的升级版本，它在特征提取部分作了改进，使其显著提高了小物体的检测效果。F-PointNet[17]提出了一种基于锥体的点云–图像融合3D检测方法，它首先在图像上生成一系列物体的2D边界框，然后将这些框投影到3D点云空间中以获得3D视锥，之后将2个PointNet应用于3D视锥体分割和3D边界框回归。MLOD[18]同样将RGB图像和激光雷达点云作为输入，首先使用区域提议网络(RPN)在点云的鸟瞰视图投影中生成3D提议，然后将3D提议边框投影到图像和BEV特征图，并将相应的图截取输入到检测头，以进行分类和边界框回归。与其他方法不同，裁剪的图像特征会使用深度信息将背景部分过滤，再输入到检测头。

2 图网络原理

图是一种对一组对象(节点)及其关系(边)建模的非线性数据结构，如图1所示，相比于其他线性表甚至树结构，它更复杂，也更抽象。图结构可以用于储存具有“多对多”逻辑关系的数据。图结构是由顶点和边构成的，顶点即为图中的数据元素，边为连接这些顶点的线，表示顶点间的关联关系；所有的顶点构成一个集合，边也构成一个集合，可以表示为

$$G=(V,E) \tag{1}$$

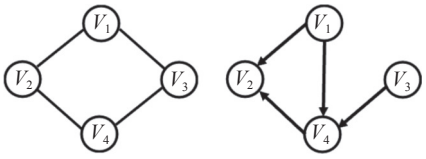


图1 无向图与有向图

Fig.1 Undirected graph and directed graph

式中：G表示图；V为图G中顶点的集合；E为图G中边的集合。

作为典型的非欧几里得数据结构，图十分契合无序的点云，可以直接用于表示点云，因此，可以将图数据直接作为网络的输入，进行目标检测。图神经网络已经被证明是一种有效的点云处理方法。

图神经网络（GNN）[19] 由 Scarselli 等在 2009 年提出。在此方法中，图中不同节点之间通过相互连接的边进行特征更新，直至每个节点达到一个相对稳定的状态，最后对节点特征进行综合作为网络的输出。由于每一个节点都存在与其相连的邻居节点，所以节点特征表示不仅包括自身节点特征，还包括其邻居节点特征，可表示为

$$z_i = f(l_i, l_{N[i]}, l_{E[i]}, z_{N[i]}) \tag{2}$$

式中：z_i为节点i的状态向量；l_i为节点i的标签；$l_{N[i]}$为与节点i相连的邻居节点的标签；$l_{E[i]}$为与节点i相连的边的标签；$z_{N[i]}$为节点i相连邻居节点在上一时间状态的特征。

图卷积神经网络 [20] 由 Thomas 等在 2017 年提出，它结合了卷积神经网络与图神经网络，是一种能对图数据进行深度学习的方法。图卷积神经网络目前主要分为两类：基于谱方法的图卷积神经网络模型和基于时空域的图卷积神经网络模型。

图卷积神经网络希望解决以图$G=(V,E)$作为输入时的特征提取问题。将每一个节点v_i的特征汇总可得到一个$N\times D$的特征矩阵 $\boldsymbol{X}$，其中，N为节点数，D为输入特征数，输出可表示为一个节点级输出$\boldsymbol{Z}$（一个 $N\times F$ 特征矩阵，其中，F为每个节点的输出特征数）。特征传播机制概括了基于时空域的图卷积神经网络模型[21] 的核心思想，如图 2 所示。

