

Relation-Shape Convolutional Neural Network for point cloud analysis

原文链接：<https://www.jiqizhixin.com/articles/2019-04-29-3>
<https://blog.csdn.net/inshallah/article/details/99446967>

一、背景

近年来，3D 点云分析在自动驾驶和机器人等领域有着诸多的应用，因此得到了各界广泛的关注。主流的点云分析任务都需要对点云的 3D 形状进行高级别的理解，比如点云识别。但从不规则分布的 3D 点中很难推理学习其隐含的 3D 形状。随着以卷积神经网络（CNN）为代表的深度学习方法的兴起，大量的研究工作致力于将 CNN 在图像分析上的巨大成功复制到点云处理领域。然而由于点云数据的不规则性，经典的图像网格卷积难以适用。

Introduction *point cloud*

Advantages

- ✓ raw sensor data, e.g., Lidar
- ✓ simple representation: $N * (x, y, z, color, normal...)$
- ✓ better 3D shape capturing

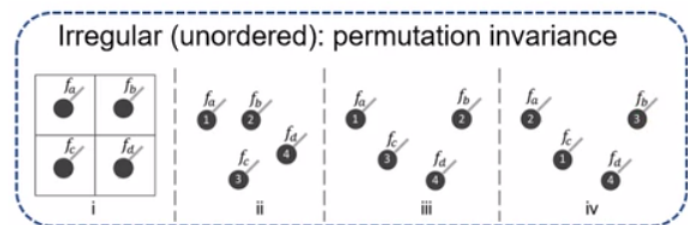
Why emerging?

- ✓ autonomous driving
- ✓ AR & VR
- ✓ robot manipulation
- ✓ Geomatics
- ✓ 3D face & medical
- ✓ AI-assisted shape design in 3D game and animation, etc.
- ✓ open problem, flexible

