自动化学报
ACTA AUTOMATICA SINICA

一种基于自监督学习的矢量球面卷积网络

陈康鑫 赵杰煜 陈豪

A Vector Spherical Convolutional Network Based on Self-supervised Learning

CHEN Kang-Xin, ZHAO Jie-Yu, CHEN Hao

在线阅读 View online: https://doi.org/10.16383/j.aas.c220694

您可能感兴趣的其他文章

深度学习在基于单幅图像的物体三维重建中的应用

Application of Deep Learning to 3D Object Reconstruction From a Single Image

自动化学报. 2019, 45(4): 657-668 https://doi.org/10.16383/j.aas.2018.c180236

基于卦限卷积神经网络的3D点云分析

Octant Convolutional Neural Network for 3D Point Cloud Analysis

自动化学报. 2021, 47(12): 2791-2800 https://doi.org/10.16383/j.aas.c200080

基于运动目标三维轨迹重建的视频序列同步算法

Synchronization of Video Sequences Through 3D Trajectory Reconstruction

自动化学报. 2017, 43(10): 1759-1772 https://doi.org/10.16383/j.aas.2017.c160584

一种基于场景图分割的混合式多视图三维重建方法

A Hybrid Multi-View 3D Reconstruction Method Based on Scene Graph Partition

自动化学报. 2020, 46(4): 782-795 https://doi.org/10.16383/j.aas.c180155

基于视觉的三维重建关键技术研究综述

Key Techniques for Vision Based 3D Reconstruction: a Review

自动化学报. 2020, 46(4): 631-652 https://doi.org/10.16383/j.aas.2017.c170502

基于概念格因子分解的零件三维CAD模型检索

Concept Lattice Factorization Based 3D Model Retrieval of Mechanical Parts

自动化学报. 2019, 45(2): 407-419 https://doi.org/10.16383/j.aas.c170205

第 49 卷 第 6 期 自 动 化 学 报 Vol. 49, No. 6
2023 年 6 月 ACTA AUTOMATICA SINICA June, 2023

一种基于自监督学习的矢量球面卷积网络

陈康鑫 [1] 赵杰煜 [1,2] 陈 豪 [1]

摘 要 在三维视觉任务中, 三维目标的未知旋转会给任务带来挑战, 现有的部分神经网络框架对经过未知旋转后的三维目标进行识别或分割较为困难. 针对上述问题, 提出一种基于自监督学习方式的矢量型球面卷积网络, 用于学习三维目标的旋转信息, 以此来提升分类和分割任务的表现. 首先, 对三维点云信号进行球面采样, 映射到单位球上; 然后, 使用矢量球面卷积网络提取旋转特征, 同时将随机旋转后的三维点云信号输入相同结构的矢量球面卷积网络提取旋转特征, 利用自监督网络训练学习旋转信息; 最后, 对随机旋转的三维目标进行目标分类实验和部分分割实验. 实验表明, 所设计的网络在测试数据随机旋转的情况下, 在 ModelNet40 数据集上分类准确率提升 75.75%, 在 ShapeNet 数据集上分割效果显著, 交并比 (Intersection over union, IoU) 提升 51.48%.

关键词 矢量球面卷积网络, 自监督学习, 三维目标分类, 三维目标部分分割

引用格式 陈康鑫, 赵杰煜, 陈豪. 一种基于自监督学习的矢量球面卷积网络. 自动化学报, 2023, **49**(6): 1354−1368

DOI 10.16383/j.aas.c220694

A Vector Spherical Convolutional Network Based on Self-supervised Learning

CHEN Kang-Xin[1] ZHAO Jie-Yu[1,2] CHEN Hao[1]

Abstract The unknown rotation of 3D objects can bring challenges to the 3D vision tasks. It is difficult for some existing neural networks to classify or segment the 3D model after the unknown rotation. Aiming at the above problems, this paper proposes a vector spherical convolutional network based on self-supervised for learning the rotation information of 3D objects. First, the 3D point cloud signal is spherically sampled and mapped to the unit sphere; then, the rotational features are extracted by the vector spherical convolution network, while the randomly rotated 3D point cloud is input into the vector spherical convolution network of the same structure to extract rotation features, and the self-supervised network is used to train and learn the rotation information; finally, target classification experiments and part segmentation experiments are performed on randomly rotated 3D objects. Experimental results show that the network designed in this paper has a 75.75% improvement in classification accuracy on the ModelNet40 dataset and a significant segmentation effect on the ShapeNet dataset with a 51.48% improvement in the intersection over union (IoU) under random rotation of the test data.

Key words Vector spherical convolutional network, self-supervised learning, 3D object classification, 3D object part segmentation

Citation Chen Kang-Xin, Zhao Jie-Yu, Chen Hao. A vector spherical convolutional network based on self-supervised learning. *Acta Automatica Sinica*, 2023, **49**(6): 1354−1368

近年来, 随着自动驾驶、无人机算法、工业机器人等应用领域的兴起, 越来越多的计算机视觉领域的研究者展开对三维图像处理的研究. 相比于传统二维图像数据处理, 三维图像数据处理更加复杂, 比如在工业机器人机械臂抓取物体任务中, 如果要求机械臂能够从任意角度抓取物体, 则需要机器人能够从任意角度获取物体的姿态, 即 6D 姿态估计任务[1−5]. 在 6D 姿态估计任务中, 三维目标的局部参考系 (Local reference frame, LRF) 大部分是通过手动标注的方式获取[6−7]. 如果能够自动地获取目标的局部参考系, 也就是规范方向, 就能够减少大量手动标注数据的工作. 因此, 需要一种神经网络能够有效地学习旋转信息, 即要求神经网络具有旋转等变性 (Rotation equivariant)[8−9].

为使神经网络具有旋转等变性, 首先要求在特征提取层具有旋转等变能力. 卷积神经网络 (Convolutional neural networks, CNN) 作为图像特征提取模块, 在二维图像处理上已经取得巨大的成功.

收稿日期 2022-09-02 录用日期 2022-12-27
Manuscript received September 2, 2022; accepted December 27, 2022
国家自然科学基金 (62071260, 62006131), 浙江省自然科学基金 (LZ22F020001, LQ21F020009) 资助
Supported by National Natural Science Foundation of China (62071260, 62006131) and Natural Science Foundation of Zhejiang Province (LZ22F020001, LQ21F020009)
本文责任编委 黄华
Recommended by Associate Editor HUANG Hua
1. 宁波大学信息科学与工程学院 宁波 315211 2. 浙江省移动网应用技术重点实验室 宁波 315211
1. Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211 2. Mobile Network Application Technology Key Laboratory of Zhejiang Province, Ningbo 315211

但是在面对三维数据时, CNN 并不能够很好的处理[9]. 首先, CNN 无法直接对三维数据进行卷积操作[10]. 如果将三维数据展开到二维平面上, 再使用二维卷积网络提取特征, 这样的三维数据在展开时会产生畸变[9, 11], 导致三维数据原始信息的部分缺失和改变. 其次, CNN 也不具有旋转等变性[12], 无法满足提取旋转信息的需求. 因此, Cohen 等[13] 提出群等变卷积网络, 并在此基础上, Cohen 等[11] 和 Esteves 等[9] 提出球面卷积 (Spherical CNNs), 球面卷积通过射线的方式将三维网格数据 (3D mesh) 投射到单位球面 (Unit spheres) 上, 再通过所设计的球面卷积去提取全局特征, 球面卷积在理论上被证明是旋转等变的, 所以能够有效提取到旋转等变特征. You 等[14] 提出 PRIN (Point-wise rotation invariant network) 网络, 使用密度自适应采样模块 (Density aware adaptive sampling, DAAS) 将三维点云信号采样到球面上, 实现三维点云数据在球面卷积上的使用. Mitchel 等[15] 提出一种莫比乌斯等变球面卷积算子, 将 $SO(3)$ 空间等变的球面卷积扩展到莫比乌斯变换群上.

由于池化操作会破坏旋转等变性[16], Hinton 等[17] 提出胶囊网络 (Capsule networks), 将传统标量网络矢量化, 使其能够保留特征的姿态信息, 保留网络的旋转等变性, 并提出胶囊层间的动态路由机制 (Dynamic routing)[18], 定义一种低级胶囊向高级胶囊转化的计算方法; 之后改进路由机制, 提出期望最大化路由算法[19], 更好地处理胶囊层之间的部分–整体关系. Zhang 等[20] 提出基于加权的核密度估计 (Kernel density estimation, KDE) 路由算法, 在路由过程, 减少了 40% 的计算量. Iqbal 等[21] 认为动态路由机制本质上类似于聚类过程, 从聚类目标函数角度进行了优化. Gu 等[22] 提出共享权值的变换矩阵, 以此提升网络仿射变换鲁棒性. Gu 等[23] 认为动态路由机制会破坏网络的等变性, 提出直接对预测矢量神经元取平均的方式得到下一层矢量神经元, 以此替代动态路由算法. Venkataraman 等[24] 使用群等变卷积构建深层次的等变胶囊网络. 姚红革等[25] 提出了一种基于期望最大算法 (Expectation-maximization, EM) 向量聚类算法的深度胶囊网络, 实现了对重叠手写数字的识别和分离.

自监督学习 (Self-supervised learning) 网络在近几年越发受到关注[26–28]. Zbontar 等[29] 认为监督学习 (Supervised learning) 是构建更加智能的通用模型道路上所面临的一个瓶颈, 实际上, 想要标记世界上的所有事物是不可能做到的, 还有一些任务根本没有足够的标记数据, 例如针对一些数据匮乏的语言的翻译系统. 自监督学习网络架构有多种形式, 文献 [30–31] 和文献 [32–34] 在 90 年代和 21 世纪初提出孪生神经网络架构 (Siamese networks), 这种网络架构在近几年重新获得大量关注. 孪生神经网络架构由同一网络的两个相同 (或几乎相同) 的副本组成. 一个网络用 x 输入, 另一个网络用 y 输入, y 是 x 的数据增强版本. 这些网络输出得到结果向量 (Embedding), 分别对应所输入的 x 和 y. 然后将这些网络头对头连接起来, 将损失计算为这两个嵌入向量之间的距离. 当模型看到同一图像的不同变形版本时, 可以调整网络参数, 以使它们的输出靠得更近.

综上所述, 为解决目前神经网络无法很好地处理三维旋转目标相关任务的问题, 提出一种具有旋转等变性的自监督矢量球面卷积网络 (Self-supervised vector spherical convolutional networks), 又可简称为规范方向网络 (Canonical orientation network, CON), 对任意旋转的三维目标可以学习其规范方向.

本文的主要贡献如下:

1) 提出了一种基于自监督学习的矢量球面卷积网络, 能够将任意旋转的三维目标规范到其规范方向上;

2) 使用 $SO(3)$ 卷积算子替代传统的变换矩阵计算预测矢量神经元, 确保网络的旋转等变性;

3) 以共享 $SO(3)$ 卷积算子权值的方式计算预测矢量神经元, 增强网络对仿射变换的鲁棒性;

4) 提出一种平均路由计算方式替换传统的动态路由机制, 能更好地汇聚全局信息, 同时增强网络对仿射变换的鲁棒性.

