苹果的无人车激光雷达处理方案

MAZE · 1 个月前

用Python打造无人驾驶车-激光雷达数据(2)

这篇文章刚开始写的时候苹果的论文还未发布，我基于当时新出的PointNet修改了一个算法。然而等我写完测试完准备发布的时候搜了一下最新的论文，发现苹果的论文提供了一种新的算法。读了一下，发现苹果的方案更加优秀，于是为了文章质量只能推翻重写，并增加苹果论文内容。

上一篇文章的一点思考

1.现在的地面数据处理方法有什么弊端？

A:当无人车处于行进状态时，车身会发生颠簸和倾斜，然而激光雷达只能提供相对位置，所以激光雷达点云会一直在坐标系中倾斜移动，因此直接对底部进行一刀切并不精确。

2.对地面数据有没有更好的处理方法？

A:可以先对地面数据进行处理，识别出道路，对相对坐标进行校准，然后再根据道路数据计算出地面平面进行过滤，可以参考论文

[1703.03613] Fast LIDAR-based Road Detection Using Fully Convolutional Neural Networks

最佳的参数与激光雷达参数（线数，数量）与物体距雷达的距离成线性关系，越远点与点之间的距离越稀疏。

4.如何过滤无用的障碍物？

利用目标障碍物的特征进行机器学习的训练。

5.激光雷达产生点云除了有xyz数据，还有强度和线圈数据，有什么用处吗？

可以用于丰富机器学习的特征数据。

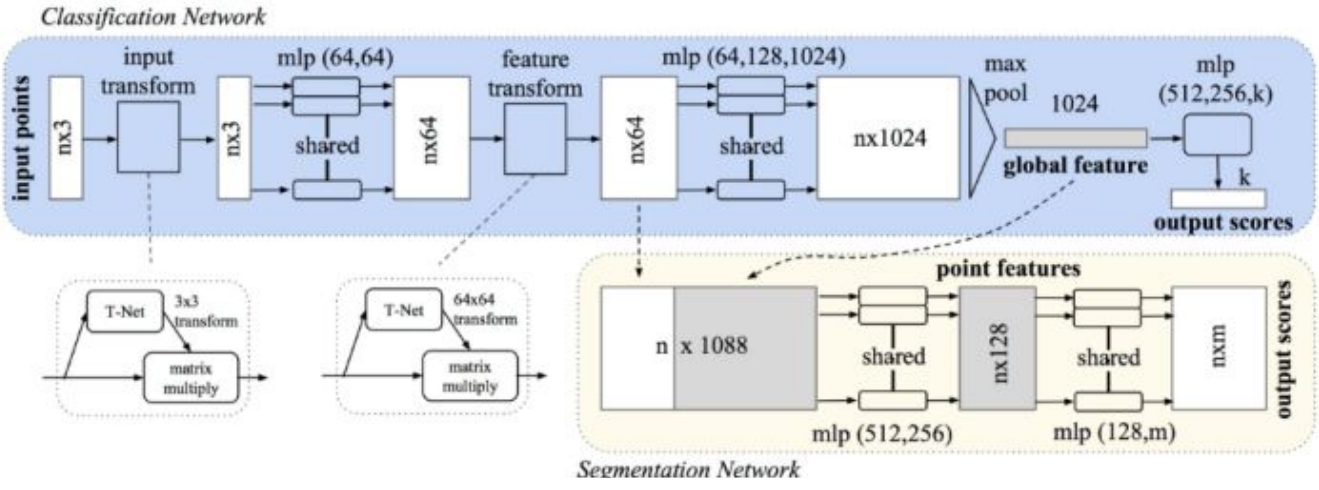
前言：

在上一篇文章中，我们已经能够成功的聚类出物体。然而，为了能够让无人驾驶做出更准确的预测和判断，我们还需要能够识别出每一种物体的类型。随着机器学习的火热，识别图像数据中的物体的算法已经非常成熟。所以，激光雷达的数据通常会处理成俯视图的图片，然后利用识别图像的成熟算法进行处理，这样通常能够花很少的精力得到一个非常不错的结果。但是，点云毕竟是一个三维的数据（激光雷达是五维），压缩成二维的图片之后，难免会造成数据的损失，同时，转换过程也会造成一定的性能影响。因此本文将讨论两种直接处理点云数据的算法。

PointNet

PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. [ref][github]

简单介绍一下，主要原理是尊重输入点中的置换不变性，使用max pooling。



应用到激光雷达点云

当你看完PointNet的论文你会发现他的目标是识别物体轮廓，并不针对激光雷达数据，因此需要对原始算法作出一点改进，来提高识别准确度。

PointNet主要的瓶颈在于两点：

1.只有xyz数据

2.可接受的点数量固定

我采用的方法是将PointNet原本的三维数据扩展到5维，使其可以容纳全部的激光雷达数据，并预先手动处理聚类的点云，通过随机抽样增加或减少已有点，使每个点云物体点是一个固定的数量，经过测试在十米的检测范围128点～256点能获得最佳结果。这样最终结果大约能获得93%+的准确度。

（就在我写完我的处理方案的时候，发现了苹果的论文，提供了一种更为合适的算法，因此我原先的方案略过，改介绍苹果的方案。）

VoxelNet

End-to-End Learning for Point Cloud Based 3D Object Detection

"""

以下是对苹果论文的翻译

"""

VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection

Yin Zhou Apple Inc

Oncel Tuzel Apple Inc

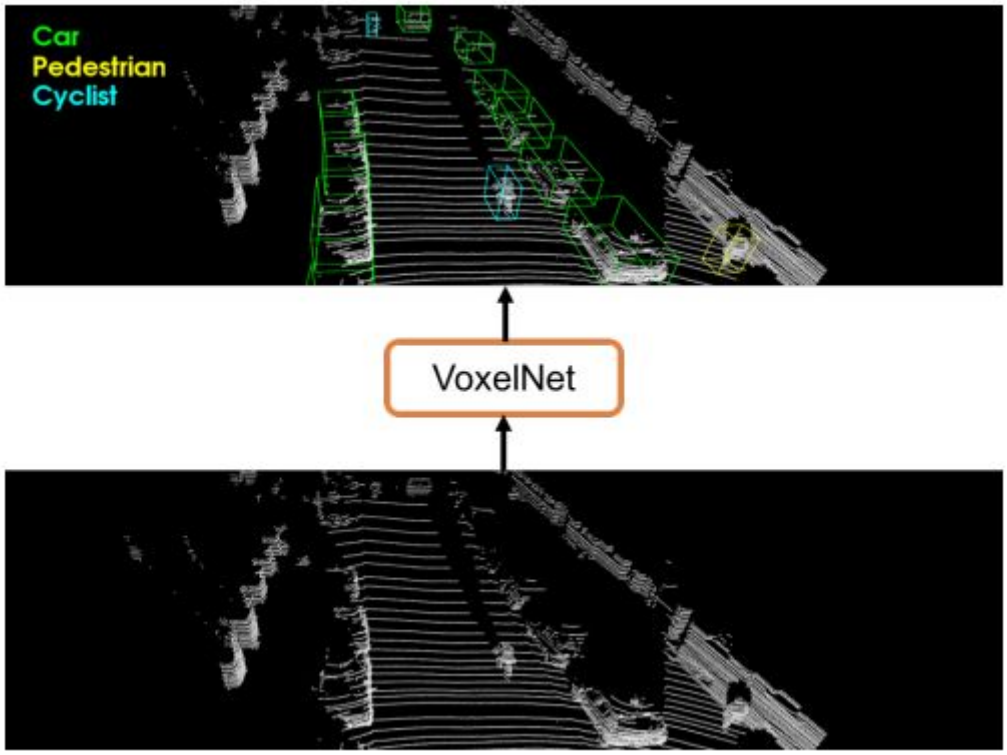


图1. VoxelNet直接在原始点云上运行(不需要特征处理)，并使用单个端到端可训练网络生成3D检测结果。

摘要

精确检测三维点云中的物体是许多应用中的核心问题，如自动导航，家庭机器人，虚拟现实等。为了将一个高度稀疏的雷达点云和region proposal network (RPN)联系起来，大多数现有的努力都集中在手工处理特征表示上，例如转换成鸟瞰图投影。在这项工作中，我们消除了对三维点云进行手动特征处理的需求，并提出了一个通用的3D检测网络VoxelNet，它将特征提取和边界框预测统一到一个单一阶段的端到端可训练深度网络中。 具体而言，VoxelNet将点云划分为等间隔的三维像素，并通过新引入的立体像素特征编码（VFE）层将每个立体像素内的一组点转换为统一的特征表示。 这样，点云被编码为描述性的体积表示，然后连接到RPN以生成检测。 在KITTI汽车检测基准测试中的实验表明，VoxelNet大大超越了先进的基于LiDAR的3D检测方法。 此外，我们的网络学习一个有效的不同几何形状的对象的区别表示，导致在仅基于LiDAR的行人和骑车人的3D检测方面令人鼓舞的结果。

1.介绍

基于点云的三维物体检测是各种现实应用的重要组成部分，如自主导航[11,14]，看家机器人[26]和增强/虚拟现实[27]。 与基于图像的检测相比，LiDAR提供可靠的深度信息，可用于精确定位物体并表征其形状[21,5]。 然而，与图像不同，LiDAR由于诸如3D空间的非均匀采样，传感器的有效范围，遮挡和相对姿态的因素而导致测量获取的点云是稀疏的，并且具有高度可变的密度。为了应对这些挑战，许多方法手动调整特征表示来进行三维物体检测。 几种方法将点云投影到透视图中，并应用基于图像的特征提取技术[28,15,22]。 其他方法将点云栅格化为三维体素网格，并用手

瓶颈，阻止了这些方法有效地利用三维形状信息和检测任务所需的不变量。 识别[20]和检测[13]任务在图像上的重大突破是得益于从手工处理特征升级到机器学习处理。

最近，Qi等人[29] 提出的PointNet[ref]是一个端到端的深度神经网络，可以直接从点云中学习点对点的特征。 这种方法在三维物体识别，三维物体部分分割和点分词语义分割任务方面展示了令人印象深刻的结果。在文献[30]中，引入了改进的PointNet版本，使网络能够学习不同尺度的局部结构。 为了获得满意的结果，这两种方法在所有输入点（一1k点）上训练特征变换器网络。 由于使用LiDARs获得的典型点云包含了大约100k个点，因此如[29,30]中所述的对体系结构进行训练会导致高计算和内存需求。 将3D特征学习网络扩展到更多点数和3D检测任务是我们在本文中讨论的主要挑战。

Region proposal network (RPN) [32]是一种高效的物体检测算法[17,5,31,24]。 然而，这种方法要求数据密集并以张量结构（例如图像，视频）组织，对于典型的LiDAR点云而言不是这种情况。 在本文中，我们弥补了三维检测任务中点集特征学习和RPN之间的差距。

我们提出了一个通用的3D检测框架VoxelNet，它可以从点云中同时学习一个有区别的特征表示，并以端到端的方式预测精确的三维边界框，如图2所示。我们设计一个新的体素特征编码（VFE）层，通过将点式特征与本地聚合特征相结合，实现了体素内的点间交互。 堆叠多个VFE层允许学习复杂的特征来表征局部3D形状信息。 具体而言，VoxelNet将点云划分为等间隔的三维像素，通过堆叠的VFE层对每个体素进行编码，然后三维卷积进一步聚合局部体素特征，将点云转化为高维体积表示。 最后，RPN消耗体积表示并产生检测结果。 这种高效的算法可以同时从体素网格上的稀疏点结构和高效的并行处理中受益。