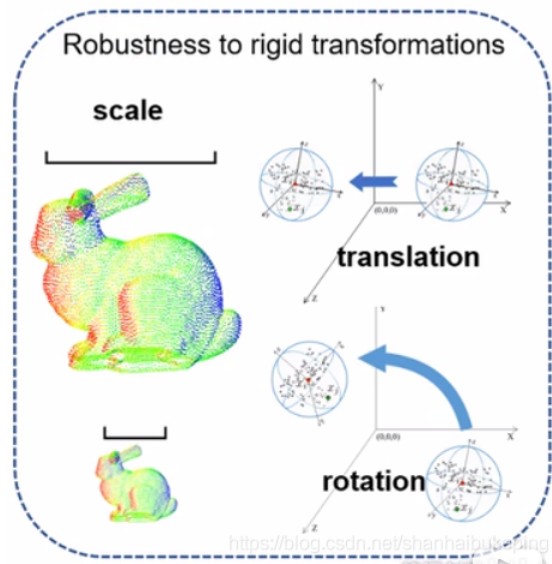


<https://blog.csdn.net/inshallah/article/details/99446967>

一般来说，在点云上进行卷积学习主要面临三大挑战：
点云由无序的点集构成，因此卷积需要对点的输入顺序具有置换不变性。

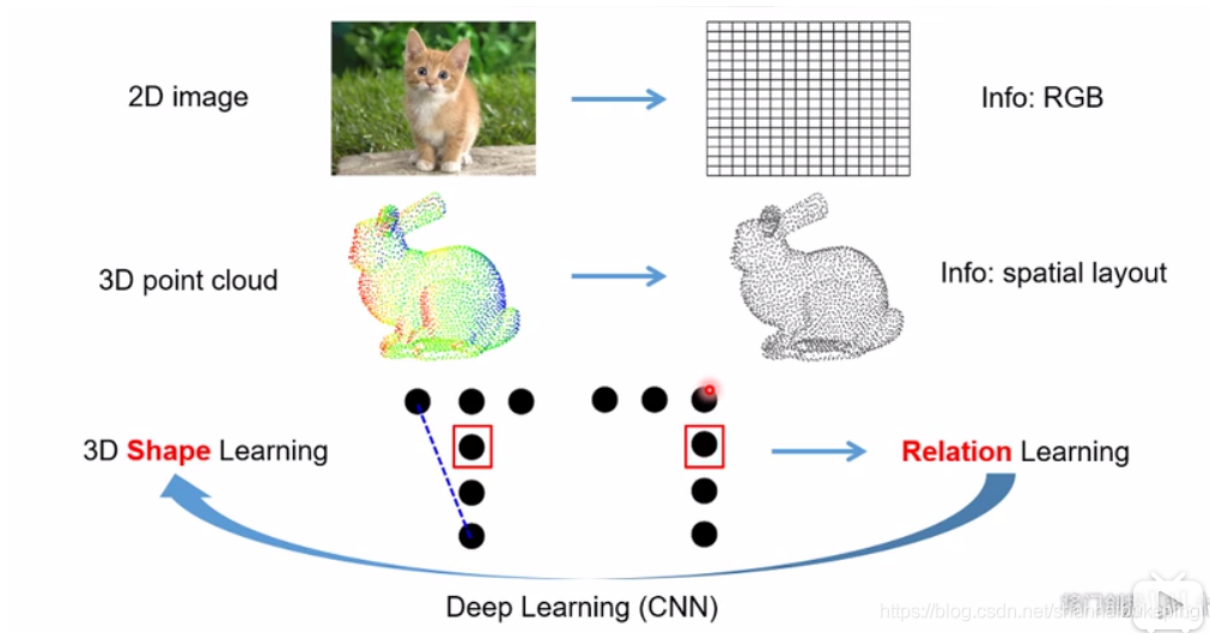


点云分布于 3D 几何空间，因此卷积应当对点云的刚体变换具有鲁棒性，比如平移、旋转等。



点云形成一个隐含的形状，因此卷积得到的表征应当具备有区分力的「形状意识」。

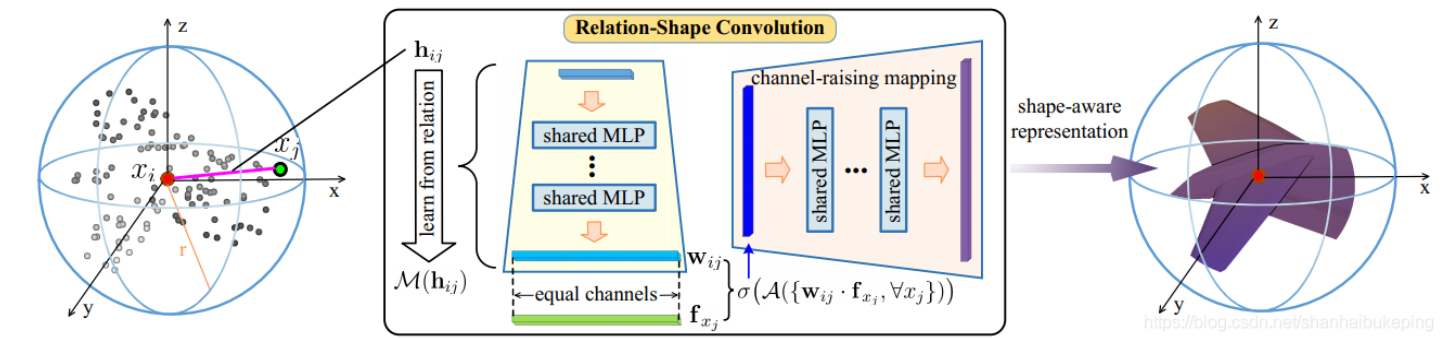
本文提出 Relation-Shape CNN (RS-CNN) 来缓解上述挑战。RS-CNN 的核心是从几何关系中推理学习 3D 形状，因为在本文看来，3D 点之间的几何关系能够有表现力的编码其隐含的形状信息。所提出的 RS-CNN 在点云分类、部件分割和法向估计三个任务上均达到了最佳水平。并且得益于对几何关系的建模，RS-CNN 能够自然地实现置换不变性以及刚体变换鲁棒性。为了验证 RS-CNN 的几何形状推理能力，本文还在 3D 点云的 2D 投影空间中做了测试，实验表明，RS-CNN 依然具备优秀的形状识别性能。



对于2D图片，如果去掉RGB，什么都看不到，所以它的关键信息是RGB；
对于3D点云，如果去掉颜色或者坐标，还是能看出来形状，所以它的关键信息是空间分布。
问题：怎么从空间分布捕捉形状或结构。
每个点之间都有关系，如果算每个点之间的关系就太复杂了。想出一个简单的方法：选取一个中心点，找中心点与其他点的关系，学习这个关系，然后聚合。

