分类号：U461.99　　学校代码:10363
密　级：公开　　学　　号：2210120163

安徽工程大学
Anhui Polytechnic University

硕士学位论文

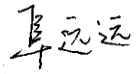
题目 基于相机和激光雷达融合的前方车辆识别研究

论文作者　阜远远
校内导师　王建平（研究员）
校外导师　赵立军（高级工程师）
专业学位类别　机 械
研究方向（领域）　自动驾驶环境感知

论文提交日期：2024 年 6 月 1 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名： 

日期：　　2024 年 6 月 1 日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 □，在 __ 年解密后适用本版权书。

本学位论文属于

不保密 □。

学位论文作者签名: 鄢远远　　指导教师签名: 王建军

日期: 2024 年 6 月 1 日　　日期: 2024 年 6 月 1 日

基于相机和激光雷达融合的前方车辆识别研究

摘 要

随着汽车智能化的快速发展，智能汽车的环境感知能力需要进一步提升，才有可能实现真正意义上的自动驾驶。对于雨雪大雾天气和交通参与者的复杂性，单一的车载传感器无法精确感知周围环境，采用多种传感器并进行信息融合可以发挥各自优势，捕获足够的行车环境信息并经过决策控制系统分析，为汽车安全高效行驶提供最优方案。本文从传感器融合的不同方面进行了四项研究。

（1）基于相机的前方车辆识别。基于 YOLOv5s 算法模型开展了优化定位损失函数、引入 CA 注意力机制和改进 PAN 网络结构的相机车辆识别研究。重新建立并训练包含 7 种类别的车辆数据集，得到训练后的最优权重，并分别在不同道路场景下对比改进模型算法与原算法之间检测框的精准度。

（2）基于激光雷达的前方车辆识别。通过引入自适应注意力机制，以 LeakyReLU 激活函数替换原 ReLU 函数的方法改进原模型，重新训练改进模型并开展消融实验，得到训练结果后将其应用于检测不同场景点云，将同一帧点云图输入改进模型和原模型，比较两种模型识别汽车点云的漏检率和误检率。

（3）相机与激光雷达联合标定。因相机与激光雷达获取信息的机理不同，需要对两种传感器联合标定，两种传感器的时间对齐采用时间戳对齐的方法。由于采集信息不同，首先对相机采用 calibration_camera_lidar 标定工具箱进行标定，然后再利用 Autoware 中的 Calibration Toolkit 标定工具实现相机与激光雷达的联合标定，得到两种传感器标定后的内参矩阵和外参矩阵。

（4）相机与激光雷达信息融合。首先将改进 Pointpillars 产生的三维检测框四元数转化为欧拉角后投影到二维伪图像，然后关联伪图像检测框和相机产生的图像检测框之间的数据，通过计算交并比，量化两个检测框之间的相似程度，通过计算检测框位置的方差评价检测框的稳定性。最后将该算法框架移植到智能驾驶试验小车上验证融合算法的鲁棒性。

关键词：YOLOv5s；Pointpillars；联合标定；多信息融合；环境感知

Research on Forward Vehicle Recognition Based on the Fusion of Camera and LiDAR

ABSTRACT

With the rapid development of automotive intelligence, the environmental perception ability of intelligent vehicles needs to be further improved in order to achieve true autonomous driving. For rainy, snowy, foggy weather and the complexity of traffic participants, a single onboard sensor cannot accurately perceive the surrounding environment. Using multiple sensors and information fusion can leverage their respective advantages, capture sufficient driving environment information, and provide the optimal solution for safe and efficient driving of vehicles through decision control system analysis. This thesis conducts four researches from different aspects of sensor fusion.

(1) Camera based front vehicle recognition. We conducted research on camera vehicle recognition based on the YOLOv5s algorithm model, which optimized the localization loss function, introduced CA attention mechanism, and improved the PAN network structure. Rebuild and train a vehicle dataset containing 7 categories, obtain the optimal weights after training, and compare the accuracy of detection boxes between the improved model algorithm and the original algorithm in different road scenes.

(2) Forward vehicle recognition based on LiDAR. By introducing an adaptive attention mechanism and replacing the original ReLU function with the LeakyReLU activation function, the original model was improved. The improved model was retrained and ablation experiments were conducted. After obtaining the training results, it was applied to detect different scene point clouds. The same frame point cloud image was input into the improved model and the original model, and the missed detection rate and false detection rate of the two models in recognizing automotive point clouds were compared.

(3) Camera and LiDAR joint calibration. Due to the different mechanisms of obtaining information between cameras and LiDAR, it is necessary to jointly calibrate the two sensors, and the time alignment of the two sensors adopts a timestamp alignment method. Due to different information collection, the camera is first calibrated using the calibration_camera_lidar calibration toolbox. Then, the calibration tool in Autoware is used to achieve joint calibration of the camera and LiDAR, obtaining the internal and external parameter matrices of the two calibrated sensors.

(4) Camera and LiDAR information fusion. Firstly, the quaternion of the 3D detection box generated by the improved Pointpillors is transformed into an Euler angle and projected onto a 2D pseudo image. Then, the data between the pseudo image detection box and the image detection box generated by the camera is correlated. By calculating the intersection and union ratio, the similarity between the two detection boxes is quantified, and the stability of the detection box is evaluated by calculating the

variance of the detection box position. Finally, transplant the algorithm framework to the intelligent driving test vehicle to verify the robustness of the fusion algorithm.

KEY WORDS: YOLOv5s;Pointpillars; Joint calibration;Multi information fusion; Environmental perception

目录

第 1 章 绪论

1.1 研究背景及意义

在日常出行工具的选择上，汽车一直是人们不愿放弃的选项之一。公安部统计数据显示，截至 2022 年底，中国汽车保有量突破 3 亿辆，相比较 2021 年增长了 7.5%，这是人类历史上第一个汽车保有量突破 3 亿辆的国家[1]。但是随着汽车保有量的增加，因交通事故死亡的人数也一直居高不下，图 1-1 展示了汽车保有量和因交通事故死亡人数随年份变化的关系，汽车保有量在近十年保持平稳的增长趋势，但是交通事故死亡人数基本维持在 6 万人左右[2]，这对于任何一个国家都是不能接受的。

图 1-1 交通事故与汽车保有量年度统计表

近年来，自动驾驶技术在计算机领域蓬勃发展，引起了研究者广泛的开发兴趣。自动驾驶技术快速发展的一个重要动力是为了避免因驾驶员操作不当而导致的交通事故[3]。据美国国家公路交通安全管理局的统计，注意力分散、疲劳驾驶等因素造成了接近 94%的交通事故[4]。为了减少交通事故的发生，提升车辆智能化水平，国内外涌现了越来越多聚焦智能网联汽车的车企，例如百度 Apollo、Uber、华为、Waymo 等，这些车企不断提升汽车的智能化水平，其自动驾驶汽车如图 1-2 所示。除此之外，越来越多的高校和科研机构也投入大量的人力物力研究自动驾驶解决方案，以加速自动驾驶汽车落地。

（a）Apollo 自动驾驶汽车

（b）Uber 自动驾驶汽车

（c）华为自动驾驶汽车

（d）Waymo 自动驾驶汽车

图 1-2 自动驾驶汽车展示图

自动驾驶汽车最迫切的需求是对前方障碍物的识别，快速准确的识别前方的障碍物来保证驾乘人员的行车安全，因此识别系统决定了自动驾驶汽车的智能化水平[6]。多传感器信息融合一直是智能汽车研究方向的重点和难点。本课题基于相机和激光雷达信息融合识别路上前方行驶的车辆，具有一定的研究价值。

相机和激光雷达各自具有独特的优点。相机能够提供丰富的颜色和纹理信息，对于理解环境的复杂性极为重要，然而其在处理深度信息上的限制却是一个挑战。相反，激光雷达可以精确地获取深度和距离信息，但无法提供颜色和纹理等细节[7]。因此，将相机和激光雷达的信息进行融合，有望在提供更全面、更准确的环境感知信息上实现突破[8-11]。不过，这一技术领域还面临着一些重大挑战，包括如何有效地对两种类型的数据进行配准和融合、如何确保传感器的时间同步，以及如何处理由于环境因素和硬件限制（例如天气、光照条件、设备成本和重量等）而产生的影响。

多传感器融合识别技术对智能网联汽车的意义，其影响力实际上远超过提高单个汽车的环境感知能力。作为一个整体系统，智能网联汽车需要相互通信和协同作业以实现最佳的交通流动性和安全性，这就需要每辆车能准确地识别并理解周围的环境。因此，生成相机与激光雷达融合识别汽车的技术，对于提高整个智能交通系统的安全性和效率，以及降低交通事故率具有重要的作用[12]。这项技术不仅能推动自动驾驶汽车的发展，提高我们的出行效率，还有可能彻底改变我们的交通方式，开启全新的交通模式。它有潜力帮助我们构建更智能、更高效、更安全的交通网络，进一步推动社会进步。

1.2 车辆目标检测研究现状

1.2.1 基于相机的车辆目标检测

基于相机的车辆目标识别主要依赖于相机采集的数据，通过对图片数据的分类达到识别的目的。由于相机的成本相对于其他环境感知传感器较低，目前已经成为机器相机首选的传感器[13]。基于相机的车辆识别绝大多数都是先提取图片中的感兴趣区域，利用当下最先进的识别算法对感兴趣区域检测识别。机器相机的车辆目标识别大致可以分为基于特征的检测方法、基于机器学习的检测方法和基于深度学习的检测方法。

（1）基于特征的检测方法

在计算机相机和图像处理的领域里，特征提取是一个至关重要的概念，它用于从图像中提取出相关的信息。这个过程包含了使用各种计算机算法来确定图像中的每个像素是否具有某种特征。基于特征的检测方法通过一些车辆的共同特征定位车辆，其中车辆底部的阴影、车辆的水平或者垂直的边缘、对称特征、纹理特征都可以作为在光照良好的条件下车辆识别的共同特征[14-16]。张建明等人利用车底部的阴影特征确定车辆可能存在的区域，根据车辆的水平边缘特征信息精确定位，通过卡尔曼滤波器实现跟踪检测到的车辆目标[17]。车辆在夜间行驶时必然会受到光照条件不足的影响，此时车辆尾灯特征是检测前方车辆最好的特征，通过对图片中的颜色通道空间处理，提取出车辆尾灯特征，根据车辆尾灯的相对位置检测出车辆[18]。

（2）基于机器学习的检测方法

这类方法的核心思想是将车辆检测问题转变为图像分类问题[19]。其主要步骤包括：首先，必须积累大量的正面样本和负面样本。正面样本包括了完整车辆的背景图像，而负面样本则是指没有任何车辆的其他图像。其次，对这些样本进行特征提取，通常采用的方法有 SIFT 特征、PCA 特征、SURF 特征、HAAR 特征和 HOG 特征等。选择适当的特征不仅能够提高分类器的训练性能，还能够降低训练过程的复杂性，提升处理速度和效率。接下来，选择合适的分类器模型，常见的算法有 SVM、AdaBoost 算法、神经网络等[20]，具体选择要根据所选特征的不同来决定。最后，进行分类器的训练，首先对收集到的样本进行特征提取，然后利用选择的分类器进行分类学习训练，得到车辆和非车辆的分类结果。特征提取和分类器选择是关键的步骤，直接影响分类器的性能。

叶丽燕等人使用级联分类器技术检测轿车尾部，使用 haar-like 特征训练弱分类器，并通过 Adaboost 算法选择最优弱分类器，为探测路况和防碰撞提供了依据[21]。余小角等人提出一种基于类 Haar 特征和 AdaBoost 分类器并结合车辆灰

度对称性验证的前车检测方法，提高了阴晴天检测率，能够快速有效实现前车检测[22]。Chen X 提出一种基于相机注意力机制和 AdaBoost 级联分类器的高性能车辆检测方法，通过构建结构 Haar 特征并使用结构 Haar 特征提取样本特征并训练 AdaBoost 级联分类器。使用相机注意机制来提取目标候选区域生成检测子窗口，用级联分类器进行区分，实现车辆检测[23]。

基于机器学习的方法具有诸多优点，包括检测速度快、准确率高以及鲁棒性强。它在各种外部环境条件下表现出色，如光照变化或背景与车辆颜色相似的情况下，检测结果依然可靠。然而，这种方法也存在一些缺点，在离线训练阶段，需要收集大量的正负样本集，而且随着数据量的增加，训练所需的时间也会相应增长。

（3）基于深度学习的检测方法

深度学习的飞速发展，越来越多的科研工作者将其应用到各个领域。在车辆检测方面，可以按检测过程分为两类。一类是以分类网络为主干网络找到候选区，再由分类或回归网络检测输出的两阶段目标检测方法，主要以 Faster-RCNN 为代表[24]，首先提出区域建议网络 RPN，快速生成候选区域，然后通过交替训练，使 RPN 和 Fast-RCNN 网络共享参数。邹斌等人在 Faster-RCNN 的基础上添加空间与通道注意力机制，使用加强的双向特征金字塔，优化非极大值抑制，最终使算法的平均精度值提升了 5.6%[25]。另一类是以 YOLO、SSD 为代表的单阶段目标检测算法，YOLO 可谓是单阶段检测算法的开篇之作，它没有真正的去掉候选区，而是将一幅图片分成 S×S 个网格（grid cell），如果某个物体的中心落在这个网格中，则这个网格就负责预测这个物体[26]。目前 YOLO 系列已经发展迭代了 7 个版本，检测性能逐渐提升，虽然 YOLOv5 没有正式的论文发表，但是各种改进版本层出不穷。胡昭华等人通过提出区域上下文模块、特征增强模块 FEM、改进多尺度检测部分提升了 YOLOv5 在交通标志检测方面的 mAP[27]。

由于相机属于二维传感器，对车辆的检测易受光照等环境的影响，如图 1-3 所示，相机在逆光、雨雾天气和隧道口等照明条件较差的环境下生成的图像质量偏低，在检测图像中汽车等障碍物时，会对最终的检测结果产生干扰，而且无法获得所检测车辆的深度信息，所以基于单一相机的相机检测存在一定的局限性。

图 1-3 相机在逆光、雨天、隧道口环境下工作图

1.2.2 基于激光雷达的车辆目标检测

（1）传统方法

基于激光雷达的车辆目标检测的传统方法包含多个关键步骤，从数据采集到最终目标检测结果的输出。首先，激光雷达传感器用于获取三维点云数据，并确保数据的坐标系与其他传感器对齐。接下来，通过点云分割技术，将连续的激光雷达点云分割成独立的物体或场景，常见的方法包括 K-means 或 DBSCAN[28]。在分割后，针对每个物体进行特征提取，以描述其形状、表面等特性，通常使用高度、密度、法线方向等特征[29-31]。随后，通过目标分类算法，将提取的特征与车辆或其他目标进行关联，其中可能使用规则和启发式方法。为了提高目标检测的准确性，可以应用滤波和后处理技术，去除噪声或错误分类，并整合目标的位置和速度信息以进行轨迹估计。最后，将检测到的车辆目标通过可视化呈现在原始图像或点云上，并输出相关信息，如位置、尺寸和速度。尽管传统方法在一些场景中可能受到限制，特别是在复杂场景和光照变化下，但它们仍然为一些实时性要求不高、资源有限的系统提供了可行的解决方案。

（2）深度学习方法

基于激光雷达的车辆目标检测的深度学习方法在近年来取得了显著的进展。这类方法主要依赖于深度神经网络，通过端到端的学习从激光雷达点云中提取丰富的特征以实现车辆目标的准确检测。首先，激光雷达点云数据被输入到卷积神经网络（CNN）或其他深度学习架构中[32-33]，学习数据中的空间和语义信息。随着网络深度的增加，逐渐形成对车辆形状、尺寸和运动模式等特征的抽象表示。在训练过程中，使用带有标签的数据集，网络学习如何映射输入点云到与目标相关的输出，如边界框和目标类别。一些先进的网络结构，如

PointNet、PointNet++、Pointpillars 以及基于图卷积网络的方法[34-36]，已经在激光雷达点云的特征学习和目标检测任务中取得了良好的性能。深度学习方法的优势在于能够自动学习适合任务的特征表示，从而提高在复杂环境中的鲁棒性和泛化能力。

1.2.3 基于相机与激光雷达融合的车辆目标检测现状

传统的目标检测方法主要使用单一传感器，如相机或激光雷达，存在着难以克服的问题，如相机在复杂环境下容易受到光照变化和遮挡的影响，而激光雷达则难以获取目标的细节信息。为了克服这些问题，学者们提出了一种将相机和激光雷达信息融合的方法[37-40]。具体而言，这种方法首先利用激光雷达获取目标的距离和形状信息，然后利用相机获取目标的外观信息。通过建立相机和激光雷达之间的对应关系，可以实现目标在三维空间中的定位和姿态估计[41]。

目前，基于相机与激光雷达融合的车辆目标检测方法主要分为两类：基于特征融合和基于多模态的深度学习方法。基于特征融合的方法通过提取相机和激光雷达的特征，然后将它们融合在一起进行目标检测。这种方法能够综合利用相机和激光雷达的优势，提高检测的精度和鲁棒性。另一种方法是基于多模态的深度学习方法，它利用深度神经网络来融合相机和激光雷达的信息。这种方法在训练过程中可以充分利用大规模的标注数据，从而提高目标检测的准确率[42-43]。此外，还有一些研究者提出了针对特定场景的目标检测方法，如在夜间或恶劣天气条件下的车辆目标检测。

按照融合阶段,可将基于特征融合的目标检测技术细分成早期融合方法和决策型特征融合[44-46]。在早期融合方法中,先将原始数据或其特征进行融合,然后进行目标检测[47]。一个更直接的方法,是通过 CNN 网络对高密集深度图和彩色图像进行联合训练,以实现目标检测。Schlosser J 等人研究了在卷积神经网络上,结合激光雷达和 RGB 视频影像进行行人探测,通过上采样和特征提取，提高了检测性能，尤其在早期到中间层融合的效果最佳[48]。

尽管早期融合方法易于实现，但其抗干扰能力相对较弱。为了解决这一问题，决策级融合方法被引入。这种方案涉及对各传感器的最终处理结果进行融合，从而防止传感器的冲突引起系统故障，并可以在传感器失效时保证正常工作。

Wang H 提出一种融合激光雷达深度信息和相机分类的实时车辆检测算法。通过激光雷达点云检测障碍物，映射到图像上，再应用 YOLO 深度学习方法。在 KITTI 数据集上表现出较高检测精度和实时性，mAP 提高 17%[49]。Li J 等提出了一种新型双视图三维目标检测网络，结合激光雷达点云和 RGB 图像。在工

程场景和自动驾驶汽车中，通过残差模块、BEV 特征提取网络和 RPN，实现了高效的目标检测和分类。通过 CARLA 模拟器和实际验证表明，该网络在复杂场景中表现出令人满意的性能[50]。Liu H 提出一种基于激光雷达和相机融合的目标检测算法，通过构建连体网络和特征层融合策略，克服传感器权衡问题。在 KITTI 数据集上实验证明，算法性能优越，尤其在中等水平上超越其它最先进算法[51]。

现有的决策级融合框架主要存在两个主要问题。一是处理速度太慢，无法满足实时性要求。二是不能充分发挥激光雷达的优势，导致夜间探测精度仍然很低，无法获得车辆的精确距离。

1.3 本文研究内容

本文采用后融合的方法实现相机与激光雷达信息融合达到环境感知的目的，相机识别采用 YOLOv5s 模型，激光雷达识别采用 Pointpillars 模型，两种传感器时空对齐之后，融合两者的检测结果，弥补了两种传感器各自的缺点，提升了智能车的环境感知能力。融合识别的基本框架如图 1-4 所示。论文各章节安排如下：