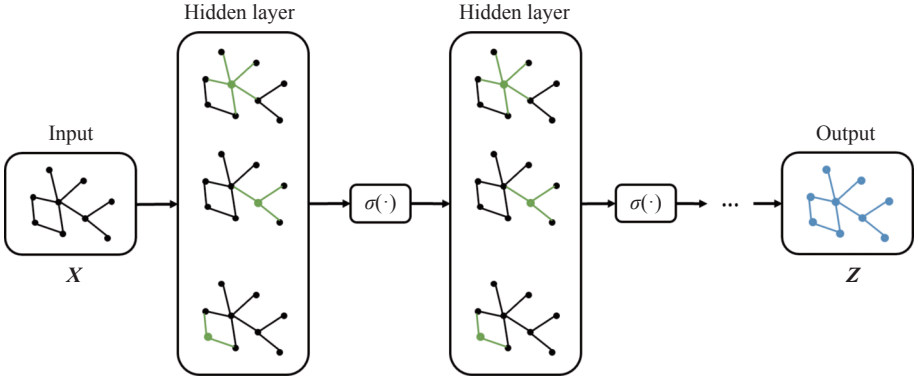


图 2　图卷积神经网络特征传播机制

Fig.2　Feature propagation mechanism of graph convolutional neural networks

3　本文方法

本文方法框架如图 3 所示。首先对原始点云数据进行随机下采样，整体降低点云的数量，对下采样后的点云再进行体素下采样，选取体素质心点进行图构建；再将构建好的点云作为网络的输入，以多层感知机（MLP）对点云进行特征提取，特征提取部分采取校准机制对顶点邻域范围内点的特征进行处理，同时使用注意力机制过滤噪声点特征；最后对提取到的特征进行分类以及回归预测，输出包含目标类别与置信度和 3D 边界框的检测结果。

3.1　点云数据预处理

对于激光雷达点云数据来说，一组数据中点的个数数量级约为 10^5，使用所有的点进行图的构建显然是不现实的。如图 3 中数据预处理部分所

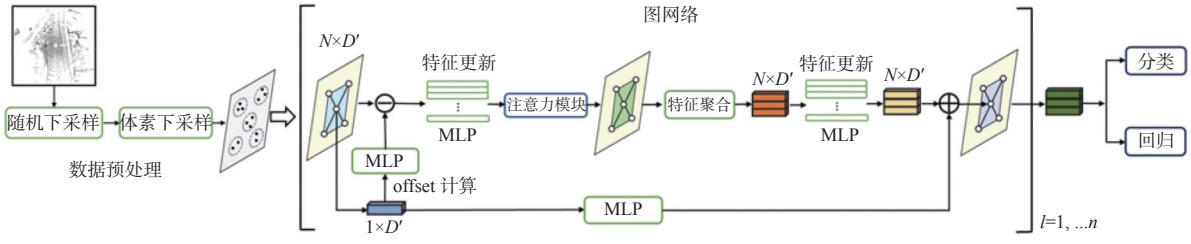


图 3　方法框架，包括数据预处理及图网络结构

Fig.3　Framework of the method, including data preprocessing and graph network structure

示，首先采用随机下采样的方式对点云数据进行下采样，在保证点云整体几何特征不变的前提下降低点云的密度；与 Point-GNN 类似，本文对随机下采样后的激光雷达点云进行体素下采样再进行图构建。将点云划分为体素网格，取每个体素中所有点计算坐标平均值，以体素质心点表示每个体素，将体素质心点作为顶点进行图构建，下采样中使用的体素仅用来降低点云的密度，而不是用来表示点云。

根据前面对图数据结构的介绍，本文采用固定半径近邻搜索的方式进行图的构建，定义点云 $\boldsymbol{S}=\{\boldsymbol{s}_1,\cdots,\boldsymbol{s}_N\}$，其中，$\boldsymbol{s}_i=(x_i,a_i)$是一个包含点 x_i 的三维坐标以及反射的激光强度 a_i 的向量。对点云进行建图，$G=(S,E)$，S 为顶点，E为点与半径 r邻域范围内其他点建立的边，可表示为

$$E=\{(\boldsymbol{s}_i,\boldsymbol{s}_j)\|x_i-x_j\|_2<r\} \tag{3}$$

K 最近邻（KNN）算法需要使用顶点与点云中其余每个点进行距离计算，它的计算复杂度与点云中点的个数成正比，因此，KNN 算法一般适用于样本总量较少的情况。相比于 KNN 算法，使用固定半径近邻搜索算法进行建图可使时间复杂度降低到$O(xN)$，x表示顶点在半径r范围内点的个数。通过这种预处理方式，在一定程度上可解决点云数据量过大以及模型预测时间较慢的问题。

3.2 图网络结构

由于图卷积神经网络在处理图数据时具有良好的性能和较高的可解释性，本文基于图卷积神经网络构建网络。网络结构如图 3 所示。

该网络通过聚合边特征对顶点特征进行更新，在第$t+1$次迭代过程中顶点和边特征可以表示为

$$\boldsymbol{s}_i^{t+1}=g\left(h\left(e_{ij}^t\right),\boldsymbol{s}_i^t\right),e_{ij}^t=f\left(\boldsymbol{s}_i^t,\boldsymbol{s}_j^t\right),\quad (i,j)\in E \tag{4}$$

式中：函数 f用来计算边的特征；函数 h 用来聚合每个顶点所对应边的特征；函数 g 使用边特征来更新顶点特征。

对顶点状态的更新可表示为

$$\boldsymbol{s}_i^{t+1}=g\left(h\left(f\left(x_j-x_i,\boldsymbol{s}_j^t\right)\right),\boldsymbol{s}_i^t\right),\quad (i,j)\in E \tag{5}$$