二、Relation-Shape CNN (RS-CNN)

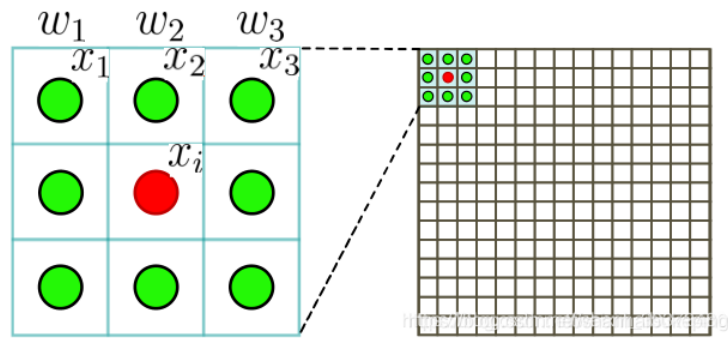
2.1模型表达



RS-CNN 的基本卷积算子如图 2 所示。为了实现局部卷积归纳学习，将局部点云子集建模为一个球形邻域，该邻域的中心点为 x_i ，其余点 x_j 为 x_i 的邻居点。在该邻域上，构建了一个一般性的卷积操作方法：

$$\mathbf{f}_{P_{\text{sub}}} = \sigma \left(\mathcal{A} \left(\{ \mathcal{T}(\mathbf{f}_{x_j}), \forall x_j \} \right) \right)^{\top}, \quad d_{ij} < r \quad \forall x_j \in \mathcal{N}(x_i), \tag{1}$$

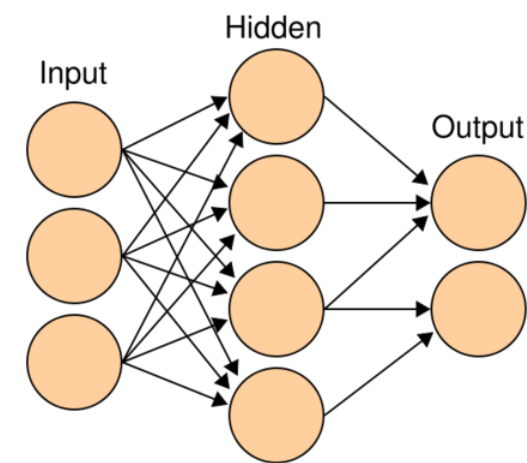
其中 \mathbf{f} 是特征向量， d_{ij} 是点 x_i 和 x_j 的 3D 欧式距离， r 是球半径。
首先使用转换函数 \mathcal{T} 对邻域中每个点进行特征变换，然后使用聚集函数 \mathcal{A} 聚集变换后的特征，使得所有的特征聚集成一个点的特征。 σ 是激活函数。
我们可以对比一下传统图像上的 2DCNN：



假设卷积核是 3X3 的，转换函数 \mathcal{T} 相当于将卷积核相应位置的值乘在了图像的对应位置，比如说卷积核左上角乘以图像对应的左上角，然后聚集函数 \mathcal{A} 相当于相加求和的操作。因此在点云上也是一样，只不过乘上去的这个值（卷积核的值）不是随机初始化的，而是通过 i 和 j 之间的关系算出来的：

$$\mathcal{T}(\mathbf{f}_{x_j}) = \mathbf{w}_{ij} \cdot \mathbf{f}_{x_j} = \mathcal{M}(\mathbf{h}_{ij}) \cdot \mathbf{f}_{x_j}. \tag{2}$$

其中映射函数 \mathcal{M} 的目标是从几何先验中学习一个高维的、有表现力的关系表达，以编码 3D 点集的空间布局，这里使用共享的**多层感知器（MLP）**实现映射函数 \mathcal{M} 。
【MLP 是一种前馈人工神经网络模型，最典型的 MLP 包括包括三层：输入层、隐层和输出层，MLP 神经网络不同层之间是全连接的（全连接的意思就是：上一层的任何一个神经元与下一层的所有神经元都有连接）】



h_{ij} 是点云中 x_i 与 x_j 的关系，这个关系可以人为地先验给出，可以是 xyz 的距离，也可以是特征维度上的距离等等。于是，模型表达式可写成：

$$\mathbf{f}_{P_{\text{sub}}} = \sigma \left(\mathcal{A} \left(\{ \mathcal{M}(\mathbf{h}_{ij}) \cdot \mathbf{f}_{x_j}, \forall x_j \} \right) \right). \tag{3}$$