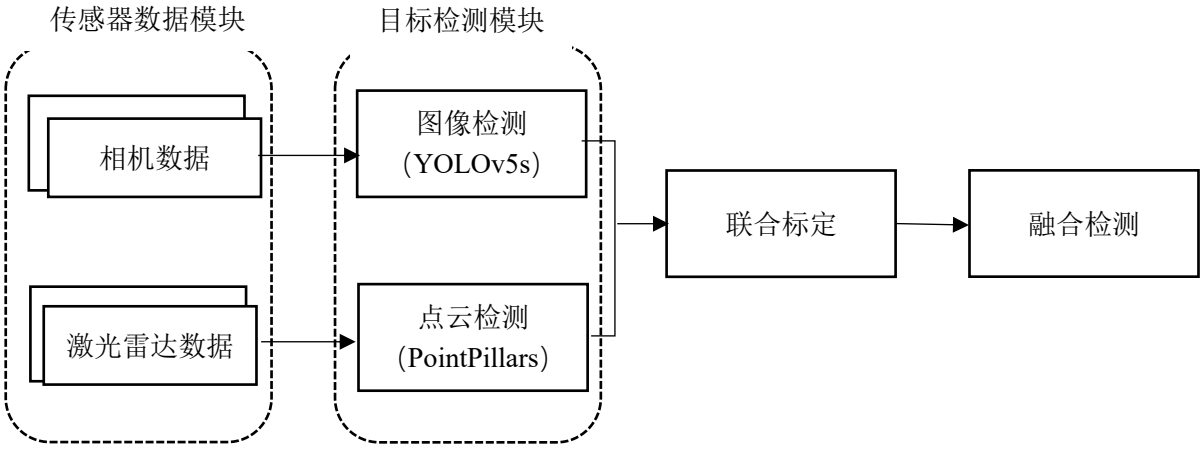


图 1-4 融合识别框架图

第一章，绪论。论述了相机与激光雷达融合识别的意义，相机目标检测的研究现状，激光雷达目标检测的研究现状和相机与激光雷达融合识别的研究现状。简述了论文各章的工作内容与行文安排。

第二章，基于激光雷达的车辆识别。介绍了激光雷达的成像原理，在 Pointpillars 模型引入注意力机制的同时替换激活函数 ReLU 为 LeakyReLU，达到尽可能不牺牲检测速度的前提下提升 Pointpillars 的检测精度。

第三章，基于相机的车辆识别。对 YOLOv5s 模型改进了损失函数，引入注意力机制和改进特征融合网络。在模型训练部分自建数据集以适应车辆检测的

要求，并对社会道路中的常见车辆类型重新分类，以实现为自动驾驶汽车的智能决策提供更多语义信息。

第四章，基于相机与激光雷达后融合的车辆识别。介绍了相机与激光雷达信息时间对齐以及坐标转换之后的空间对齐。实现了点云 3D 检测框到 2D 检测框的转换，通过相似性加权稳定性融合方法输出最终可视化的融合结果。先在 KITTI 数据集实验测试融合效果，然后部署到智能驾驶试验小车验证算法的鲁棒性。

第五章，总结和展望。总结概括了全文所做的工作，对相机与激光雷达融合识别汽车方面在未来可能的发展做出了更深层次的展望。

第 2 章 基于相机的车辆识别

汽车行业的科研工作者致力于不断提升相机识别智能车周围环境的速度和精度，虽然相机产生的数据是 2D 图像，视场有限，只聚焦于有限视场，且没有深度值信息，但是可以得到一个清晰、可选高分辨率的图像，这就可以实现对车身周围物体形状和类别的高精度感知。AlexNet 让人们再一次看到了深度学习的未来，越来越多的业界人士不断在机器相机领域深耕，目前已经出现多种多样的物体检测算法。

2.1 车辆识别算法

卷积神经网络的兴起推动了车辆识别技术的飞速发展。车辆识别算法主要分为两类：一种是以 FasterRCNN 为代表的两阶段算法，它将识别检测过程分为两个步骤。首先，在图片上生成候选区域，然后对这些候选区域进行分类。然而，由于生成候选区域的阶段需要大量的卷积操作，消耗了大量的计算资源，因此 FasterRCNN 很难满足实时检测的要求，不适用于部署在自动驾驶车辆上进行前方车辆的识别。另一种是以 SSD 和 YOLO 为代表的单阶段物体检测算法，这种算法取消了候选框的生成过程，而是利用一个卷积网络直接将整个图像分成多个区域，然后预测区域内目标的边界框和概率[52]。这种方法在减少计算资源的同时也提高了检测速度。尤其是 YOLO 检测算法，已经迭代了 8 个版本，识别的精度和速度已经能满足实时检测的要求，目前大多数的研究者使用的多为 YOLOv5 以及 YOLOv5 之前的版本，表 2-1 是 YOLO 至 YOLOv5 算法对比，可以看出 YOLOv5 的 FPS 已经达到 120。

表 2-1 YOLO 系列算法对比

算法名称	骨干网络	mAP(%)			FPS
		COCO	VOC 2007	VOC 2012	
YOLO	VGG-16	21.6	63.4	57.9	45
YOLOv2	Darknet-19	57.9	78.6	73.5	40
YOLOv3	Darknet-53	43.5	82.3	\	51
YOLOv4	CSPDarknet-53	43.5	\	\	83
YOLOv5	BottleneckCSP	54	\	\	120

YOLOv5 算法分为四个版本：s、m、x、l，按照大小排序，速度逐渐减慢，但精度逐渐提高。该算法由四个部分组成：输入端、Backbone、Neck、Prediction，具体的网络结构图如图 2-1 所示。在 YOLOv5s 中，图像经过数据增强，尺寸变换后进入 Backbone 网络，通过卷积操作提取图像特征；接着，在 Neck 部分获取强语义信息和定位信息；最后，利用回归计算预测物体位置和置

信度。

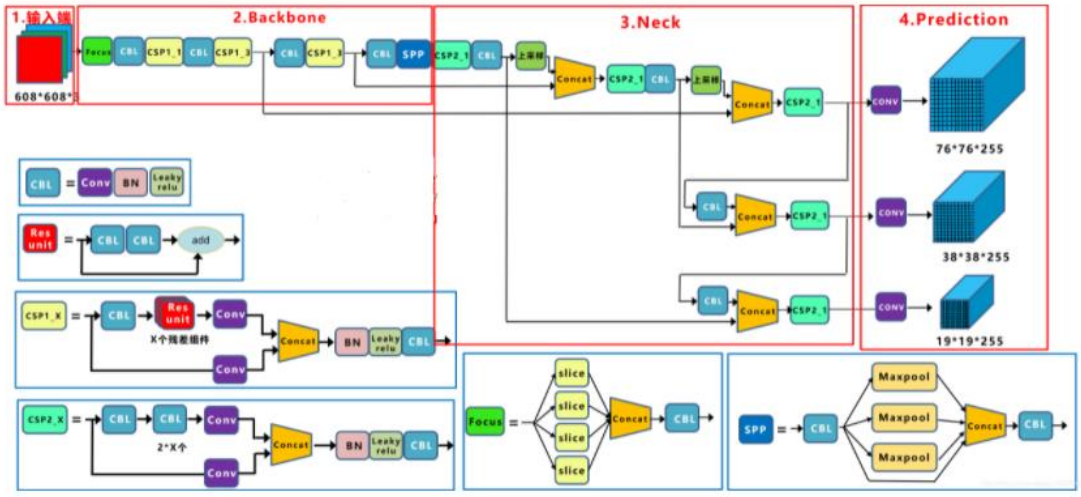


图 2-1 YOLOv5 的网络结构图

2.2 YOLOv5s 的改进

YOLOv5s 算法具备识别 80 种类物体的能力，不断受到专家学者的优化和改进，以提升检测速度和精度，满足实际需求。颜学坤等人在网络颈部检测层引入了通道注意力模块，增强对小目标的检测性能[53]。邓天民等简化了主干网络中的卷积层数量，增强了特征融合能力，提高了识别精度[54]。邱天衡等改进了特征金字塔结构，增加了检测层以提高特征融合效果[55]。刘超阳应用了 CIoU 损失函数和 DIoU_nms 来提升目标识别效果[56]。刘志杰等在 YOLOv5s 模型 Backbone 的 SPPF 前增加了一层 CBAM(卷积块注意力模块)，并使用 ACON(是否获得)代替 ReLU 作为网络的激活函数，提高了检测精度，可以有效地应用于复杂场景下的车辆检测[57]。沈轩静等在模型的主干部分引入坐标注意力模块指导模型提高对遮挡条件下车辆的位置信息关注度。通过融入双向尺度连接和加权特征融合对模型的特征融合组件改进，对 YOLOv5s 的预测头进行解耦，将回归和分类任务分配给两个独立的分支，提高了车辆检测精度[58]。

2.2.1 改进损失函数

损失函数在深度学习中的作用是衡量模型预测与实际标签之间的差异，它是训练过程中优化模型参数的关键部分。在目标检测任务中，损失函数用于衡量模型预测的边界框与实际目标边界框之间的距离以及目标类别的预测准确性。在 YOLOv5s 中，损失函数起着关键作用，帮助模型不断调整参数以最小化预测边界框与真实边界框之间的差异，同时确保正确地分类检测到的目标。YOLOv5s 的损失函数通常由边界框坐标损失、目标类别损失和置信度损失三部

分组成，这些损失函数共同构成了模型训练的目标，帮助模型逐步提高检测准确性[59-60]。

（1）分类损失（classification loss）

在 YOLOv5s 中，同一个目标可能对应不同的类别，例如“路人”和“学生”，存在各个类别的预测概率之和大于 1 的可能性。二元交叉熵损失函数能有效解决上述分类损失问题。这种损失函数可以衡量模型对每个类别的预测概率与实际标签之间的差异，确保模型能够学习并调整每个类别的预测概率，以提高分类准确性，分类损失函数的数学表达式如式 2-1 和 2-2 所示。

$$y_i = Sigmoid(x_i) = \frac{1}{1+e^{-x_i}} \tag{2-1}$$

$$Lclass = -\sum_{n=1}^{N} y_i^* \log(y_i) + (1-y_i^*)\log(1-y_i) \tag{2-2}$$

式 2-1 和 2-2 中的 N 表示类别总个数，x_i 表示预测值，y_i 表示在经过激活函数之后计算的类别的概率，y_i^* 表示真实值（0 或 1），$Lclass$ 表示分类损失。

（2）定位损失（localization loss）

定位损失是衡量预测框与真实框位置之间距离差异的标准。YOLOv5s 原模型使用 GIoU（Generalized Intersection over Union）作为定位损失函数，GIoU 损失的作用是衡量预测框和目标框之间的位置和大小的匹配程度，从而指导模型在训练过程中更好地调整目标框的位置和尺寸，以提高目标检测的精度。

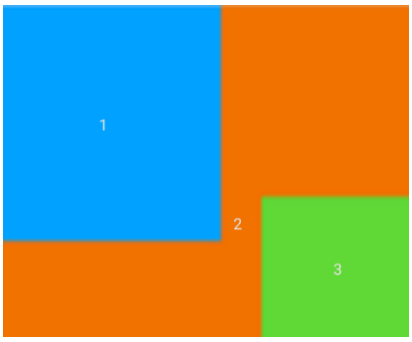


图 2-2 损失函数示意图

图 2-2 中的矩形 1 代表预测框面积，3 代表真实框面积，为方便说明图中未画出矩形 1 与矩形 3 重叠面积。最小方框 C 由 1、2、3 的面积和组成。真实框与预测框距离的差异可以用最小方框 C 的面积除以预测框与真实框重叠面积得到。GIoU 损失计算公式可以由式 2-3、2-4、2-5、2-6 和 2-7 表示。

$$x_1^c = min(x_1^B, x_1^{Bgt}), x_2^c = max(x_2^B, x_2^{Bgt}) \tag{2-3}$$

$$y_1^c = min(y_1^B, y_1^{Bgt}), y_2^c = max(y_2^B, y_2^{Bgt}) \tag{2-4}$$

$$C = (x_2^c - x_1^c) * (y_2^c - y_1^c) \tag{2-5}$$

$$\mathrm{GIoU(B,B}_{gt})=\mathrm{IoU(B,B}_{gt})-\frac{|C-(B\cup B_{gt})|}{|C|} \tag{2-6}$$

$$LGIoU(B,B_{gt})=1-GIoU(B,B_{gt})=1-IoU(B,B_{gt})+\frac{|C-(B\cup B_{gt})|}{|C|} \tag{2-7}$$

式 2-3 中 B 为预测框中心点的位置，B_{gt}为真实框中心点的位置，x_1^c与x_2^c分别代表可以覆盖预测框与真实框最小矩形 X 轴的最小横坐标与最大横坐标，y_1^c与y_2^c分别代表可以覆盖预测框与真实框最小矩形 Y 轴的最小纵坐标与最大纵坐标。

本文将原始模型中的 GIoU 更换为 CIoU 作为 YOLOv5s 的损失函数，这意味着在计算 IoU 损失后引入了额外的损失项以及一个调整因子，用于考虑预测框和目标框之间的匹配关系。该调整因子考虑了预测框和目标框的长宽比之间的匹配程度。具体而言，CIoU 损失函数的定义如式 2-8 所示。

$$B_{CIoU}=1-IoU+\frac{\rho^2(b,b^{ot})}{c^2}\alpha v \tag{2-8}$$

其中，v是预测框和真实框长宽比例差值的归一化，可以由式 2-9 表示。

$$v=\frac{4}{\pi^2}(arctan\frac{\omega^{ot}}{h^{ot}}-arctan\frac{\omega}{h})^2 \tag{2-9}$$

α是权重系数，ω 和 h 分别是锚框的宽和高，可以表示为式 2-10。

$$\alpha=\frac{v}{(1-IoU)+v} \tag{2-10}$$

CIoU 在考虑中心点距离和重叠面积的基础上进一步考虑了长宽比，以便预测框更好的与真实框相匹配，但是 CIoU 的梯度在宽高比的区间属于[0,1]时会爆炸，因为ω^2+h^2的值会很小，所以令这个值为 1 即可。

CIoU 相较 GIoU 是一种全面的损失函数。EIoU 对其继续改进，将纵横比因子拆开分别计算目标框的长和宽，表达式如式 2-11 所示。

$$\begin{aligned}EIoU&=L_{IoU}+L_{dis}+L_{asp}\\&=1-IoU+\frac{\rho^2(b,b^{gt})}{c^2}+\frac{\rho^2(w,w^{gt})}{C_w^2}+\frac{\rho^2(h,h^{gt})}{C_h^2}\end{aligned} \tag{2-11}$$

式 2-11 中c_ω和c_h为覆盖预测框和真实框的最小封闭矩形的宽和高。EIoU 是在 CIoU 的基础上进一步的改进，它将边界框的纵横比损失项分解为预测的宽度和高度与实际最小外接框的宽度和高度之间的差异。通过这种方法，模型能更加专注于优化与目标框重叠较多的高质量锚框，减少对边界框回归影响较小的锚框。

（3）置信度损失（confidence loss）

每个检测框的置信度反映了其可信度水平，高置信度意味着预测框更可靠，

而低置信度则表示预测框与真实框之间的差距更大。为了建立概率分布，本文选择 sigmoid 函数来替代 softmax 函数。此外，本文采用了二元交叉熵损失函数（Binary Cross-Entropy Loss，BCE Loss）来度量模型的预测与实际情况之间的差异。

2.2.2 引入注意力机制

人类在浏览图像时，通常会将相机焦点放在感兴趣的区域，以便获取所需目标的详细信息，而忽略其他无关信息，以节省时间和精力。这种特性启发了将注意力机制引入深度学习模型的想法。

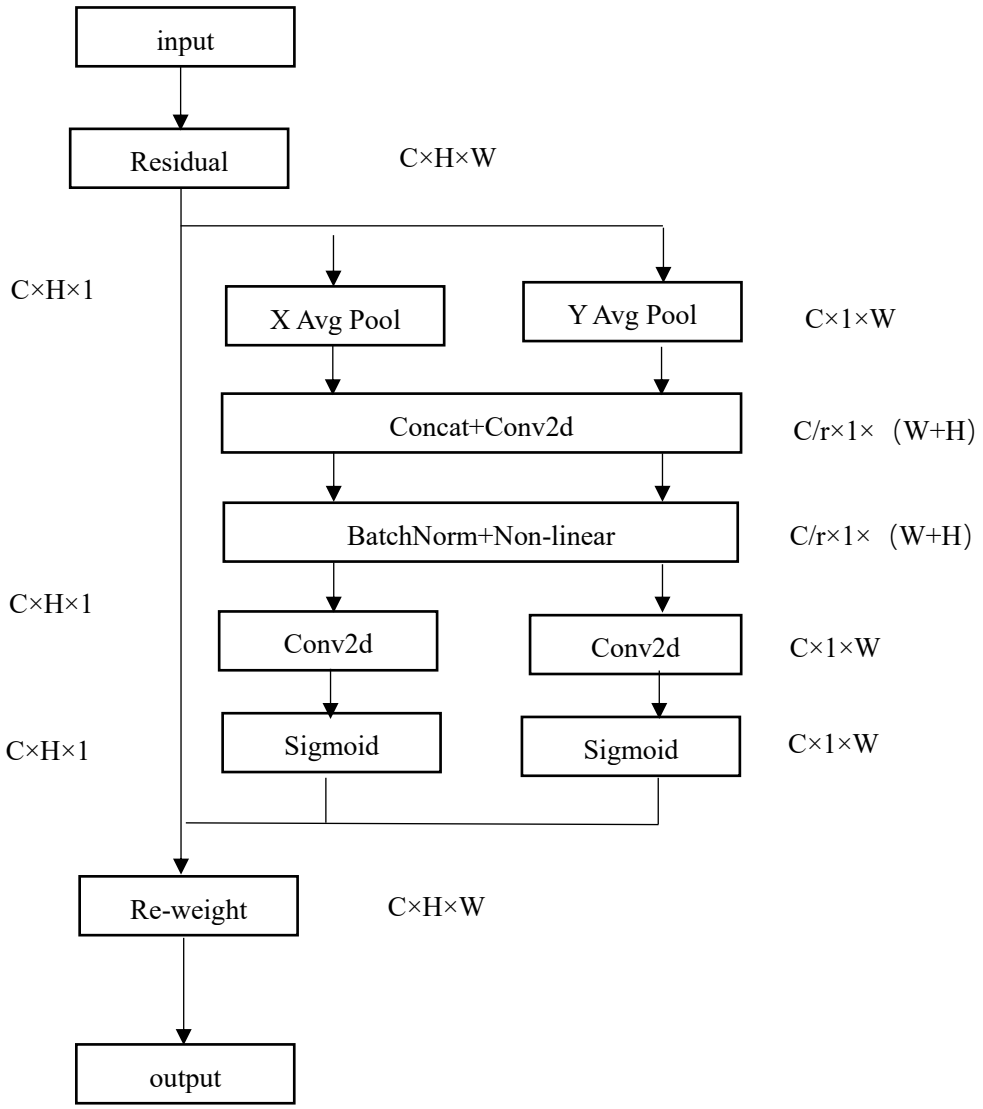


图 2-3 CA 注意力机制网络结构

该机制通过比较当前输入序列与输出向量的相似度来确定应该在哪些区域集中注意力。相似度越高，意味着相关区域的注意力权重越高。注意力权重是

根据当前序列对计算得出的，而不是整个网络模型的权重。本文将引入 CA（Coordinate Attention）注意力机制到 YOLOv5s，CA 注意力机制网络结构如图 2-3 所示。CA 注意力机制将通道注意力拆分成两个 1 维特征编码的过程，分别在两个空间方向上聚合特征。

2.2.3 改进特征融合网络

YOLOv5s 的 Neck 部分采用了 FPN+PAN 的结构。FPN（Feature Pyramid Network）是自顶向下的，通过上采样将高层特征信息传递到低层，并与低层特征融合，以改善低层特征的传播。与之不同的是，PAN（Path Aggregation Network）是自底向上传达强定位特征，两者从不同的主干层对不同的检测层进行参数聚合。本文将 YOLOv5s 的 FPN 改进为重复加权双向特征金字塔（BiFPN），以实现一种简单快速的多尺度融合方式，使网络更加关注重要的层次，减少一些不必要层的节点连接。

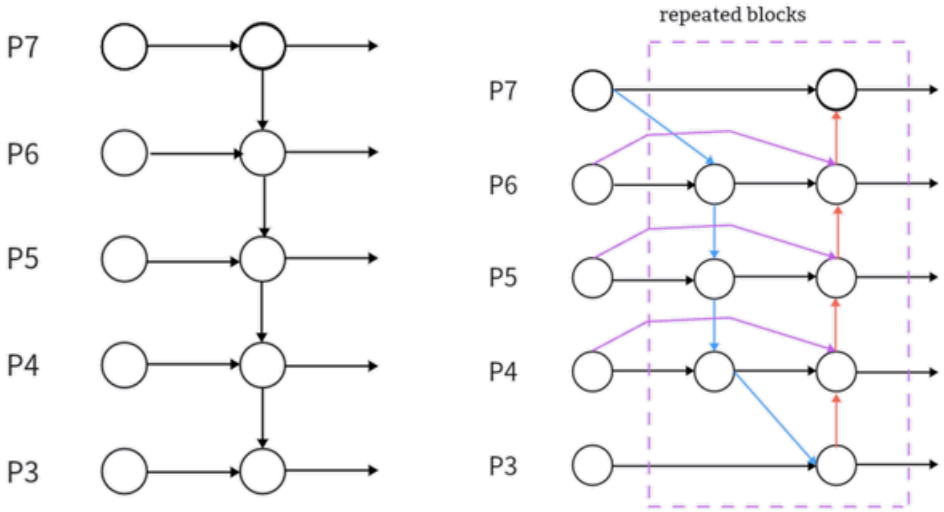


图 2-4 FPN 网络结构（左）与 BiFPN 网络结构（右）

BiFPN 是 FPN 的扩展，它通过在同一级别的输入和输出节点之间添加额外的连接来提高特征融合效率，而不增加额外的复制成本。图 2-4 右侧 BiFPN 网络结构中，蓝色箭头表示自顶向下的通路，传递高层语义信息；红色箭头表示自底向上的通路，传递低层位置信息；紫色箭头则是在同一级别的输入和输出节点之间新添加的连接。