我们评估VoxelNet的鸟瞰图检测和完整的3D检测任务，由KITTI基准[11]提供。 实验结果表明，VoxelNet大大超越了先进的基于LiDAR的三维检测方法。 我们还证明，VoxelNet在LiDAR点云中检测行人和骑车者方面取得了令人鼓舞的成果。

1.1. 相关工作

3D传感器技术的快速发展促使研究人员开发高效的表示来检测和定位点云中的物体。 一些早期的特征表示方法是[39,8,7,19,40,33,6,25,1,34,2]。 当丰富和详细的三维形状信息可用时，这些手工处理的特征可以产生令人满意的结果。 然而，他们无法适应更复杂的形状和场景，并无法从数据中学习所需的不变性，导致无法控制的情景（如自主导航）只能取得有限的成功。

由于图像提供了详细的纹理信息，所以许多算法从2D图像中获取3D边界框[4，3，42，43，44，36]。 然而，基于图像的3D检测方法的准确度受到深度估算的准确度的限制。

几种基于LIDAR的3D对象检测技术利用了立体像素网格的表示方法。 [41，9]用6个统计量对每个非空像素进行编码，这些统计量是从包含在体素中的所有点导出的。 [37]融合了多个本地统计数

通过计算鸟瞰图中的多通道特征地图和正视图中的圆柱坐标来引入LiDAR点云的多视图表示。 其他几项研究将点云投影到透视图上，然后使用基于图像的特征编码方案[28,15,22]。

也有几种多模式融合方法结合了图像和LiDAR来提高检测的准确性[10,16,5]。 与只有LiDAR的3D检测相比，这些方法提供了改进的性能，特别是对于小物体（行人，骑自行车的人）或物体远的时候，因为相机能够比LiDAR提供更多数量级的测量数据。 然而，这些方案需要一个与LiDAR同步并校准的附加摄像头，限制了它们的使用场景，并使解决方案对传感器故障模式更为敏感。 在这项工作中，我们专注于LiDAR检测。

1.2. 贡献

- 我们提出了一种基于点云的三维检测VoxelNet的新型端到端可训练深度架构，可直接在稀疏3D点上运行，并避免手动特征工程引入的信息瓶颈。
- 我们提出了一种有效的方法来实现VoxelNet，它既可以从体素网格上的稀疏点结构和高效的并行处理中受益。
- 我们对KITTI基准进行实验，并展现VoxelNet在基于LiDAR的汽车，行人和骑自行车者检测基准方面产生了最新的成果。

2. VoxelNet

在本节中，我们将解释VoxelNet的体系结构，用于训练的损失函数，以及用于实现网络的高效算法。

2.1. VoxelNet架构

生成3D检测。

所提出的VoxelNet由三个功能块组成：（1）特征学习网络，（2）卷积中间层，（3）区域提议网络[32]，如图2所示。我们在下面详细介绍VoxelNet部分。

2.1.1特征学习网络

立体像素分区 给定一个点云，我们将三维空间细分为等距体素，如图2所示。假设点云分别包含沿Z，Y，X轴的范围D，H，W的三维空间。 我们相应地定义大小为vD，vH和vW的每个体素。 得到的三维像素网格的大小为 $D' = D/v_D$ ， $H' = H/v_H$ ， $W' = W/v_W$ 。 在这里，为了简单起见，我们假设D，H，W是vD，vH，vW的倍数。

分组 我们根据它们所在的体素对这些点进行分组。由于距离，遮挡，物体的相对姿态以及非均匀采样等因素，LiDAR点云是稀疏的，在整个空间中点密度变化很大。 因此，在分组之后，体素将包含可变数目的点。 图2显示了一个例子，其中Voxel-1具有比Voxel-2和Voxel-4更多的点，而Voxel-3没有任何点。

随机抽样 通常，高分辨率LiDAR点云由~100k点组成。 直接处理所有点不仅会增加计算平台上的内存/效率负担，而且整个空间的高度可变的点密度可能会对检测造成偏见。 为此，我们从包含多于T个点的那些体素中随机地抽样一个固定的数目T。 这个抽样策略有两个目的，（1）计算节省（详见2.3节）；（2）减少体素间点的不平衡，减少抽样偏差，增加训练变量。

堆叠体素特征编码 关键的创新点是VFE层链。 为简单起见，图2说明了一个体素的分层特征编码过程。为了不失一般性，我们在下一段使用VFE Layer-1来描述相关细节。 图3显示了VFE Layer-1的架构。

图3.立体像素特征编码层

$V = \{P_i = [x_i, y_i, z_i, r_i]^T \in R^4\}_{i=1...t}$ 表示一个包含 $t \leq T$ LiDAR 点的非空立体像素， P_i 包含了第i个点的XYZ坐标， r_i 是收到的反射率。我们首先计算局部均值作为V中所有点的质心，表示为 (v_x, v_y, v_z) 。然后我们使用相对偏移量w.r.t和质心来增量每个点 P_i 并获得输入特征集 $V_{in} = \{\hat{P}_i = [x_i, y_i, z_i, x_i - v_x, y_i - v_y, z_i - v_z]^T \in R^7\}_{i=1...t}$ 。下一步，每个 $\hat{P}_i$ 通过一个全链接网络（FCN）转换成一个特征空间，在特征空间中我们可以从点特征 $f_i \in R^m$ 中聚合信息来编码包含在立体像素中的表面形状。FCN由线性层，批量归一化（BN）层和整流线性单元（ReLU）层组成。在获得了逐点特征表示之后，我们在所有与V有关的 f_i 上使用基于元素的最大池化（MaxPooling）来获得V的局部聚合特征 $\tilde{f} \in R^m$ 。最终，我们使用 $\tilde{f}$ 增量每一个 f_i 来形成点范围链接的特征 $f_i^{out} = [f_i^T, \tilde{f}^T]^T \in R^{2m}$ 。因此我们获得了输出特征集 $V_{out} = \{f_i^out\}_{i...t}$ 。所有非空体积像素都以相同的方式编码，并且它们在FCN中共享相同的一组参数。