1 基于自监督学习的矢量球面卷积网络

本文提出一种基于自监督学习的矢量球面卷积网络 (下称 CON 网络), 用于学习三维目标的旋转信息. 具体网络训练流程如图 1 所示.

如图 1 所示, 虚线框内为 CON 网络结构, CON 网络可以分为三维点云球面信号采样模块和矢量球面卷积模块. 其中, 三维点云球面信号采样模块对原始点云信号进行采样, 将点云信号映射到球面上; 矢量球面卷积模块使用球面卷积提取三维信号的特征, 并设计多个矢量层获取姿态胶囊, 最后得到旋转信息.

自监督网络采用孪生神经网络结构进行训练, 上下两个网络分支结构组成一致, 权值共享. 虚线框内的网络为本文设计的 CON 网络, 用于提取旋转特征, 将其设为映射 g. 则绿色虚线框算法过程可以用 $g(P)$ 表示, 蓝色虚线框算法过程可以用 $g(W)$ 表示.

又设旋转操作符 L_Q, $L_Q \cdot P$ 表示对 P 进行旋

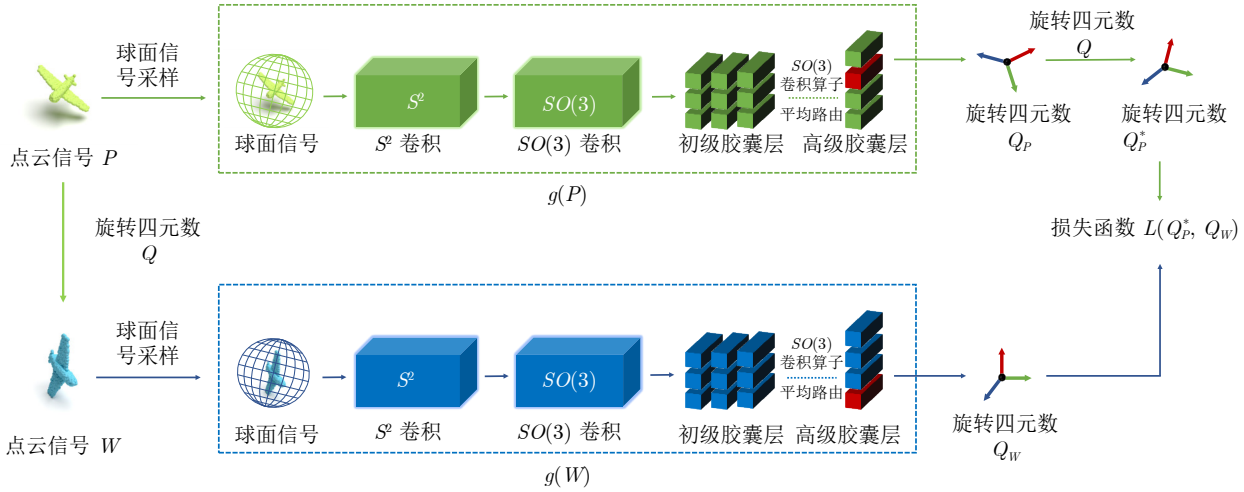


图 1 自监督矢量球面卷积网络训练流程图

Fig. 1 Self-supervised vector spherical convolutional network

转操作. 三维点云信号 P 和对其进行随机旋转后的点云信号为 $W = L_Q \cdot P$, P 和 W 分别输入两个网络分支中, 得到对应的旋转特征, 即旋转四元数 $Q_P = g(P)$ 和 $Q_W = g(W)$. 则根据第 1.4 节的旋转等变性定义, 应有:

$$Q_P^* = L_Q \cdot g(P) = g\left(L_Q \cdot P\right) = g(W) = Q_W \tag{1}$$

由此定义自监督网络的损失函数 $L\left(Q_P^*, Q_W\right)$. 即网络的训练目标为, 通过随机旋转的数据增强方式, 训练自监督网络得到一个满足式 (1) 的映射 g. 表 1 中列出了本文常用符号, 可供查阅.

表 1 常用符号表

Table 1 Table of common symbols

序号	符号	说明
1	(a_i, b_j, c_k)	球面网格坐标
2	(α_n, β_n, h_n)	点云用球面坐标表示
3	S^2	单位球面
4	$SO(3)$	三维旋转群
5	g	表示 CON 网络运算过程
6	f	指 S^2 或 $SO(3)$ 信号
7	L_R	旋转操作符
8	ψ	卷积核
9	$\boldsymbol{h}$	矢量神经元

1.1 三维点云球面信号采样模块

为方便后续球面卷积模块处理特征, 需要将不规则的点云信号转化为球面信号.

给定输入的点云信号, 先把每个点的坐标从欧几里得参考系转化成球面参考系, 然后再在球面坐标系统上构建量化网格. 每个点都可以用球面坐标表示[14, 35], 即 $x_n = (\alpha_n, \beta_n, h_n) \in S^2 \times H$, n 表示第 n 个点, α_n 表示方位角, β_n 表示倾角, h_n 表示到圆心的径向距离. 沿着固定的方位角和倾角, 可以划分多个不同径向距离的网格. 设有 K 个这样的网格, 则其可以被看作是映射到单位球面 S^2 的 (α, β) 位置的 K 值信号通道, K 是通道数, K 值球面信号 $f: S^2 \times H \to \mathbf{R}^K$.

对于每个球面网格的索引由该球面网格的中心 (a_i, b_j, c_k) 表示, 其中, $a_i \in [0, 2\pi]$, $b_j \in [0, \pi]$, $c_k \in [0, 1]$, $(i, j, k) \in I \times J \times K$, $a_i = (i/I) \cdot 2\pi$, $b_j = (j/J) \cdot \pi$, $c_k = k/K$. a_i, b_j, c_k 分别表示方位角、倾角和径向距离. 这里的 $I \times J \times K$ 表示空间分辨率, 也称为带宽.

为了避免等角离散化采样方式会导致输入信号在欧拉空间上的不一致性, 引入密度自适应采样[14]方式对点云信号进行处理.

使用以下公式计算球面信号 $f: S^2 \times H \to \mathbf{R}^K$:

$$f\left(a_i, b_j, c_k\right) = \frac{\sum\limits_{n=1}^{N} w_n \cdot \left(\xi - \|h_n - c_k\|\right)}{\sum\limits_{n=1}^{N} w_n} \tag{2}$$

其中 w_n 表示正则化因子:

$$w_n = \mathbf{1}\left(\|\alpha_n - a_i\| < \xi\right) \cdot \mathbf{1}\left(\|\beta_n - b_j\| < \eta\xi\right) \cdot \mathbf{1}\left(\|h_n - c_k\| < \xi\right) \tag{3}$$

$\mathbf{1}$ 是阈值函数, 当括号内不等式成立时, 函数取值为 1, 否则为 0. ξ 是预定义的卷积核阈值宽度, η 是密度自适应采样因子, 由此自适应调整采样密度, 减少畸变.

1.2 矢量球面卷积模块

在第 1.1 节中, 得到输入三维点云信号的球面

信号, 接下来利用球面卷积提取三维信号的特征, 通过多个 S^2 卷积层和 $SO(3)$ 卷积层提取三维点云特征, 然后进行矢量化, 构建初级胶囊层, 再通过 $SO(3)$ 卷积算子和平均路由的方式, 推理下一层矢量神经元, 构造多个高级胶囊层, 得到姿态胶囊, 最后通过 *soft-argmax* 层得到旋转信息.

1.2.1 球面卷积模块

球面卷积[9, 11] 能够提取分布在球面上的信号特征. 下面给出一些球面卷积的相关定义.

定义 1. 单位球面 S^2

S^2 是一个二维流形, 是到固定中心点距离为 1 的点 $\boldsymbol{x} \in \mathbf{R}^3$ 的集合. $\boldsymbol{x}$ 可以由方位角 $\alpha \in [0, 2\pi]$ 和极角 $\beta \in [0, \pi]$ 表示: $\boldsymbol{x}(\alpha, \beta)$.

定义 2. 球面信号

球面信号是定义在 S^2 上的 K 值函数, $f: S^2 \to \mathbf{R}^K$, K 是通道数.

定义 3. 三维旋转群 $SO(3)$

$SO(3)$ 是一个三维流形, 也是三维旋转群, 或称为特殊正交群, 其可以用 ZYZ 顺序旋转的欧拉角表示. 给定欧拉角三元组 (α, β, γ), 对应的旋转矩阵 R 可以由绕 z 轴的旋转 $R_z(\cdot)$ 和绕 y 轴的旋转 $R_y(\cdot)$ 的乘积来表示: $R(\alpha, \beta, \gamma) = R_z(\alpha)R_y(\beta)R_z(\gamma)$.

$$R = R(\alpha, \beta, \gamma) = R_z(\alpha)R_y(\beta)R_z(\gamma) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \cdot \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

三维数据点可以用向量 $\boldsymbol{x}$ 表示, 则 $\boldsymbol{x}$ 的旋转就可以用矩阵–向量乘法描述: $R\boldsymbol{x}$.

定义 4. S^2 卷积

设球面信号 $f: S^2 \to \mathbf{R}^K$, 卷积核 $\psi: S^2 \to \mathbf{R}^K$, *表示卷积运算, K 是特征数, S^2 卷积可以表示为:

$$[\psi * f](R) = \langle L_R\psi, f\rangle = \int_{S^2} \sum_{k=1}^{K} \psi_k\left(R^{-1}\boldsymbol{x}\right) f_k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \tag{5}$$

L_R 表示通过 $R \in SO(3)$ 对 f 进行任意角度的旋转, 等价于对输入 $\boldsymbol{x} \in S^2$ 左乘 R^{-1}, 即 $[L_R f](\boldsymbol{x}) = f(R^{-1}\boldsymbol{x})$ [9, 11].

定义 5. $SO(3)$ 卷积

类似地, 定义 $SO(3)$ 空间上的信号 $f: SO(3) \to \mathbf{R}^K$, 扩展 L_R 旋转操作符到 $SO(3)$ 空间: $[L_R f](Q) = f\left(R^{-1}Q\right)$.

这里, $R, Q \in SO(3)$, $R^{-1}Q$ 表示两个旋转矩阵相乘, 即旋转的复合.

设信号 f 和卷积核 ψ, $f, \psi: SO(3) \to \mathbf{R}^K$.

由此, 可以定义 $SO(3)$ 卷积:

$$[\psi * f](R) = \langle L_R\psi, f\rangle = \int_{SO(3)} \sum_{k=1}^{K} \psi_k\left(R^{-1}Q\right) f_k(Q)\mathrm{d}Q \tag{6}$$

定义 6. 球面信号快速傅里叶变换 (Fast Fourier transform, FFT)

为减少计算量, 提高运算速度, 利用球面信号 FFT[36] 把球面信号 f 和卷积核 ψ 变换到谱域进行计算, 这样可以减少计算量, 提高运算速度.