2.3 训练模型

2.3.1 数据集建立

本文研究的目的在于提升车辆检测的准确性。鉴于 YOLOv5s 模型在 COCO

数据集上的表现不佳，因此创建一个新的数据集，其中包含 2025 张来自日常场景的车辆图片。考虑到现实生活中车辆种类的多样性，将这些车辆分为七类：小型车（tinycar）、短途载人车辆（midcar）、中长途载人车辆（bigcar）、小型运输车辆（smalltruck）、大型运输车辆（bigtruck）、油罐车（oiltruck）以及特殊车辆（specialcar）。图 2-5 展示了数据集中各类车辆的外观。通过这种分类，本文能够针对不同类型和距离的车辆设置不同的提示语，从而提高实际车辆检测跟踪的效果。例如，当检测到特殊车辆（如消防车、救护车）时，本文的驾驶系统将持续向驾驶员发送警告信息，提醒他们保持安全距离并注意礼让，以提升驾驶体验的舒适性和文明程度。

图 2-5 数据集中的车辆类别

数据集的标注是利用 makesense 在线标注工具完成的，它对原始图像进行了边界框标注，并生成了符合 YOLO 格式的标注文件。数据集标注完成之后在 YOLOv5s 算法框架中，新建数据集并更改一个新的名称，例如“newdata”， 标注完成的数据和原始图片数据放在对应的文件夹并对训练集、验证集和测试集进行划分。模仿“coco.yaml”中的文件内容，修改其中的类别以及数量粘贴生成新的“newdata.yaml”文件。完成数据集的建立后，开始训练改进 YOLOv5s 模型。图 2-6 展示了训练过程中的训练批次图，其中包含了数据集中的 7 个车辆类别，显示了网络在训练中充分利用了训练数据。

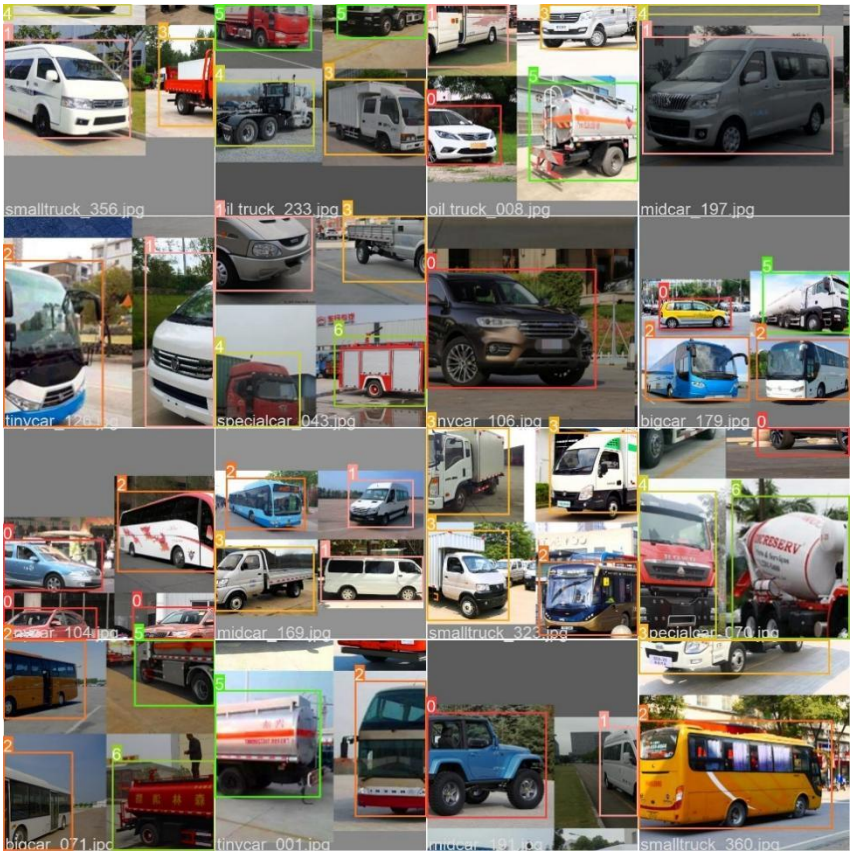


图 2-6 训练批次图

2.3.2 模型训练

训练 YOLOv5s，除了准备数据集外，选择合适的硬件配置也是至关重要的一步。通过搭载适当的硬件设备，如高性能的中央处理器（CPU）、图形处理器（GPU）以及足够的内存资源，可以有效提升模型训练的效率和性能。本文训练模型使用计算机配置的部分参数如表 2-2 所示。

表 2-2 计算机硬件部分参数

名称	型号
操作系统	Windows10
处理器型号	12th Gen Intel(R) Core(TM) i7-12700F
处理器频率	2.10 GHz
显卡型号	GEFORCE RTX 3060 Ultra oc
显卡容量	32GB
存储容量	500G

在训练 YOLOv5s 模型之前，超参数优化是必不可少的。良好的初始参数设置可以帮助模型更快地收敛。YOLOv5s 原模型采用遗传算法对超参数进行优化，这一过程不仅可以帮助模型更快地收敛，还有助于获得更优质的最终性能，具体的超参数设置如表 2-3 所示。

表 2-3 超参数设置

超参数			
初始学习率	0.01	锚框长宽比	4
循环学习率	0.01	焦损失伽马	0
学习率动量	0.937	色调	0.015
权重衰减系数	0.0005	饱和度	0.7
预热学习	3	亮度	0.4
预热学习动量	0.8	旋转角度	0
预热初始学习率	0.1	平移	0.1
定位损失系数	0.05	图像缩放	0.5
分类损失系数	0.5	图像剪切	0
正样本权重	1	透明度	0
有无物体系数	1	进行左右翻转概率	0.5
有无物体正样本权重	1	马赛克数据增强	1
定位损失阈值	0.2	图像混叠概率	0
进行上下翻转概率	0	复制粘贴概率	0

2.3.3 实验与分析

对原模型 YOLOv5s 训练 1300 轮，引入注意力机制、改进特征融合网络且损失函数分别为 EIoU 和 CIoU 的 YOLOv5s 训练 1300 轮得到的 PR 曲线如图 2-7、图 2-8 所示，P 代表精度（Precision），R 代表召回率（Recall），PR 曲线与坐标轴围成的面积越大，表示同等条件下模型训练结果越好。图 2-8 比图 2-7 有更好的 PR 曲线，说明改进 YOLOv5s 训练结果优于原算法模型。图 2-8 左图是损失函数为 CIoU 训练 1300 轮得到的结果，右图是损失函数为 EIoU 训练 1300 轮得到的结果，对比左右两图的 PR 曲线可得，当把原模型的损失函数改为 CIoU 时，训练的结果略优于 EIoU。

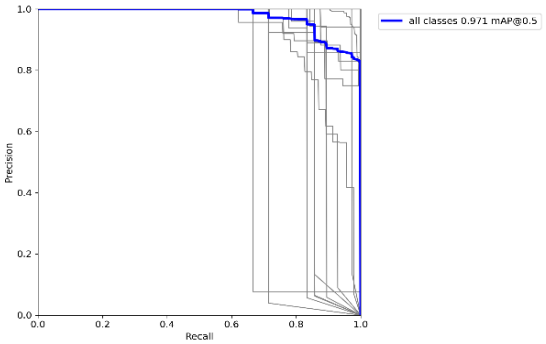


图 2-7 GIoU 作为损失函数的 PR 曲线

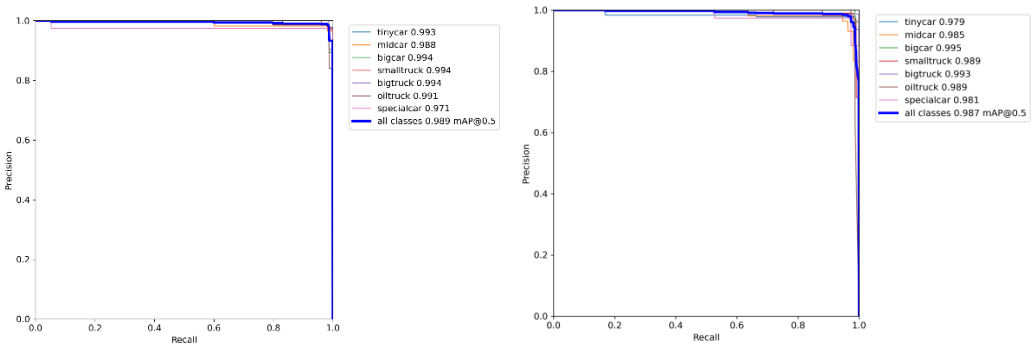


图 2-8 CIoU（左）与 EIoU（右）作为损失函数的 PR 曲线

为验证改进算法的合理性和有效性，本文采用了一种省力且高效的方法来研究改进模型相较原模型具有哪些改变：消融实验。在这些实验中，通过增添或删除一些模块来分析算法模型的影响。以 YOLOv5s 原模型为基准，分别测试了改进损失函数、引入注意力机制以及改进特征融合网络对模型性能的影响。然后，评估这些交叉改进算法是否能够提升模型的性能。通过一系列实验，得到了表 2-4 所示的结果。

表 2-4 改进 YOLOv5s 消融实验结果

损失函数	CA 注意力	特征融合网络	P	R	mAP@0.5
√	×	×	0.921	0.917	0.922
×	√	×	0.935	0.928	0.938
×	×	√	0.932	0.933	0.924
√	√	×	0.931	0.935	0.943
×	√	√	0.950	0.953	0.958
√	√	√	0.979	0.979	0.989

在目标检测和分类任务中，TP（True Positives，真正例）、FP（False Positives，假正例）和 FN（False Negatives，假负例）是关键的性能指标，用于评估模型的性能。它们反映了模型预测与真实标签之间的关系。True Positives 表示模型正确地将正类别样本（目标或感兴趣的类别）预测为正类别。False Positives 表示模型错误地将负类别样本（不是目标或不感兴趣的类别）预测为正类别。False Negatives 表示模型错误地将正类别样本预测为负类别。

FN 表示模型未能检测到实际存在的正类别样本，将其错误地分类为负类别。这三个指标通常用于计算其它性能度量，如 Precision（精度）、Recall（召回率）、F1 Score 和 mAP（平均精度）等。这些指标有助于评估模型的准确性、召回率、综合性能以及识别目标的能力。

精度（Precision）衡量了模型在所有被分类为正类别的样本中，有多少是真正的正类别。Precision 的计算公式可以表示为式 2-12。

$$P = \frac{TP}{TP + FP} \tag{2-12}$$

召回率（Recall）衡量了模型成功地检测到了多少正类别样本。Recall 的计算公式可以表示为式 2-13。

$$R = \frac{TP}{TP + FN} \tag{2-13}$$

P 和 R 两个指标通常被视为权衡，因为增加 Precision 通常会降低 Recall，反之亦然。mAP 是对不同类别或对象的平均精度进行平均的值，通常用于多类别目标检测问题。mAP 是一种常用于衡量目标检测模型性能的标准指标，它兼顾了检测的准确性和召回率，并且适用于多类别问题。模型的高 mAP 值通常表示其在多个类别的物体检测任务中表现出良好的性能。

从表 2-4 可以看出，同时改进损失函数、特征融合网络和引入 CA 注意力机制比其他单独改进其中一种或两种的效果更好。在精度和召回率同时达到 0.979 时，当 IoU 阈值为 0.5 时，mAP 值达到了 0.989，相较于单独改进损失函数提升了 6.7%。

本文评估改进后的算法在车辆检测任务上的表现，与当前主流的单阶段物体检测算法 YOLOv3、YOLOv4、YOLOv5s 以及 SSD 进行了对比试验。本文使用了 mAP@0.5 和 FPS 两种评价指标，在建立的数据集上进行了评估，对比试验结果如表 2-5 所示。

表 2-5 改进算法与其他算法对比试验结果

检测算法名称	mAP@0.5	FPS
YOLOv3	88.3	31.8
YOLOv4	90.2	43.5
YOLOv5s	92.6	72.8
SSD	85.1	74.9
本文改进算法	97.3	71.7

由表 2-5 可知，改进后的 YOLOv5s 在检测前方车辆的数据集上平均精度均值达到了 97.3%。相比于 YOLOv3、YOLOv4、YOLOv5s 和 SSD，其精度分别提高了 9%、7.1%、4.7%和 12.2%。而在 FPS 方面，相对于 YOLOv3、YOLOv4 的提升分别达到了 39.9%、28.2%，相对于 YOLOv5s 和 SSD 略有降低。综合考虑速度和精度，改进后的 YOLOv5s 算法在车辆实时检测方面表现出了良好的性能，既提高了检测精度，又基本保持了检测速度。

2.3.4 可视化实验结果比较

在比较改进后的 YOLOv5s 模型和原模型的检测速度时，首先确保在相同的硬件设备和软件环境下进行测试，并使用深度学习框架提供的性能分析工具多次重复实验以测量推理时间，以获得更准确的结果。在实际检测中，改进

YOLOv5s 模型与原模型的检测速度没有差异。YOLOv5s 原模型包含 213 层，改进后（改进损失函数、引入注意力机制和改进特征融合网络）的模型包含 270 层。表 2-6 展示了 YOLOv5s 模型改进前后的摘要对比，改进算法模型与原算法模型每秒进行的浮点运算数（GFLOPs）都是 15.9 亿次。因此，本文改进 YOLOv5s 没有影响模型检测车辆的速度。

表 2-6 YOLOv5s 改进前后模型摘要对比

模型摘要	神经网络层数	参数	模型计算量
原 YOLOv5s	213	7029004	15.9
本文改进 YOLOv5s	270	7038508	15.9

视频流的目标检测是通过对视频中的图片逐帧检测实现的。由表 2-6 可知，在检测速度方面，改进 YOLOv5s 算法模型与原模型相同。在检测端应用本文训练完成的最优模型权重参数，然后将车辆在不同道路场景下行驶视频输入到原模型和改进 YOLOv5s 模型的检测端，随机截取视频中的某一帧进行检测，如图 2-9 所示。

（a）市区十字交叉道路

（b）校园主干道路

（c）地上地下停车场

（d）夜间市区内部道路

（e）市区隧道道路

（f）雨雪天气道路

图 2-9 不同道路环境下车辆识别结果

图 2-9 分别是市区、校园、停车场、隧道和雨雪天气场景的 YOLOv5s 检测图，①和③是 YOLOv5s 原模型的车辆检测图，②和④是本文改进 YOLOv5s 模型车辆检测图，图（a）、（b）、（c）是光线较好的白天，可以看出②的预测框普遍比①的预测框概率高，④的预测框概率普遍比③的预测框概率高。图（d）、（e）、（f）是光线不好的夜晚、隧道、雨雪天场景，其中图（d）中的①出现了同向车道车辆重复检测情况，将对向车道地面误检测为车辆的情况；图（e）中的③出现了预测类别错误的情况，图中的 truck 在原模型训练时分类的类别为 car；图（f）中①、③距离摄像头最近的车的预测框的概率均低于②、④预测框的概率。综上，改进 YOLOv5s 提高了汽车预测精度，可以减少误检情况的发生，提升汽车摄像头传感器感知外界环境能力。

2.3 本章小结

本章主要研究了基于改进 YOLOv5s 的前方车辆识别算法。首先通过引入注意力机制、改进损失函数和改进特征融合网络对原始的 YOLOv5s 算法模型改进，然后通过消融实验得出同时做上述三方面的改进能在不影响模型检测速度的情况下提升检测精度的结论，分别在市区、校园、停车场、隧道和雨雪天气五种道路场景下对比了改进模型与原模型在检测精度以及准确性方面的区别，实验结果表明，本章改进算法模型在多种场景下的识别效果均优于原算法模型。

第 3 章 基于激光雷达的车辆识别

3D 目标检测技术在自动驾驶系统中扮演着重要的角色，用于环境感知模块。准确识别道路上的车辆、行人、骑车人等物体对车辆规划和控制至关重要。为了实现这一目标，自动驾驶汽车需要利用多种传感器，其中激光雷达是至关重要的一种。激光雷达可以通过扫描仪测量周围环境的距离，直接生成稀疏的三维点云信息，在三维目标检测任务中具有天然优势[61]。传统方法通常涉及对点云信息进行下采样，然后去除地面，并使用欧氏距离聚类 (DBSCAN) 等方法结合三维边界框对目标进行检测[62]。然而，传统方法在部署过程中需要复杂的参数调整，难以在实际应用中得到有效应用。随着深度学习技术和并行计算单元的快速发展，基于深度学习的端到端三维目标检测方法已成为当前的研究重点。

3.1 激光雷达成像原理

激光雷达在工作时通过向周围的视场角发射多条激光束，这些激光束被障碍物或物体反射，并由激光接收器捕获。由于激光的发射和接收数据量相当大，因此可以利用激光返回的点云数据构建目标物体的三维轮廓[63]。激光雷达的探测技术主要分为 TOF（飞行时间法）和相干探测方法，本文所使用的激光雷达基于 TOF 原理工作，工作原理如图 3-1 所示。

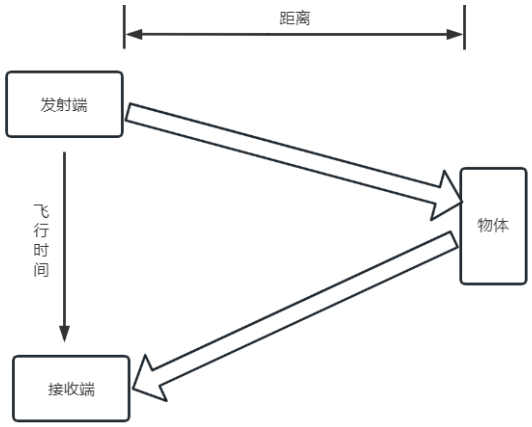


图 3-1 激光雷达 TOF 工作原理

图 3-2 是二维激光雷达和三维激光雷达的实物图。二维激光雷达扫描结果是空间中的一条线，无法获知物体的高度信息，不利于前方车辆的识别判断。三维激光雷达在垂直地面方向存在一定视角，在发射激光时，能够有效获取物体的高度信息形成具有深度的点云数据，随着激光雷达成本的降低，目前大多数车载激光雷达采用三维激光雷达。