我们使用VFE-i (c_{in}, c_{out}) 来表示将输入特征维度 c_{in} 转化为输出特征维度 c_{out} 的第i VFE层。线性层学习一个大小为 $c_{in} \times (c_{out}/2)$ 的矩阵，并且逐点连接得到维度 c_{out} 的输出。

由于输出特征结合了逐点特征和局部聚合特征，所以堆叠VFE层对体素内的点交互进行编码，并使最终特征表示学习描述性的形状信息。 通过FCN将VFE-n的输出转换为 R^C 并应用元素级的Maxpool（其中C是体素特征的尺寸）获得体素级的特征，如图2所示。

稀疏张量表示 通过仅处理非空体素，我们获得体素特征的列表，每个体素特征与特定的非空像素的空间坐标唯一关联。 得到的体素特征列表可以表示为一个稀疏的4D张量，大小为 $C' \times D' \times H' \times W'$ ，如图2所示。虽然点云包含了约100k个点，但是超过90%的体素通常是空的。 将非空体素特征表示为稀疏张量，大大减少了反向传播时的内存使用和计算代价，是实现高效实现的关键一步。

2.1.2卷积中间层

我们使用ConvMD $(c_{in}, c_{out}, k, s, p)$ 来表示一个M维卷积算子，其中 c_{in} 和 c_{out} 是输入和输出通道的数量，k，s和p是对应于内核大小的M维向量， 步幅和填充大小。 当M维的大小相同时，我们使用一个标量来表示大小，例如 k代表 $k = (k, k, k)$ 。

每个卷积中间层依次应用3D卷积，BN层和ReLU层。 卷积中间层在逐渐扩大的感受域内聚合体素的特征，为形状描述增加更多内容。 卷积中间层滤波器的详细尺寸将在第3节中解释。

2.1.3局部提案网络(Region Proposal Network)

图4.区域提案网络架构

最近，局部提案网络[32]已经成为表现最好的目标检测框架的重要组成部分[38,5,23]。在这项工作中，我们对[32]中提出的RPN体系结构进行了几个关键的改进，并将其与特征学习网络和卷积中间层结合起来，形成一个端到端的可训练流水线。

RPN的输入是由卷积中间层提供的特征图。该网络的体系结构如图4所示。该网络有三个完全卷积层的块。每个块的第一层通过步长为2的卷积将特征图采样一半，然后是步长1的卷积序列（×q表示应用q个滤波器）。在每个卷积层之后，应用BN和ReLU操作。然后，我们将每个块的输出上采样到一个固定的大小并串联构造高分辨率的特征图。最后，该特征图被映射到期望的学习目标：（1）概率评分图和（2）回归图。

2.2. 损失函数

把 $\{a_i^{pos}\}_{i=1...N_{pos}}$ 作为 N_{pos} 正锚点(positive anchor)集合， $\{a_i^{neg}\}_{i=1...N_{neg}}$ 作为 N_{neg} 消极锚点(negative anchor)集合。我们用 $(x_c^g, y_c^g, z_c^g, l^g, w^g, h^g, \theta^g)$ 参数化三维参考标准边界盒(3D ground truth box)，其中 x_c^g, y_c^g, z_c^g 代表中心位置， l^g, w^g, h^g 是边界盒的长宽高， θ^g 是围绕Z轴的偏航旋转角度。为了从参数化为 $(x_c^a, y_c^a, z_c^a, l^a, w^a, h^a, \theta^a)$ 的一个匹配正锚点中检索参考标准边界盒，我们定义了包含7个与中心位置 $\Delta x, \Delta y, \Delta z,$ ，三个维度 $\Delta l, \Delta w, \Delta h$ 和旋转角度 $\Delta\theta$ 的相对应的回归目标的残差向量 $u^* \in R^7$ ，可以用以下公式计算：

$$\Delta x = \frac{x_c^g - x_c^a}{d^a} \text{ ，} \Delta y = \frac{y_c^g - y_c^a}{d^a} \text{ ，} \Delta z = \frac{z_c^g - z_c^a}{dh^a} \text{ ，}$$

$$\Delta l = log(\frac{l^g}{l^a}) \text{ ，} \Delta w = log(\frac{w^g}{w^a}) \text{ ，} \Delta h = log(\frac{h^g}{h^a}) \text{ ，(1)}$$

$$\Delta\theta = \theta^g - \theta^a$$

其中 $d^a = \sqrt{(l^a)^2 + (w^a)^2}$ 是锚箱底部的对角线。与[32, 38, 22, 21,4, 3, 5]不同之处在于，这里我们的目的是直接预测定向的3D边界盒，并使用对角线 d^a 均匀归一化 Δx 和 Δy 。我们定义损失函数如下：

$$L = \alpha \frac{1}{\sum} L_{\cdot\cdot}(p^{pos}, 1) + \beta \frac{1}{\sum} L_{\cdot\cdot}(p^{neg}, 0) + \frac{1}{\sum} L_{\cdot\cdot}(u_i, u_i^*) \text{ (2)}$$