在此基础上，本文方法在聚合特征时引入注意力机制u_a。由于图构建算法中的半径r为固定值，它限制了图卷积核的感受野区域，若r值太小则不能很好地聚合邻域点的特征，若r值过大，会引入无关噪声点信息，本文方法通过使用注意力机制处理图卷积层中难以选择r值的缺点，在特征聚合时调整顶点特征权重占比，对顶点特征进行细化。引入注意力机制可以过滤邻域中噪声点的特征，使其不对图卷积结果产生影响。假设α_{ij}为节点s在下一步迭代中的加权因子（重要性），它在迭代计算更新后的节点特征时对节点本身进行约束，可表示为

$$\alpha_{ij}=u_a\left(x_j-x_i,w_j\boldsymbol{s}_j,w_i\boldsymbol{s}_i\right),\quad j\in R(i) \tag{6}$$

其中，w为可学习的权重系数。对α_{ij}使用 softmax 函数进行归一化：

$$\alpha_{ij}=\frac{\exp\left(\alpha_{ij}\right)}{\sum_{j\in R(i)}\exp\left(\alpha_{ij}\right)} \tag{7}$$

故图卷积层的输出可表示为

$$\boldsymbol{s}_i^{t+1}=g\left(h\left(f\left(x_j-x_i,\alpha_{ij}\boldsymbol{s}_j^t\right)\right),\boldsymbol{s}_i^t\right),\quad (i,j)\in E \tag{8}$$

值得注意的是，在节点特征迭代时，由于相对坐标对于点云整体具有平移不变性，因此，网络的输入是点与点之间的相对坐标，但是，顶点特征在不同维度变换时会产生一些位移误差，故需考虑如何减少这种位移误差。虽然相对坐标会发生变化，但是，点云的局部结构还是相似的，所以，本文引入坐标校准机制。由于顶点特征在$t+1$次迭代时已包含上一时刻的自身顶点特征信息，因此，更新顶点特征时考虑邻居节点在t次迭代时的坐标特征，通过这种方法可以在$t+1$次迭代时将坐标与t次迭代时对齐，以尽量减少维度变换带来的点云平移误差，坐标校准模块具体设计见图 4 中坐标偏移计算部分。本文中位移误差Δ_{o}的计算可表示为

$$\Delta_{\mathrm{o}}=M_{\mathrm{MLP}}\left(\boldsymbol{s}_i^t\right) \tag{9}$$

网络以多层感知机（MLP）进行顶点特征提取以及学习相邻节点之间相对坐标差的空间关系，所以，最终顶点特征更新可表示为

$$\boldsymbol{s}_i^{t+1}=M_{\mathrm{MLP},g}\left(\max_h\left(M_{\mathrm{MLP},f}\left(x_j-x_i+\Delta_{\mathrm{o}},\alpha_{ij}\boldsymbol{s}_j^t\right)\right),\boldsymbol{s}_i^t\right),\quad (i,j)\in E \tag{10}$$

具体的图网络结构参数如图 4 所示，包括特征提取、偏移误差计算以及预测输出，网络输入为$N\times4$的张量，通过多层感知机将特征逐层升维至$N\times300$，进行特征细化，对升维后的特征进行误差计算，将偏移特征与高维特征进行拼接得到$N\times303$的特征张量，再进一步提取特征并与高维