给出球面信号 FFT 的定义:

$$\widehat{\psi * f}^l = \widehat{\psi}^{l\dagger}\hat{f}^l \tag{7}$$

其中,

$$\hat{f}^l = \begin{cases} \displaystyle\int_{S^2} f(\boldsymbol{x})Y_M^l(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, & \boldsymbol{x} \in S^2 \\ \displaystyle\int_{SO(3)} f(\boldsymbol{x})D_{MM'}^l(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, & \boldsymbol{x} \in SO(3) \end{cases} \tag{8}$$

ψ^l 是可学习的傅里叶系数, $l, M, M' \in N^+$, $-l \le M, M' \le l$, $Y_M^l(\boldsymbol{x})$ 是球谐函数, 用来对 S^2 采用广义的傅里叶展开.

定义 7. 球面信号逆快速傅里叶变换 (Inverse fast Fourier transform, IFFT)

在谱域空间计算完成后, 需要把信号输出再转换回 $SO(3)$ 空间, 通过球面信号 IFFT 操作完成. 给出球面信号 IFFT 的定义:

$$f(\boldsymbol{x}) = \sum_{l=0}^{B-1}(2l+1)\sum_{M=-l}^{l}\sum_{M'=-l}^{l}\hat{f}_{MM'}^l U_X^l(\boldsymbol{x}) \tag{9}$$

这里, X 是 $SO(3)$ 流形或者 S^2 流形 (Manifold).

X 是 S^2 流形的时候, $U_X^l(\boldsymbol{x})$ 是球谐函数 $Y_M^l(\boldsymbol{x})$; X 是 $SO(3)$ 流形的时候, $U_X^l(\boldsymbol{x})$ 是维格纳 D 函数 $D_{MM'}^l(\boldsymbol{x})$. 球谐函数 $Y_M^l(\boldsymbol{x})$ 和维格纳 D 函数 $D_{MM'}^l(\boldsymbol{x})$ 有如下关系:

$$D_{MM'}^l(\alpha, \beta, \gamma) = Y_M^l(\alpha, \beta)\mathrm{e}^{\mathrm{i}\gamma} \tag{10}$$

根据以上定义, 球面信号先通过 S^2 卷积模块得到 $SO(3)$ 空间的特征, 再通过 $SO(3)$ 卷积模块进一步提取特征.

1.2.2 矢量球面卷积层

在球面卷积得到 $SO(3)$ 空间特征以后, 通过胶囊网络的形式构建多个矢量层, 对得到的 $SO(3)$ 空间特征进一步提取旋转信息.

原始的胶囊网络在获取预测矢量神经元的时候, 使用的是变换矩阵的方式[18]. 为了保持旋转等变性, 本文设计一种共享权值的 $SO(3)$ 卷积算子用来计算预测矢量神经元. 层间计算方式如图 2 所示.

设 $\boldsymbol{h}_i^l : SO(3) \to \mathbf{R}^{d^l}$, $\boldsymbol{h}_j^{l+1} : SO(3) \to \mathbf{R}^{d^{l+1}}$, 分别代表 l 层第 i 个矢量神经元和 $l+1$ 层第 j 个矢量神经元, d 表示矢量神经元的长度, $i \in \{0, \cdots, N_l - 1\}$, $j \in \{0, \cdots, N_{l+1} - 1\}$, N 表示矢量神经元的个数.

根据第 1.2.1 节定义的 $SO(3)$球面卷积 $[\psi * \boldsymbol{h}](R)$, 将信号 f 用信号 $\boldsymbol{h}_i^l$ 替换:

$$\boldsymbol{h}_{j|i}^{l+1}(R) = \left[\psi_j^{l+1} * \boldsymbol{h}_i^l\right](R) \tag{11}$$

其中, ψ_j^{l+1} 是对应的 $SO(3)$ 球面卷积核, 由此得到预测矢量神经元 $\boldsymbol{h}_{j|i}^{l+1}$.

同样地, 由于传统动态路由算法破坏了网络的旋转等变性[22], 本文设计平均路由的方式计算下一层矢量神经元, 如式 (12) 所描述:

$$\boldsymbol{s}_j^{l+1} = \frac{1}{N_l} \sum_{i=0}^{N_l - 1} \boldsymbol{h}_{j|i}^{l+1} \tag{12}$$

$\boldsymbol{s}_j^{l+1}$ 表示第 $l+1$ 层第 j 个矢量神经元. 如式 (12) 所示, 对所有的 $\boldsymbol{h}_{j|i}^{l+1}$ 预测矢量神经元取平均值, 可以得到 $\boldsymbol{s}_j^{l+1}$. 然后通过式 (13) 的压缩函数 (Squashing function), 将矢量神经元的模长压缩至 $0 \sim 1$ 范围.

$$\boldsymbol{h}_j^{l+1} = \frac{\left\|\boldsymbol{s}_j^{l+1}\right\|^2}{1 + \left\|\boldsymbol{s}_j^{l+1}\right\|^2} \frac{\boldsymbol{s}_j^{l+1}}{\left\|\boldsymbol{s}_j^{l+1}\right\|} \tag{13}$$

由此, 得到下一层矢量神经元. 按照这样的方式搭建多层矢量球面卷积层, 最终获取姿态胶囊. 具体算法步骤如下:

算法 1. 平均路由算法

输入. 矢量神经元 $\boldsymbol{h}_i^l$.

输出. 矢量神经元 $\boldsymbol{h}_j^{l+1}$.

1) 对第 l 层的所有胶囊 i: $\boldsymbol{h}_{j|i}^{l+1} \leftarrow \psi_j^{l+1} * \boldsymbol{h}_i^l$;

2) 对第 $l+1$ 层的所有胶囊 j: $\boldsymbol{s}_j^{l+1} \leftarrow \frac{1}{N_l} \sum_{i=0}^{N_l - 1} \boldsymbol{h}_{j|i}^{l+1}$;

3) 对第 $l+1$ 层的所有胶囊 j: $\boldsymbol{h}_j^{l+1} \leftarrow squashing(\boldsymbol{s}_j)^i$;

4) 返回 $\boldsymbol{h}_j^{l+1}$.

1.2.3 soft-argmax 操作

在本节中, 将介绍如何通过姿态胶囊, 获取到旋转信息.

在通过矢量球面卷积模块搭建多层矢量层后, 获取到姿态胶囊, 本文实验中设计的维度是 $1 \times 16 \times 48 \times 48 \times 48$, 表示 1 通道, 长度 16, 带宽为 24 的胶囊. 胶囊网络的模长代表该姿态胶囊代表的姿态概率, 概率值越大, 说明其越能代表特征的旋转信息. 因此, 通过选取姿态胶囊中模长最大的姿态胶囊, 获得对应的位置信息, 表示在 $SO(3)$ 空间上的坐标位置, 然后取对应的旋转信息, 在 $SO(3)$ 空

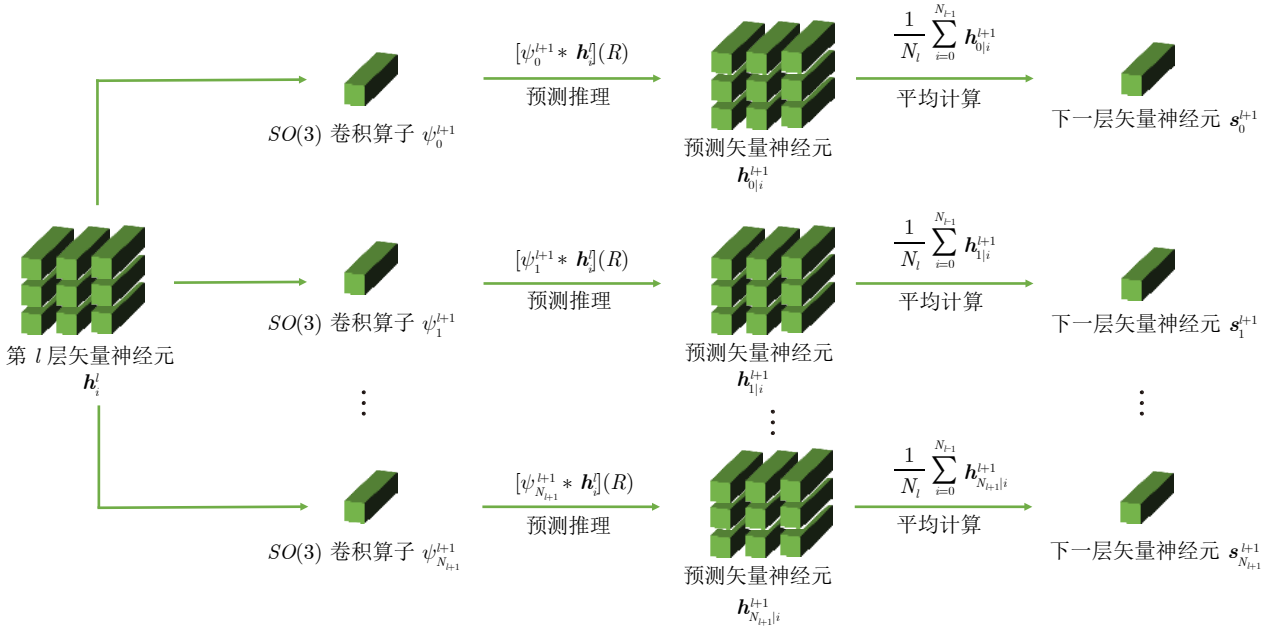


图 2 矢量球面卷积层间计算方法流程图

Fig. 2 Vector spherical convolution interlayer calculation method

间以欧拉角的形式保存. 如式 (14) 所描述, C_R 表示坐标, Φ 表示姿态胶囊, 利用 $soft\text{-}argmax$ 函数, 获取模长最大的姿态胶囊所对应的 $SO(3)$ 空间上对应的坐标.

$$C_R = soft\text{-}argmax(\Phi) = \sum_{i,j,k} softmax(\|\Phi\|)(i, j, k) \tag{14}$$

1.3 损失函数

网络整体采用孪生网络架构, 即使用两个结构相同的矢量球面卷积网络提取特征, 如图 1 所示.

设三维点云样本集合 S, 有两个点云数据 $P, W \in S$. $P = \{s_{P_i} \in \mathbf{R}^3 \mid s_{P_i} = (x, y, z)^{\mathrm{T}}\}$, $W = \{s_{W_i} \in \mathbf{R}^3 \mid s_{W_i} = (x, y, z)^{\mathrm{T}}\}$, W 是 P 经过随机旋转矩阵 $R \in SO(3)$ 得到的, 即 $W = L_R \cdot P$.

通过第 1.2.2 节的矢量层, 可以获得对应的旋转信息. 设 g 代表矢量球面卷积网络, 第一个网络分支学习 P 对应的旋转矩阵 $R_P = g(P)$, 第二个网络分支学习 W 对应的旋转矩阵 $R_W = g(W)$.

理想状态下, 由于网络是旋转等变的, 那么 $R_W = RR_P = R_P^*$.

因此, 网络训练目标设计成最小化 R_P^* 和 R_W 之间的差距, 即 $L(R_P^*, R_W)$.