该卷积方法聚集了点 x_i 和所有邻居点 x_j 之间的几何关系表达，因此可以对 3D 点的空间分布进行显式的推理，进而有区分力的反映其隐含的 3D 形状。其中几何先验 h_{ij} 可以灵活设置，因为映射函数 M 能将 h_{ij} 映射为高维的关系向量，以实现与特征 f_{x_j} 进行通道对齐。在经典图像 CNN 中，随着图像分辨率的降低，特征通道数会逐渐增加以提升表达能力。基于此，我们在 f_{Sub} 上增加共享的 MLP 以实现通道提升映射。

图 4 展示了 RS-CNN 应用于点云分类和点云分割的网络框架。将所提出的几何关系卷积方法搭配均匀的点云下采样算法（下采样是通过构造一个三维栅格，然后在每个栅格内用所有点的重心近似显示其他点，这样栅格内所有点就用一个重心点来表示，大大的减少了数据量，特别是在配准，曲面重建等工作之前作为预处理，可以很好的提高程序的运行速度。），可以很容易实现类似图像 CNN 的多层神经网络，相关操作简单高效。

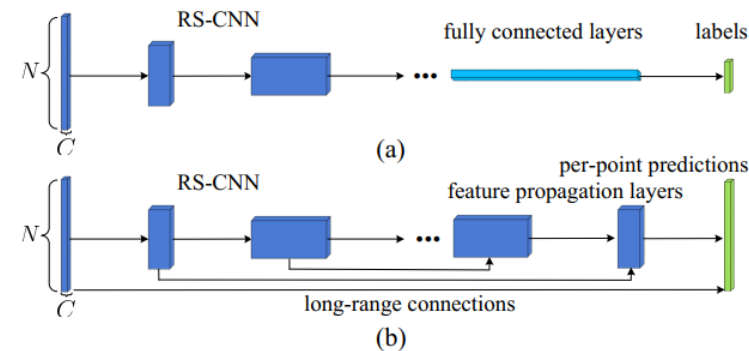
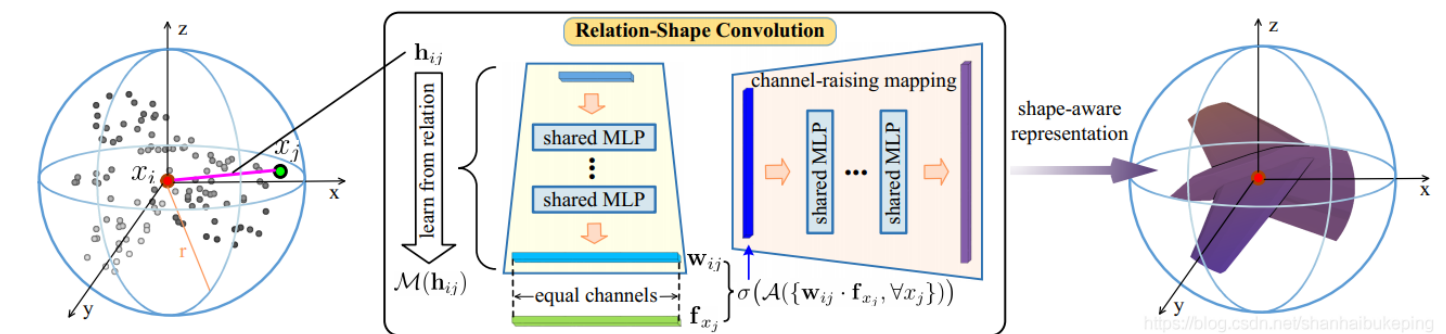


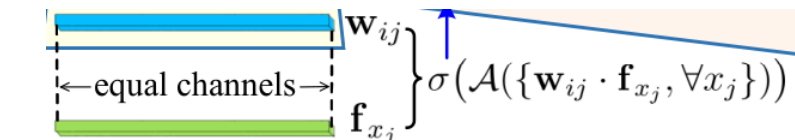
Figure 4. The architectures of RS-CNN applied in the classification (a) and segmentation (b) of point cloud. N is the number of points and C is the channel number. <https://blog.csdn.net/shanhaibukeping>