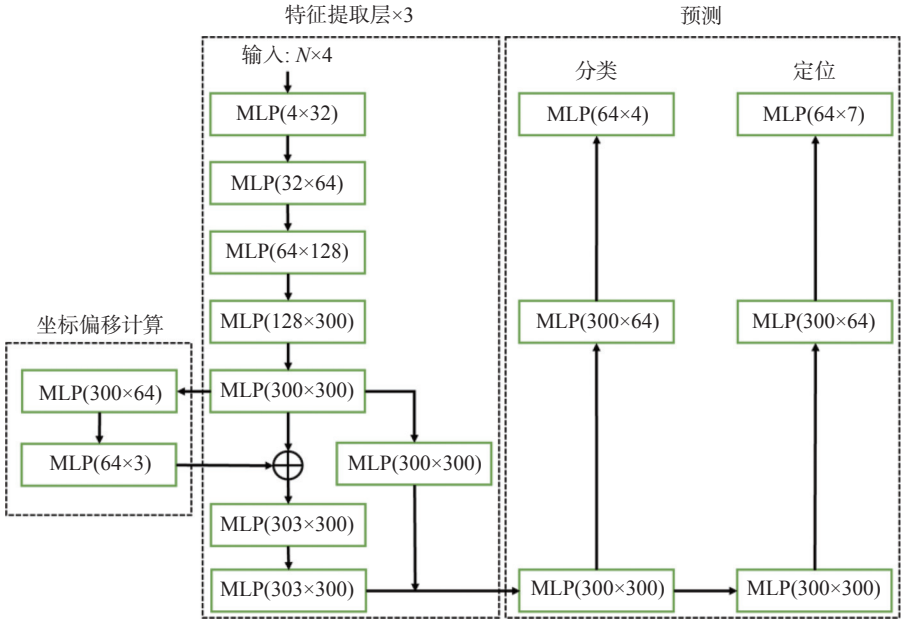


图 4　图网络结构详细参数

Fig.4　Detailed parameters of the graph network structure

顶点特征进行加和操作。最后同样使用多层感知机对特征降维并进行结果预测；分类模块输出类别信息，定位模块输出目标 3D 检测框的相关信息。

3.3　损失函数

本文方法定义损失函数主要由 3 部分组成：分类损失、回归损失、定位损失。其中，分类损失被定义为平均交叉熵损失，可表示为

$$l_{\mathrm{cls}}=-\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M}y_c^i\log p_c^i \tag{11}$$

式中，y_c^i为样本i的类别标签；p_c^i为样本i属于类别c的预测概率。

对于回归损失 l_{reg}，定义真实 3D 真值边界框为 $b_g=(x_g,y_g,z_g,\theta_g,l_g,w_g,h_g)$，$(x_g,y_g,z_g)$为边界框中心点的坐标，$(l_g,w_g,h_g)$为边界框的长宽高，$\theta_g$为框的偏转角；实际的预测框为$b_c=(x_c,y_c,z_c,\theta_c,l_c,w_c,h_c)$，故预测框和真值框的残差为$\delta_b=\left(\delta_x,\delta_y,\delta_z,\delta_\theta,\delta_l,\delta_w,\delta_h\right)$。

$$\begin{aligned}&\delta_x=\frac{x_g-x_c}{l_c},\delta_y=\frac{y_g-y_c}{w_c},\delta_z=\frac{z_g-z_c}{h_c},\delta_\theta=\log\left(\frac{\theta_g}{\theta_c}\right)\\&\delta_l=\log\left(\frac{l_g}{l_c}\right),\delta_w=\log\left(\frac{w_g}{w_c}\right),\delta_h=\log\left(\frac{h_g}{h_c}\right)\end{aligned} \tag{12}$$

基于式(12)，对于定位损失，本文使用 Huber 损失 l_{H} 来定位需预测的对象，定位损失可表示为

$$l_{\mathrm{loc}}=-\frac{1}{N}\sum_{i=1}^{N}(p_i\in b_c)\sum_{\delta\in\delta_b}l_{\mathrm{H}}\left(\delta-\delta_g\right) \tag{13}$$

综上所述，完整的损失函数为

$$l_t=\alpha l_{\mathrm{cls}}+\beta l_{\mathrm{loc}}+\gamma l_{\mathrm{reg}} \tag{14}$$

式中，权重参数α，β和γ用于调整每个损失项的相对权重占比。

4　实验及结果分析

实验在 KITTI 数据集[22]上进行，数据集包括 7481 个训练样本和 7518 个测试样本，数据集标注类别包括 car(汽车)、pedestrian(行人)、cyclist(骑行者)。在 KITTI 数据集中，根据标注框的高度、是否被遮挡以及遮挡程度定义了 3 种场景，分别为 Easy，Moderate，Hard，其中，Easy 为最小边框高度大于 40 像素，完全可见，截断小于 15%；Moderate 为最小边框高度大于 25 像素，部分遮挡，截断小于 30%；Hard 为最小边框高度大于 25 像素，严重遮挡，截断小于 50%。

本文方法采用 Pytorch 框架编写程序，实验在 256G 内存，Intel(R) Xeon(R) Silver 4216 CPU@2.10 GHz 处理器和 16G 内存，Tesla T4 GPU 以及安装有 Ubuntu 20.04 操作系统的计算机上开展。设置 batch size 为 2，以 3 层图网络结构进行端到端的训练，损失函数的权重占比分别设置为$\alpha=0.1$，$\beta=10$，$\gamma=5\mathrm{e}-7$。对于汽车类别，设置锚点框的长宽高分别为 3.9，1.6，1.5 m，行人和骑行者分别为 0.8，0.6，1.73 m 和 1.76，0.6，1.76 m。网络的优化器为 ADAM 优化器。在网络中使用多层感知机进行特征提取，使用 ReLU 激活函数以