图 3-2 二维激光雷达（左）、三维激光雷达（右）

三维激光雷达两个相邻激光发射器在垂直角度固定，最上方和最下方的发射器构成垂直视角 FoV。激光雷达工作时，激光发射器随机械式旋转结构按照固定角度进行 360°旋转。根据 ToF 原理，可以计算出反射点在极坐标下的距离和角度，从而计算出点的x、y、z坐标，激光雷达极坐标到三维坐标的映射示意图如图 3-3 所示，假设其中一点 P，极坐标到三维坐标的转换关系如式子 3-1、3-2 和 3-3 表示。

$$x = R \times \cos\beta \times \sin\alpha \tag{3-1}$$

$$y = R \times \cos\beta \times \cos\alpha \tag{3-2}$$

$$z = R \times \sin\beta \tag{3-3}$$

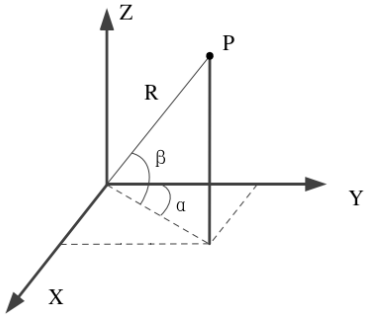


图 3-3 激光雷达极坐标到三维坐标的映射

激光雷达扫描的一帧点云图片如图 3-4 所示，激光雷达探测距离较远，点云包含几何结构信息，在自动驾驶汽车方面非常适用于感知车身外部的环境信息。

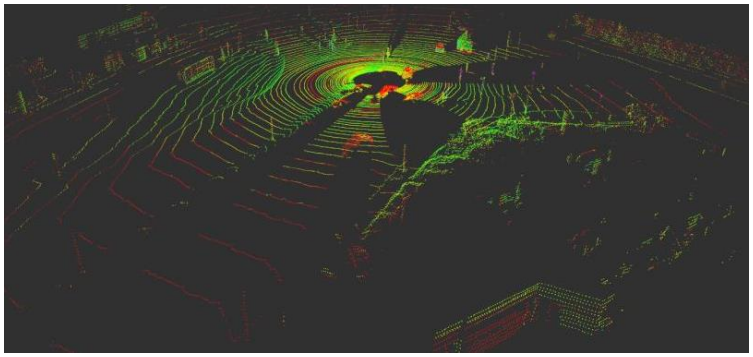


图 3-4 激光雷达扫描得到的点云

3.2 Pointpillars 模型介绍

激光雷达具备环境感知的能力，而基于激光雷达的车辆检测算法主要分为传统点云处理算法和深度学习算法两类。传统的点云处理算法通过对原始点云进行降采样、地面滤波、点云聚类等操作来实现障碍物识别功能，但这种算法的鲁棒性较差，不适用于实时目标检测。而基于深度学习的目标检测算法通过训练神经网络来直接预测原始点云的 3D 目标。目前，基于点云的 3D 检测框架主要可以分为两大类：一类是先将点云转换成密集表示的图片形式，然后像处理图片一样处理点云；另一类是先将点云处理成有序、密集的表示，即基于体素的检测器[64]。基于体素的检测器的思路是将点云转换成均匀分布的体素，然后采用类似于稀疏卷积网络的 CNN 方式进行处理。

基于体素（Voxel）的方法通常将输入的点云分割为规则的 3D Voxel 网格，然后使用具有稀疏 3D 卷积的编码器来学习跨多个级别的几何表示。在编码器之后，通过标准的 2D CNN 将特征馈送到检测头之前进行多尺度特征融合。相比之下，基于 pillar 的方法将 3D 点云投影到 BEV（Bird's Eye View）平面上，形成 2D 伪图像，然后直接在基于 2D CNN 的特征金字塔网络（FPN）上构建 Neck 网络以融合多尺度特征。

本文通过改进 OpenPCDet 框架算法中的 Pointpillars[65]，使其更适合用于车辆检测。Pointpillars 是一个源自工业界的模型，其核心思想是基于图像处理框架。它直接将点云从俯视图的角度划分为一系列的 Pillar（柱体），形成类似于图像的数据结构。然后，通过使用 2D 的检测框架进行特征提取和密集的框预测，得到检测框，从而使该模型在速度和精度方面达到了很好的平衡。

图 3-5 是 Pointpillars 的网络结构拓扑图，可以看出主要包含将输入的点云转换为稀疏的伪图像特征形式的支柱特征网络(Pillar Feature Net)、使用 2D 的 CNN 处理伪图像特征得到高维度特征的骨干网络(Backbone)、检测和回归 3D 边界框的检测头(Detection)三部分。

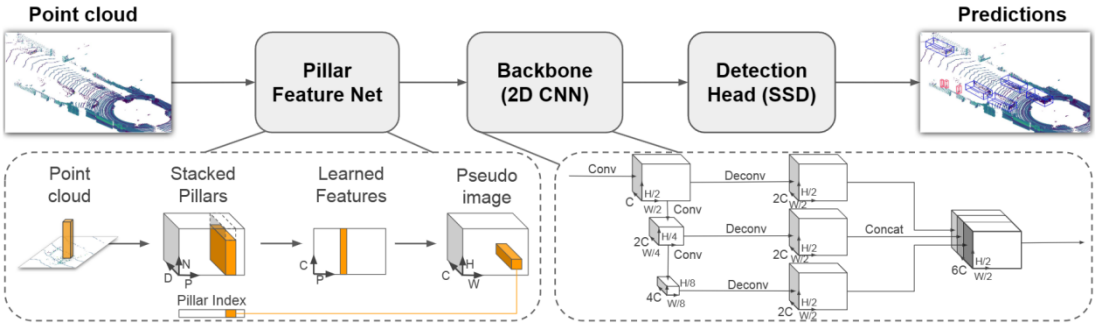


图 3-5 Pointpillars 网络结构拓扑

支柱特征网络直接以俯视图的形式获取输入的 3D 原始点云信息。在点云

中假设存在 N*3 个点的信息，所有的这些点都位于 kitti lidar 坐标系 X、Y、Z 中（单位为米，其中 X 向前，Y 向左，Z 向上）。然后，这些点会被分配到均匀大小的 X-Y 平面立方柱体中，这个立方柱体被称为 pillar。以图 3-6 KITTI 的第 23 帧图片为例，对应的点云和伪图像点云如图 3-7 所示。

图 3-6 KITTI 数据集第 23 帧图像

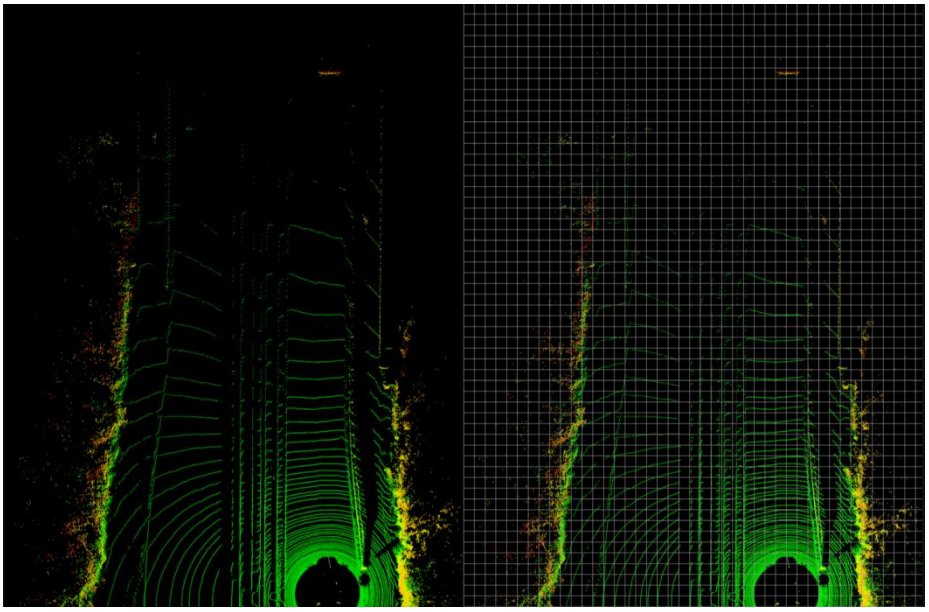


图 3-7 FOV 视角原始点云（左）和 Pillar 后的伪图像点云（右）

支柱特征网络负责将点云数据映射到一个规则的三维网格中，形成称为支柱（pillars）的栅格结构。这个栅格通常包括水平和垂直方向的栅格单元，每个栅格单元包含一组点云数据，表示为一个特征向量。支柱特征网络的主要目标是提取每个栅格单元中的特征信息，以捕捉物体的位置、形状和语义信息。为了实现这一目标，支柱特征网络使用卷积神经网络（CNN）或其他深度学习架构来处理每个栅格单元中的点云数据。该网络将每个栅格单元的点云数据转化为高维的特征向量，这些特征向量将传递给后续的骨干网络用于进一步处理。

骨干网络是 Pointpillars 的核心组件之一，它接收来自支柱特征网络的高维特征向量，通常以伪图像（pseudo-image）的形式组织。这些伪图像通常是多个通道的特征图。2D CNN 骨干网络被用来处理这些伪图像特征，类似于传统的图像处理任务。这允许模型在伪图像上执行卷积、池化和全连接等操作，以更好地捕获物体的相机特征。骨干网络通常负责两个主要任务：特征提取和尺度

自适应。特征提取阶段用于从伪图像中提取有关物体的重要特征，而尺度自适应阶段则允许模型在不同尺度上检测物体，以适应不同大小的目标。

检测头是 Pointpillars 的最后一步，它接收来自骨干网络的高维特征表示，并用于预测每个栅格单元中是否存在物体，以及物体的位置、形状和类别信息。通常，检测头包括两个子网络：分类头（Classification Head）和回归头（Regression Head）。分类头用于确定每个栅格单元是否包含物体，并预测物体的类别。回归头用于预测物体的精确位置和形状信息。 检测头通常包括全连接层、卷积层和其他神经网络组件，以从高维特征中提取有关物体的相关信息。

这三个组件共同工作，使 Pointpillars 能够从点云数据中有效地检测三维物体。支柱特征网络负责将点云数据进行栅格化和特征提取，骨干网络进一步处理特征并适应不同尺度，而检测头则进行最终的物体检测和边界框回归。这种多层次处理允许 Pointpillars 在不同场景中实现高精度的三维物体检测。

点云数据是 4 维度的数据包含 $(x，y，z，r)$，其中 x、y、z 是该点在点云中的坐标，r 代表了该点的反射强度（与物体材质和激光入射角度等有关）；并且将所有点放入每个 pillar 中的时候不需要像 voxel 那样考虑高度，可以将一个 pillar 理解为就是一个 z 轴上所有 voxel 组成在一起的。

在进行数据增强的时候，需要对 pillar 中的数据进行增强操作，需要将每个 pillar 中的点增加 5 个维度的数据，包含 x_c，y_c，z_c，x_p 和 y_p，其中下标 c 代表了每个点云到该 点所对应 pillar 中所有点平均值的偏移量，下标 p 代表了该点距离所在 pillar 中心点的 x，y 的偏移量。所有经过数据增强操作后每个点的维度是 9 维；包含了 x、y、z、x_c、y_c、z_c、x_p、y_p 和 r。

经过上述操作之后，就可以把原始的点云结构（N*3）变换成了（D，P，N），其中 D 代表了每个点云的特征维度，也就是每个点云 9 个特征，P 代表了所有非空的立方柱体，N 代表了每个 pillar 中最多会有多少个点。

在实现的过程中，每个 pillar 的长宽是 0.16 米，只会截取前视图的部分，进行训练，因为 KITTI 数据集的标注是根据 2 号相机进行标注的，所有 X 轴的负方向（即车的后方）是没有标注数据的，所以会截取掉后面的数据；同时为了保证检测的可靠性，距离太远的点，由于点云过于稀疏，也会被截取。所以在 Pointpillars 的实现中，点云空间选取范围 x、y、z 的最小值是=[0, -39.68,-3]；x、y、z 选取的最大值是[69.12, 39.68, 1]。其中每个 pillar 中的最大点云数量是 32，如果一个 pillar 中的点云数量超过 32，那么就会随机采样，选取 32 个点；如果一个 pillar 中的点云数量少于 32；那么会对这个 pillar 使用 0 样本填充。在经过映射后，就获得了一个（D，P，N）的张量；接下来这里使用了一个简化版的 pointnet 网络对点云的数据进行特征提取（即将这些点通过 MLP 升维，然

后跟着 BN 层和 Relu 激活层），得到一个（C，P，N）形状的张量，之后再使用 maxpool 操作提取每个 pillar 中最能代表该 pillar 的点。在经过上述操作编码后的点，需要重新放回到原来对应 pillar 的x、y位置上生成伪图象数据。

经过上面的映射操作，骨干网络将原来的 pillar 提取最大的数值后放回到相应的坐标后，就可以得到类似于图像的数据了；只有在有 pillar 非空的坐标处有提取的点云数据，其余地方都是 0 数据，所以得到的一个（batch_size，64，432，496）的张量还是很稀疏的。然后对得到的张量数据使用 2D 中的特征提取手段进行多尺度的特征提取和拼接融合。

PiontPillars 中的检测头采用了类似 SSD 的检测头设置，直接使用了一个网络训练车、人、自行车三个类别。因此在检测头的先验框设置上，一共有三个类别的先验框，每个先验框都有两个方向分别是 BEV 视角下的 0 度和 90 度，每个类别的先验框只有一种尺度信息；分别是车 [3.9，1.6，1.56]、人[0.8，0.6，1.73]、自行车[1.76，0.6，1.73]。在 anchor 匹配 GT 的过程中，使用的是 2D IOU 匹配方式，直接从生成的特征图也就是 BEV 视角进行匹配；不需要考虑高度信息。其中每个 anchor 都需要预测 7 个参数，分别是（x，y，z，w，l，h，θ），其中 x、y、z 预测一个 anchor 的中心坐标在点云中的位置， w、l、h 分别预测了一个 anchor 的长宽高数据，θ 预测了 box 的旋转角度。为了解决两个完全相反的 box 的角度预测问题，Pointpillars 的检测头还添加了一个基于 softmax 的方向分类来预测 box 的两个朝向信息。

每个三维检测框需要 7 个变量 $(x，y，z，w，l，h，\theta)$ 表示，其中 $(x，y，z)$ 表示检测框中心坐标，$(w，l，h)$表示检测框的尺度，(θ)表示检测框的朝向信息。在检测框的回归任务中要学习 7 个变量的偏移量，Pointpillars 产生了三个损失，分别是定位损失、分类损失和角度损失。地面真值（ground truth）和锚点（anchors ）之间的本地化回归残差定义为式子 3-4 和 3-5。

$$\begin{gathered}\Delta x=\frac{x^{gt}-x^{a}}{d^{a}},\Delta y=\frac{y^{gt}-y^{a}}{d^{a}},\Delta z=\frac{z^{gt}-z^{a}}{h^{a}}\\ \Delta w=\log\frac{w^{gt}}{w^{a}},\Delta l=\frac{l^{gt}}{l^{a}},\Delta h=\log\frac{h^{gt}}{h^{a}}\\ \Delta\theta=\sin(\theta^{gt}-\theta^{a})\end{gathered}\tag{3-4}$$

$$d^{a}=\sqrt{(w^{a})^{2}+(l^{a})^{2}}\tag{3-5}$$

其中，上标gt代表地面真值框（ground truth），上标α代表锚框（anchors）定位损失函数采用 Smooth L1 函数。

$$L_{loc}=\sum_{b\in(x,y,z,w,l,h,\theta)}SmoothL1(\Delta b)\tag{3-6}$$

对于目标分类任务，Pointpillars 采用了聚焦损失（Focal loss），聚焦损失是一种用于解决类别不平衡问题的损失函数，特别是在目标检测任务中，其中正样本（目标存在）通常远少于负样本（目标不存在）。在 Pointpillars 中聚焦损失的数学表达式如 3-7 所示。

$$L_{cls} = -\alpha_a (1-p^a)^{\gamma} \log p^a \tag{3-7}$$

其中 p^{α} 表示锚框的类别概率，$\alpha = 0.25$，$\gamma = 2$。

三维检测框需要考虑检测框的方向，要使用Softmax分类损失来学习目标的朝向，通常需要将目标朝向离散化为不同的类别，并使用 Softmax 损失函数来预测目标所属的类别，该损失函数用L_{dir}表示。

总的损失函数用式子 3-8 表示。

$$L = \frac{1}{N_{pos}} (\beta_{loc} L_{loc} + \beta_{cls} L_{cls} + \beta_{dir} L_{dir}) \tag{3-8}$$

其中N_{pos}是真阳性锚框的数量，$\beta_{loc} = 2$，$\beta_{cls} = 1$，$\beta_{dir} = 0.2$。

3.3 Pointpillars 模型改进

Pointpillars 是一种广泛用于处理点云数据的关键算法，用于检测和定位物体。为了应对不断变化的挑战和需求，学者们致力于不断改进 Pointpillars。

Xu 等提出了一种改进的基于支柱的 3D 目标检测网络，该网络具有两阶段支柱特征编码（Ts-PFE）模块，该模块同时考虑了 pillar 之间和 pillar 内部的关系特征[66]。这种新方法增强了模型识别数据局部结构和全局分布的能力，从而改善了遮挡和重叠场景中对象之间的区别，并最终减少了分割不足和错误检测问题。Song 等提出了一种新的检测管道，从点云中提取多分辨率柱级特征，以使检测方法更具尺度感知能力，使用空间注意机制来突出特征图中的对象激活，提高检测性能[67]。Duan 提出了环境感知框架 RGB-PVRCNN，利用 V2I 通信技术提高交叉路口自动驾驶车辆的环境意识[68]。

3.3.1 引入注意力机制

作为神经网络中的一项关键技术，注意力机制在计算机相机中得到了广泛应用，包括分类、检测和分割等任务。通过这种机制，网络的注意力集中在关键细节上，更多地关注感兴趣的区域或特征，从而提高检测精度和鲁棒性。注意力机制也被广泛应用于点云对象检测领域。常用的方法有自适应注意力、通道注意力、空间注意力及其组合。自适应注意力允许模型根据输入数据的不同部分分配不同程度的注意力。空间注意力允许网络对点云中特定位置的信息进行优先级排序，而渠道注意力意味着网络可以更多地关注特定渠道的特征。

本文的重点内容是检测识别汽车，为了使 Pointpillars 能够更有效地捕捉点

云中各部分之间的复杂关系，从而增强特征表示的能力，本文引入自适应注意力机制。自适应注意力机制是一种动态调整注意力权重的方法。与传统的固定注意力计算方法不同，自适应注意力机制可以根据输入数据和任务需求动态改变权重，以便更有效地捕捉关键信息。

自适应注意力机制通过引入可训练的参数来学习如何自动调整注意力权重。这些参数可能与输入的特征、全局信息或任务目标有关。通过在训练过程中优化这些参数，模型可以学会自适应地关注不同部分的信息。

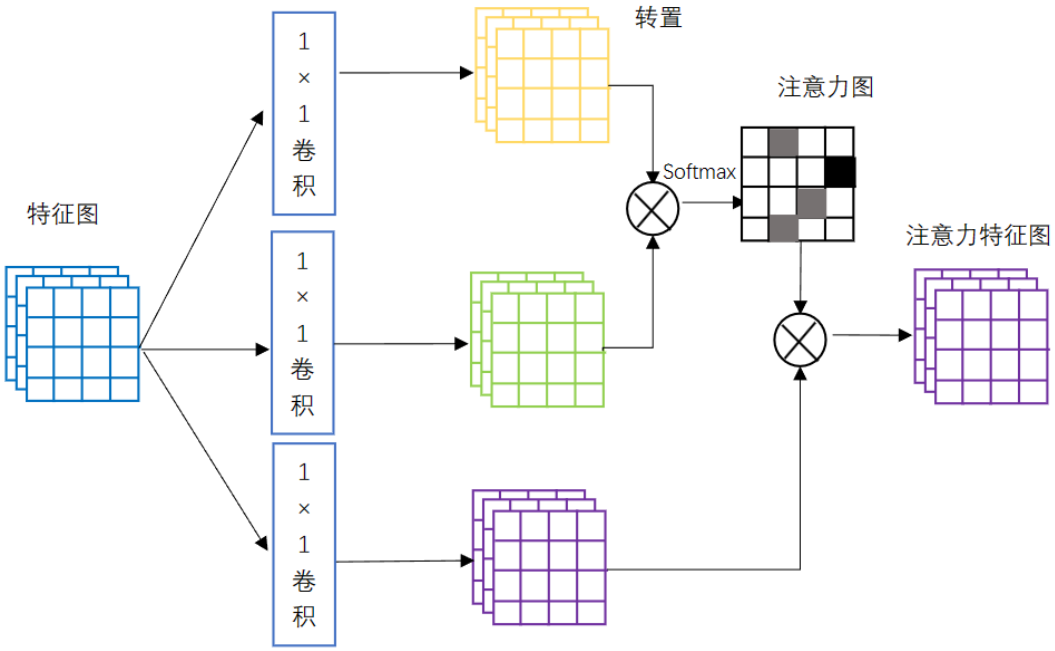


图 3-8 自适应注意力网络

自适应注意力机制的网络结构图如图 3-8 所示，用 3 个 1×1 卷积对输入的原始特征图分别进行 3 次处理得到 3 个新的特征图，新的 3 个特征图分别记作 Q(query)、K(key)、V(value)。利用多层感知机、拼接和点积等相似度函数对查询（Q）和键（K）进行相似度分析，得到它们之间的关系矩阵。通过 Softmax 函数将关系矩阵转换为注意力矩阵，将该矩阵与值（V）进行加权求和，从而生成注意力特征图的输出。

Pillar 的数据可以用数学公式表示为式子 3-9。

$$P = \{P_i = [f_i^1 \cdots, \cdots f_i^j] \epsilon R^9\}_{i=1\cdots n, j=1\cdots d} \tag{3-9}$$

