反卷积网络在驾驶场景中的点云车辆检测和跟踪

车辆检测和跟踪是在城市情景下开发自主驾驶应用的核心要素。最近的基于图像的深度学习（DL）技术正在这些感知任务中获得突破性的成果。然而，DL研究尚未在激光雷达测距仪上处理三维点云。这些传感器在自动驾驶汽车中是非常普遍的，因为尽管没有像图像那样提供语义丰富的信息，但是在恶劣的天气条件下，它们的性能比传感器更加稳健。本文提出了仅用于三维激光雷达信息的全天候探测和跟踪系统。我们的检测步骤使用一个卷积神经网络（CNN），该卷积神经网络接收由三维信息提供的3D信息，并返回一个关于它是否属于车辆的每个点的分类。然后对经过分类的点云进行几何处理，以生成用于估计周围车辆的位置和速度的多假设扩展卡尔曼滤波器（MH-EKF）实现的多目标跟踪系统的观测值。系统主要对KITTI跟踪数据集进行评估，我们展示了基于CNN的车辆检测器在标准几何方法中提供的性能提升。基于激光雷达的方法使用基于图像的检测器所需数据的4％，具有相似的竞争结果。

自主驾驶（AD）现在已经成为现实。这个成功的主要原因是双重的。一方面，机器学习和计算机视觉等相关领域的研究进展正在对车辆环境的场景进行高度的理解。另一方面，新的硬件和车载传感器正在为社区提供足够的数据来开发新的，强大的感知算法以及实时处理这些算法的能力。然而，在完全自主的汽车（AV）在我们的城市完全自由行驶之前，还有很长的路要走。城市交通非常具有挑战性和活力，行人，骑车人和其他车辆等众多干预元素。

然而，在完全自主的汽车（AV）在我们的城市完全自由行驶之前，还有很长的路要走。 城市交通非常具有挑战性和活力，行人，骑自行车者，其他车辆甚至街头家具等众多介入元素。 为此，准确的检测和跟踪算法是AV中至关重要的元素。 这些系统必须足够强大，能够在保证司机，行人和其他道路安全的情况下，对任何可能的情况进行识别，理解和采取行动。

随着深度学习技术的出现，涉及诸如对象检测，语义分割或运动捕捉等任务的基于图像的场景理解已经经历了令人印象深刻的性能和准确度提升[1] [2] [3] [4]。然而，在苛刻的环境条件下，基于图像的方法在实际驾驶情况下可能遭受高性能下降。大雨，雪，雾，甚至是夜景。为了避免这种情况并提高健壮性，必须在AV中包含冗余。这通常是通过建立依赖其他传感器（如雷达或3D激光雷达测距仪）的自主感知系统来解决的。激光雷达传感器特别适用于AD目的，因为它们提供非常准确的空间环境信息，对恶劣的气候条件非常有效，并且其性能几乎与场景照明无关。然而，部署在3Dlidar点云上的深度学习方法远没有达到2D-RGB图像上的成功性能。这主要是由于问题维数变化引起的计算负担，以及缺少注释的训练数据。

我们提出了一个强大的和准确的车辆检测和跟踪系统，只使用三维激光雷达信息作为输入。所开发的系统的草图如图1所示。所提出的方法的主要核心是一个量身定制的全卷积网络（FCN）[3]训练来检测特征范围内的车辆和三维点云的反射，数据由Velodyne 64-HDL传感器来提供。我们的FCN通过对每个3D点是否属于车辆进行逐点分类来完成这项任务。然后对正样本进行聚类并获得2D车辆姿态。这是通过在将簇投影到地平面之后选择最佳的配合边界框（x，y，θ）来执行的。这个2D信息最终被馈送到基于多假设扩展卡尔曼滤波器（MH-EKF）的跟踪器，其随着提取的3D特征（例如对应的群集的高度）提供3D跟踪上的最终结果。

KITTI数据集由德国卡尔斯鲁厄理工学院和丰田美国技术研究院联合创办，是目前国际上最大的自动驾驶场景下的算法评测数据集。该数据集用于评测立体图像(stereo)，光流(optical flow)，视觉测距(visual odometry)，3D物体检测(object detection)和3D跟踪(tracking)等计算机视觉技术在车载环境下的性能。KITTI包含市区、乡村和高速公路等场景采集的真实图像数据，每张图像中最多达15辆车和30个行人，还有各种程度的遮挡与截断。整个数据集由389对立体图像和光流图，39.2 km视觉测距序列以及超过200k 3D标注物体的图像组成[1] ，以10Hz的频率采样及同步。