其中 p_i^{pos} 和 p_j^{neg} 分别表示正锚点 a_i^{pos} 和负锚点 a_j^{neg} 的softmax输出，而 $u_i \in R^7$ 和 $u_i^* \in R^7$ 则是正锚点 a_i^{pos} 的回归输出(regression output)和参考标准值(ground truth)。公式前两项为 $\{a_i^{pos}\}_{i=1...N_{pos}}$ 和 $\{a_i^{neg}\}_{i=1...N_{neg}}$ 的归一化分类损失， L_{cls} 代表二进制交叉熵损失， α, β 是平衡相对重要性的正常数。公式最后一项是回归损失，我们在此使用SmoothL1 方法 [12, 32]。

2.3. 高效实现

GPU对处理稠密张量结构进行了优化。而直接使用点云的问题在于，点在空间上是稀疏分布的，每个体素有不同数量的点。 我们设计了一种将点云转换为稠密张量结构的方法，其中堆叠的VFE操作可以在点和体素上并行处理。

图5.高效实现的例子

图5是这个方案的总结。我们初始化K×T×7维张量结构来存储体素输入特征缓冲区，其中K是非空体素的最大数量，T是每个体素的最大点数，以及7是每个点的输入编码维度。在处理之前，这些点是随机的。对于点云中的每个点，我们检查相应的体素是否已经存在。该查找操作在O(1)中使用散列表（其中体素坐标被用作散列键）有效地完成。如果体素已经初始化，如果有少于T个点，我们将点插入体素位置，否则该点将被忽略。如果体素未初始化，我们初始化一个新体素，将其坐标存储在体素坐标缓冲区中，并将该点插入此体素位置。体素输入特征和坐标缓冲区可以通过点列表上的单遍构造，因此其复杂度为O(n)。为了进一步提高内存/计算效率，可以仅存储有限数量的体素（K）并忽略来自具有少量点的体素的点。

在构建体素输入缓冲器之后，堆叠的VFE只涉及可以在GPU上并行计算的点级和体素级密集操作。 请注意，在VFE中的连接操作之后，我们将与空点相对应的特征重置为零，使得它们不影响

3. 训练细节

在本节中，我们将解释VoxelNet的实现细节和训练过程。

3.1 网络细节

我们的实验装置基于KITTI数据集的LiDAR规范[11]。

车辆检测 对于这个任务，我们考虑沿Z，Y，X轴分别在[-3,1]×[-40,40]×[0,70.4]米范围内的点云。投影在图像边界之外的点被删除[5]。我们选择立体像素大小为 $v_D = 0.4, v_H = 0.2, v_W = 0.2$ 米，这使得 $D' = 10, H' = 400, W' = 352$ 。我们设每个非空体素中随机采样的最大数为T=35。我们使用两个VFE层，VFE-1(7,32)和VFE(32,128)。最后FCN映射VFE-2输出到 R^{128} 。因此我们的特征学习网络生成一个形状为128×10×400×352的稀疏张量。为了汇总体素特征，我们依次使用三个卷积中间层Conv3D(128, 64, 3,(2,1,1), (1,1,1)), Conv3D(64, 64, 3, (1,1,1), (0,1,1)), 和Conv3D(64, 64, 3, (2,1,1), (1,1,1))，这产生了尺寸为 $64 \times 2 \times 400 \times 352$ 的4D张量。整形之后，RPN的输入是尺寸为128×400×352的特征图，尺寸对应于3D张量的通道数，高度和宽度。图4显示了这个任务的详细网络架构。 与[5]不同的是，我们只使用一个锚点大小，$l^a = 3.9$，$w^a = 1.6$，$h^a = 1.56$ 米，以 $z_c^a = -1.0$ 米为中心，分别旋转0和90度。我们的锚点匹配标准如下：如果一个锚点与参考标准具有最高的交叉重合(IoU）和或者与参考标准的IoU高于0.6（在鸟瞰图中），则锚点被认为是正的。 如果它与所有的参考标准边界盒之间的IoU小于0.45，则认为锚是负的。 我们把锚定在0.45≤IoU≤0.6的锚点忽略。 在公式2，我们设α= 1.5和β= 1。

行人和骑自行车者的检测 输入范围(我们的经验观察表明，超出这个范围，从行人，骑自行车的人返回的LiDAR变得非常稀疏，因此检测结果将是不可靠的。)分别沿Z，Y，X轴为[-3,1]×[-20,20]×[0,48]米。我们使用与汽车检测相同的体素大小，其使得D = 10，H = 200，W = 240。为了获得更多的LiDAR点以获得更好的形状信息，我们设置T = 45。特征学习网络和卷积中间层与用于汽车检测任务的网络相同。对于RPN，我们对图4中的块1进行一次修改，将第一个2D卷积中的步长从2更改为1来。这将允许更精细的锚定匹配的分辨率，对于检测行人和骑车人来说是必需的。我们使用锚定尺寸 $l^a = 0.8$，$w^a = 0.6$，$h^a = 1.73$ 米，以 $z_c^a = -0.6$ 米为中心，0和90度旋转用于行人检测，使用锚定尺寸 $l^a = 1.76$，$w^a = 0.6$，$h^a = 1.73$ 米，$z_c^a = -0.6$ 米为中心，0和90度旋转用于骑自行车者检测。具体的锚点匹配标准如下：如果与参考标准边界盒具有最高的IoU，或者其与参考标准的IoU高于0.5，则将锚点分配为正的。如果与参考标准边界盒的IoU小于0.35，则锚被认为是负的。对于0.35≤IoU≤0.5的锚点，我们忽略。

在训练期间，我们使用随机梯度下降（SGD），前150个学习阶段的学习率(learning rate)为0.01，最后10个学习阶段的学习率降低到0.001。我们batchsize 使用16点云。