式中：P 代表每一帧点云的特征矩阵，n 代表点柱的数量，d 代表每个点柱的数据维度。

首先将 P 输入利用三个 1×1 的多层感知机制，对每个点柱进行映射到高维特征空间操作，得到的三个新点柱特征矩阵为Q_p、K_p以及V_p，其公式表示为式 3-10。

$$Q_p, K_p, V_p = MLP_q(P), MLP_k(P) Q_p \epsilon R^{m\times n}, K_p \epsilon R^{m\times n}, V_p \epsilon R^{m\times n} \tag{3-10}$$

接着需要求解Q_p与K_p之间的关系矩阵，首先将K_p进行转置，对K_p^T和Q_p作乘积计算其相似性，得到的关系矩阵 A 用式 3-11 表示。

$$A = Q_p \times K_p^T, A \epsilon R^{n \times n} \tag{3-11}$$

利用 Softmax 对相关性矩阵 A 进行归一化，即得到式 3-12。

$$A_n = soft\max(A) \tag{3-12}$$

最后将A_n与V_p做矩阵乘积操作，如下面公式所示，与原始的pi特征进行相加，从而完成了将一个点柱的特征用所有相关点柱进行信息增强的目的。

$$R_s = A_n \times V_p + P, R_s \epsilon R^{n \times d} \tag{3-13}$$

最终得到经过点柱特征增强后的特征矩阵R_s。

2D 卷积坍缩模块在 Pointpillars 目标检测中的作用是将来自三维体素表示的特征映射到二维的俯视图（BEV）平面上，以便进行后续的目标检测和定位。这个模块有助于将三维空间中的目标信息转化为适用于二维卷积神经网络（CNN）的特征表示。

（1）输入体素特征：输入到 2D 卷积坍缩模块的是体素特征，通常表示为（B，C，Z，Y，X），其中 B 是批量大小，C 是通道数，Z 是高度通道数，Y 是垂直维度上的分辨率，X 是水平维度上的分辨率。

（2）降维和展平：首先，模块会将高度通道进行降维，然后将体素特征展平为（B，C*Z，Y，X）的形状，将 3D 特征转化为 2D 特征。

（3）卷积操作：使用定义的 2D 卷积块（如 BasicBlock2D）对展平后的特征进行卷积操作，从（B，C*Z，Y，X）的特征映射得到（B，C，Y，X）的 BEV 特征表示。

（4）输出 BEV 特征：最终的输出是经过卷积和特征加强的 BEV 特征表示，形状为（B，C，Y，X）。这些特征可以传递到下一阶段的网络，如 3D CNN，用于目标检测和定位。

在 2D 卷积坍缩模块引入注意力模块后，处理后的 BEV 特征 bev_features 输入到自适应注意力模块 attention 中，获取增强的特征表示。通过引入注意力机制，模型可以自动学习并关注汽车形态的重要特性。这种对细节的关注有助于减少误检和漏检，提高整体检测准确性。由于自适应注意力机制可以根据任务动态调整权重，因此有助于模型更好地适应未见过的场景和汽车类型，从而提高泛化能力。

3.3.2 激活函数的替换

激活函数在神经网络中起着关键作用，它们引入了非线性，使得神经网络能够学习和执行复杂的任务，捕捉非线性关系。没有激活函数，神经网络将仅

仅是线性变换的组合，限制了模型的表示能力。Pointpillars 将 ReLU 函数作为激活函数， ReLU（Rectified Linear Activation）激活函数用式 3-14 定义。

$$f(x) = \max(0, x) \tag{3-14}$$

如果输入值大于 0，它就返回输入值；如果小于或等于 0，它就返回 0。

ReLU 的主要优点是计算效率高，并且有助于减轻梯度消失问题。ReLU 的输出对于负输入是0，可能会导致一些神经元在训练过程中永远不会被激活，称为“死神经元”问题。本文将 ReLU 激活函数改为 LeakyReLU 激活函数，LeakyReLU 用式 3-15 定义。

$$f(x) = \begin{cases} x, \ if \ x > 0 \\ ax, \ if \ x \leqslant 0 \end{cases} \tag{3-15}$$

其中α是一个小的正常数，通常设置为 0.01。

3.4 训练模型

本研究采用了 KITTI 数据集进行实验评估 。实验基于 OpenPCDet 框架实现，这个框架方便应用 KITTI 数据集和改进 Pointpillars 算法实验。本文实验用到的硬件环境为 ：GPU 型号为 GEFORCE RTX 3060 Ultra oc，32G 显存，12th Gen Intel(R) Core(TM) i7-12700F 2.10 GHz，2.0T 硬盘；软件环境为：Ubuntu 20.04 LTS，Python 3.7，Cuda 11.3，Pytorch 1.10。

表 3-1 Pointpillars 的部分参数

参数	含义	
输入体素尺寸（Input Voxel Size）	0.16m × 0.16m × 4.0m	
柱状表示参数（Pillar Representation Parameters）	高度	0.2m
	宽度	0.2m
	深度	4.0m
Voxel 特征提取网络（Voxel Feature Extractor Network）	包含三个卷积层的 Voxel 特征提取网络	
稀疏 3D 卷积（Sparse 3D Convolution）	卷积核大小	3×3×3
	卷积核数量	64
锚定框参数（Anchor Box Parameters）	多个不同大小和宽高比的锚定框	
学习率（Learning Rate）	一个初始学习率，并采用学习率衰减策略	
批量大小（Batch Size）	2	
正则化参数（Regularization Paramaters）	L2 正则化	0.01
	丢弃率	0.3

模型的参数是通过训练过程中学习的，每个参数都对模型的输出产生一定的影响。一些参数可能对模型的性能和准确性有更大的影响，而另一些参数可能对模型的稳定性或泛化能力有更大的影响。因此，对于特定的模型和任务，

评估参数的重要性可能需要进行详尽的实验和分析。Pointpillars 的部分参数如表 3-1 所示。

3.5 消融实验

本文的训练数据集与 Pointpillars 官方训练数据集保持一致，仍然是在 KITTI 数据集上训练，划分 KITTI 数据集之后训练 1400 个 epoch，分别训练得到原始训练的权重，引入自适应注意力机制的权重，替换激活函数的权重和同时引入自适应注意力机制和替换激活函数的权重，对它们进行定性定量分析。

学习率是深度学习训练中的一个重要超参数，它控制模型参数在每次训练迭代中更新的幅度。学习率越大，模型参数更新得越快，但可能会导致训练不稳定；学习率越小，模型参数更新得越慢，但可能需要更长的时间才能收敛到最优解，Pointpillars 的学习率如图 3-9 所示。

TensorBoard 可以实时显示训练损失、准确率、学习率等指标的变化趋势，帮助开发者了解模型的训练进展情况。在下图中，“meta_data/learning_rate”表示当前训练迭代中使用的学习率的数值。这个值会随着训练的进行而动态调整，以确保训练过程收敛到最佳模型。

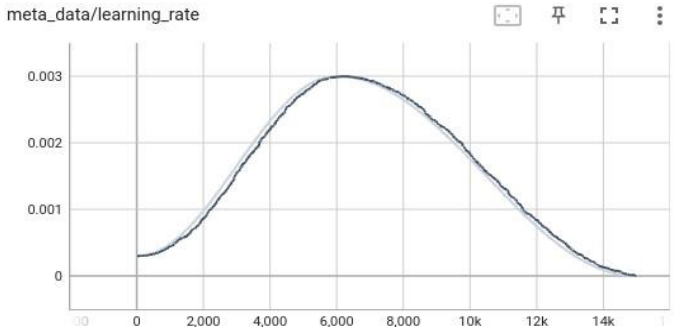


图 3-9 Pointpillars 学习率

改变 Pointpillars 模型，通常需要重新评估和调整学习率。但是引入注意力机制是一种模型结构的改进，通常不会直接影响学习率的选择。学习率的选择通常是在训练模型之前通过实验和验证来确定的，以找到合适的学习率值，以便在训练过程中获得最佳性能。由于 Leaky ReLU 是 ReLU 的变种，它在输入小于零时具有一个小的负数斜率，而在输入大于零时行为与 ReLU 相同。因此，它们之间具有一定的相似性，不会引入太大的变化。这里使用与原来的学习率相同的初始学习率。使用与原来相同的学习率可以使初始训练更加稳定，因为模型在新的激活函数下可能需要适应较小的变化。

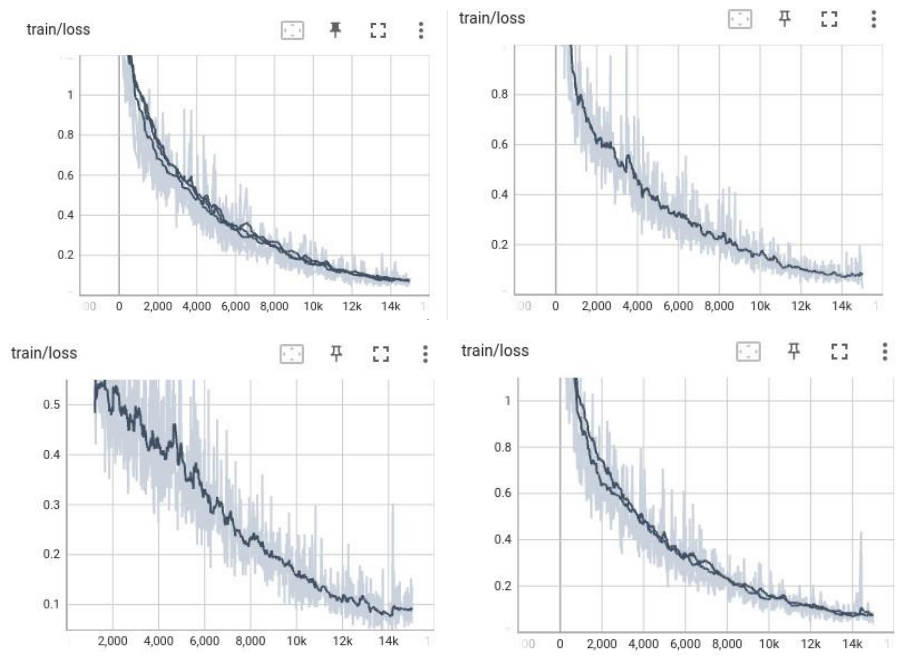


图 3-10 训练损失结果

图 3-10 分别是 Pointpillars 原模型、单独引入注意力机制、单独改进激活函数、同时引入注意力机制和改进激活函数之后训练 1400 轮得到的训练损失结果图，单独引进损失函数和单独改进激活函数的损失在训练 10k 次之前都有明显的振荡，同时引进注意力机制与改进激活函数相较于其他两种情况更稳定，最终的损失也处于合理的范围。

3.5.1 消融实验结果分析

（1）精度结果分析

PointPillars 在 Ubuntu20.04 上的训练，对同一数据集训练 1400 个 epoch 得到的训练结果如表 3-2、表 3-3 所示。

表 3-2 Pointpillars 消融实验结果

Pointpillars 改进内容	精度提升	Map Car/IOU=0.5/3dbox		
		简单	中等	困难
原模型	/	84.92	75.32	69.64
引入自适应注意力机制	0.86%	85.65	77.68	69.98
改进激活函数	0.11%	85.01	75.77	69.33
引入自适应注意力机制和改进函数	1.90%	86.54	77.96	70.21

表 3-3 Pointpillars 消融实验结果

Pointpillars 改进内容	精度提升	Map Car/IOU=0.7/3dbox		
		简单	中等	困难
原模型	/	57.96	53.63	51.69
引入自适应注意力机制	0.29%	58.13	54.41	51.83
改进激活函数	0.15%	58.05	53.66	51.79
引入自适应注意力机制和改进函数	2.10%	59.16	54.85	52.03

引入注意力机制和改进激活函数都可以提升 Pointpillars 检测汽车的精度，单纯的引入自适应注意力机制和改进激活函数精度提升并不明显，但是对原模型同时改进这两部分对于汽车的检测精度可以提升 2%，这在点云检测时可以更准确的检测出属于汽车的点云并绘制出 3 维包络框。

（2）速度结果分析

在测量推理时间之前，对模型进行预热，以确保模型的参数被加载到 GPU 内存中，并且 GPU 已经达到了最佳状态。在相同的硬件环境下进行完全相同数据集推理的实验对比，得到的平均推理速度如表 3-4 所示。

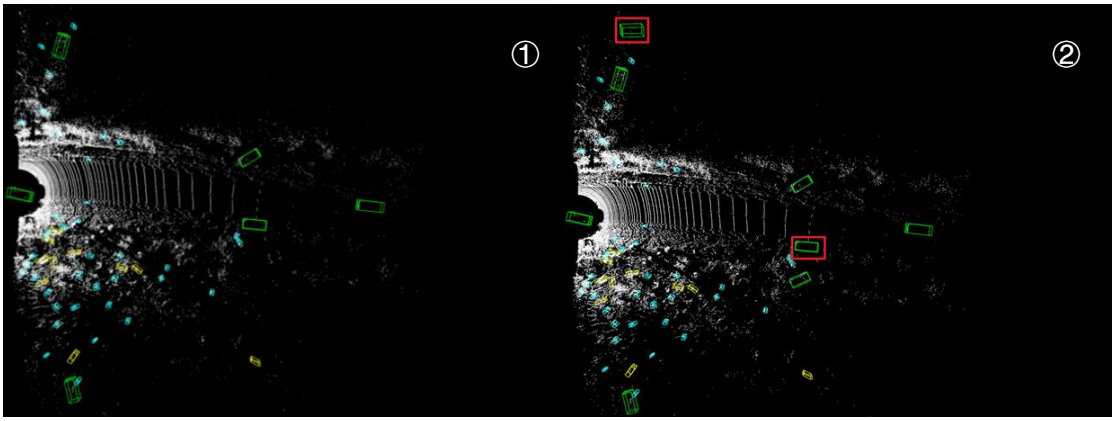
表 3-4 中原模型的平均推理速度在本文实验设备上能达到 75 FPS/S，单独引入注意力及之后的平均推理速度相较原模型减少了 5 FPS/S，单独改进激活函数后的算法模型相较原模型平均推理速度提高了 2 FPS/S，同时做这两方面的改进相较原模型的推理速度降低了 2 FPS/S，73 FPS/S 相较 75 FPS/S 的推理速度略微降低，但在本质上，同时做本文两方面的改进对模型的平均推理速度影响不大。

表 3-4 Pointpillars 消融实验平均推理速度

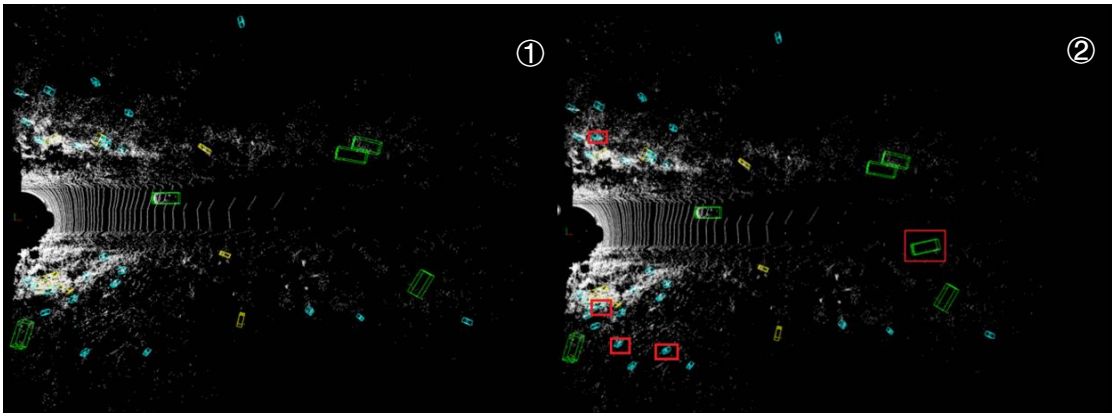
Pointpillars 改进内容	平均推理速度(FPS/S)
原模型	75
引入自适应注意力机制	70
改进激活函数	77
引入自适应注意力机制和改进函数	73

3.5.2 可视化结果与分析

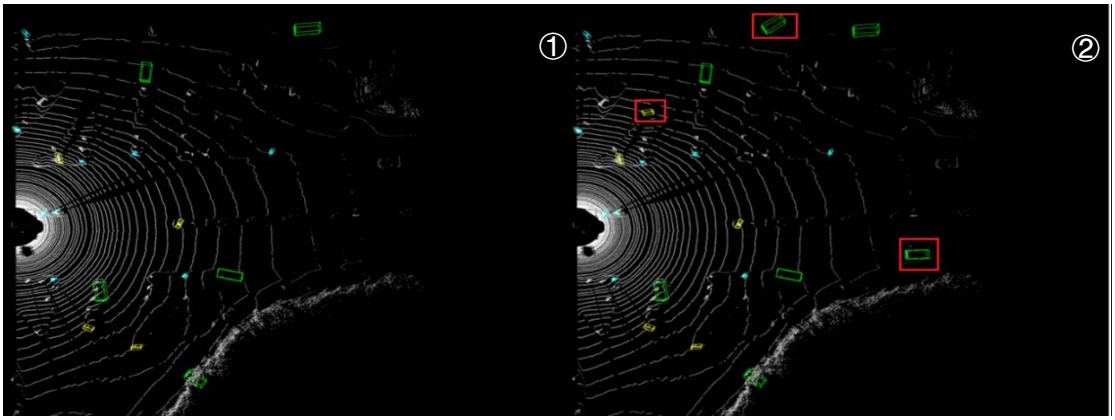
对训练完成后的 Pointpillars 分别对同一帧点云 bin 文件识别得到的效果如图 3-11 所示。



（a）



(b)



（c）

图 3-11 Pointpillars 对同一帧点云文件识别结果

图 3-11 中（a）、（b）、（c）是对 KITTI 数据集中不同点云 bin 文件的检测结果对比图，左边是 Pointpillars 原模型的检测结果，右边是改进 Pointpillars 模型的检测结果，右图红色框是原模型没有检测出来的目标。检测结果表明，引进注意力机制和改进激活函数能提升 Pointpillars 的检测精度，减少漏检率，点云检测精度的提升对实现相机与激光雷达融合识别汽车研究起着重要作用。

3.6 本章小结

本章基于改进 Pointpillars 实现汽车点云的识别。通过引入自适应注意力机

制和替换激活函数改进 Pointpillars 模型提升检测车辆点云的能力。通过消融实验可得，在 Car/IOU=0.7/3dbox 和 Car/IOU=0.5/3dbox 的情况下本文改进算法模型相较原模型的检测精度均能提升 2%。KITTI 数据集的可视化结果显示，在检测同一帧点云时，本文改进算法模型的漏检率比原模型明显降低。

第 4 章 相机与激光雷达联合标定

相机和激光雷达作为车辆感知传感器得到外界的信息数据后，需要对两者的数据进行融合，以判断前方是否存在汽车。时间对齐和空间对齐是融合不可或缺的重要一环。时间对齐是指相机采集的数据与激光雷达采集的数据要在同一时间戳下，这样在信息融合时才能避免出现时间不一致导致的信息不匹配的问题；空间对齐是指相机和激光雷达不仅在汽车要有相对汽车确定的物理坐标关系，而且要保证像素坐标系与三维坐标系之间能相互转化，确保能准确检测外界环境中的车辆。相机与激光雷达标定完成之后，通过对两者的信息融合，即可检测前方车辆。

4.1 时间对齐

搭载相机和激光雷达的自动驾驶汽车在道路行驶时，由于两种传感器是独立的且采集数据的频率不同。虽然能对两者进行物理上的空间对齐将其坐标关联起来，但是采样频率的不同会导致两者在同一时间戳下得到外界信息的数据不同，即点云信息与图像信息不匹配，融合的精度将受到极大的影响。所以在对两者进行信息融合之前必须先将两种传感器的时间对齐，保证每一帧图像对应当前时刻的点云信息，提高相机与激光雷达信息融合的精度[69]。相机与激光雷达的时间对齐可以通过下面的方法实现。

（1）外部同步信号：一种可行的方法是使用外部同步信号，如 GPS 信号或外部时钟源。这些信号可以提供高精度的时间信息，以确保相机和激光雷达的数据采集是同步的。这要求传感器硬件支持外部信号接口，以接收来自同步源的信号。

（2）内部硬件时钟：许多相机和激光雷达设备内置了高精度的硬件时钟。您可以配置这些设备，使它们在硬件级别上同步，以确保它们的时间戳是一致的。这种方法通常提供了可靠的时间同步，但需要确保设备正确配置。