及BetchNorm层，防止梯度消失以及过拟合问题出现。

实验结果采用通用的KITTI数据集的评价方法，选取准确率P、召回率R以及平均准确度m_{AP}作为性能评价指标。

准确率

$$P=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{P}}} \tag{15}$$

召回率

$$R=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{N}}} \tag{16}$$

平均准确度

$$m_{\mathrm{AP}}=\int_0^1 P(R)\,\mathrm{d}R \tag{17}$$

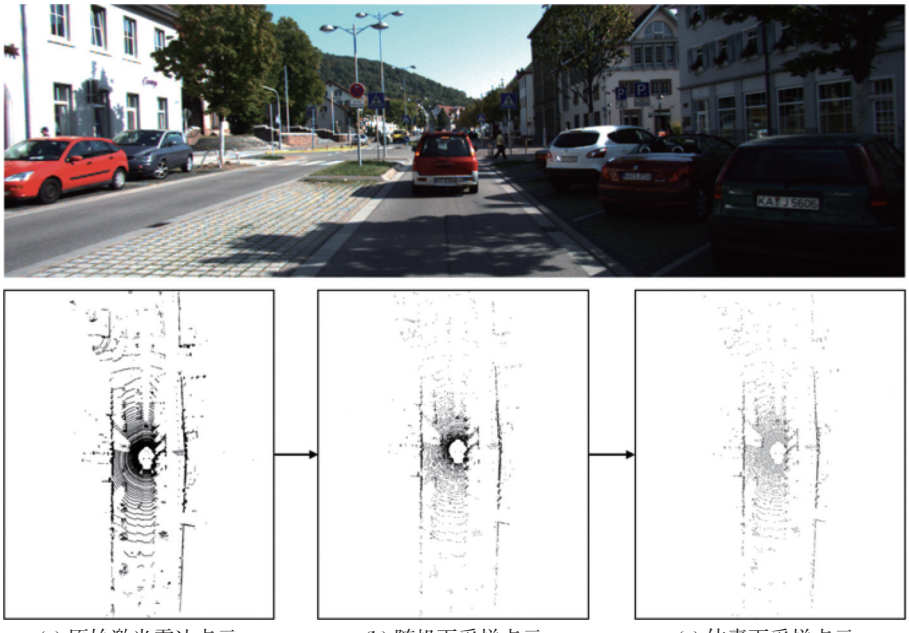
式中：T_{P}为真正例(true positive)；F_{P}为假正例(false positive)；F_{N}为假负例(false negative)；$P(R)$为不同召回率对应的准确率。

4.1 点云预处理结果

对KITTI数据集提供的点云数据进行预处理，将下采样后的点云作为网络的输入。下采样的结果如图5所示。图5(a)为原始激光雷达点云；图5(b)为随机下采样后的点云；图5(c)为图5(b)基础上进行体素下采样后的点云。

4.2 图网络层数对检测结果的影响分析

在图神经网络中，顶点状态通过图网络层迭代更新，不断进行优化。本文研究了不同图网络层数(顶点特征迭代次数)对目标为汽车时的检测结果的影响，结果如表1所示。将图3中图网络层数l分别设置为1，2，3，4层，进行检测结果对比，可以发现，网络层数设置为3层的结果优于2层和1层的，这说明顶点特征迭代次数较少时，顶点在进行邻域聚合时的感受野不够广，局部特征无法通过边很好地向顶点聚合，需要加深网络以扩大感受野；但是，当图网络层数设置为4层时性能又有所下降，这说明随着网络深度进一步加深会造成训练的困难，这是由于层数变深造成过平滑[23-24]，从而导致图中节点的向量表示逐渐趋于相等。最终，本文方法使用3层图网络结构。



(a) 原始激光雷达点云 (b) 随机下采样点云 (c) 体素下采样点云

图5 点云预处理结果

Fig.5 Preprocessing results of point clouds

4.3 注意力机制对检测结果的影响分析

目前，不同算法在进行图构建时均会不可避免地引入噪声点信息，这也是图构建的难点之一。本文采用固定半径近邻算法进行图构建，顶点邻域中同样会包括背景噪声点。为使顶点特征更精确，在顶点特征聚合时引入了注意力机制，通过注意力机制降低特征聚合时邻域中噪声点特征占比。研究了注意力机制对目标为汽车时的检测结果的影响，结果如表2所示。结果表明，使用3层图网络结构时，使用注意力机制进行特征聚合可以有效过滤邻域中的噪声点特征，提升模型检测准确度。