包括自动驾驶场景中的识别，跟踪，道路识别等。

我们在Kitti跟踪基准[5]上测试我们的系统，其中由于基于图像的2D评估测量，仅激光雷达的方法受到严重的惩罚。然而，我们展示了我们的方法的竞争力，并验证了使用基于CNN的激光雷达探测器对其他几何方法的假设。

相关工作：

深度学习技术和更具体的卷积神经网络（CNNs）在经典的计算机视觉问题，如对象分类[1]，[6]，检测[2]，[7]和语义分割[3]。 然而，有线电视网络还没有部署其可能的超范围激光雷达点云。接下来我们回顾一下在这种三维稀疏点云中检测物体的方法，以及深度学习方法如何接近这个任务。

激光雷达点云中的经典物体检测。

关于在雷达产生的点云中检测物体的文献很多。 最常见的是分割方法，试图将更近的点聚类在一起，并对结果组进行分类[8]，[9]，[10]，[11]。 这些方法通常适用于单层（2D）和多层（3D）激光雷达。 对于后者，投票计划也被用来垂直融合单层集群，获得基于部分的对象模型[12]，[13]。 在自主驾驶应用中，分割方法先前倾向于去除地平面[14,15]，从而简化了聚类和分类步骤。 这个启发式算法对于计算被检测对象的边界框很有用，将在III-C节中展示。 其他细微的聚类方法在预处理的三维像素上创建图形，在分类步骤之后利用它们的连通性[16]，[17]，[18]。

最近的方法用滑动窗口方法直接审查3D范围扫描空间。 例如，Vote3D [19]将稀疏激光雷达点云在具有不同特征的网格编码，例如强度的均值和方差，占用等级以及其他三种不同的形状因子。 结果表示以滑动方式进行扫描使用不同大小和方向的3D窗口，使用SVM和投票方案对最终候选人进行分类。 对于点云分类的分类，从RGB算法继承的标准方法是手工制作自旋图像，形状模型或几何统计等特征。 最常用的3D特征的细节可以在[20]中找到。 学习过程也被用来通过稀疏编码获得有用的特征，如[21]中的工作。

3D激光雷达目标检测的深度学习。

随着特征学习的趋势，并意识到CNN模型的成功，一些作者正在卷积三维激光雷达点云。例如，通常用于视频分析的3D卷积（将其第三维用于时间变量）已被应用于[22]中的3D车辆检测。然而，由于数据的高维度和稀疏性，将这种方法部署在点云上意味着高的计算负担，这对于在线应用是不实际的。重新构建卷积是一个解决方案。通过这种方式，Vote3D最近已经在[23]中扩展，通过用作为投票权重的新型稀疏3D卷积代替SVMs来预测检测分数。其他方法设计和应用稀疏卷积，如[24]，[25]。

另一个采用的方法是获得3D点云的等效二维表示，以应用众所周知和优化的二维卷积工具。这样，[26]建立了一个前视图表示，其中每个元素编码一个地面测量的距离和3D点的高度。在这个表示的基础上，他们应用一个完全卷积网络训练来预测每个点的物体，同时执行每个车辆的3D边界框的回归。类似地，我们将3D点归类为属于车辆或背景，尽管我们的图像式激光雷达表示包括关于点的距离和反射率的直接信息，并且我们使用更先进的去卷积结构，如将在III-A和III -B。

最近的演变[26]结合他们的激光雷达信息的正面视图表示与鸟瞰图，以生成精确的三维边界框提案[27]。这些方法后来在基于区域的融合网络中与RGB图像进行融合，从而获得了最先进的结果，以检测挑战基底组织。然而，这种方法并不能满足我们在工作中要求的激光雷达的要求。

车辆的检测和跟踪

我们将检测激光雷达点云中的车辆的任务重新定义为一个每点分类问题，其中我们想要获得每个样本作为车辆的概率，因此：p（k | pi），其中k∈{vehicle，no -车辆}; i∈1，... N，pi∈R3代表欧几里得空间。