（3）时间戳校准：如果传感器没有内部同步功能，可以使用时间戳校准方法。这涉及记录每个传感器数据的时间戳，并在数据后处理阶段进行校准，以对齐数据。这需要在实际数据采集期间测量延迟并校准时间戳，以确保数据对齐。

（4）使用时间同步协议：通过使用时间同步协议，如网络时间协议（NTP）或精确时间协议（PTP），可以通过网络连接同步传感器的时钟。这需要网络连接并确保网络延迟较小，以确保高精度的时间同步。这种方法特别适用于分布

式传感器系统。

（5）软件同步：可以在数据后处理阶段使用软件同步方法。通过分析数据集中的时间戳信息，可以将相机和激光雷达数据对齐到一个共同的时间基准。这通常需要复杂的数据处理算法，但可以在不同传感器之间实现时间同步。

相机的工作频率为60Hz，激光雷达的工作频率为10Hz，激光雷达的工作频率低于相机。数据接收系统在接受图像数据和点云数据时会附上每一时刻的时间戳，因此本文基于时间戳的方式对两种传感器进行时间对齐，由于雷达的工作频率较低，所以选取激光雷达的时间戳作为时间对齐的基准，缓存相机采集的图像数据，当系统接收到点云信息时，从缓存信息中找到与该点云信息时间戳最接近的图像信息，然后将两者的信息送入预先写好的融合处理函数处理，完成相机与激光雷达的时间对齐。两种传感器的时间对齐示意图如图4-1所示。

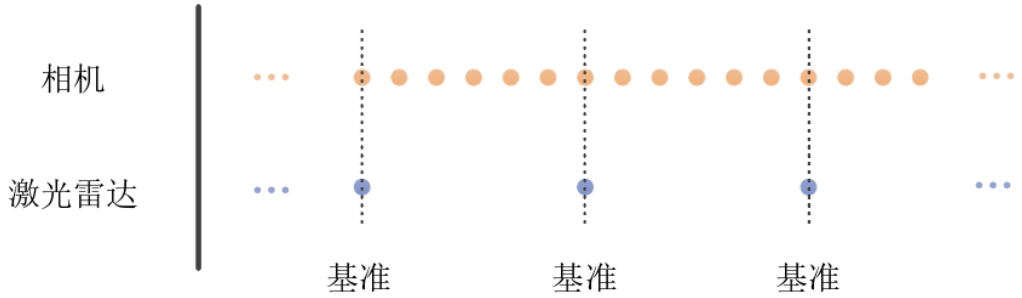


图4-1 相机与激光雷达时间对齐示意图

4.2 空间对齐

当涉及到相机和激光雷达的空间同步时，本文采用了一种高度精确且专业的方法。具体而言，我们通过将激光雷达的坐标系映射到相机的坐标系来实现这两种传感器之间的空间同步。这个过程不仅仅是一种对齐方式，它代表了一个精细的过程，旨在确保激光雷达捕获的点云数据与相机捕获的图像数据在三维空间中具有高度的一致性。

4.2.1 激光雷达坐标系

激光雷达扫描得到的点云以激光雷达为中心分布，由于和相机安装位置的不同，需要对两种传感器的坐标系转换，转换关系如图4-2所示。

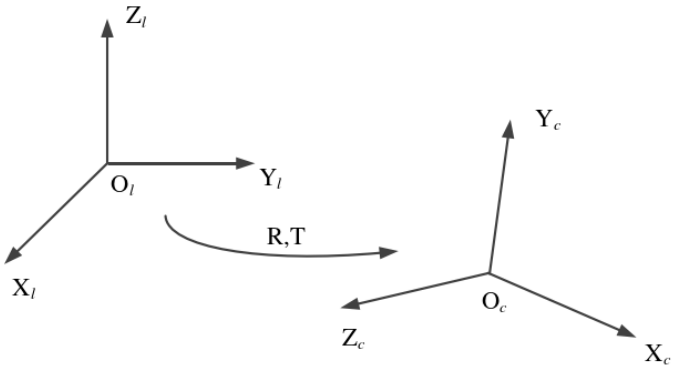


图 4-2 传感器坐标系的转换

图 4-2 可以看出，激光雷达坐标系首先进行旋转操作，以O_l为旋转原点，将激光雷达的三维坐标X_l、Y_l、Z_l与相机的三维坐标X_c、Y_c、Z_c一一对应，然后以O_l为中心，将O_l平移至O_c，此时坐标系$O_lX_lY_lZ_l$与坐标系$O_cX_cY_cZ_c$完全重合。坐标系的旋转包含绕X轴、Y轴、Z轴三部分，以绕Z轴旋转为例进行说明，旋转过程如图 4-3 所示。

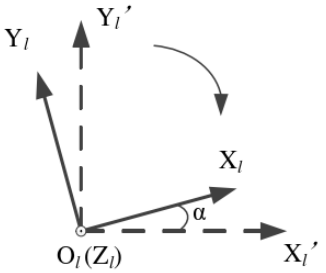


图 4-3 X、Y轴绕Z轴转动示意图

图 4-3 中首先固定Z轴，再将X_l、Y_l旋转至X_l'、Y_l'，此时顺时针旋转的角度记为α，假定顺时针转动的方向为正，旋转公式如式 4-1 所示。

$$\begin{aligned} X_l' &= X_l\cos\alpha - Y_l\sin\alpha \\ Y_l' &= X_l\sin\alpha + Y_l\cos\alpha \end{aligned} \tag{4-1}$$

将上述式子写成矩阵的形式表示为 4-2。

$$\begin{bmatrix} X_l' \\ Y_l' \\ Z_l' \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_l \\ Y_l \\ Z_l \end{bmatrix} \tag{4-2}$$

同理，假定绕Y轴顺时针旋转角度为β，绕X轴顺时针旋转角度为γ，写成矩阵形式如式 4-3 和式 4-4 所示。

$$\begin{bmatrix} X_l' \\ Y_l' \\ Z_l' \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} X_l \\ Y_l \\ Z_l \end{bmatrix} \tag{4-3}$$

$$\begin{bmatrix} X_l' \\ Y_l' \\ Z_l' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix} \begin{bmatrix} X_l \\ Y_l \\ Z_l \end{bmatrix} \tag{4-4}$$

对X轴、Y轴、Z轴的三个旋转矩阵相乘，就可以得到激光雷达坐标系转换

为相机坐标系的旋转矩阵 R，具体的转换过程如式 4-5 所示。

$$\begin{aligned} R &= \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix} \\ &= \begin{bmatrix} \cos\alpha\cos\beta & -\sin\alpha\cos\gamma+\cos\alpha\sin\beta\sin\gamma & \sin\alpha\sin\gamma+\cos\alpha\sin\beta\cos\gamma \\ \sin\alpha\cos\beta & \cos\alpha\cos\gamma+\sin\alpha\sin\beta\sin\gamma & -\cos\alpha\sin\gamma+\sin\alpha\sin\beta\cos\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix} \end{aligned} \tag{4-5}$$

激光雷达坐标系旋转之后还不能与相机坐标系完全重合，因为两者的原点不相同，所以需要对旋转后的激光雷达坐标系进行平移变换，假设 X 轴、Y 轴、Z 轴三个坐标轴方向上的平移量为 Δx、Δy、Δz，此时的平移矩阵可以用式 4-6 表示。

$$T = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \tag{4-6}$$

激光雷达坐标系旋转和平移之后与相机坐标系完全重合，转换后的坐标矩阵为式 4-7。

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = R \begin{bmatrix} x_l \\ y_l \\ z_l \end{bmatrix} + T \tag{4-7}$$

4.2.2 相机坐标系

相机采集的数据来源是三维世界，最终以像素的方式成像。相机由相机坐标系 $O_cX_cY_cZ_c$、图像坐标系 $O_iX_iY_iZ_i$ 和像素坐标系 UOV 三种坐标系，三者之间的关系如下图所示，相机获取的外界三维数据先转换到图像坐标系再转换到像素坐标系然后得到成像结果，内参矩阵是相机坐标系转换到像素坐标系的重要一环。

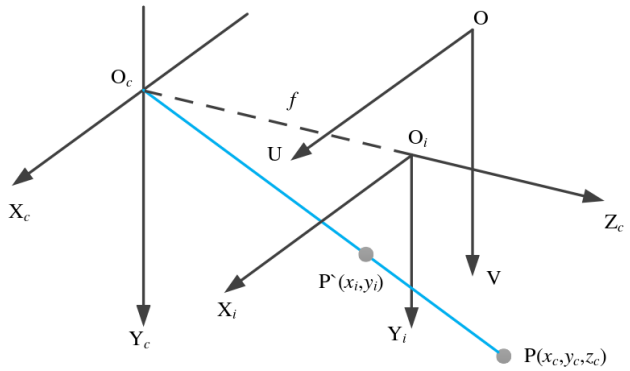


图 4-4 相机的坐标系转换示意图

图 4-4 O_c 是相机的光心，f 是相机的焦距，假设相机坐标系中的一点 $P(x_c, y_c, z_c)$ 在图像坐标系中的投影为点 $P'(x_i, y_i)$，根据三角形相似的数学原

理可表示为式 4-8。

$$\frac{x_i}{x_c}=\frac{y_i}{y_c}=\frac{f}{z_c} \tag{4-8}$$

变换该公式可得式 4-9 和式 4-10。

$$x_i=f\frac{x_c}{z_c} \tag{4-9}$$

$$y_i=f\frac{y_c}{z_c} \tag{4-10}$$

整理可得相机坐标系转换为图像坐标系的转换矩阵为式 4-11。

$$Z_c\begin{bmatrix}X_i\\Y_i\\1\end{bmatrix}=\begin{bmatrix}f&0&0\\0&f&0\\0&0&1\end{bmatrix}\begin{bmatrix}X_c\\Y_c\\Z_c\end{bmatrix} \tag{4-11}$$

图像坐标系与像素坐标系单位不同，两者转换之前要进行单位换算，然后将坐标原点平移至与像素坐标原点重合，其转换公式如式 4-12 和式 4-13 所示。

$$u=\frac{x_i}{dx}+u_0 \tag{4-12}$$

$$v=\frac{y_i}{dy}+v_0 \tag{4-13}$$

其中， dx、dy 分别代表每一行和每一列一个像素占的长度单位是多少，μ_0、v_0 分别代表图像坐标与像素坐标之间相差的横向与纵向的像素数。因此图像坐标系与像素坐标系之间的转换关系为式 4-14。

$$\begin{bmatrix}U\\V\\1\end{bmatrix}=\begin{bmatrix}\frac{1}{dx}&0&u_0\\0&\frac{1}{dy}&v_0\\0&0&1\end{bmatrix}\begin{bmatrix}X_i\\Y_i\\1\end{bmatrix} \tag{4-14}$$

整理可得相机坐标系和像素坐标系之间的转换关系为式 4-15。

$$Z_c\begin{bmatrix}U\\V\\1\end{bmatrix}=\begin{bmatrix}\frac{1}{dx}&0&u_0\\0&\frac{1}{dy}&v_0\\0&0&1\end{bmatrix}\begin{bmatrix}f&0&0\\0&f&0\\0&0&1\end{bmatrix}\begin{bmatrix}X_c\\Y_c\\Z_c\end{bmatrix}=\begin{bmatrix}\frac{f}{dx}&0&u_0\\0&\frac{f}{dy}&v_0\\0&0&1\end{bmatrix}\begin{bmatrix}X_c\\Y_c\\Z_c\end{bmatrix}=G\begin{bmatrix}X_c\\Y_c\\Z_c\end{bmatrix} \tag{4-15}$$

上述式子中，G 为相机的内参矩阵。

得到相机的内参矩阵之后，将转换到相机坐标系的激光雷达坐标系进一步转换到像素坐标系，实现激光雷达与相机的空间同步，具体的转换矩阵如式 4-

16 所示。

$$Z_c\begin{bmatrix}U\\V\\1\end{bmatrix}=\begin{bmatrix}\frac{1}{dx}&0&u_0\\0&\frac{1}{dy}&v_0\\0&0&1\end{bmatrix}\left(R\begin{bmatrix}x_l\\y_l\\z_l\end{bmatrix}+T\right)\begin{bmatrix}x_l\\y_l\\z_l\\1\end{bmatrix}=\begin{bmatrix}\frac{f}{dx}&0&u_0&0\\0&\frac{f}{dy}&v_0&0\\0&0&1&0\end{bmatrix}\begin{bmatrix}R&T\\0_{1\times3}&1\end{bmatrix}\begin{bmatrix}x_l\\y_l\\z_l\\1\end{bmatrix} \quad (4\text{-}16)$$

4.3 搭建联合标定实验平台

本文的实验平台分为硬件实验平台和软件实验平台，硬件实验平台主要是搭载相机与激光雷达的智能驾驶试验小车，主要由线控底盘组成，包括线控驱动系统、电池管理系统、车架车身系统、线控转向系统，整车控制 VCU 等，智能驾驶试验小车的组成部分如图 4-5 所示。

图 4-5 智能驾驶试验小车俯视图

为了支持硬件实验平台的智能驾驶试验小车，设计并实现了相应的软件实验平台。该软件平台负责实时数据处理、传感器数据融合。智能驾驶试验小车用到的部分计算机软件版本如表 4-1 所示。

表 4-1 计算机软件版本

软件名称	软件版本
Linux 系统	Ubuntu 20.04
ROS 系统	Noetic
Python	3.7
Cuda	10.2
Cudnn	8.0.5
Pytorch	1.7.0
OpenCV	3.4.2

4.4 相机与激光雷达联合标定

本文相机与激光雷达融合识别框架部署应用在智能驾驶试验小车，智能驾

驶试验小车主要用于自动驾驶的环境感知、规划控制、智能决策方面。该小车搭载了一颗速腾聚创 16 线激光雷达与一个海康威视 2K 超清 USB 摄像机，两者在小车上的相对位置如图 4-6 所示。

图 4-6 相机与激光雷达的相对位置

海康威视 2K 超清 USB 摄像头图像清晰、细腻，采用低畸变镜头，适用于广角场景低照度， 0.1 Lux 支持自动电子增益功能，亮度自适应，实验用的海康威视 2K 摄像头部分参数如表 4-2 所示。

表 4-2 海康威视 2K 摄像头参数

尺寸	帧率	分辨率	焦距	像素尺寸	视角
80.16×48.6×42.18 mm	60 FPS	2560×1440	3.6 mm	2.2×2.2 um	D:88°H:50°V:79°

速腾聚创 16 线激光雷达主要面向自动驾驶汽车环境感知领域，其内置 16 组激光元器件，同时发射和接收高频率激光束，360°旋转能提供精确的三维空间点云数据。速腾聚创 16 线激光雷达的主要参数如表 4-3 所示。

表 4-3 速腾聚创 16 线激光雷达部分参数

名称	参数
线数	16
测距能力	150m(80m@10%NIST)

水平视场角	360°
水平角分辨率	0.2°
帧率	10Hz
出点数	300000pts/s
激光波长	905nm
精度	±2cm
垂直视场角	30°
垂直角分辨率	2.0°
转速	600rpm
以太网输出	100Mbps

4.4.1 相机内参标定

在进行相机与激光雷达联合标定之前，单独标定相机是必要的前置步骤。相机标定的独立性为联合标定提供了基础，通过此过程获得的相机内外参数以及建立的相机模型为后续联合标定提供了关键信息。通过单独标定相机，能够准确获得相机的内参（如焦距、主点等）和外参（位置和朝向）[70]。这使得整个联合标定系统更加可靠，减少了复杂性。通过这一独立而深入的相机标定过程，为后续与激光雷达的联合标定提供了可信的基础，确保系统能够精确理解相机与激光雷达之间的相对关系，为多传感器融合应用提供可靠的基础。

在标定相机内参之前，需要准备棋盘格标定板，本实验采用纸张尺寸大小为 A2 尺寸的标定板，通过 Python 脚本制作数目为 12×9、边长为 45mm 的棋盘格标定文件，打印出来测量其实际尺寸与设计尺寸误差小于 0.5mm，满足相机标定板的要求。

对 USB 接口的相机标定采用 calibration_camera_lidar 标定工具箱，由于标定工具箱标定的是棋盘格的角点，把角点的数目改为 11×8 之后打开相机和标定工具箱，棋盘格标定板放在相机视野内合适的位置，使相机能尽可能多的采集到棋盘格标定板上的角点。不断变换位置让标定箱采集足够的数据后 CALIBRATE 按钮由灰色变为蓝色，按下此键，点击 SAVE 保存之后查看标定结果。

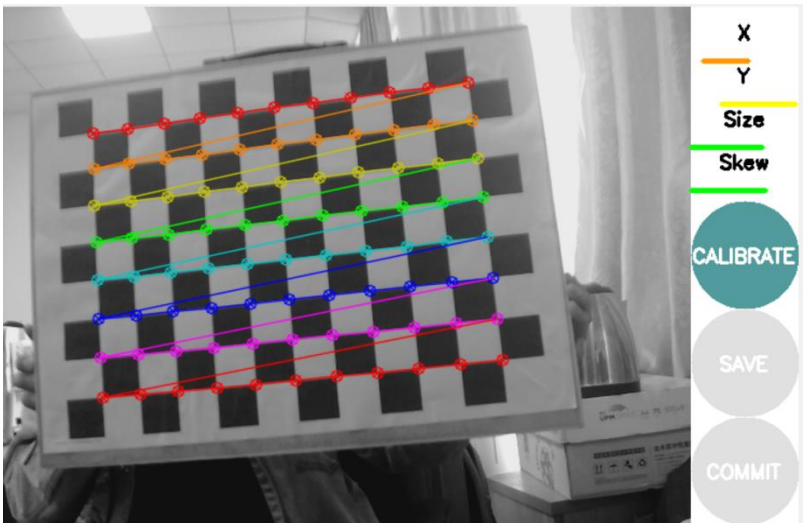


图 4-7 相机内参标定过程图

图 4-7 中 X、Y、Size 和 Skew 这四个参数分别代表了标定板在图像中的水平位置、垂直位置、大小以及倾斜程度。标定完成之后，查看相机的内参矩阵为：

$$G = \begin{bmatrix} 773.409058 & 0.000000 & 344.814996 \\ 0.000000 & 774.160256 & 228.755456 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

4.4.2 相机与激光雷达联合标定

相机内参标定完成之后，需要确定相机与激光雷达在三维世界的相对位置，即外参的确定。Autoware 是一款基于 Ros 开源的自动驾驶框架，1.10 版本之前内置相机与激光雷达联合标定工具箱，之后的版本取消了这个工具箱。由于实验安装的 Autoware 为 1.13 版本，单独下载配置了 Calibration Toolkit，两者同时运行实现联合标定。

Autoware 使用离线标定的方式对相机和激光雷达联合标定，固定好两者在三维空间的相对位置，启动激光雷达调试出点云图之后再启动 USB 相机，把准备好的棋盘格标定板放在相机视野内且能被激光雷达扫面到，通过 Ros 命令录制相机和激光雷达的数据包，在录制 Ros 数据包的过程中，不断调整棋盘格标定板的位置，尽可能多的让标定板的位置被两种传感器充分扫描到。

由于激光雷达的话题名称为 velodyne_points，Autoware 标定时识别的激光雷达的话题名称只能是 points_raw，在标定之前用一个脚本把激光雷达的话题名称改为能被识别的 points_raw，转换话题名称脚本的部分代码如下所示。

```
#初始化节点
rospy.init_node(&amp;apos;transfer_topic_from_ego_lid1ar_to_points_raw")
self.points_raw_pub = rospy.Publisher("/points_raw , PointCloud2, queue_size=100)
```

```
#订阅 lidar 节点
rospy.Subscriber("/rslidar_points",PointCloud2, self.callback_lidar)
#设置循环的频率
rate = rospy.Rate(10)
def callback_lidar(self,data):
#重设 frame_id
data.header.frame_id="rslidar"
#发布消息
self.points_raw_pub.publish(data)
# print(data)
```

打开 Autoware 和 Calibration Toolkit 标定工具箱，选择联合标定，Autoware 导入之前录制好的数据包开始播放，激光雷达和相机的画面会出现在 Calibration Toolkit 中，当标定箱中相机和激光雷达同时检测到棋盘格标定板时，调整激光雷达扫描的位置，让激光雷达的线束以最好的角度投影至标定板，完成一幅联合标定图。