表 1　不同图网络层数的检测平均准确度对比

Tab.1　The average precision comparison of graph network detection with different layers

图网络层数	对汽车的检测平均准确度/%		
	Easy	Moderate	Hard
1	78.48	69.81	60.78
2	78.95	70.15	61.03
3	80.03	71.35	61.44
4	79.73	71.28	61.32

表 2　是否使用注意力机制的检测平均准确度对比

Tab.2　The average precision comparison with or without attention mechanism

注意力机制	对汽车的检测平均准确度/%		
	Easy	Moderate	Hard
无注意力机制	78.87	70.17	60.56
有注意力机制	80.03	71.35	61.44

4.4　坐标校准机制对检测结果的影响分析

本文研究了坐标校准机制对目标为汽车时的检测结果的影响，结果如表 3 所示。将坐标校准机制实现为一个易于扩展的坐标偏移计算模块，分别对引入该模块和没有该模块的模型进行评估，可以发现引入坐标校准机制后，模型检测准确度有明显提升。这说明引入坐标校准机制后，相对坐标在特征迭代时对中心顶点位置特征的依赖变少，更多地依赖点云自身的结构特征，由式(9)计算得到的Δ_o可以有效抵消顶点特征在维度变化时引入的坐标偏移误差。

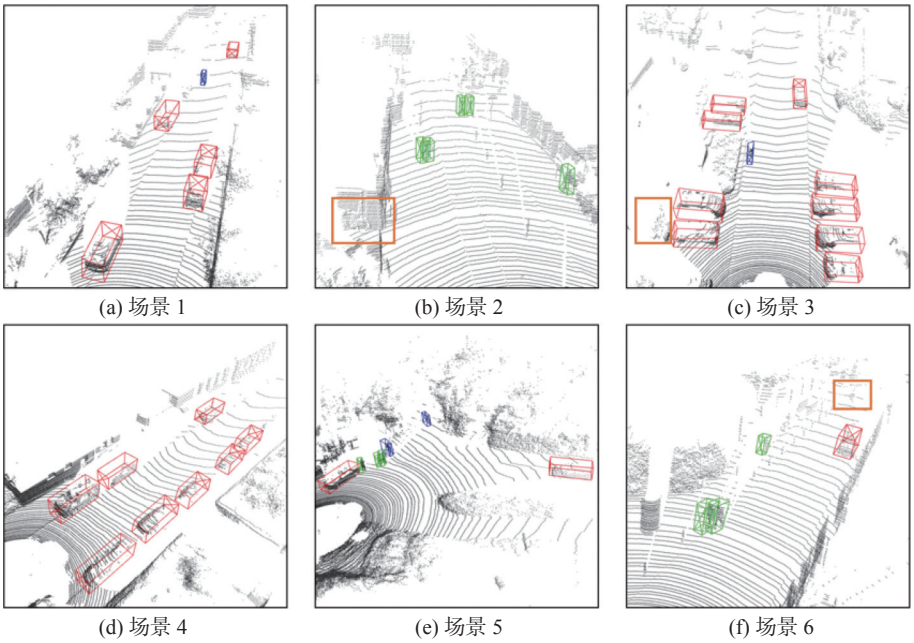
表 3　是否使用坐标校准机制的检测平均准确度对比

Tab.3　The average precision comparison with or without coordinate calibration mechanism

坐标校准机制	对汽车的检测平均准确度/%		
	Easy	Moderate	Hard
无坐标校准机制	79.67	71.02	60.86
有坐标校准机制	80.03	71.35	61.44

4.5　典型场景下的 3D 目标检测结果

图 6 为本文方法在 KITTI 数据集中一些典型场景下的检测结果，主要检测场景中的汽车、行人和骑行者。图 6(a)为汽车与骑行者同时存在的场景，且骑行者目标距离激光雷达较远；图 6(b)为行人多且密集的场景；图 6(c)和(d)为汽车多且密集的场景，同时存在车辆之间的遮挡；图 6(e)和(f)为汽车与行人同时存在的场景，且存在不同目标之间的遮挡。结果表明，本文方法对以上场景中较近的、清晰的目标可以准确检测；对于密集目标也均可准确识别；对于距离较远、难以分辨的行人等较小目标也可以进行有效检测并取得



(a) 场景 1　(b) 场景 2　(c) 场景 3

(d) 场景 4　(e) 场景 5　(f) 场景 6

（汽车、骑行者和行人分别以红色、蓝色和绿色 3D 框显示）

(car, cyclist and pedestrian are shown in red, blue and green bounding boxes respectively)