如果训练点云少于4000个，从头开始训练我们的网络将不可避免地会出现过度配合。 为了减轻这个问题，我们介绍三种不同形式的数据增量。 增量的训练数据可以在运行中生成，而不需要存储在磁盘上[20]。

定义一个集合 $M = \{p_i = [x_i, y_i, z_i r_i]^T \in R^4\}_{i=1,...,N}$ 作为由N个点组成的整个点云。我们参数化一个三维边界盒 $b_i = (x_c, y_c, z_c, l, w, h, \theta)$,其中 x_c, y_c, z_c 是中心点， l, w, h 是长宽高， θ 是围绕Z轴的偏移角度。我们定义 $\Omega_i = \{P | x \in [x_c - l/2, x_c + l/2], y \in [y_c - w/2, y_c + w/2], z \in [z_c - h/2, z_c + h/2], P \in M\}$ 作为包含 b_i 内所有点的集合，其中 $P = [x, y, z, r]$ 表示整个集合中的特定LiDAR点。

增量数据的第一种形式是将独立于每个地面真实3D边界框以及边界盒内的那些LiDAR点的扰动独立地应用。具体来说，我们围绕z轴旋转 b_i 和对应的 Ω_i ,旋转角度为一个均匀分布的随机变量 $\Delta\theta \in [-\pi/10, +\pi/10]$ ，然后我们给 b_i 的XYZ分量和 Ω_i 中的每个点增加一个平移 $(\Delta x, \Delta y, \Delta z)$,其中 $\Delta x, \Delta y, \Delta z$ 独立绘制于一个均值为0标准差为1的高斯分布。为了避免物理上的不可能结果，我们对扰动后的任意两个边界盒做碰撞测试，如果碰撞被发现，我们将恢复到原来的状态，由于扰动被应用与每个参考标准边界盒和相关的LiDAR点，因此网络能够从原始数据中获得比实际更多的变化。

第二，我们对所有的参考标准边界盒 b_i 和整个点云M应用全局缩放。具体来说，我们将每个 b_i 的XYZ坐标和三个维度以及M中所有点的XYZ坐标乘以一个均匀分布于[0.95,1.05]的随机变量。 引入全局尺度增量提高了网络的鲁棒性，用于检测具有各种尺寸和距离的对象，如基于图像的分类[35,18]和检测任务[12,17]所示。

最后，我们将全局旋转应用到所有的参考标准边界盒 b_i 和整个点云M上。沿着Z轴和(0,0,0)周围应用旋转。 全局旋转偏移量取决于均匀分布[-π/ 4，+π/ 4]的采样。 通过旋转整个点云，我们可以模拟车辆转弯。

4. 实验

我们在KITTI 3D物体检测基准[11]上评估VoxelNet，其中包含7,481个训练图像/点云和7,518个测试图像/点云，覆盖三类：汽车，行人和骑车者。 对于每个分类，根据三个难度级别来评估检测结果：简单，中等和难度，根据对象大小，遮挡状态和截断级别确定检测结果。 由于测试集的参考标准不可用，并且对测试服务器的访问是有限的，我们使用[4，3，5]中描述的协议进行综合评估，并将训练数据细分为训练集和验证集，这形成了3,712个训练数据样本和3,769个数据样本用于验证。 分割操作避免了来自相同序列的样本被包括在训练集和验证集中[3]。 最后，我们还使用KITTI服务器提供测试结果。

对于Car类，我们将所提出的方法与几种性能最好的算法进行比较，包括基于图像的方法：

MV [5]。 Mono3D [3]，3DOP [4]和MV [5]使用预先训练的模型进行初始化，而我们仅使用KITTI提供的LiDAR数据从头开始训练VoxelNet。

为了分析端到端学习的重要性，我们实现了源自VoxelNet体系结构的强大基线，但是使用手工处理特征而不是所提出的特征学习网络。我们称这个模型为手工处理的基线[hand-crafted baseline (HC-baseline)]。 HC基线使用[5]中描述的以0.1m分辨率计算的鸟瞰特征。与[5]不同，我们将高度通道的数量从4个增加到16个，以获取更详细的形状信息 - 进一步增加高度通道的数量不会导致性能改进。我们使用Conv2D(16, 32, 3, 1, 1), Conv2D(32, 64, 3, 2, 1), Conv2D(64, 128, 3, 1, 1)这些有着相似尺寸的2D卷积层替换VoxelNet的卷积中间层。最后的RPN在VoxelNet和HC-基线中是相同的。 HC基线和VoxelNet中的参数总数非常相似。我们使用第3节中描述的相同的训练程序和数据增强来训练HC基线。

4.1. 在KITTI验证集上评估

指标 我们遵循官方的KITTI评估协议，其中IoU门限为0.7级轿车，0.5级为行人和骑车人。 IoU阈值对于鸟瞰和全3D评估都是相同的。 我们比较方法使用平均精度（AP）度量。

鸟瞰图中的评估 评估结果如表1所示。VoxelNet在所有三个难度级别上始终优于所有竞争方法。相比最先进的[5]，HC基线也取得了令人满意的性能，这表明我们的区域提案网络（RPN）是有效的。 对于鸟瞰视图中的行人和骑车者检测任务，我们将提议的VoxelNet与HC基线进行比较。 对于这些更具挑战性的类别，VoxelNet产生的AP比HC基线高得多，这表明端到端学习对于基于点云的检测是必不可少的。

我们想要指出的是，[21]分别报告了简单，中等和困难水平分别为88.9％，77.3％和72.7％，但是这些结果是根据6000个训练框架和1500个验证框架的不同分割得到的， 它们不能与表1中的算法直接比较。因此，我们在表中没有包括相关的结果。