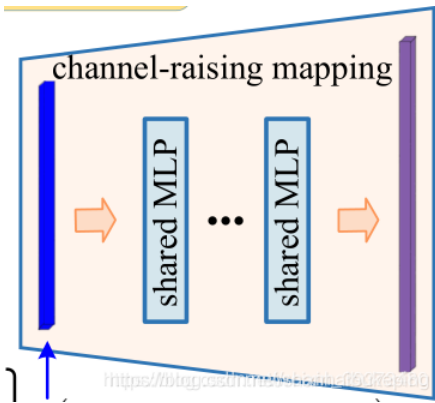
2.2模型结构



我们可以看到模型实际上包含了三个操作：

- 1.先随机采样，对于每一个采样点建立一个邻域，并在里面挑选足够的邻居点，即上图的最左边的球就是由采样点得到的邻域集合。具体的采样细节同pointnet++，使用了最远点采样（FPS，该方法比随机采样的优势在于它可以尽可能的覆盖空间中的所有点。）
假设一共有 n 个点,整个点集为 $N = \{f_1, f_2, \dots, f_n\}$, 目标是选取 n_1 个起始点做为下一步的中心点:
 - 随机选取一个点 f_i 为起始点，并写入起始点集 $B = \{f_i\}$;
 - 选取剩余 $n-1$ 个点计算和 f_i 点的距离，选择最远点 f_j 写入起始点集 $B = \{f_i, f_j\}$;
 - 选取剩余 $n-2$ 个点计算和点集 B 中每个点的距离,将最短的那个距离作为该点到点集的距离，这样得到 $n-2$ 个点到点集的距离，选取最远的那个点写入起始点 $B = \{f_i, f_j, f_k\}$,同时剩下 $n-3$ 个点,重复上面步骤直到选取 n_1 个起始点为止.
- 2.第二个操作即计算邻域中邻居点与中心点之间的关系，即上图左侧那条紫色的线，这个关系首先是low level的先验关系，即两个点之间的欧式距离、特征维度的距离；再通过这样的先验关系带入神经网络中算出高维high level的关系，输出维度 w_{ij} 和 f_{x_j} 维度相同，因此可以直接相乘作为转换函数 T 。然后通过聚合函数 A 将所有邻居点的特征聚合。





三、实验结果

为了验证 RS-CNN 的有效性，在主流的点云分析任务上进行了测试，包括点云分类、部件分割和法向预测。ModelNet40 上的分类结果如表 1 所示，在仅使用 3D 坐标 xyz 和 1k 个稀疏点作为输入的情况下，RS-CNN 仍然实现了最佳分类效果。

Table 1. Shape classification results (%) on ModelNet40 bench- mark (nor: normal, “-”: unknown).