由于欧拉角表示的旋转矩阵在沿着旋转轴旋转的时候, 第二次旋转的角度为 $\pm 90°$, 使得第一次和第三次旋转的旋转轴相同的现象称为万向锁问题或者奇异点问题[37].

因此, 本文将旋转矩阵转变为旋转四元数, 对损失函数进行优化.

四元数定义类似于复数, 区别在于四元数有三个虚部. 所有的 $q \in H$ (H 表示汉密尔顿代数空间 (Hamilton algebra)), 形式如下:

$$q = s\mathbf{1} + x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}, \quad s, x, y, z \in \mathbf{R} \tag{15}$$

其中, $\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -\mathbf{1}$. $s \in \mathbf{R}$ 看作四元数 q 的实部, 用标量表示; $\boldsymbol{v} = (x, y, z)^{\mathrm{T}} \in \mathbf{R}^3$ 看作四元数 q 的虚部, 用三维向量表示. 则四元数常表示成标量–向量有序对形式:

$$q = [s, \boldsymbol{v}], \boldsymbol{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad s, x, y, z \in \mathbf{R} \tag{16}$$

四元数模长定义为: $\|q\| = \sqrt{s^2 + x^2 + y^2 + z^2}$

类似于共轭复数, 四元数也有其共轭形式: $\bar{q} = s\mathbf{1} - x\boldsymbol{i} - y\boldsymbol{j} - z\boldsymbol{k}$.

则定义单位四元数: 满足模长 $\|q\| = \sqrt{s^2 + x^2 + y^2 + z^2} = 1 = q \cdot \bar{q}$ 的四元数称为单位四元数, 且 $\bar{q} = q^{-1}$.

单位四元数可以紧凑地表示三维旋转, 而且不会有万向锁问题和奇异点问题[38].

任意向量 $\boldsymbol{v}$ 沿着以单位向量定义的旋转轴 $\boldsymbol{u}$ 旋转 θ 角度之后, 得到向量 $\boldsymbol{v}'$, 令 $t = [0, \boldsymbol{v}]$, $t' = [0, \boldsymbol{v}']$, 分别代表向量 $\boldsymbol{v}$ 和向量 $\boldsymbol{v}'$ 的纯四元数形式, $q = [\cos(\theta/2), \sin(\theta/2)\boldsymbol{u}]$, 则 t' 可以由式 (17) 计算得到:

$$t' = qtq^* = qtq^{-1} \tag{17}$$

式 (17) 表示了如何使用四元数进行旋转操作.

下面给出四元数与旋转矩阵之间的转换. 设有旋转矩阵:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{18}$$

则四元数 $q = s\mathbf{1} + x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} = [s, \boldsymbol{v}]$, $(s, x, y, z \in \mathbf{R})$ 可由式 (19) 计算得到:

$$q_0 = \frac{\sqrt{1 + r_{11} + r_{22} + r_{33}}}{2}, \quad q_1 = \frac{r_{32} - r_{23}}{4q_0},$$
$$q_2 = \frac{r_{13} - r_{31}}{4q_0}, \quad q_3 = \frac{r_{21} - r_{12}}{4q_0} \tag{19}$$

其中, $q_0 \neq 0$, $1 + r_{11} + r_{22} + r_{33} > 0$, 即 $1 + \mathrm{tr}(R) > 0$.

第 1.3 节的损失函数是 $L(R_P^*, R_W)$, 通过式 (19) 将旋转矩阵转化为四元数后, 定义损失函数形式: $L(Q_P^*, Q_W)$.

$SO(3)$ 流形中的测地距离可以定义成两个旋转矩阵 R_P^* 和 R_W 之间的角距离, 即在 $SO(3)$ 流形中 R_P^* 到 R_W 的最短距离.

设 $p = Q_P^*$ 和 $q = Q_W$ 是 R_P^* 和 R_W 对应的单位四元数, 即旋转四元数. 则两个旋转四元数之间的差异可以表述为: $r = pq^*$, 这里的 q^* 指的是 q 的共轭四元数. 令 $r = [\cos(\theta/2), \boldsymbol{u}\sin(\theta/2)]$, $p = p_1\mathbf{1} + p_2\boldsymbol{i} + p_3\boldsymbol{j} + p_4\boldsymbol{k}$, $q = q_1\mathbf{1} + q_2\boldsymbol{i} + q_3\boldsymbol{j} + q_4\boldsymbol{k}$, 则:

$$\cos\frac{\theta}{2} = p_1q_1 + p_2q_2 + p_3q_3 + p_4q_4 \tag{20}$$

由式 (20) 可以得到损失函数:

$$L(Q_P^*, Q_W) = \theta = 2\arccos(|\langle p, q\rangle|) \tag{21}$$

其中, $\langle p, q\rangle = p_1q_1 + p_2q_2 + p_3q_3 + p_4q_4$, $|\cdot|$ 表示取绝对值.

1.4 旋转等变性

下面从理论证明, 本文所提出的网络具有旋转等变性.

定义 8. 等变映射

设 G 是一个群, X 是一个集合, $G \times X \mapsto X$

称为 G 在 X 上的一个 (左) 群作用, 即 X 是一个 G-集合.

如果对任意的 $g \in G$ 和任意的 $x \in X$, 有:

$$f(g \cdot x) = g \cdot f(x) \tag{22}$$

则称函数 $f: X \mapsto Y$ 是等变的[13, 39].

设 T 是群 G 的一个线性表示, 则有:

$$f(T(g) \cdot x) = T(g) \cdot f(x) \tag{23}$$

定义 9. 等变网络

如果一个网络的所有层都是等变的, 那么这个网络可以被称为等变网络[40].

因为等变性是可传递的, 所以由等变层组成的网络也是等变的. 如旋转等变的网络, 如果输入向量旋转一定的角度, 那么等变网络的输入也会旋转一样的角度[41−42].

定理 1. 球面卷积在 S^2 空间和 $SO(3)$ 空间的操作是旋转等变的.

证明. L_Q 和 L_R 是旋转操作符, 根据式 (5) 和式 (6) 相关球面卷积的定义, 球面卷积旋转等变性证明如下:

$$\begin{aligned}&[\psi * [L_Q \boldsymbol{h}]](R) = \langle L_R \psi, L_Q \boldsymbol{h} \rangle = \left\langle L_{Q^{-1}R} \psi, \boldsymbol{h} \right\rangle = \\ &[\psi * \boldsymbol{h}]\left(Q^{-1}R\right) = [L_Q[\psi * \boldsymbol{h}]](R)\end{aligned} \tag{24}$$

实际上 S^2 卷积可以看作是 $SO(3)$ 卷积的一种特殊情况, 具体推导见 Spherical CNNs [11]. □

定理 2. 矢量球面卷积层是旋转等变的.

证明. 如式 (11) 所示, 矢量球面卷积层在获取预测矢量神经元的时候是通过球面卷积的方式, 所以预测操作是旋转等变的.

设平均路由计算过程为 g, 则有: $\boldsymbol{s}_j^{l+1} = g\left(\boldsymbol{h}_{j|i}^{l+1}\right) = 1/N_l \sum_{i=0}^{N_l-1} \boldsymbol{h}_{j|i}^{l+1}$. 则:

$$\begin{aligned}L_R g\left(\boldsymbol{h}_{j|i}^{l+1}\right) &= L_R\left(\frac{1}{N_l}\sum_{i=0}^{N_l-1} \boldsymbol{h}_{j|i}^{l+1}\right) = \\ &\frac{1}{N_l}\left(L_R \sum_{i=0}^{N_l-1} \boldsymbol{h}_{j|i}^{l+1}\right) = \\ &\frac{1}{N_l}\left(\sum_{i=0}^{N_l-1} (L_R \boldsymbol{h}_{j|i}^{l+1})\right) = g\left(L_R \boldsymbol{h}_{j|i}^{l+1}\right)\end{aligned} \tag{25}$$

在获取下一层矢量神经元 $\boldsymbol{s}_j^{l+1}$ 部分, 取消动态路由机制, 使用平均预测矢量神经元 (见式 (12)) 来获取 $\boldsymbol{s}_j^{l+1}$, 如式 (25) 所示, 由于对旋转等变的 $\boldsymbol{h}_{j|i}^{l+1}$ 矢量神经元取平均操作不改变旋转等变性, 所以 $\boldsymbol{s}_j^{l+1}$ 保持旋转等变性.

又因为二范数操作和两个非零等变特征图的除法 (见式 (13)) 都是等变的[13], 所以压缩函数操作也是旋转等变的.

因此, 矢量球面卷积层是旋转等变的. □

定理 3. CON 网络是旋转等变网络.

证明. 根据定义 9 等变网络, 因为 CON 网络满足定理 1 和定理 2, 所以定理 3 成立. □

1.5 规范方向实验

为说明 CON 网络规范三维旋转目标的能力, 设计规范方向实验, 实验框架如图 3 所示.

CON 网络使用图 1 所示的自监督网络框架训练完成后, 为验证其获取三维目标规范方向的能力, 设计规范方向实验进行验证.

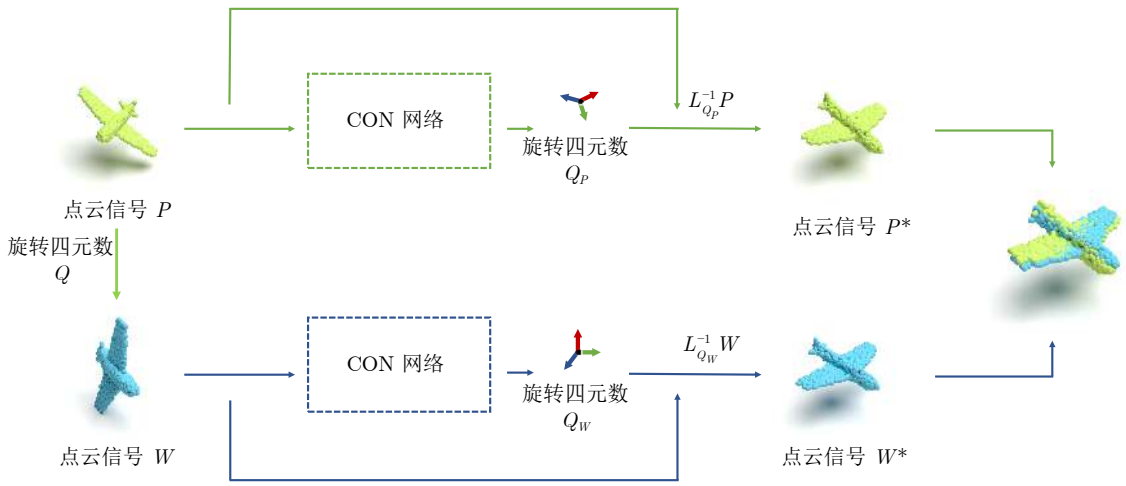


图 3 规范方向实验框架图

Fig. 3 Canonical orientation experiment framework diagram

假设点云信号 P, W, L_Q旋转操作符, $W = L_Q P$, g 表示 CON 网络计算过程, 则 $L_{Q_P} = g(P)$, $L_{Q_W} = g(W)$. 假设网络是严格旋转等变的, 有 $L_Q L_{Q_P} = L_{Q_W}$, 即损失函数为 0.

$$L_{Q_W}^{-1} W = (L_Q L_{Q_P})^{-1} W = L_{Q_P}^{-1} L_Q^{-1} L_Q P = L_{Q_P}^{-1} P \tag{26}$$

通过式 (26) 可知, CON 网络输出的旋转四元数代表 CON 网络认为的规范方向到点云信号的距离. 如图 3 所示, 点云信号 P 和 W 经过已经训练好的 CON 网络, 分别得到对应的旋转四元数 Q_P 和 Q_W, 利用得到的旋转四元数将对应的点云信号旋转到规范方向.