A范围数据的2D表示

为了有效地利用成功的深卷积体系结构，我们将点云通过G（P）∈RH×W投影到图像表示。 这个过程如图2所示。

为了得到变换G（·），我们首先将三维笛卡尔点云投影到球面坐标sph（pi）= {φi，θi，ρi}。根据Velodyne HDL64 speci fi cations，仰角θ被表示为一个H∈R64向量，分辨率Δθ分别为上部激光射线1/3度和下半部1/2度。此外，G（·）需要限制视角的方位角，φ∈[-40.5,40.5]以避免出现未标记的车辆点，因为Kitti跟踪基准仅对前置摄像头观看元素标记。根据制造商，方位分辨率被设定为Δφ= 0.18度的值，因此位于W∈R451。每个H，W对每个投影点的范围（ρ）和反射率编码，所以我们的输入数据表示最终位于图像空间G（P）∈R64×451×2。

为了获得学习过程所需的等效地面实况表示，我们使用在RGB空间中给出的Kitti轨迹。这些轨迹被转换成3D并且在标记内点点之后获得P \*。这些地面真实3D类标签也被编码为G（·）图像空间中的一个通道，像素值为1，背景值为2，车辆值为2。在“车辆”课程中，我们考虑与汽车，面包车和卡车相关的基蒂班。然而，基准的评估方法没有考虑到卡车类别，这可能会对我们的Kitti测量的性能造成不利影响。

为了获得一个有用的输入，我们的基于CNN的车辆探测器，我们投影三维点云原始数据，以包含范围和反射信息的特征像图像表示，通过G（·）。

用于学习提出的分类任务的Groundtruth是通过首先将基于图像的Kitti轨迹投影到3D 信息上，然后在选定的点上再次应用G（·）（一种三维到二维的变换）来获得的。

B.用于车辆检测的反卷积网络

为了实现这个结构车辆分类，我们建立了图3所示的反卷积结构。 在这里，我们透露一些我们设计的重要见解。 为了减少 Velodyne 表示的垂直和水平维度的不平衡，并获得更易处理的中间特征图，在第一卷积层中，我们施加两倍于垂直的水平。 最初的卷积滤波器尺寸也根据在新的表示中观察到的车辆的形状来设计，以便获得与其一致的接收域。 此外，为了解决每个类别的样本数量之间的不平衡，我们惩罚正面车辆样本的错误分类ω如公式2。

由于我们的收缩膨胀设计可能会在狭窄的层面上提供一个信息瓶颈，所以我们引入跳过连接来连接两个部分的等价特征映射。 这些链接通过从较高部分向后传播更纯的渐变来帮助较低层的学习过程。 最后，我们在网络的不同分辨率下陈述分类问题，以获得对学习过程的直接而精细的控制。 这是通过包括中间损失，以更快地引导网络正确解决方案，在这些中间层面引入新的有价值的梯度。

因此，网络是通过以下损失函数引导的端到端反向传播来训练的：

其中r表示中间损失控制位置，λr是每个分辨率损失的权重，Y，Y分别是这些分辨率下的预测和地面实况类别。 在我们的方法中，Lr是一个多级加权交叉熵损失（WCE）[28]，定义为：

以上都是用于车辆的分类呢

C.获取车辆边界框

设计网络的输出Y是在G（·）∈R64×451空间中的一个矩阵，其中每个像素表示对应的3D点属于车辆的概率。为了获得跟踪和评估步骤所需的车辆边界框，我们首先将反变换G-1（Y）应用于网络输出。通过这种方法，我们可以得到等价的三维点云P∈R3，它通过一个欧氏阈值进行聚类。对于每个产生的聚类，提取一组特征来构建EKF观测矢量，例如质心，高度和“车辆性质”（由网络给予该聚类的分类平均值计算）。然后将群集（3D的点云数据）投影到地平面上的极坐标网格上，（这时又变成2维的数据）计算每个只有传感器的最近点的方位角。这个过程的目的是获得每个车辆的外部周长，这将减轻任何一个二维定向矩形模型的任务。？