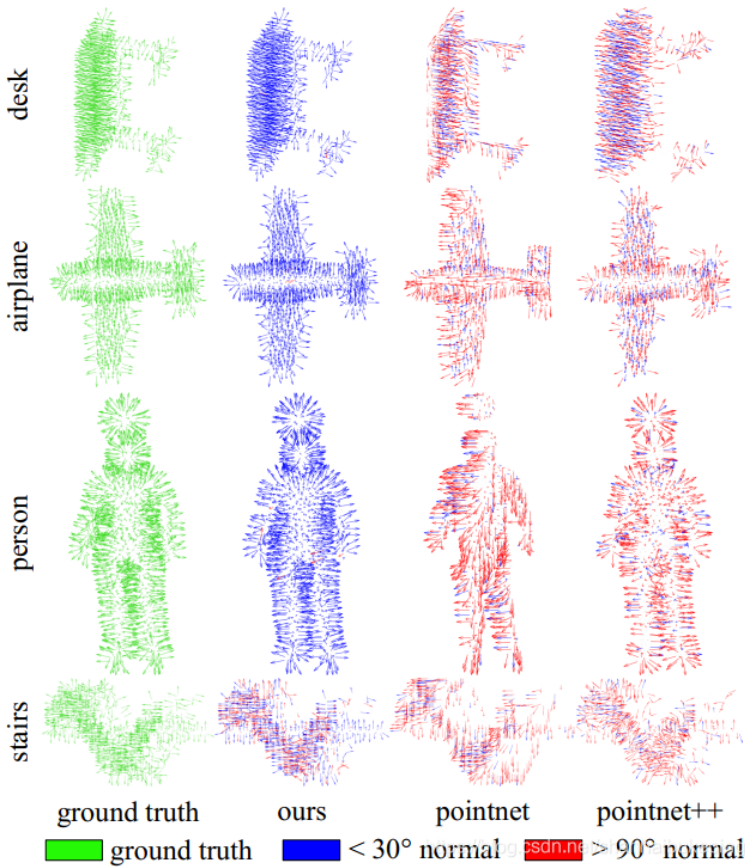
method	input	#points	acc.
Pointwise-CNN [11]	xyz	1k	86.1
Deep Sets [51]	xyz	1k	87.1
ECC [33]	xyz	1k	87.4
PointNet [26]	xyz	1k	89.2
SCN [47]	xyz	1k	90.0
Flex-Conv [4]	xyz	1k	90.2
Kd-Net(depth=10) [18]	xyz	1k	90.6
PointNet++ [28]	xyz	1k	90.7
KCNet [32]	xyz	1k	91.0
MRTNet [3]	xyz	1k	91.2
Spec-GCN [40]	xyz	1k	91.5
PointCNN [23]	xyz	1k	91.7
DGCNN [44]	xyz	1k	92.2
PCNN [1]	xyz	1k	92.3
Ours	xyz	1k	93.6
SO-Net [21]	xyz	2k	90.9
Kd-Net(depth=15) [18]	xyz	32k	91.8
O-CNN [41]	xyz, nor	-	90.6
Spec-GCN [40]	xyz, nor	1k	91.8
PointNet++ [28]	xyz, nor	5k	91.9
SpiderCNN [48]	xyz, nor	5k	92.4
SO-Net [21]	xyz, nor	5k	93.4

分割：

Table 2. Shape part segmentation results (%) on ShapeNet part benchmark (nor: normal, “-”: unknown).

method	input	class mIoU	instance mIoU	air plane	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor bike	mug	pistol	rocket	skate board	table
Kd-Net [16]	4k	77.4	82.3	80.1	74.6	74.3	70.3	88.6	73.5	90.2	87.2	81.0	94.9	57.4	86.7	78.1	51.8	69.9	80.3
PointNet [24]	2k	80.4	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
RS-Net [11]	-	81.4	84.9	82.7	86.4	84.1	78.2	90.4	69.3	91.4	87.0	83.5	95.4	66.0	92.6	81.8	56.1	75.8	82.2
SCN [44]	1k	81.8	84.6	83.8	80.8	83.5	79.3	90.5	69.8	91.7	86.5	82.9	96.0	69.2	93.8	82.5	62.9	74.4	80.8
PCNN [1]	2k	81.8	85.1	82.4	80.1	85.5	79.5	90.8	73.2	91.3	86.0	85.0	95.7	73.2	94.8	83.3	51.0	75.0	81.8
SPLATNet [34]	-	82.0	84.6	81.9	83.9	88.6	79.5	90.1	73.5	91.3	84.7	84.5	96.3	69.7	95.0	81.7	59.2	70.4	81.3
KCNet [30]	2k	82.2	84.7	82.8	81.5	86.4	77.6	90.3	76.8	91.0	87.2	84.5	95.5	69.2	94.4	81.6	60.1	75.2	81.3
DGCNN [41]	2k	82.3	85.1	84.2	83.7	84.4	77.1	90.9	78.5	91.5	87.3	82.9	96.0	67.8	93.3	82.6	59.7	75.5	82.0
Ours	2k	84.0	86.2	83.5	84.8	88.8	79.6	91.2	81.1	91.6	88.4	86.0	96.0	73.7	94.1	83.4	60.5	77.7	83.6
PointNet++ [26]	2k,nor	81.9	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
SyncCNN [47]	mesh	82.0	84.7	81.6	81.7	81.9	75.2	90.2	74.9	93.0	86.1	84.7	95.6	66.7	92.7	81.6	60.6	82.9	82.1
SO-Net [19]	1k,nor	80.8	84.6	81.9	83.5	84.8	78.1	90.8	72.2	90.1	83.6	82.3	95.2	69.3	94.2	80.0	51.6	72.1	82.6
SpiderCNN [45]	2k,nor	82.4	85.3	83.5	81.0	87.2	77.5	90.7	76.8	91.1	87.3	83.3	95.8	70.2	93.5	82.7	59.7	75.8	82.8