CON 网络对同一个模型的不同方向上的副本, 理应能够将它们旋转到同一个规范方向. 因此, 如图 3 所示, 将飞机模型信号 P^* 和 W^* 合并渲染.

1.6　分类与部分分割实验

为进一步说明 CON 网络对旋转三维目标的规范方向能力, 使用相对简单的 PointNet[43] 点云处理网络进行分类与部分分割实验.

点云网络结构图如图 4 所示, 输入维度 $n \times 3$ 的点云数据, 经过多个 T-Net 变换[43]、全连接层、最大池化操作后, 得到 $1 \times 1\,024$ 的全局特征, 再经过 $512 \times 256 \times k$ 的全连接层, 得到分类分数, k 是分类类别数; 同时, 在第二次 T-Net 变换后, 得到逐点特征, 将全局特征与其拼接, 得到 $n \times 1\,088$ 的特征, 再通过 $512 \times 256 \times 128 \times 128 \times m$ 的全连接层, 得到分割分数, m 是分割类别数.

如图 5 所示, 分类分割实验的输入数据先经过 CON 网络规范方向之后, 再通过 PointNet 网络进行分类和部分分割实验.

2　实验及结果分析

为说明所提出的 CON 网络获取任意旋转三维目标的规范方向的能力, 设计规范方向实验, 可视化规范方向模型.

同时也设计分类和部分分割两个下游任务, 进一步说明 CON 网络学习旋转信息的能力.

实验数据集主要使用 ModelNet40 数据集和 ShapeNet[44] 数据集, 每次使用的旋转操作符均为随机生成的任意角度的旋转.

2.1　规范方向实验结果分析

CON 网络采用 ModelNet40 数据集进行训练.

ModelNet40 数据集一共有 40 个类别 (包含飞

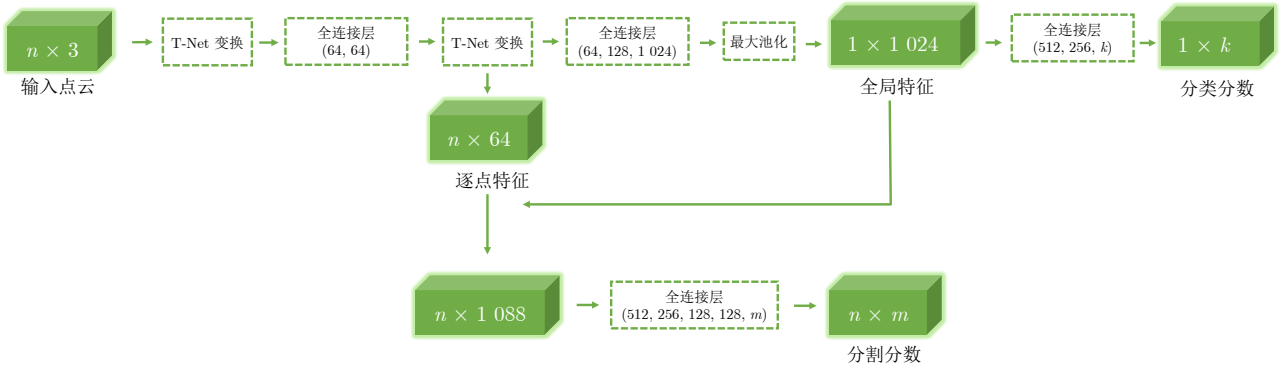


图 4　点云网络结构图

Fig. 4　PointNet architecture

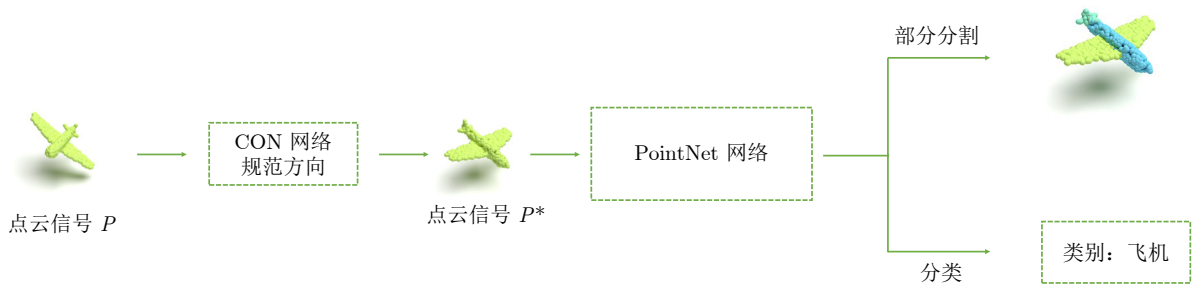


图 5　分类分割实验框架图

Fig. 5　Classification and segmentation experiment framework diagram

机、汽车、台灯等), 12 311 个模型, 其中的 9 843 个模型用于训练, 余下的 2 468 个模型用于测试.

CON 网络使用 PyTorch 深度学习框架实现, 在 NVIDIA GeForce 3080 显卡上实验, 操作系统为 Ubuntu LTS 20.04, 使用 Adam 优化器[45] ($\beta_1 = 0.9$, $\beta_2 = 0.999$), 学习率 0.0005, 批量大小为 16, 一共训练 100 轮.

网络包含一层 S^2, 一层 $SO(3)$ 层, 一层初级胶囊层, 一层高级胶囊层, 一层姿态胶囊层. 输入信号通道为 4; S^2 层和 $SO(3)$ 的带宽为 32, 通道为 40; 初级胶囊维度为 $5 \times 8 \times 48 \times 48 \times 48$, 经过 5 个 $SO(3)$ 卷积算子及平均路由运算, 得到的高级胶囊维度为 $5 \times 16 \times 48 \times 48 \times 48$, 再经过 1 个 $SO(3)$ 卷积算子及平均路由运算, 得到的姿态胶囊维度为 $1 \times 16 \times 48 \times 48 \times 48$, 然后通过 soft-argmax 层, 得到对应的姿态胶囊维度 $1 \times 1 \times 48 \times 48 \times 48$. 网络总体参数量为 112 691.

本文网络总体结构相对精简, 只使用了一层 $SO(3)$ 卷积. 多层的 $SO(3)$ 理论上能够一定程度上进一步地提取特征, 但是也会使计算量增大, 同时过多的堆叠 $SO(3)$ 反而会导致网络性能下降. 在进行多次调整网络参数后, 将一开始设计的多层 $SO(3)$ 和多层胶囊层的网络参数量从超过 30 万降到 112691, 同时保证网络的性能最佳.

CON 网络使用图 1 所示的自监督网络框架训练完成后, 为验证其获取三维目标规范方向的能力, 设计规范方向实验进行验证. 图 6 展示了 CON 网络对 ModelNet40 数据集上的多个模型规范方向的实验结果, 可以看到 CON 网络对两个不同方向的模型, 能够将它们旋转到一个十分接近的规范方向上, 说明了 CON 网络出色的规范方向的能力.

同时, 进一步测试了 CON 网络的迁移能力. 在 ModelNet40 数据集上训练的 CON 网络, 去规范 ShapeNet 数据集的模型方向, 结果如图 7 所示. 可以看到, 虽然 CON 网络没有见过 ShapeNet 数据集的模型, 但是仍然能够很好地得到它们的规范方向, 体现了 CON 网络良好的泛化能力.

2.2 分类与部分分割实验结果分析

为验证 CON 网络对旋转的处理能力, 使用相对简单的 PointNet[43] 点云处理网络进行分类与部分分割实验.

如图 5 所示, 分类分割实验的输入数据先经过 CON 网络规范方向之后, 再通过 PointNet 网络进行分类和部分分割实验.

对于是否旋转数据集, 分为 NR (No rotation) 和 AR (Arbitrary rotation) 两种方式, NR/NR 表示的是训练和测试阶段均不旋转数据集; NR/AR 表示的是训练阶段不旋转数据集, 测试阶段随机旋转数据集; AR/AR 表示的是训练和测试阶段均随机旋转, 实际上就是在训练的时候使用旋转的数据增强.

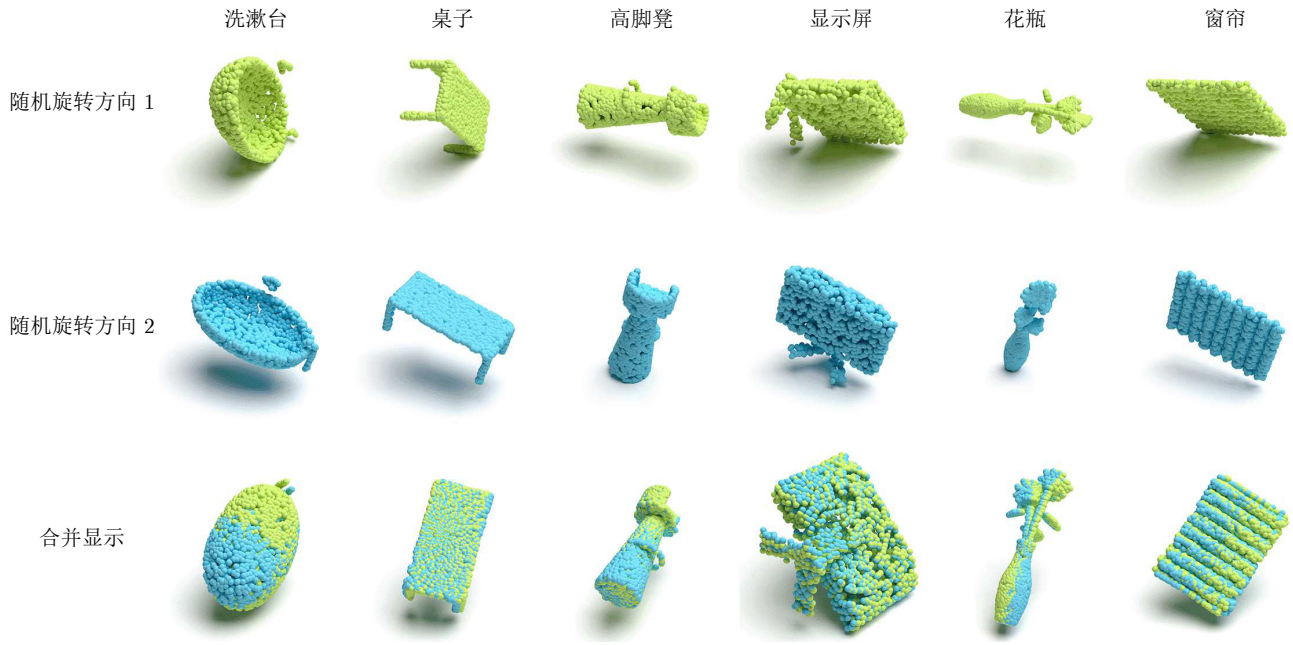


图 6 ModelNet40 规范方向实验可视化结果

Fig. 6 ModelNet40 canonical orientation experiment visualization results

2.2.1　分类实验

表 2 中, NR/NR 和 NR/AR 的对比实验数据均来自 SPRIN[14], AR/AR 是重新复现后的实验数据, 各网络均采用任意角度的随机旋转进行实验.