3D评估 与只需要精确定位2D平面上的物体的鸟瞰检测相比，3D检测是一个更具挑战性的任务，因为它需要3D空间中更精细的形状定位。 表2总结了对比情况。 对于Car类，VoxelNet在所有难度级别上明显优于AP中的所有其他方法。 具体而言，仅使用LiDAR，VoxelNet在简单，中等和高效率方面明显优于基于LiDAR + RGB的最新方法MV（BV + FV + RGB）[10]，分别为10.68％，2.78％和6.29％ 对应简单，中等，和困难等级。 HC基线达到与MV [5]方法相似的精度。

在鸟瞰图评估中，我们还将VoxelNet与HC-基线进行比较，以进行3D行人和骑行者检测。 由于3D姿态和形状的高度变化，成功检测这两个类别需要更好的3D形状表示。 如表2所示，对于更具挑战性的3D检测任务，VoxelNet的改进性能得到了强化（从鸟瞰图提高8％到3D检测提高约12％），这表明VoxelNet在捕获3D形状信息方面比 手工处理的更高效。

4.2. 在KITTI测试集上评估

我们通过将检测结果提交给官方服务器来评估KITTI测试集上的VoxelNet。 结果总结在表3中.VoxelNet在所有任务（鸟瞰图和三维检测）以及所有困难方面明显优于先前发表的最先进的[5]。 我们想要指出的是，KITTI基准测试中列出的许多其他领先方法都使用RGB图像和LiDAR点云，而VoxelNet仅使用LiDAR。

我们在图6中给出了几个3D检测示例。为了更好的可视化，使用LiDAR检测到的3D盒被投影到RGB图像上。 如图所示，VoxelNet在所有类别中提供高度精确的三维边界框。

在TitanX GPU和1.7Ghz CPU上，VoxelNet的处理时间为225ms，体素输入特征计算需要5ms，特征学习网络需要20ms，卷积中间层需要170ms，区域处理网络需要30ms。

5. 结论

基于LiDAR的3D检测中大多数现有的方法依赖于手工特征表示，例如鸟瞰图投影。在本文中，我们消除了手工特征处理的瓶颈，并提出了VoxelNet，这是一种新颖的基于点云的3D检测端到端可训练深度架构。我们的方法可以直接在稀疏的3D点上操作，并有效地捕捉3D形状信息。我们还介绍了VoxelNet的一个高效实现，它可以从点云稀疏性和体素网格上的并行处理中受益。我们在KITTI汽车检测任务上的实验表明，VoxelNet大大超越了先进的基于LiDAR的3D检测方法。在更具挑战性的任务中，例如行人和骑自行车的3D检测，VoxelNet也展示了令人鼓舞的结果，表明它提供了更好的3D表示方法。未来的工作包括将VoxelNet扩展到联合LiDAR和基于图像的端到端3D检测，以进一步提高检测和定位精度。

致谢：我们感谢同事 Russ Webb，Barry Theobald和Jerremy Holland 宝贵的意见。

参考

[1] P. Bariya and K. Nishino. Scale-hierarchical 3d object recognition in cluttered scenes. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 1657–1664, 2010. 2

[2] L. Bo, X. Ren, and D. Fox. Depth Kernel Descriptors for Object Recognition. In IROS, September 2011. 2

[3] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun. Monocular 3d object detection for autonomous driving. In IEEE CVPR, 2016. 2, 5, 6, 7

[4] X. Chen, K. Kundu, Y. Zhu, A. Berneshawi, H. Ma, S. Fidler, and R. Urtasun. 3d object proposals for accurate object class detection. In NIPS, 2015. 2, 5, 6, 7

[5] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-view 3d object detection network for autonomous driving. In IEEE CVPR, 2017. 1, 2, 3, 4, 5, 6, 7, 8

[6] C. Choi, Y. Taguchi, O. Tuzel, M. Y. Liu, and S. Ramalingam. Voting-based pose estimation for robotic assembly using a 3d sensor. In 2012 IEEE International Conference on Robotics and

[7] C. S. Chua and R. Jarvis. Point signatures: A new representation for 3d object recognition. International Journal of Computer Vision, 25(1):63–85, Oct 1997. 2

[8] C. Dorai and A. K. Jain. Cosmos-a representation scheme for 3d free-form objects. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(10):1115–1130, 1997. 2

[9] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In 2017 IEEE International Conference on Robotics and Automation (ICRA), pages 1355–1361, May 2017. 1, 2

[10] M. Enzweiler and D. M. Gavrila. A multilevel mixture-ofexperts framework for pedestrian classification. IEEE Transactions on Image Processing, 20(10):2967–2979, Oct 2011. 3

[11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In Conference on Computer Vision and Pattern Recognition (CVPR), 2012. 1, 2, 5, 6

[12] R. Girshick. Fast r-cnn. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), ICCV '15, 2015. 5, 6

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 580–587, 2014. 1

[14] R. Gomez-Ojeda, J. Briales, and J. Gonzalez-Jimenez. Plsvo: Semi-direct monocular visual odometry by combiningpoints and line segments. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4211–4216, Oct 2016. 1

[15] A. Gonzalez, G. Villalonga, J. Xu, D. Vazquez, J. Amores, and A. Lopez. Multiview random forest of local experts combining rgb and lidar data for pedestrian detection. In IEEE Intelligent Vehicles Symposium (IV), 2015. 1, 2

[16] A. Gonzlez, D. Vzquez, A. M. Lpez, and J. Amores. Onboard object detection: Multicue, multimodal, and multiview random forest of local experts. IEEE Transactions on Cybernetics, 47(11):3980–3990, Nov 2017. 3