图 6　不同场景下的点云可视化检测结果

Fig.6　Point cloud detection results in different scenarios

良好的效果。在点云图作了可视化显示，将汽车、行人以及骑行者目标分别以红色、绿色和蓝色3D框标识。同时本文方法也存在一些误检、漏检的情况，导致检测失败的主要原因是：a. 目标距离激光雷达过远，无法获得足够多的点云信息。如图6(f)中橘色框标出的骑行者。b. 存在严重遮挡情况，激光雷达扫描出的点云无法构成一个较完整目标。如图6(c)中橘色框中汽车。c. 目标距离墙壁等障碍物过近，无法从点云中获取精确的轮廓信息。如图6(b)中橘色框标出的骑行者。

4.6 定量评估及与其他方法的比较

将本文方法与基于点云−图像融合的方法如AVOD，MV3D，MLOD和F-PointNet，基于欧几里得变换的方法BirdNet+以及基于网格的方法VoxelNet均进行了比较，各种方法在Easy，Moderate，Hard这3种场景中，分别对汽车、行人和骑行者3类目标检测的性能及算法效率如表4所示，其中，N/A表示该方法未公开该类别检测结果。相比于一些基于点云−图像融合的方法，本文方法取得了一些准确度的提升，但推理时间偏慢，在汽车类别的检测结果优于MV3D和MLOD，与F-PointNet检测结果接近；在行人以及骑行者类别的检测结果远优于AVOD，明显改善了对小目标的检测效果。相比于基于欧几里得变换的方法BirdNet+，本文方法也实现了准确度上的领先。相比于VoxelNet，算法效率有一定差距，但从准确度指标进行比较，仅在行人类别上的检测不占优势，其他两类的检测准确度较接近，总体来说，本文方法的平均准确度优于VoxelNet。通过计算可知，本文方法在Easy，Moderate，Hard场景中对汽车、行人和骑行者的检测平均准确度分别是70.94%，54.73%和41.38%，单帧计算耗时457 ms，其中，数据加载及预处理耗时44 ms，固定半径近邻图构建耗时103 ms，图网络层迭代耗时287 ms，目标类别以及3D框预测耗时23 ms。结果表明，本文方法可以很好地检测复杂交通场景中的各类目标。

表4 KITTI数据集3D目标检测平均准确度(m_{AP})对比

Tab.4 Average precision(m_{AP}) comparison of 3D object detection on the KITTI dataset

方法	单帧计算耗时/s	模式	对汽车的检测m_{AP}/%			对骑行者的检测m_{AP}/%			对行人的检测m_{AP}/%		
			Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
AVOD[16]	0.080	LiDAR+Image	76.39	66.47	60.23	57.19	42.08	38.29	36.10	27.86	25.76
MV3D[15]	0.360	LiDAR+Image	68.35	54.54	49.16	N/A	N/A	N/A	N/A	N/A	N/A
MLOD[18]	0.120	LiDAR+Image	77.24	67.76	62.05	68.81	49.43	42.84	47.58	37.4	35.07
F-PointNet[17]	0.170	LiDAR+Image	81.20	70.39	62.19	71.96	56.77	50.39	51.21	44.89	40.23
BirdNet+[5]	0.110	LiDAR	76.15	64.04	59.79	65.67	53.84	49.06	41.55	35.06	32.93
VoxelNet[7]	0.230	LiDAR	81.97	65.46	62.85	67.17	47.65	45.11	57.86	53.42	48.87
本文	0.457	LiDAR	80.03	71.35	61.44	68.76	51.36	44.08	46.26	40.23	37.66

本文方法的检测准确度与当前3D目标检测最优算法的相比，仍有一定差距，有待于进一步提升。造成这种差别的原因如下：其一，本文使用随机下采样和体素下采样算法对点云进行下采样，相比于其他基于视锥体、球形体素的采样方法，目标点被稀释，导致模型准确度下降，需要考虑使用更有效的点云预处理方法，尽量多的保留前景点；其二，本文采用图顶点分类方法进行预测和分类，没有对不同尺度的特征进行聚合，这同样会导致模型准确度降低；其三，相比其他点云−图像融合的算法，本文方法仅使用激光雷达点云作为输入，若考虑融合其他传感器数据可使检测准确度进一步提升。对于目标检测算法，模型推理速度至关重要。目前图网络的缺点之一在于运算量巨大，这是由于图网络特性导致的。将完整点云图送入网络后，每一次特征传递的过程