如表 2 所示, 在没有使用规范方向的时候, PointNet 在测试数据集旋转的情况下仅取得 12.47% 的准确率, 其他一些主流点云处理网络效果也不佳. 但是加了 CON 网络之后, 准确率显著提升到 88.22%, 显著高于部分无法处理三维旋转目标的网络, 如 PointNet++[46]、LDGCNN[47] 等. PRIN[49] 和 SPRIN[14] 是目前较为先进的处理点云不变特征的网络, 对比本文网络, 可以看到本文网络在分类准确率上取得了一定的提升. 虽然在不旋转测试数据集合的场景下, CON网络没有取得最高的准确率, 但是也足够反映出 CON 网络规范方向对 PointNet 网络处理旋转目标的帮助.

在 NR/NR 实验中, CON+PointNet 的准确率反而比 PointNet 更低, 实际上这是因为 CON 网络做了不必要的规范方向操作导致的. NR/NR 的模式下, PointNet 网络训练测试的数据集是完全一致的, 没有任何的角度偏差. 而通过 CON 网络规范方向的模型, 即使是同一个角度的模型, CON 网络给出的规范方向也是不完全相同的. 这是由于 CON 网络本身的误差导致的. 因此就会造成 CON+PointNet 在 NR/NR 模式下没有直接用 PointNet 效果好.

实验还对比了部分网络在旋转的数据增强 (AR/AR) 下的效果, 可以看到 PointNet 网络和 PointNet++ 网络即使在旋转的数据增强下, 仍然表现不佳. 这是由于 PointNet 网络依赖于在训练阶段见过的模型, 而旋转的数据增强不可能把无限的不同旋转角度的模型都放入网络进行训练, 因此在训练阶段遇到没有见过的旋转角度模型, PointNet 网络的表现就会很差. 而同样是通过旋转的数据增强方式训练的自监督的 CON 网络, 能够通过数据增强指导 CON 网络学习到模型的规范方向, 从而在下游任务中取得良好的性能.

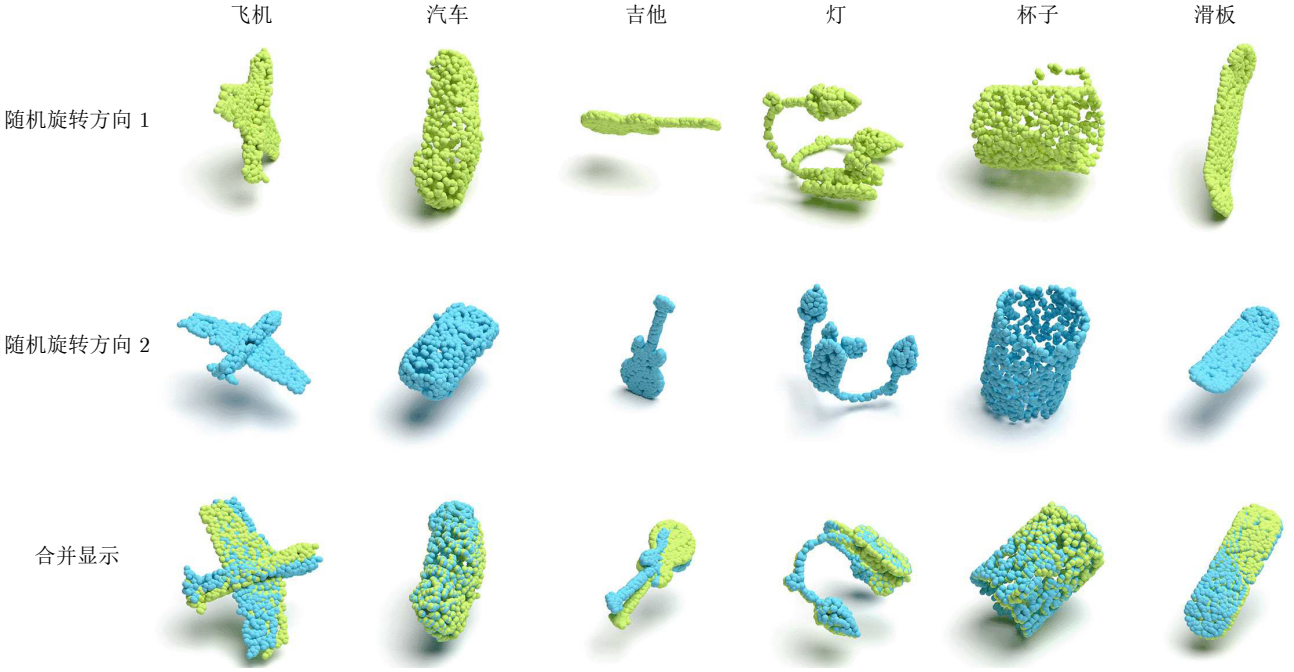


图 7　ShapeNet 规范方向实验可视化结果

Fig. 7　ShapeNet canonical orientation experiment visualization results

表 2　分类准确度 (%)

Table 2　Classification accuracy (%)

	PointNet[43]	PointNet ++[46]	Spherical CNN[11]	LDGCNN[47]	SO-Net[48]	PRIN[49]	SPRIN[14]	CON+PointNet
NR/NR	88.45	89.82	81.73	92.91	**94.44**	80.13	86.01	86.79
NR/AR	12.47	21.35	55.62	17.82	9.64	70.35	86.13	**88.22**
AR/AR	21.92	31.72	—	—	—	73.32	86.21	**88.27**

2.2.2 部分分割实验

本文同样使用 CON+ PointNet 网络设计部分分割实验, 验证 CON 网络对处理旋转三维目标网络的性能提升.

部分分割实验使用 ShapeNet 数据集. ShapeNet 数据集一共有 16 个大类, 表示飞机、杯子、耳机等模型类别; 一共有 50 个小类, 代表每个大类对应的小类, 如飞机有三个小类, 表示飞机的机翼、机身、机尾部件. 数据集一共有 16 881 个模型, 其中训练集 12 137 个, 验证集 1 870 个, 测试集 2 874个. 旋转进行实验.

对比了多个网络的分割实验结果, 同时重新复现了 PointNet、PointNet++、PRIN、SPRIN 的实验结果, 并将它们的分割结果可视化展示在图 8 中.

表 3 中的指标是部分分割里常用的交并比 (Intersection over union, IoU), IoU 越高, 说明分割效果越好. 表中的 avg. inst. 和 avg. cls. 分别表示实例平均交并比和类别平均交并比, 同时表中给出了

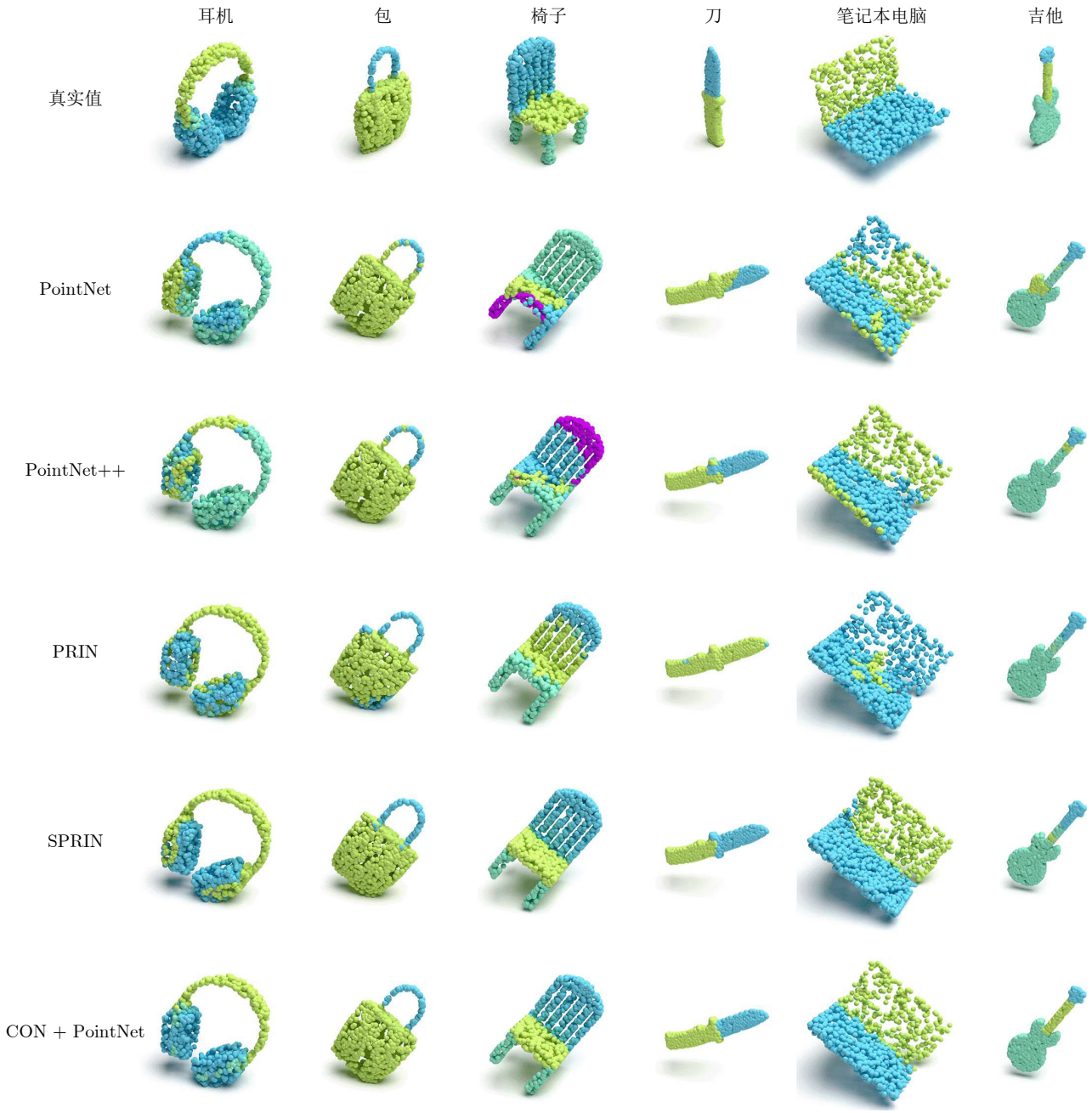


图 8 部分分割实验可视化结果

Fig. 8 Part segmentation experiment visualization results

表 3　部分分割实验结果 IoUs (%)