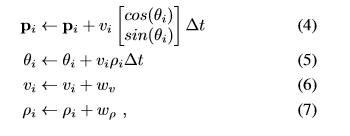
定向边界框的配合过程首先在[-π/4，π/4] 区间执行边界框的角度扫描。 对于每一个我们投射模拟的2D Velodyne射线来获得在盒子上几何等效的影响。 然后选择最佳的二维匹配框作为实际车辆检测点与模拟的最小均方距离之一。 最后提取跟踪器车辆观测向量的有用特征，如二维边界框的宽度，长度和中心等，并利用之前计算出的簇高来恢复三维框。

D.基于激光雷达的车辆跟踪

我们实现了一个多假设扩展卡尔曼滤波器（MH-EKF）来跟踪边界框根据适用于轮式车辆在道路环境中的逼真的运动模型。 当车辆在道路平面上行驶时，二维边界框（BB）被考虑用于跟踪。 由于真实的车辆尺寸和质心不能通过简单的检测来测量，所以我们将BB起点定位在最接近的可见角处，这确实是可测量的。 对于每个BB，我们开始一个MH-EKF，跟踪它的2D位置，方向和速度。 

其中p，（x，y）和π，（x，y，θ）是BB的位置和姿态，v是线速度， 在局部x方向上的速度，ρ是曲率半径的倒数（因此角速度是ω=vρ）。

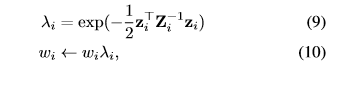
由于检测到的边界框的几何信息是有限的，我们建立了多个运动的假设：即一个沿着主横轴移动，另一个横过它。最初，我们为所有的假设赋予均匀的权重wi = 1 / N，i∈[1，...，N]。 每个EKF估计xi根据运动模型正常演变。



其中←代表时间更新; 和测量模型y = h（xi）+ v，描述如下



在这些模型中，w =（wv，wρ）和v是白色的高斯过程，是减去帧的组成，πV是本车的姿态，这是通过简单的测距法已知的，πS是传感器在车辆上的安装姿势。 测量y =（xS，yS，θS）与传感器帧中的检测算法的结果相匹配。在每次新的观察中，根据当前的假设可能性λi来更新权重。



，即其中zi = y-h（xi）是当前测量的更新，并且Zi是其协方差矩阵。 权重系统地归一化，使得Pwi = 1。最后，当权重下降到阈值τ以下时，其假设被丢弃。 初始检测后的一些观察结果，每个滤波器只剩下一个假设。这个基本的方案是修改与可见的角落的管理：在部分遮挡或车辆超车的情况下，我们可能不得不将最初检测到的角落（可能已经看不见）与最近的可见角落切换。 平凡地更新（x，y）状态到新的可见角落，保持所有其他状态不变。

结论

我们测量了基于CNN的激光雷达探测器在Kitti跟踪基准测试中提供的MH-EKF跟踪器的性能提升。 另外，我们提供DeepLidar探测器获得的精确度/召回率的见解，以及图4中可见的整个系统的定性评估。

Kitti跟踪基准包括8000个Velodyne扫描组成的训练集，分为21个不同的覆盖不同城市环境的序列。为此，提供了在相应的RGB图像上定义的3D轨迹。 另外，在一个测试集中给出了11,095次扫描，分组为29个序列，没有提供注释。 跟踪基准中的任何序列都没有给出Velodyne数据时间戳。由于我们的跟踪器将观测与车辆测距综合在一起，因此我们不得不按照制造商的指定，在10Hz下创建模拟Velodyne数据的合成时间戳。

1. 完整的工作系统

我们使用MatConvNet设计并训练了我们的深度学习模型。网络用He的方法初始化[29]，并使用标准参数β1= 0.9和β2= 0.999的Adam优化。为了保持激光雷达信息的几何特性，只进行水平面反射，数据增加的几率为50％。训练过程在单个NVIDIA K40 GPU上进行20万次迭代，每次迭代20个Velodyne扫描。在最初的15万次迭代中，学习率固定为10-3，之后减半。我们选择不平衡调整器ω为25，损失调整器λr分别为1，0.7和0.5。