中都有极大的运算量，尤其是图构建以及图网络层，故还需考虑如何加快模型推理速度。

5　结束语

本文将基于图方法的图注意力卷积神经网络应用于自动驾驶场景的 3D 目标检测。将激光点云数据以图结构数据的形式表示，区别于基于投影和体素化等传统方法，避免对点云进行繁琐而复杂的预处理。此外，本文中引入的注意力机制考虑到节点特征在特征聚合时的权重占比，可以得到更精确的特征。实验证明，本文方法可以实时产生精确的边界框位置以及分类结果，表明了该方法的有效性。未来的改进方向，主要是改进数据预处理方法以解决图网络运算复杂的问题；以及研究如何高效地从点云中提取特征，从而加快模型推理速度。

参考文献：

[1] LIU L K, CHAN S H, NGUYEN T Q. Depth reconstruction from sparse samples: representation, algorithm, and sampling[J]. IEEE Transactions on Image Processing, 2015, 24(6): 1983–1996.

[2] GRIGORESCU S, TRASNEA B, COCIAS T, et al. A survey of deep learning techniques for autonomous driving[J]. Journal of Field Robotics, 2020, 37(3): 362–386.

[3] LI B, ZHANG T L, XIA T. Vehicle detection from 3D lidar using fully convolutional network[C]//Proceedings of 2016 Robotics: Science and Systems. Ann Arbor:MIT Press Journals, 2016.

[4] MEYER G P, LADDHA A, KEE E, et al. LaserNet: an efficient probabilistic 3D object detector for autonomous driving[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019: 12677–12686.

[5] BARRERA A, BELTRÁN J, GUINDEL C, et al. BirdNet+: two-stage 3D object detection in LiDAR through a sparsity-invariant bird ’s eye view[J]. IEEE Access, 2021, 9: 160299–160316.

[6] YANG B, LUO W J, URTASUN R. PIXOR: real-time 3D object detection from point clouds[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 7652–7660.

[7] ZHOU Y, TUZEL O. VoxelNet: end-to-end learning for point cloud based 3D object detection[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 4490–4499.

[8] YAN Y, MAO Y X, LI B. SECOND: sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[9] QI CHARLES R, SU H, 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. Honolulu: IEEE, 2017: 652–660.

[10] 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. Long Beach: Curran Associates Inc. , 2017: 5105–5114.

[11] SHI S S, WANG X G, LI H S. PointRCNN: 3D object proposal generation and detection from point cloud[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019: 770–779.

[12] 马帅, 刘建伟, 左信. 图神经网络综述 [J]. 计算机研究与发展, 2022, 59(1): 47–80.

[13] SHI W J, RAJKUMAR R. Point-GNN: graph neural network for 3D object detection in a point cloud[C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020.

[14] ZARZAR J, GIANCOLA S, GHANEM B. PointRGCN: Graph convolution networks for 3D vehicles detection refinement[EB/OL].(2019-11-27). https://doi.org/10.48550/arXiv.1911.12236

[15] CHEN X Z, MA H M, WAN J, et al. Multi-view 3D object detection network for autonomous driving[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 1907–1915.

[16] KU J, MOZIFIAN M, LEE J, et al. Joint 3D proposal generation and object detection from view aggregation[C]//Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. Madrid: IEEE, 2018: 1–8.

[17] QI C R, LIU W, WU C X, et al. Frustum PointNets for

3D object detection from RGB-D data[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 918–927.

[18] DENG J, CZARNECKI K. MLOD: a multi-view 3D object detection based on robust feature fusion method[C]//Proceedings of 2019 IEEE Intelligent Transportation Systems Conference. Auckland: IEEE, 2019: 279–284.

[19] SCARSELLI F, GORI M, TSOI A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2009, 20(1): 61–80.

[20] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[C]//Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview. net, 2017.

[21] 徐冰冰, 岑科廷, 黄俊杰, 等. 图卷积神经网络综述 [J]. 计算机学报, 2020, 43(5): 755–780.

[22] GEIGER A, LENZ P, STILLER C, et al. Vision meets robotics: the KITTI dataset[J]. The International Journal of Robotics Research, 2013, 32(11): 1231–1237.

[23] LI Q M, HAN Z C, WU X M. Deeper insights into graph convolutional networks for semi-supervised learning[C]//Proceedings of the 32nd AAAI Conference on Artificial Intelligence. New Orleans: AAAI Press, 2018.

[24] ZHAO L X, AKOGLU L. PairNorm: tackling oversmoothing in GNNs[C]//Proceedings of the 8th International Conference on Learning Representations. Addis Ababa:ICLR, 2020.

（编辑：黄 娟）