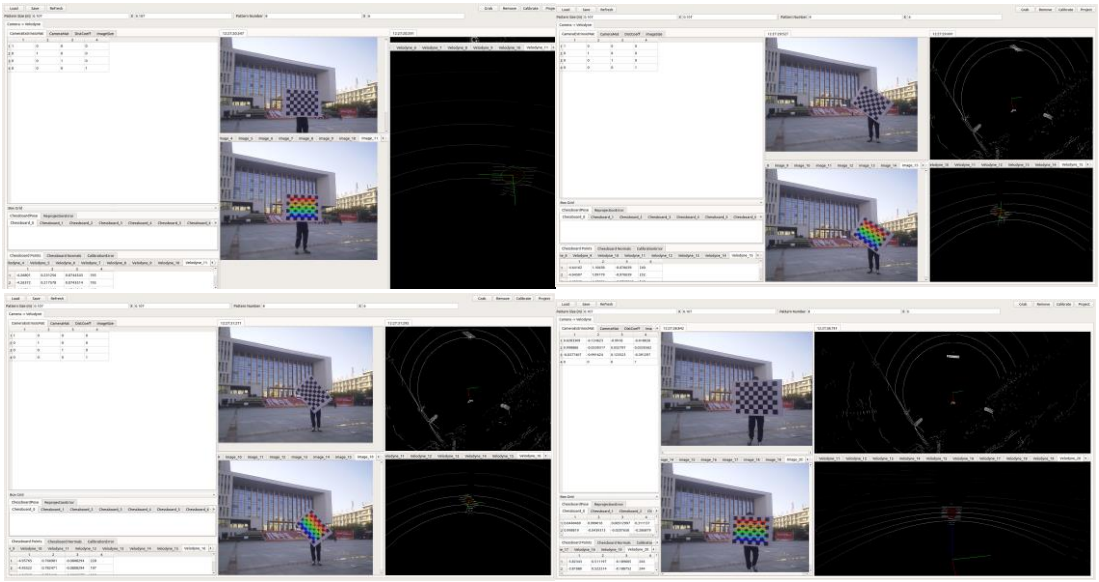


图 4-8 联合标定过程图

图 4-8 是标定过程中随机截取的四幅过程图，展示激光雷达在扫描在不同方向的标定板上。标定 50 幅图之后生成相机与激光雷达联合标定的标定文件，得到的相机与激光雷达联合标定参数如表 4-4 所示。

表 4-4 相机与激光雷达联合标定内外参结果

联合标定外部参数	旋转矩阵 R	$\begin{bmatrix} 0.0543 & 0.6335 & 0.7769 \\ 0.0824 & -0.7645 & 0.6268 \\ 0.9973 & 0.0309 & -0.1037 \end{bmatrix}$
	平移矩阵 T	$[-0.5509 \quad 1.0573 \quad 0.0068]$
	重投影误差	$0.0133 pixel$

标定完成之后点击标定工具箱中的 Project 投影，会根据计算结果和激光雷达数据生成的图像对应位置，以红色散点表示，如果红色散点分布到标定板上，则说明正确，如果红色散点分布在标定板外或者没有红色散点，说明标定效果很差，需要重新标定。图 4-9 是标定后投影结果的部分效果图，红色散点全部位于标定板上，相机与激光雷达的联合标定效果满足实验需求。

图 4-9 联合标定效果图

4.5 本章小结

本章主要介绍了本文智能驾驶试验小车的时空标定。介绍了相机坐标系、图像坐标系、像素坐标系和激光雷达坐标系之间的转换关系。由于两种传感器能提供高精度的时间戳，所以本文选择时间戳同步作为时间标定的方法。空间同步采用 Calibration Toolkit 标定工具箱对两种传感器进行空间同步，得到联合标定参数后验证标定效果，激光雷达的散点都能完整的投影到标定板的棋盘格内，标定结果满足本文要求，可以为后续融合实验提供数据支持。

第 5 章 相机与激光雷达融合识别

本文采取的相机与激光雷达融合识别的方式为目标级融合，通过融合两种传感器单独的检测框输出最终的检测结果。这通常涉及使用 ROS 的相机驱动程序（usb_cam）和激光雷达驱动程序（Velodyne）来订阅和发布相应的传感器数据。本文的实验基于 ROS 操作系统编写 ROS 节点实现相机采集的图像数据和激光雷达采集的点云数据的融合。ROS 采用发布与订阅（Publish/Subscribe）模型进行消息传递，融合识别系统可以将相机和激光雷达的数据发布到特定的话题（Topic），其他节点可以订阅这些话题来获取数据。在 ROS 操作系统中封装改进后的 YOLOv5s 和 Pointpillars 为两个 ROS 节点，配置相应的 launch 文件，允许图像传感器接收图像，点云传感器接收点云，进行模型的推理并发布检测结果。

5.1 ROS 操作系统介绍

使用 ROS（Robot Operating System）系统进行相机与激光雷达融合和目标识别在机器人和自主系统领域具有广泛的应用。这是因为 ROS 提供了一个开源、模块化、通信友好、硬件支持广泛以及拥有庞大用户社区的平台，使得这一融合过程更加容易和高效。ROS 的模块化架构允许用户独立开发和测试不同功能组件，然后将它们轻松集成到一个系统中。消息传递机制允许不同节点之间实时共享数据，对于多个传感器数据的协同工作至关重要。此外，ROS 还提供了丰富的现成库和工具，用于处理图像、点云、机器学习等任务，简化了开发过程。它支持各种不同型号的相机和激光雷达，适用于多个操作系统，因此具有高度的灵活性和扩展性。最重要的是，ROS 拥有庞大的社区支持，可以为开发者提供支持、建议和解决方案。综上所述，ROS 系统为相机与激光雷达融合和目标识别提供了强大的基础架构，使其成为机器人和自主系统开发的首选平台之一。

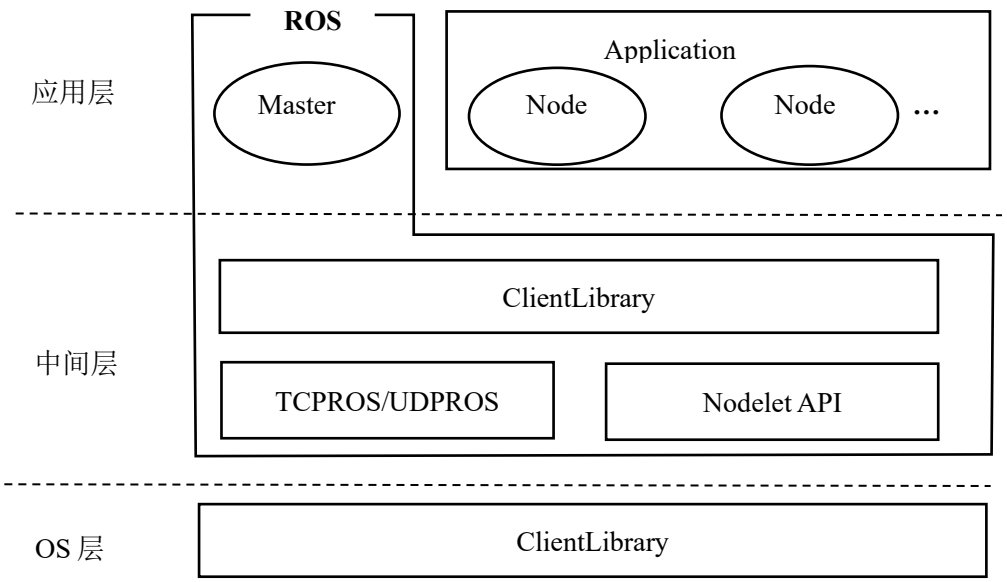


图 5-1 ROS 架构图

图 5-1 是 ROS 架构图，Master 充当着 ROS 系统的中心，负责管理节点的注册、名称服务和参数服务器。ROS 节点是独立的执行单元，可执行各种任务，如传感器数据获取、算法计算、执行器控制等。这些节点之间通过主题进行消息传递，节点可以发布和订阅主题来实现数据交流。主题携带特定消息类型的数据，这些消息类型定义了数据的结构和内容，为不同节点之间的通信提供了一致性。此外，ROS 还支持服务，节点可以提供和调用服务以实现请求-响应式通信，通常用于执行复杂的计算或控制任务。参数服务器允许节点在运行时共享配置参数，以便进行动态配置和参数调整。最后，ROS 软件以包的形式组织，每个包包含了相关的节点、主题、消息、服务和配置文件，从而使代码的组织和重用更加便捷。

5.2 ROS 实现图像与点云检测框融合流程

通过 ROS 实现图像检测框架与点云检测框架之间的连接的流程如图 5-2 所示，首先采用 Autoware 对相机和激光雷达两种传感器实现时间与空间对齐。图像和激光雷达点云识别节点分别经历原始数据输入、预处理、特征提取之后送入检测网络提取出目标的坐标信息，两种传感器在对应的识别框架得到检测框的坐标信息之后再融合节点开始目标级融合，通过检测框信息匹配关联策略得到目标的完整信息，最后有通过融合节点发布目标融合检测框信息。

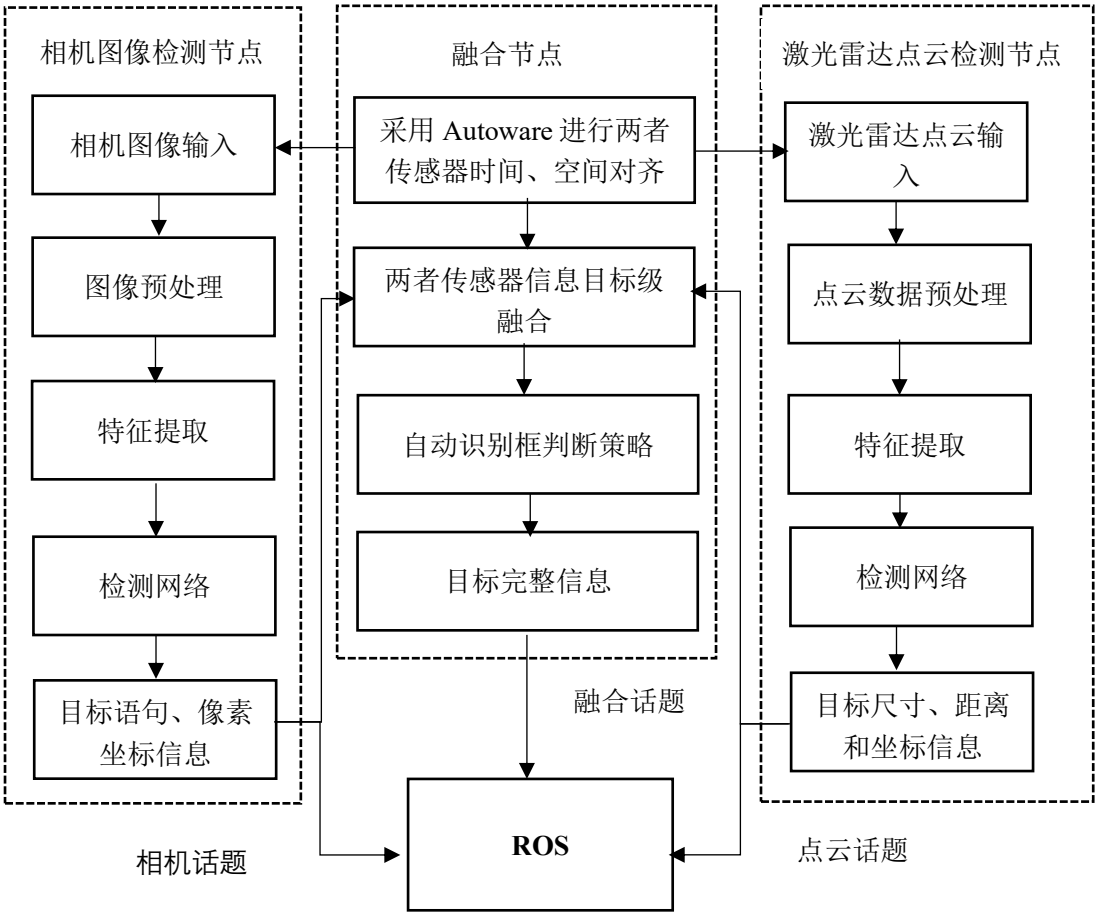


图 5-2 图像与点云融合检测的详细流程图

5.3 3D 检测框转换为 2D 检测框

将 3D 点云转换为 2D 伪图像时，通常需要考虑点云中的物体姿态信息。四元数（quaternion）和欧拉角（Euler angles）都用于表示物体的旋转姿态，但它们有不同的优点和用途。欧拉角是一种直观的方式来表示物体的旋转，通常包括绕三个轴（通常是 X、Y 和 Z 轴）的旋转角度。欧拉角具有直观性，易于理解，因为它们对应于人类感知的旋转概念，如俯仰、偏航和滚动。

在 3D 点云转换为 2D 伪图像时，通常与 2D 图像的坐标系和旋转表示更容易对齐。将四元数转换为欧拉角可以更好地与 2D 图像相匹配。

激光雷达通过改进的 Pointpillars 对扫描得到的点云识别之后，会得到三维包络检测框，每个三维包络检测框的姿态信息通常可以由一个四元数表示，四元数转换为欧拉角之后方便把 3D 包络框投影到 2D 图像上。四元数用于表示 3D 空间中的旋转，表示为$q=(x，y，z，w)$，通常表示为 $q=xi+yj+zk+w$，其中x、y、z和w是四元数的四个组成部分，而i、j和k是四元数的基。四元

数转换为欧拉角的公式如式 5-1、5-2 和 5-3 所示。

$$\varphi=\operatorname{atan2}(2(\omega z+xy),1-2(y^2+z^2)) \tag{5-1}$$

$$p=\operatorname{asin}(2(\omega y-zx)) \tag{5-2}$$

$$r=\operatorname{atan2}(2(\omega x+yz),1-2(x^2+y^2)) \tag{5-3}$$

式 5-1 中φ表示偏航角，式 5-2 中p表示俯仰角，式 5-3 中r表示滚动角。

得到 3D 包络检测框的方向信息之后自定义局部坐标（包络检测框的每个角点坐标）转换全局坐标函数通过包络框中心坐标和偏航角信息将包络框坐标转换为全局鸟瞰图坐标。这里用二维欧几里得平面上的旋转变换实现局部坐标到全局坐标的转换，矩阵公式 5-4 用于将局部坐标绕 Z 轴旋转一个角度ϕ：

$$R(\phi)=\begin{vmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{vmatrix} \tag{5-4}$$

在许多目标检测和跟踪任务中，通常使用鸟瞰视图来表示场景中的目标位置，这有助于简化问题并提供更清晰的相机信息，特别适用于移动机器人、自动驾驶车辆等应用。将包络检测框坐标转换到全局鸟瞰视图中，可以使得不同目标的位置信息更容易进行比较和分析，同时还能够避免物体在视角变化时的尺度问题。在此过程中主要经过下面三个步骤完成:

（1）将包络检测框的中心点$[x，y]$进行平移，以适应全局坐标系中的参考平面；

（2）将包络检测框的宽度$width$和长度$length$转换为鸟瞰视图中的坐标，通常只保留横向的宽度和纵向的长度；

（3）将局部坐标旋转，以确定包围盒在鸟瞰视图中的朝向。

5.4 检测框数据关联

IOU（Intersection over Union）是一种常用的检测框关联方法，用于确定两个检测框之间的重叠程度。在点云的 3D 检测框和图像检测得到的 2D 检测框之间的关联中，IOU 可以帮助确定它们是否属于同一个目标。在实现时间同步和空间同步之后，将点云的 3D 检测框投影到 BEV 图像上，获得 2D 检测框。这可以通过将 3D 框的底部中心点投影到 BEV 图像上，并根据框的高度和宽度确定 2D 框的位置和大小。同时，图像检测已经提供了 2D 检测框。对于每个点云的 2D 检测框和图像检测的 2D 检测框，计算它们之间的 IOU。定义一个 IOU 阈值，用于确定两个检测框是否属于同一个目标。如果 IOU 大于等于阈值，则认为它们属于同一个目标；否则，它们被认为是不同的目标。遍历点云的 2D 检测框和图像检测的 2D 检测框，根据计算的 IOU 值和设定的阈值，将它们关联起来。

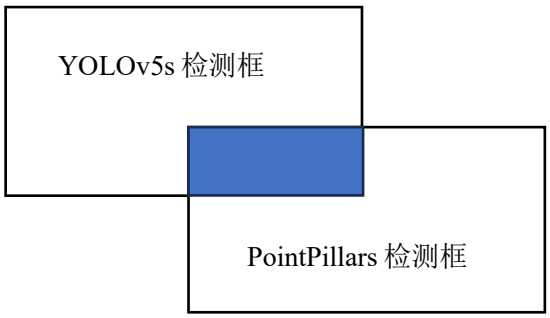


图 5-3 两种传感器检测框关联的结构图

图 5-3 是两种传感器检测框关联的结构图，计算 YOLOv5s 检测框与 Pointpillars 检测框之间的交并比，即上图蓝色部分面积与两个检测框面积总和之比，如果比值大于设定的阈值，则认为两个检测框属于同一目标，重新绘制融合检测框，如果比值小于设定的阈值，在无法匹配关联检测框之后在该帧图像中舍弃置信度较小的检测框，保留置信度较大的检测框。

把点云识别到的 3D 包络框变换成 2D 包络检测框投影到图像上，需要计算 YOLOv5s 识别得到的包络框与 Pointpillars 得到的包络框之间的交并比 IOU。为了确定重叠部分在 x 和 y 方向上的位置，首先计算两个边界框的交叉区域的坐标范围，通过图像包络检测框和点云转换到图像的包络检测框的宽度和高度计算相机包络框和激光雷达包络框的面积，计算交叉区域的面积，这是包络框之间的重叠部分的面积，计算交并比 (IOU)，这是交叉区域的面积与总面积的比值，本文将 IOU 的最小值设置为 0.5，如果大于 0.5，保留识别框。相机和激光雷达融合识别的伪代码如下。

```
导入必要的库和模块

定义函数 quaternion2euler，将四元数转换为欧拉角：
根据四元数的 x、y、z、w 计算偏航角 (yaw)
返回偏航角 (以度为单位)

定义函数 local2global，将局部点转换为全局点：
使用给定的质心和偏航角将局部点转换为全局点
返回全局点

定义函数 calc_angle，计算两个向量之间的角度：
使用内积计算两向量之间的角度
返回角度

定义函数 iou，计算两个边界框之间的交并比：
根据两个边界框的坐标计算它们的交集和并集
```

返回交并比

定义类 fusion 以进行传感器数据融合：
初始化方法：
初始化必要的变量和数据结构
初始化 ROS 节点和订阅者
定义 camera_callback 方法，处理改进 YOLO 的检测结果
定义 image_callback 方法，获取图像数据
定义 lidar_callback 方法，处理改进 Pointpillars 的检测结果并进行传感器校准
定义 bbox2globalBEV 方法，将 LiDAR 边界框转换为全局 2D BEV 坐标
定义 select_corners 方法，选择 LiDAR 边界框的角点
定义 calib_sensor 方法，进行摄像头和 LiDAR 的校准，并发布融合后的检测结果

主程序：
如果 ROS 没有关闭：
创建 fusion 类的实例
使 ROS 持续运行，直到节点被关闭

5.5 相似性加权稳定性融合策略

为了提高不同检测器输出结果的融合效果，本文引入一种创新的融合方法，即相似性加权稳定性融合方法。这一方法以相似性权重和检测框的稳定性为核心，旨在综合考虑目标检测框之间的相似性，同时提高对环境动态变化的鲁棒性。相较于传统融合方法，这种方法更为灵活，能够适应不同场景和任务的需求。

考虑相似性权重有助于在融合中更加重视那些在形状和位置上相似的框。这样的设计能够使融合结果更准确地反映相似目标的位置和形状，从而提高对相似目标的分辨率。相似性权重（*SimWeight*）是相似性加权稳定性融合方法的关键组成部分。通过计算交并比（*IoU*），*SimWeight* 能够量化两个检测框之间的相似程度。其公式如式 5-5 所示。

$$SimWeight = \frac{IoU(Box_1, Box_2)}{2 - IoU(Box_1, Box_2)} \tag{5-5}$$

其中，Box_1 表示 YOLOv5s 检测框，Box_2 表示 Pointpillars 检测框，$IoU(Box_1, Box_2)$表示两个框的交并比。相似性权重的引入使得融合方法能够更加重视形状和位置上相似的目标，从而提高对相似目标的准确性。

框的稳定性（*FusionWeight*）考虑了框在空间上的变化程度，以增强对动态环境的适应性。该稳定性通过计算框位置的方差来度量，其公式表示为式 5-6。

$$FusionWeight = \alpha \cdot SimWeight + (1-\alpha) \cdot (1 - Var(Box_1, Box_2)) \quad (5\text{-}6)$$