对于聚类步骤，我们将3D点分组，最大距离为1米。之后，丢弃小于25点的组或者小于50cms的无线电。其余集群及其边界框然后转换成ROS格式，并保存为我们的跟踪器的输入观察。每个检测到的车辆被分配一个双假设MHEKF：一个假设假设沿着最长的矩形维度运动;另一个穿过它。每个MH-EKF的建立如Tab。我按（按顺序）显示了假设的数量，修剪阈值，每个假设的初始平均值和sigma以及过程和测量噪声的sigma。定向观测噪声被设置为矩形的最大可能误差π2，并且通过取决于模型拟合误差和聚类尺寸的因子进行动态调整：



其中rp和rv是实际点和虚拟点的范围，n是点数，w和l是虚拟矩形的宽度和长度，k是通过实验设置为100的调整参数。所有公制单位 （rp，rv，w，l）用米表示。

1. 实验

我们首先评估我们的逐点卷积检测器的性能。为了避免和修改所提出的架构的泛化能力，我们在训练过程中执行了4倍交叉验证步骤。通过这种方式，我们训练相同的体系结构，每次选择不同的序列，组合一个大约1,000个样本的验证集，其方式是所得到的集合中的车辆与非车辆点的比率保持相似。平均结果表明，我们的检测器能够从验证集中分类Velodyne 3D点，平均精度为82.3％，回忆率为87.6％。请注意，这些措施是有点明智的，并不是指车辆的数量，而是指每次扫描正确分类的平均点数。然而，这个结果证明了训练后的模型能够根据我们的输入表示来更新车辆的一般化信息，这使得能够训练具有Kitti训练集的全部8000个样本的最终模型。另外，我们还测量了DeepLidar车辆探测器对追踪任务的贡献。为此，我们使用三种不同的检测模块评估整个系统的性能：

几何：是我们的基准检测方法，它使用完整的原始Velodyne信息作为输入。根据文献[14]，它最初执行地面地震消除，并对其余点应用聚类算法。然后如第III-C节所述提取边界框，以获得最终的检测结果。然而，由于没有关于所创建的集群的车辆的可靠信息，因此引入了附加的几何约束。直到车辆的一个角落被识别出来，才形成轨道。

DeepLidar：是在整套训练集上训练的深层模型。因此，我们展示了由Kitti评估服务器提供的测试数据集的结果。 •

CNN-GT：其目标是在理想的激光雷达车辆检测下设置跟踪器容量的上限。为此，它使用我们的数据表示的基本事实作为预测来模拟卷积检测器的完美输出。由于在这种方法的检测步骤中没有引入噪声，所以将小于25点的阈值丢弃集群降低到4个。仅在训练集中进行评估，因为没有为测试序列提供地面实况。

我们系统的定性结果。 所有图像均来自Kitti Tracking基准测试集，因此我们的Deep Lidar探测器之前没有任何图像。 在列中，图像显示原始输入点云，深度检测器输出，最终跟踪车辆和提交给评估的RGB投影边界框。 请注意，尽管激光雷达提供的信息稀少，但我们的系统能够在复杂的城市环境中检测（红色点）和跟踪（绿色方块）车辆，即使它们部分被遮挡（最下面一行）也是如此

池化：

解锁：在最小化池中，最大池操作是不可逆的，但是我们可以通过记录一组开关变量中每个池区中最大值的位置来获得一个近似的逆。 在解卷过程中，解卷操作使用这些开关将来自上层的重构放置到合适的位置，从而保持刺激的结构。 参见图1（下图），了解该过程。

纠正Recti fi cation：convnet使用relu非线性，纠正特征映射从而保证特征映射总是正面的。 为了在每一层获得有效的特征重建（也应该是正的），我们将重构的信号通过非线性非线性1

过滤：使用已知滤波器来对前一层的特征映射进行卷积。 为了近似地反转这个，deconvnet使用相同过滤器的转置版本（和其他autoencoder模型，比如RBMs），但是应用于整形图，而不是下面图层的输出。 实际上，这意味着垂直和水平方向上的每个过滤器。

顶部：一个deconvnet层（左）连接到一个convnet层（右）。 deconvnet将从下面的图层重建近似版本的convnet特征。 底部（Bottom）：在deconvnet中使用交换机进行解除操作的示例，该交换机记录在convnet中进行池化时，在每个池区（彩色区域）中记录本地max的位置。 黑色/白色条纹在特征映射中为负面/正面激活。