Table 3　Part segmentation experimental results IoUs (%)

	随机旋转																		不旋转	
	avg. inst.	avg. cls.	飞机	包	帽子	汽车	椅子	耳机	吉他	刀	灯	笔记本电脑	摩托车	马克杯	手枪	火箭	滑板	桌子	avg. inst.	avg. cls.
PointNet[43]	31.30	29.38	19.90	46.25	43.27	20.81	27.04	15.63	34.72	34.64	42.10	36.40	19.25	49.88	33.30	22.07	25.71	29.74	83.15	78.95
PointNet++[46]	36.66	35.00	21.90	51.70	40.06	23.13	43.03	9.65	38.51	40.91	45.56	41.75	18.18	53.42	42.19	28.51	38.92	36.57	84.63	81.52
RS-Net[50]	50.38	32.99	38.29	15.45	53.78	33.49	60.83	31.27	9.50	43.48	57.37	9.86	20.37	25.74	20.63	11.51	30.14	66.11	84.92	81.41
PCNN[51]	28.80	31.72	23.46	46.55	35.25	22.62	24.27	16.67	32.89	39.80	52.18	38.60	18.54	48.90	27.83	27.46	27.60	24.88	85.13	81.80
SPLATNet[52]	32.21	38.25	34.58	68.10	46.96	19.36	16.25	24.72	88.39	52.99	49.21	31.83	17.06	48.56	21.20	34.98	28.99	28.86	84.97	82.34
DGCNN[53]	43.79	30.87	24.84	51.29	36.69	20.33	30.07	27.86	38.00	45.50	42.29	34.84	20.51	48.74	26.25	26.88	26.95	28.85	85.15	82.33
SO-Net[48]	26.21	14.37	21.08	8.46	1.87	11.78	27.81	11.99	8.34	15.01	43.98	1.81	7.05	8.78	4.41	6.38	16.10	34.98	84.83	81.16
SpiderCNN[54]	31.81	35.46	22.28	53.07	54.2	22.57	28.86	23.17	35.85	42.72	44.09	55.44	19.23	48.93	28.65	25.61	31.36	31.32	**85.33**	**82.40**
SHOT+PointNet[55]	32.88	31.46	37.42	47.30	49.53	27.71	28.09	16.34	9.79	27.66	37.33	25.22	16.31	50.91	25.07	21.29	43.10	40.27	32.75	31.25
CGF+PointNet[56]	50.13	46.26	50.97	70.34	60.44	25.51	59.08	33.29	50.92	71.64	40.77	31.91	23.93	63.17	27.73	30.99	47.25	52.06	50.13	46.31
RIConv[57]	79.31	74.60	78.64	78.70	73.19	68.03	86.82	71.87	89.36	82.95	74.70	76.42	56.58	88.44	72.16	51.63	66.65	77.47	79.55	74.43
Kim 等[58]	79.56	74.41	77.53	73.43	76.95	66.13	87.22	**75.44**	87.42	80.71	78.44	71.21	51.09	90.76	73.69	53.86	68.10	78.62	79.92	74.69
Li 等[59]	82.17	78.78	81.49	80.07	85.55	74.83	88.62	71.34	90.38	82.82	80.34	81.64	68.87	92.23	74.51	54.08	74.59	79.11	82.47	79.40
PRIN[49]	71.20	66.75	69.29	55.90	71.49	56.31	78.44	65.92	86.01	73.58	66.97	59.29	47.56	81.47	71.99	49.02	64.70	70.12	72.04	68.39
SPRIN[14]	82.67	79.50	82.07	**82.01**	76.48	75.53	88.17	71.45	90.51	**83.95**	79.22	83.83	**72.59**	**93.24**	78.99	**58.85**	74.77	80.31	82.59	79.31
CON+PointNet	**84.39**	**80.86**	**82.27**	79.14	**85.88**	**76.44**	**90.42**	73.24	**90.96**	82.81	**82.99**	**95.64**	69.51	91.93	**79.74**	55.60	**75.33**	**81.81**	84.06	81.22

ShapeNet 数据集中 16 个大类模型各自的 IoU. 表中首行的随机旋转和不旋转指的是测试集是否随机旋转, 在训练的时候一律不旋转训练集.

PointNet、PointNet++、DGCNN[53] 和 SpriderCNN[54] 是目前较为先进的点云处理网络, 不过它们在测试集随机旋转的情况下, 分割效果显著下降.

SHOT[55] 和 CGF[56] 以局部参考帧的形式表示不规则的点云信号的局部几何信息, You 等[14] 设计了 SHOT/CGF + PointNet 形式, 将 SHOT/CGF 处理过的特征送入 PointNet 进行分割实验, 如表 3 所示, 部分分割实验结果并不理想.

RIConv 网络[57]、Kim 等[58] 以及 Li 等[59] 同样是面对处理旋转的三维目标, 不过在训练阶段采用了绕 z 轴旋转的数据增强的方法, You 等[14] 重新在无数据增强的情况下训练, 得到数据如表 3 所示.

PRIN/SPRIN 是目前较为先进的提取点云不变特征的网络, 这两个网络在处理旋转三维目标的时候有着较为出色的表现. 如表 3 所示, 本文的 CON+PointNet 网络的平均实例交并比和平均类别交并比在测试数据集随机旋转的情形下取得了较好的数据, 高于 PRIN/SPRIN 网络, 在表中对比网络中表现为最优.

同时在 16 个大类的模型中, 大部分类别模型的部分分割 IoU 也是拿到最好数据, 如表中加粗部分数据显示.

而且在随机旋转的情形下, 获得的平均实例交并比和平均类别交并比 (84.39% 和 80.86%) 接近不旋转情形下的最优数据 (SpiderCNN, 85.33% 和 82.40%).

这些数据表明, CON 网络规范方向对 PointNet 网络处理三维旋转目标的能力有着显著的提升.

图 8 展示了部分分割实验的可视化结果, 第一行是真实值, 下面几行是各个不同模型在面对旋转的三维目标情形下的实验结果.

2.2.3　与其他主流网络的结合

CON 网络作为一个前置网络模块, 可以看作是一个数据预处理过程, 与后续网络的结合是独立的, 理论上后面可以接任意的点云处理网络.

DGCNN 和 PointNet++ 是比 PointNet 更加先进的点云处理网络, 因此结合 CON 网络后处理旋转模型的能力理应比 PointNet 网络表现好.

对此, 本文做了 CON+DGCNN 和 CON+PointNet++ 的分类与分割实验, 实验结果如表 4 和表 5 所示. 由实验结果可知, CON 在结合了更加先进的点云处理网络以后, 整体网络的性能有显著的提升. 在分类实验中, DGCNN 的性能本来就优于 PointNet++, 在结合 CON 网络后同样表现的比 CON+PointNet++ 优异. 但是在分割实验中, CON+PointNet++ 的性能要优于 CON+DGCNN.

3　结束语

本文提出一种基于自监督学习的矢量球面卷积

表 4 与主流网络结合的分类准确度 (%)

Table 4 Classification accuracy in combination with mainstream networks (%)

	PointNet[43]	PointNet++[46]	DGCNN[53]	CON+DGCNN	CON+PointNet++	CON+PointNet
NR/NR	88.45	89.82	90.20	**88.32**	87.27	86.79
NR/AR	12.47	21.35	16.36	**89.86**	89.21	88.22
AR/AR	21.92	31.72	29.73	**89.93**	89.30	88.27

表 5 与主流网络结合的部分分割实验结果 IoUs (%)

Table 5 Experimental results of part segmentation combined with mainstream networks IoUs (%)

	随机旋转																		不旋转	
	avg. inst.	avg. cls.	飞机	包	帽子	汽车	椅子	耳机	吉他	刀	灯	笔记本电脑	摩托车	马克杯	手枪	火箭	滑板	桌子	avg. inst.	avg. cls.
PointNet[43]	31.30	29.38	19.90	46.25	43.27	20.81	27.04	15.63	34.72	34.64	42.10	36.40	19.25	49.88	33.30	22.07	25.71	29.74	83.15	78.95
PointNet++[46]	36.66	35.00	21.90	51.70	40.06	23.13	43.03	9.65	38.51	40.91	45.56	41.75	18.18	53.42	42.19	28.51	38.92	36.57	84.63	81.52
DGCNN[53]	43.79	30.87	24.84	51.29	36.69	20.33	30.07	27.86	38.00	45.50	42.29	34.84	20.51	48.74	26.25	26.88	26.95	28.85	85.15	82.33
CON+PointNet	84.39	80.86	82.27	79.14	85.88	76.44	90.42	73.24	90.96	82.81	82.99	95.64	69.51	**91.93**	79.74	55.60	75.33	81.81	84.06	81.22
CON+PointNet++	**85.77**	**82.30**	**84.12**	**80.66**	**88.90**	**76.51**	90.37	**78.65**	90.15	**83.01**	83.62	95.45	**71.26**	91.67	80.77	**60.36**	**77.23**	**84.01**	**86.02**	**83.41**
CON+DGCNN	85.21	81.36	83.71	79.02	86.91	74.21	**93.22**	74.43	**91.90**	82.31	**84.24**	**96.53**	70.22	90.86	**81.37**	58.28	76.96	83.27	85.73	82.62

网络, 通过球面卷积提取三维目标的特征, 并通过胶囊网络进一步提取特征并且保证旋转等变性, 同时使用自监督网络架构, 无需额外标签信息的情形下自主训练网络. 该网络训练完成后, 可以获取任意旋转方向的三维目标的规范方向. 这种规范方向的能力能够给以往无法处理三维旋转目标的网络 (如 PointNet) 提供帮助, 实验表明, 本文所提出的网络作为前置网络处理三维旋转目标确实能够给 PointNet 网络做分类或部分分割任务带来很大的提升. 同时本文实验还验证了本文所提出的网络具有良好的泛化能力, 在不同数据集上也能够有良好的表现. 本文所设计的网络针对点云数据, 而三维数据有着多样的形式, 如体素 (Voxel) 或者网格 (Mesh) 等. 因此, 今后将考虑处理更加复杂的网格形式的三维数据.