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, June 2016. 2, 6

[18] A. G. Howard. Some improvements on deep convolutional neural network based image classification. CoRR, abs/1312.5402, 2013. 6

3d scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(5):433–449, 1999. 2

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 25, pages 1097–1105. Curran Associates, Inc., 2012. 1, 6

[21] B. Li. 3d fully convolutional network for vehicle detection in point cloud. In IROS, 2017. 1, 2, 5, 7

[22] B. Li, T. Zhang, and T. Xia. Vehicle detection from 3d lidar using fully convolutional network. In Robotics: Science and Systems, 2016. 1, 2, 5, 7

[23] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollar. Focal´loss for dense object detection. IEEE ICCV, 2017. 4

[24] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In ECCV, pages 21–37, 2016. 2

[25] A. Mian, M. Bennamoun, and R. Owens. On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes. International Journal of Computer Vision, 89(2):348–361, Sep 2010. 2

[26] Y.-J. Oh and Y. Watanabe. Development of small robot for home floor cleaning. In Proceedings of the 41st SICE Annual Conference. SICE 2002., volume 5, pages 3222–3223 vol.5, Aug 2002. 1

[27] Y. Park, V. Lepetit, and W. Woo. Multiple 3d object tracking for augmented reality. In 2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, pages 117–120, Sept 2008. 1

[28] C. Premebida, J. Carreira, J. Batista, and U. Nunes. Pedestrian detection combining RGB and dense LIDAR data. In IROS, pages 0–1. IEEE, Sep 2014. 1, 2

[29] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2017. 1

[30] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. arXiv preprint arXiv:1706.02413, 2017. 1

[31] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017. 2

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems 28, pages 91–99. 2015. 2, 3, 4, 5

[33] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration.

Blake. Real-time human pose recognition in parts from single depth images. In CVPR 2011, pages 1297–1304, 2011. 2
[35] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014. 6
[36] S. Song and M. Chandraker. Joint sfm and detection cues for monocular 3d localization in road scenes. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3734–3742, June 2015. 2
[37] S. Song and J. Xiao. Sliding shapes for 3d object detection in depth images. In European Conference on Computer Vision, Proceedings, pages 634–651, Cham, 2014. Springer International Publishing. 1, 2
[38] S. Song and J. Xiao. Deep Sliding Shapes for amodal 3D object detection in RGB-D images. In CVPR, 2016. 1, 2, 4,5
[39] F. Stein and G. Medioni. Structural indexing: efficient 3-d object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2):125–145, 1992. 2
[40] O. Tuzel, M.-Y. Liu, Y. Taguchi, and A. Raghunathan. Learning to rank 3d features. In 13th European Conference on Computer Vision, Proceedings, Part I, pages 520–535, 2014.2
[41] D. Z. Wang and I. Posner. Voting for voting in online point cloud object detection. In Proceedings of Robotics: Science and Systems, Rome, Italy, July 2015. 1, 2
[42] Y. Xiang, W. Choi, Y. Lin, and S. Savarese. Data-driven 3d voxel patterns for object category recognition. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2015. 2
[43] M. Z. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(11):2608–2623, 2013. 2
[44] M. Z. Zia, M. Stark, and K. Schindler. Are cars just 3d boxes? jointly estimating the 3d shape of multiple objects. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 3678–3685, June 2014. 2

"""

以上是对苹果论文的翻译

"""

那么苹果的方案好在哪呢？

我认为主要是三点：

2.避免了手动处理原始点云，保留了完整的数据，提高训练与检测精度。

3.应用端到端网路，提高了检测效率。

PointNet 作者提供了开源的代码，相关改进方法可以参考Github的 issue里的讨论。

苹果的VoxelNet并没有提供相关代码，我会在下一篇文章尝试根据论文中的信息用TensorFlow搭建一个模型试试。

无人驾驶车　苹果公司 (Apple Inc.)　人工智能

5 条评论

李铀

用识别的方法处理点云，会牺牲掉一部分点云的物理含义

1 个月前

1 个月前

李铀 回复 **MAZE**（作者） 查看对话

是指如果识别失效，则错过了一个可能障碍物

1 个月前

MAZE（作者） 回复 **李铀** 查看对话

是这样的，尤其是较远距离的物体，由于点云太稀疏，并且只有朝向车本体方向的点云，识别率是比较低的。所以一般结合摄像头数据一起用。你还有什么更好的方案吗？

1 个月前

李铀 回复 **MAZE**（作者） 查看对话

一般来说，会有好的和安全的两套方案

1 个月前

文章被以下专栏收录

人工智能笔记

人工智能从入门到AI统治世界

进入专栏

推荐阅读

无人驾驶的技术安全风险可以避免吗？

今年6月底，特斯拉自动驾驶汽车发生了一起致命意外，一时间无人驾驶的安全性被推到了... 查看全文 >

罗韵 · 1 年前

Pony.ai 首次公开无人车路测视频

Pony.ai 首次公开无人车路测视频。本次发布的视频包括三部分：8字弯 – 超越人类极限的控制：视频中，Pony.ai 无人车在时速30km左右的速度下绕着一个8字连续自动驾驶10多圈没有偏移撞到

量产激光雷达 —— 速腾聚创在增量无人驾驶领域会成为下一个大疆

激光雷达做为最具想象力的「机器人的眼睛」，凭借着探测距离远、测量精度高、响应快的优势，... 查看全文 >

炫姐姐 · 9 个月前 · 发表于 深圳湾 | shenzhenware

刚刚，英伟达发布最强无人车AI芯片，以及一系列自动驾驶新产品

夏乙 发自 凹非寺 量子位 出品 | 公众号 QbitAI英伟达再次带来新“核弹”。在刚刚结束的英伟... 查看全文 >

量子位 · 17 天前 · 发表于 量子位