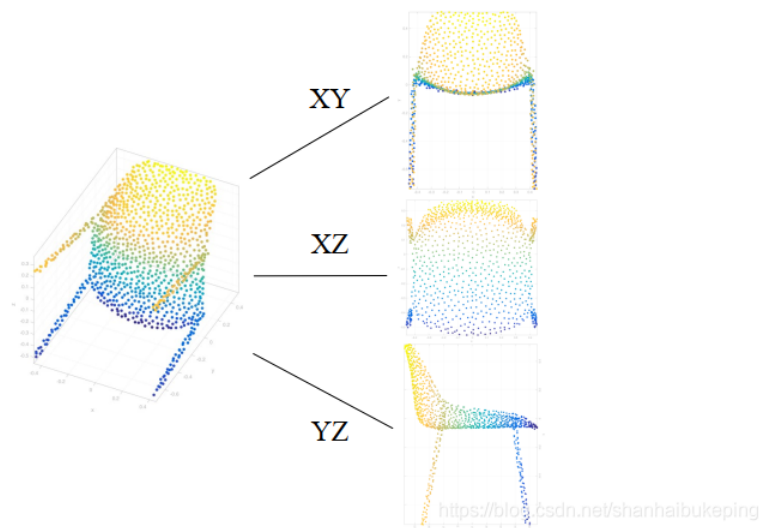
ModelNet40 上的法向预测结果如图 7 所示。与 PointNet 以及 PointNet++相比，RS-CNN 可以取得更加准确的法向预测结果。尽管如此，RS-CNN 仍然难以有效推理棘手的形状，比如旋转楼梯以及错综复杂的植物。



公式 (3) 中的几何先验 可以进行灵活地定义，在 ModelNet40 上测试了五个比较直观的例子，结果如表 6 所示。可以看到，仅仅使用 3D 欧式距离作为低维几何关系 (model A)，RS-CNN 依然能够取得 92.5% 的精度，这很让人印象深刻。同时使用欧氏距离、特征距离与点本身的特征，效果会最好：93.6%。

model	low-level relation h	channels	acc.
A	(3D-Ed)	1	92.5
B	(3D-Ed, $x_i - x_j$)	4	93.0
C	(3D-Ed, $x_i - x_j, x_i, x_j$)	10	93.6
D	(3D-cosd, x_i^{nor}, x_j^{nor})	7	92.8
E	(2D-Ed, $x'_i - x'_j, x'_i, x'_j$)	10	≈ 92.2

此外，为了测试 RS-CNN 的几何形状推理能力，我们强制置零某一维的坐标值，即将 3D 点云投影到 2D 空间 (model E，图 9)，得到的分类精度均接近 92.2。这证明了 RS-CNN 不仅可以从 3D 点云中学习 3D 形状，还能从 2D 投影空间中推理 3D 形状。



为了验证所提出的几何关系卷积的鲁棒性，设置几何先验 h_{ij} 为 3D 欧式距离，然后在 ModelNet40 上进行鲁棒性测试，结果如表 7 所示。虽然几何关系 h_{ij} 能够做到旋转不变，但网络初始输入的特征 xyz 仍然会受到旋转的影响。针对这一问题，我们引入法向将每一个局部点集旋转到的法向和采样点确定的局部坐标系中，实现了旋转不变。但该旋转会给形状识别带来困难，因此分类精度会有所下降。

method	acc.	perm.	+0.2	-0.2	90°	180°
PointNet [26]	88.7	88.7	70.8	70.6	42.5	38.6
PointNet++ [28]	88.2 [†]	88.2	88.2	88.2	47.9	39.7
Ours	90.3 [†]	90.3	90.3	90.3	90.3	90.3

[†] The accuracy drops a lot mainly because the forcible normalization of each local point subset could bring difficulty for shape recognition.

论文：Relation-Shape Convolutional Neural Network for Point Cloud Analysis
论文地址：<https://arxiv.org/abs/1904.07601>
项目主页：<https://yochengliu.github.io/Relation-Shape-CNN/>