References

1 Piga N A, Onyshchuk Y, Pasquale G, Pattacini U, Natale L. ROFT: Real-time optical flow-aided 6D object pose and velocity tracking. *IEEE Robotics and Automation Letters*, 2022, **7**(1): 159−166

2 Gao F, Sun Q, Li S, Li W, Li Y, Yu J, et al. Efficient 6D object pose estimation based on attentive multi-scale contextual information. *IET Computer Vision*, 2022, **16**(7): 596−606

3 Peng W, Yan J, Wen H, Sun Y. Self-supervised category-level 6D object pose estimation with deep implicit shape representation. In: Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2022. 2082−2090

4 Huang W L, Hung C Y, Lin I C. Confidence-based 6D object pose estimation. *IEEE Transactions on Multimedia*, 2022, **24**: 3025−3035

5 Li X, Weng Y, Yi L, Guibas L J, Abbott A L, Song S, et al. Leveraging SE(3) equivariance for self-supervised category-level object pose estimation from point clouds. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2021. 15370−15381

6 Melzi S, Spezialetti R, Tombari F, Bronstein M M, Stefano L D, Rodol E. Gframes: Gradient-based local reference frame for 3D shape matching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2019. 4629−4638

7 Gojcic Z, Zhou C, Wegner J D, Wieser A. The perfect match: 3D point cloud matching with smoothed densities. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2019. 5545−5554

8 Hao Z, Zhang T, Chen M, Zhou K. RRL: Regional rotate layer in convolutional neural networks. In: Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2022. 826−833

9 Esteves C, Allen-Blanchette C, Makadia A, Daniilidis K. Learning SO(3) equivariant representations with spherical CNNs. In: Proceedings of European Conference on Computer Vision. Berlin, DE: Springer, 2018. 52−68

10 Chen Y, Zhao J Y, Shi C W. Mesh convolution: A novel feature extraction method for 3D nonrigid object classification. *IEEE Transactions on Multimedia*, 2021, **23**: 3098−3111

11 Cohen T S, Geiger M, Köhler J, Wellin M. Spherical CNNs. In: Proceedings of International Conference on Learning Representations. Vancouver, CA: 2018. 1−15

12 Gerken J E, Carlsson O, Linander H, Ohlsson F, Petersson C, Persson D. Equivariance versus augmentation for spherical images. In: Proceedings of International Conference on Machine Learning. New York, USA: PMLR, 2022. 7404−7421

13 Cohen T, Welling M. Group equivariant convolutional networks. In: Proceedings of International Conference on Machine Learning. New York, USA: PMLR, 2016. 2990−2999

14 You Y, Lou Y, Shi R, Liu Q, Tai Y W, Ma L Z, et al. Prin/sprin: On extracting point-wise rotation invariant features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, **44**(12): 9489−9502

15 Mitchel T W, Aigerman N, Kim V G, Kazhdan M. Möbius convolutions for spherical CNNs. In: Proceedings of ACM SIGGRAPH Annual Conference. New York, USA: ACM, 2022. 1−9

16 Mazzia V, Salvetti F, Chiaberge M. Efficient-capsnet: Capsule network with self-attention routing. *Scientific Reports*, 2021, **11**(1): 1−13

17 Hinton G E, Krizhevsky A, Wang S D. Transforming auto-encoders. In: Proceedings of International Conference on Artificial Neural Networks. Berlin, DE: Springer, 2011. 44–51

18 Sabour S, Frosst N, Hinton G E. Dynamic routing between capsules. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2017. 3856–3866

19 Hinton G E, Sabour S, Frosst N. Matrix capsules with EM routing. In: Proceedings of International Conference on Learning Representations. Vancouver, CA: 2018. 16–30

20 Zhang Z, Xu Y, Yu J, Gao S H. Saliency detection in 360 videos. In: Proceedings of European Conference on Computer Vision. Berlin, DE: Springer, 2018. 488–503

21 Iqbal T, Xu Y, Kong Q Q, Wanfg W W. Capsule routing for sound event detection. In: Proceedings of European Signal Processing Conference. Piscataway, USA: IEEE, 2018. 2255–2259

22 Gu J, Tresp V. Improving the robustness of capsule networks to image affine transformations. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2020. 7285–7293

23 Gu J, Tresp V, Hu H. Capsule network is not more robust than convolutional network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2021. 14309–14317

24 Venkataraman S R, Balasubramanian S, Sarma R R. Building deep equivariant capsule networks. In: Proceedings of International Conference on Learning Representations. Vancouver, CA: 2020. 1–12

25 Yao Hong-Ge, Dong Ze-Hao, Yu Jun, Bai Xiao-Jun. Fully overlapped handwritten number recognition and separation based on deep EM capsule network. *Acta Automatica Sinica*, 2022, **48**(12): 2996–3005
(姚红革, 董泽浩, 喻钧, 白小军. 深度 EM 胶囊网络全重叠手写数字识别与分离. 自动化学报, 2022, **48**(12): 2996–3005)

26 Saha S, Ebel P, Zhu X X. Self-supervised multisensor change detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2022, **60**: 1–10

27 Gong Y, Lai C I, Chung Y A, Glass J R. SSAST: Self-supervised audio spectrogram transformer. In: Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2022. 10699–10709

28 Sun L, Zhang Z, Ye J, Peng H, Zhang J W, Su S, et al. A self-supervised mixed-curvature graph neural network. In: Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2022. 4146–4155

29 Zbontar J, Jing L, Misra I, LeCun Y, Deny S. Barlow twins: Self-supervised learning via redundancy reduction. In: Proceedings of International Conference on Machine Learning. New York, USA: PMLR, 2021. 12310–12320

30 Becker S, Hinton G E. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 1992, **355**(6356): 161–163

31 Goldberger J, Hinton G E, Roweis S, Salakhutdinov R. Neighbourhood components analysis. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2004. 513–520

32 Bromley J, Bentz J W, Bottou L, Guyon I, LeCun Y, Moore C, et al. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 1993, **7**(4): 669–688

33 Hadsell R, Chopra S, Lecun Y. Dimensionality reduction by learning an invariant mapping. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2006. 1735–1742

34 Chopra S, Hadsell R, Lecun Y. Learning a similarity metric discriminatively, with application to face verification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2005. 539–546

35 Spezialetti R, Salti S, Stefano L D. Learning an effective equivariant 3D descriptor without supervision. In: Proceedings of International Conference on Computer Vision. Piscataway, USA: IEEE, 2019. 6400–6409

36 Driscoll J R, Healy D M. Computing Fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics*, 1994, **15**(2): 202–250

37 Thomas F. Approaching dual quaternions from matrix algebra. *IEEE Transactions on Robotics*, 2014, **30**(5): 1037–1048

38 Busam B, Birdal T, Navab N. Camera pose filtering with local regression geodesics on the riemannian manifold of dual quaternions. In: Proceedings of International Conference on Computer Vision Workshops. Piscataway, USA: IEEE, 2017. 2436–2445

39 Cohen T S, Geiger M, Weiler M. A general theory of equivariant CNNs on homogeneous spaces. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2019. 9142–9153

40 Zhao Y, Birdal T, Lenssen J E, Menegatti E, Guibas L J, Tombari F, et al. Quaternion equivariant capsule networks for 3D point clouds. In: Proceedings of European Conference on Computer Vision. Berlin, DE: Springer, 2020. 1–19

41 Kondor R, Trivedi S. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In: Proceedings of International Conference on Machine Learning. New York, USA: PMLR, 2018. 2747–2755

42 Lenssen J E, Fey M, Libuschewski P. Group equivariant capsule networks. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2018. 8858–8867

43 Qi C R, Su H, Mo K, Guibas L J. PointNet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2017. 77–85

44 Chang A X, Funkhouser T A, Guibas L J, Hanrahan P, Huang Q X, Li Z, et al. ShapeNet: An information-rich 3D model repository. arXiv preprint arXiv: 1512.03012, 2015.

45 Kingma D P, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv: 1412.6980, 2014.

46 Qi C R, Yi L, Su H, Guibas L J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2017. 5099–5108

47 Zhang K, Hao M, Wang J, Silva C W, Fu C L. Linked dynamic graph CNN: Learning on point cloud via linking hierarchical features. arXiv preprint arXiv: 1904.10014, 2019.

48 Li J, Chen B M, Lee G H. SO-Net: Self-organizing network for point cloud analysis. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2018. 9397–9406

49 You Y, Lou Y, Liu Q, Tai Y W, Ma L Z, Lu C W, et al. Pointwise rotation-invariant network with adaptive sampling and 3D spherical voxel convolution. In: Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI, 2020. 12717–12724

50 Huang Q, Wang W, Neumann U. Recurrent slice networks for 3D segmentation of point clouds. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2018. 2626–2635

51 Atzmon M, Maron H, Lipman Y. Point convolutional neural networks by extension operators. *ACM Transactions on Graphics*, 2018, **37**(4): 71–84

52 Su H, Jampani V, Sun D, Maji S, Kalogerakis E, Yang M H, et al. SPLATNet: Sparse lattice networks for point cloud processing. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2018. 2530–2539

53 Wang Y, Sun Y, Liu Z, Sarma S E, Bronstein M M, Solomon J M. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics*, 2019, **38**(5): 1–12

54 Xu Y, Fan T, Xu M, Zeng L, Qiao Y. SpiderCNN: Deep learning on point sets with parameterized convolutional filters. In: Proceedings of European Conference on Computer Vision. Berlin, DE: Springer, 2018. 90–105

55 Tombari F, Salti S, Stefano L D. Unique signatures of histograms for local surface description. In: Proceedings of European Conference on Computer Vision. Berlin, DE: Springer, 2010. 356–369

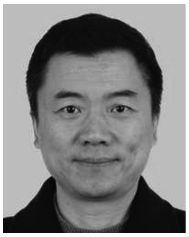
56 Khoury M, Zhou Q Y, Koltun V. Learning compact geometric features. In: Proceedings of International Conference on Computer Vision. Piscataway, USA: IEEE, 2017. 153–161

57 Zhang Z, Hua B S, Rosen D W, Yeung S K. Rotation invariant convolutions for 3D point clouds deep learning. In: Proceedings of International Conference on 3D Vision. Piscataway, USA: IEEE, 2019. 204–213

58 Kim S, Park J, Han B. Rotation-invariant local-to-global representation learning for 3D point cloud. In: Proceedings of Annual Conference on Neural Information Processing Systems. New York, USA: MIT Press, 2020. 8174–8185

59 Li X Z, Li R H, Chen G Y, Fu C W, Cohen-Or D, Heng P. A rotation-invariant framework for deep point cloud analysis. *IEEE Transactions on Visualization and Computer Graphics*, 2021, **28**(12): 4503–4514

陈康鑫 宁波大学信息科学与工程学院硕士研究生. 主要研究方向为深度学习和计算机视觉.
E-mail: kxchenxy@outlook.com
(**CHEN Kang-Xin** Master student at the Faculty of Electrical Engineering and Computer Science, Ningbo University. His research interest covers deep learning and computer vision.)



赵杰煜 宁波大学信息科学与工程学院教授. 1985 年和 1988 年获浙江大学学士和硕士学位、1995 年伦敦大学博士学位. 主要研究方向为深度学习和计算机视觉. 本文通信作者.
E-mail: zhao_jieyu@nbu.edu.cn
(**ZHAO Jie-Yu** Professor at the Faculty of Electrical Engineering and Computer Science, Ningbo University. He received his bachelor and master degrees from Zhejiang University in 1985 and 1988, and his Ph.D. degree from Royal Holloway University of London in 1995. His research interest covers deep learning and computer vision. Corresponding author of this paper.)

陈 豪 宁波大学信息科学与工程学院博士研究生. 主要研究方向为三维重建, 模式识别和机器学习.
E-mail: 1901100014@nbu.edu.cn
(**CHEN Hao** Ph.D. candidate at the Faculty of Electrical Engineering and Computer Science, Ningbo University. His research interest covers 3D reconstruction, pattern recognition, and machine learning.)