式 5-6 中α是超参数，用于调整相似性权重的影响。*Var* 计算了目标框位置的方差，为考虑框的稳定性提供了量化指标。通过引入框的稳定性，融合方法在面对目标位置和形状的动态变化时表现更加鲁棒。

融合过程通过综合考虑相似性权重和框的稳定性，得到的融合结果如式 5-7 所示。

$$FusedBox = FusionWeight \cdot Box_1 + (1 - FusionWeight) \cdot Box_2 \quad (5\text{-}7)$$

通过展开可以得到式 5-8，明确展示了超参数α与相似性权重的关系。

$$FusedBox = \alpha \cdot SimWeight \cdot Box_1 + (1 - \alpha \cdot SimWeight) \cdot Box_2 \quad (5\text{-}8)$$

本文提出的融合方法着眼于综合考虑相似性权重和框的稳定性，其中相似性权重 *SimWeight* 的引入通过计算 *IoU* 进一步加深了对相似目标的关注。超参数 α 的引入能够调整相似性权重的影响程度，从而增强了方法的适应性。在进行融合实验时，通过将检测框输入到设计的融合方法中，得到了融合结果 *FusedBox*。

5.6 KITTI 数据集图像与点云融合识别实验

车载 KITTI 数据集捕获了 2 个灰度相机、2 个彩色相机、一个 Velodyne64 线激光雷达、4 个光学镜头和一个 GPS 导航系统的数据。相比其他数据集，KITTI 数据集包含了丰富的数据，因此在自动驾驶领域被广泛使用。本研究中使用的是 16 线 3D 激光雷达，它与 KITTI 数据集中的 Velodyne64 线激光雷达有着不同的参数，具体对比见表 5-1。

表 5-1 KITTI 数据采集平台 64 线激光雷达与本文 16 线激光雷达对比

名称	64 线激光雷达	16 线激光雷达
线数	16	64
测距能力	150m	120m
水平视场角	360°	360°
水平角分辨率	0.2°	0.2°
垂直视场角	26.8°	30°
垂直分辨率	0.4°	0.53°
精度	±2cm	±2cm
帧率	10Hz	20Hz
出点数	220 万点/秒	32 万点/秒

KITTI 的图像数据和点云数据都是静态的，包含每一帧的时间戳，在实现图像与点云的融合之前，需要处理 KITTI 数据为可播放的视频文件。把 KITTI 数据通过 kitti2bag 转换成 ROS 可播放的 bag 文件，执行 launch 文件的同时播放 KITTI 数据的 rosbag，打开可以得到如图 5-4 所示的结果。

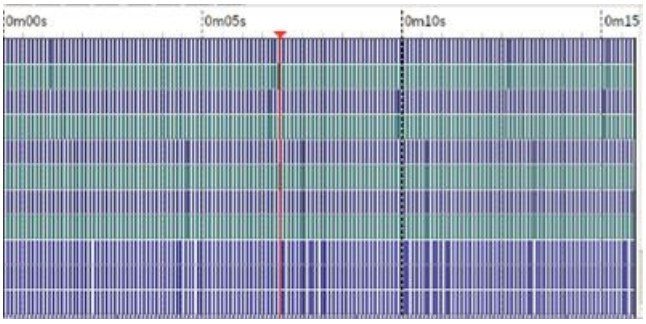


图 5-4 KITTI 数据集的数据包格式

rqt 是 ROS 中的一个强大工具，用于图形化地监视和控制 ROS 系统的各个方面。它提供了一种可视化界面，允许用户轻松地查看和操控 ROS 节点、主题、服务、参数等核心 ROS 组件。图 5-5 是 KITTI 数据集在进行图像与点云融合识别实验的 ROS 节点通信图，KITTI 数据的图像数据和点云数据以指定的话题分别发布到融合项目的图像检测节点和点云检测节点，图像采用 YOLOv5 检测，得到的检测框消息传递到融合节点，点云采用 Pointpillars 检测得到的检测框传递到融合节点，在融合节点对两种检测框处理完成之后以可视化窗口的形式展示出来。

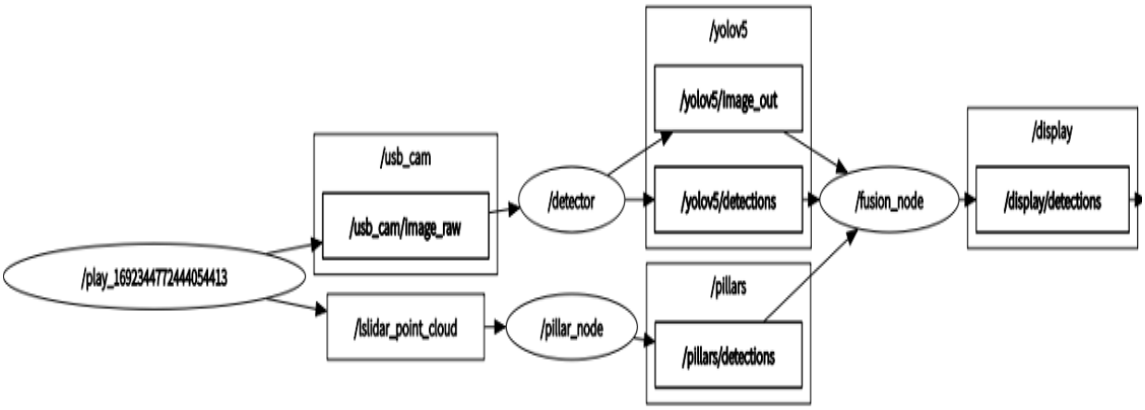


图 5-5 KITTI 数据集图像与点云融合的 ROS 节点通信图

准备工作完成之后，将部分 KITTI 数据集导入系统，测试本文融合算法的完整性与鲁棒性，随机截取 rviz 中的其中一帧如图 5-6 所示，为了使检测框更加清晰，本文均放大了 KITTI 数据集实验和智能驾驶试验小车的部分结果，放大内容为融合算法的检测框（图中红色①部分）和改进 YOLOv5s 算法的检测框（图中红色②部分）。

图 5-6 KITTI 数据集融合实验 rviz 可视化

图 5-6 中左下角橙色检测框是改进 YOLOv5s 算法的检测结果，红色检测框是本文融合算法实验结果，右边白色部分是激光雷达扫描得到的点云，绿色三维检测框是改进 Pointpillars 得到的检测结果。对比橙色检测框可以得知，融合算法的检测结果优于单独改进 YOLOv5s 的检测结果。由于本文主要是检测前方车辆，右边三维检测框有部分属于行人的检测框没有投影到伪图像，所以在最终的融合检测框没有显示这部分内容。

5.7 校园智驾小车融合实验

把相机与激光雷达融合识别的模型框架部署到校园智能驾驶小车上，启动相机和激光雷达，在 rviz 中的可视化如图 5-7 所示。

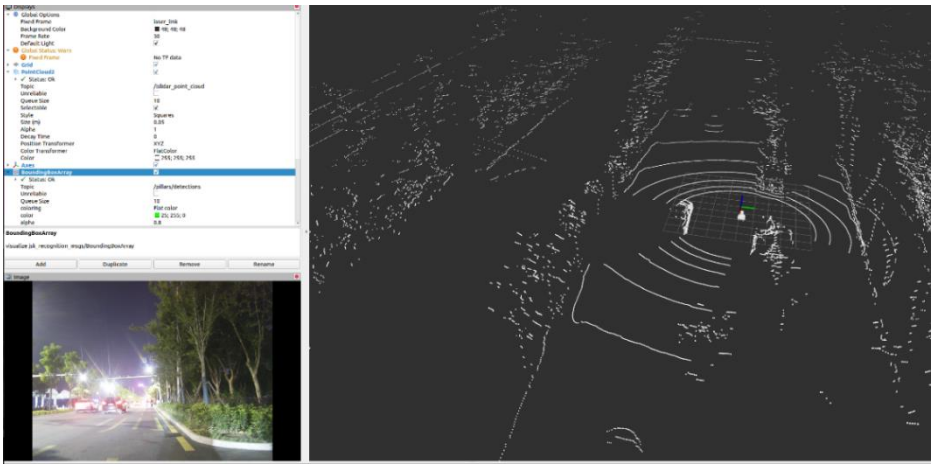
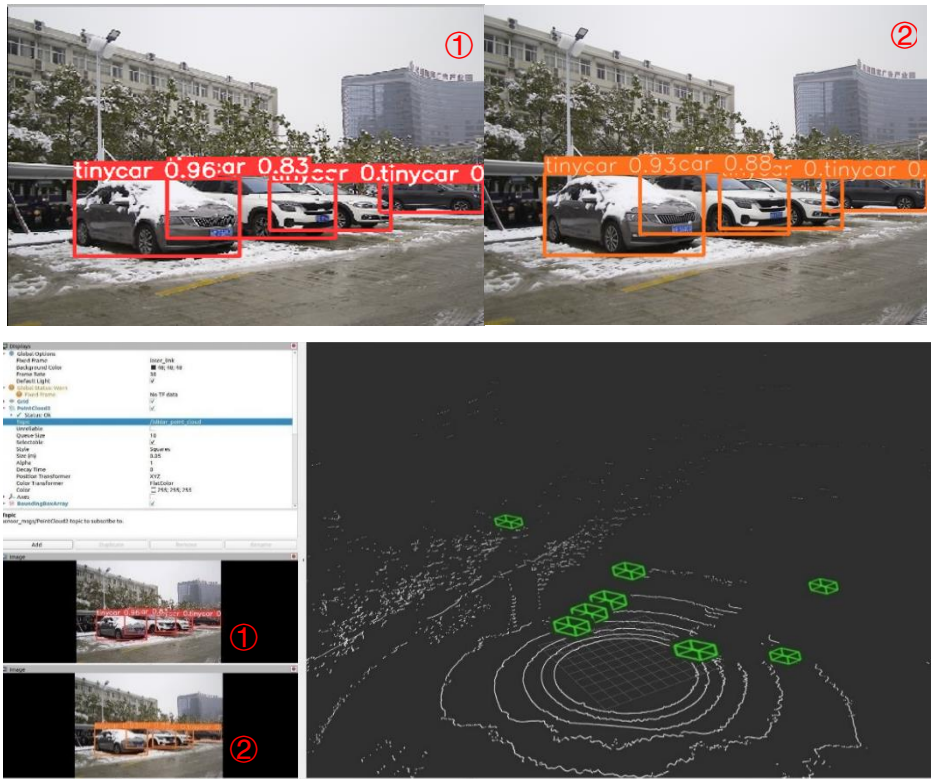


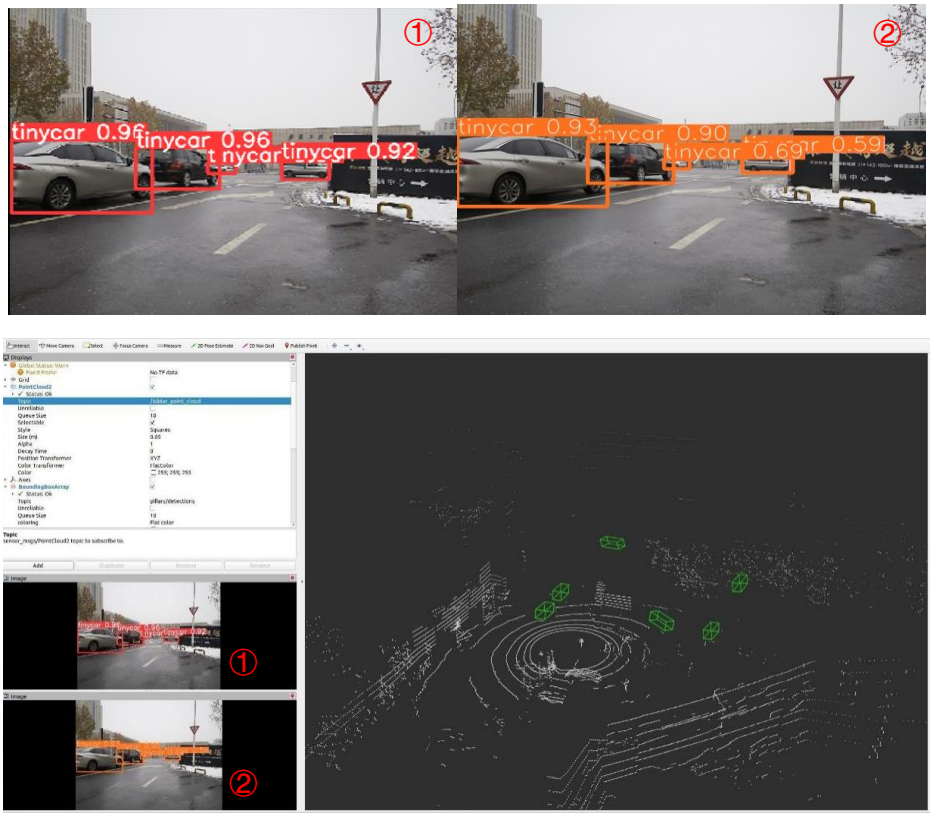
图 5-7 相机和激光雷达同时启动 rviz 可视化

通过 rostopic 查看相机和激光雷达的话题名如下图所示，在 launch 文件和

rviz 中修改当前对应的话题，然后在 rviz 中的可视化如图 5-8 所示。



(a)　雨雪道路



(b)　红绿灯路口

(c)　城区快速路

(d) 室外停车场

图 5-8 不同场景传感器融合可视化

图 5-8 中（a）、（b）、（c）、（d）分别是雨雪道路、红绿灯路口、城区快速路

和室外停车场四种场景、图中黄色框是改进 YOLOv5s 的预测框，红色框是相机和雷达相似性加权稳定性融合算法得到的预测框，红色框的预测概率明显高于黄色框，红色预测框的对汽车框选的精准度高于黄色框。说明融合算法识别前方车辆相比改进的 YOLOv5s 算法更精确。Pointpillars 可以检测出图像中不存在的汽车三维点云，但是由于三维点云转换成 BEV 视角，不在图像范围内的车辆目标在融合过程中被过滤掉了，换成广角摄像头，融合结果将会出现更多的车辆目标。

5.8 实验结果分析

为了定量分析智能小车融合识别前方汽车的表现，利用查全率m_r和准确率m_p对本文相似性加权稳定性融合算法进行量化评价，计算公式如式 5-9 和式 5-10 所示。

$$m_r = \frac{O_r}{O_a} \tag{5-9}$$

$$m_p = \frac{O_t}{O_a} \tag{5-10}$$

上面公式中O_r识别到的车辆目标数，O_t是识别到前方车辆目标且标签正确目标数，O_a是当前帧图像中车辆目标数。在光线良好的白天、光线不好的夜晚及雨雪天气场景下分别随机截取 1000 帧数据，利用上面的评价指标，完成算法的评估，相机算法与融合算法的目标检测率如表 5-2 所示。

表 5-2 前方车辆检测查全率和准确率表

方案	工况	查全率(m_p)	准确率(m_r)
纯相机(原 YOLOv5s 算法)	光线良好工况	88.60%	89.70%
本文融合算法		95.30%	98.80%
纯相机(原 YOLOv5s 算法)	光线不好工况	81.80%	85.70%
本文融合算法		92.50%	94.90%

从上表可以看出，在雨雪雾等恶劣天气会给相机成像带来影响的工况下，纯相机方法对于前方车辆的查全率和准确率都低于本文采用相机与激光雷达两种传感器对前方汽车的识别能力，这说明了单用相机一种传感器对前方汽车的环境感知是有缺陷的，激光雷达作为感知前方汽车的另一种传感器可以弥补相机存在的缺陷，即在光线不好的工况下，融合算法也可以探测前方车辆的位置，从而确定自身车辆的行驶安全。

5.9 本章小结

本章主要介绍了相机与激光雷达的融合方法以及融合结果。三维点云检测框投影到 BEV 伪图像生成二维检测框，与相机生成图像的二维检测框完成数据关联后利用相似性加权稳定性融合方法融合两种检测框，生成新的检测框。在 KITTI 数据集上验证了本章融合算法的有效性后将该算法移植到智能驾驶试验小车上，结果表明，本文提出的两种传感器信息融合可以高效完成汽车实时检测任务。分别评估了在光线良好的白天、光线不好的夜晚及雨雪天气场景下的 1000 帧数据，结果进一步证明了无论是在光线良好的工况还是在光线不好的工况，本文融合算法都优于原 YOLOv5s 算法。

第 6 章 结论与展望

6.1 本文总结

为了提高车辆智能驾驶的环境感知能力，本文提出了一种相机图像数据与激光雷达点云数据后融合的方法。获取相机信息导入改进 YOLOv5s 算法模型，获取激光雷达信息导入 Pointpillars 算法模型，应用时间戳对齐的方法处理两种传感器的信息，经坐标系转换后实现融合识别，并在 KITTI 数据集和校园真实场景下用智能试验小车进行算法验证，具体研究结论如下：

（1）基于改进 YOLOv5s 算法模型实现相机对前方车辆的识别。通过同时改进损失函数、引入注意力机制和改进特征融合网络，在不损失检测速度的情况下使车辆检测的平均精度均值 mAP 达到了 0.989。将改进算法模型得到的最优权重应用于检测层，分别在市区、校园、停车场、隧道和雨雪天气道路场景下进行前方车辆识别试验，结果显示改进算法模型的识别效果均优于原模型。

（2）基于改进 Pointpillars 算法模型实现激光雷达对前方车辆的有效识别。本文引入自适应注意力机制和优化激活函数方法实现了 Pointpillars 算法的改进，通过 KITTI 数据集验证，改进算法模型可以在保持检测速度的前提下将车辆检测平均精度均值 mAP 提升 2%。

（3）实现了相机与激光雷达的时空标定。本文基于时间戳的方式对两种传感器进行时间对齐，采用 Autoware 中的标定工具箱联合标定智能试验小车上的相机与激光雷达，标定效果显示激光雷达点云能完全准确的投影到图像坐标系中。

（4）相机与激光雷达检测信息融合车辆识别。将改进 YOLOv5s 二维检测框架与改进 Pointpillars 三维检测框架相结合，引入相似性加权稳定性融合方法实现了两种传感器采集信息的有效融合。在 ROS 系统中对改进 YOLOv5s 模型和本文融合检测模型以 KITTI 数据集的方式验证了算法的可行性。

（5）智能驾驶实验小车在多种复杂道路场景下验证算法的鲁棒性。把融合算法模型移植到配备相机与激光雷达的智能驾驶试验小车上，分别在雨雪道路、红绿灯路口、城区快速路和室外停车场四种场景进行实验，结果表明，采用本文研究的融合算法在几种环境下均能精准有效识别前方车辆。

6.2 未来展望

本文提出改进 YOLOv5s 和改进 Pointpillars 融合识别汽车，结果证明不论

是白天还是晚上，都能较好的识别出路面的汽车，基于检测框的后融合算法相较于数据直接融合减少了运算量，但是仍然存在一些问题，下面是对本文工作的一些展望，以供后续研究参考：

（1）更轻量化的检测框架。YOLOv5s 和 Pointpillars 虽然可以做到实时检测，但是两种模型中的参数较多，耗费较多的计算资源，两种模型可以在保证检测精度的基础上进一步轻量化，轻量化的框架可以节省算力，进一步提升检测的速度。

（2）相机与激光雷达更准确的时空标定。本文采用的是相机与激光雷达联合标定的方法确定内外参，由于是人工标定，每一帧的标定效果与标定的帧数都会对标定结果产生影响，未来的工作可以采用自动标定和定位的方法改善相机与激光雷达之间的准确配准，进一步提高融合系统的性能。

（3）使用更高规格的激光雷达。本文虽然在 KITTI 数据集验证了融合识别效果，但是在校园智能驾驶小车上搭载的是 16 线激光雷达，未验证 64 线激光雷达与相机融合的识别效果。

（4）验证更多工况下的融合识别能力。本文针对汽车行驶的工况是在直线道路，但现实生活中还有弯道、坡道、岔路口等多种道路场景，对于这些道路环境需要进一步验证算法的鲁棒性。

参考文献

[1] 中华人民共和国国家统计局.中国统计年鉴[M].北京:中国统计出版社,2022.

[2] 焦学文.强化安全管理，避绝交通事故[J].交通与运输,2017,33(02):76-77.

[3] 王聪,胡文,李文博,等.社会认知自动驾驶[J/OL].机械工程学报:1-21[2023-11-02].

[4] 陈见哲.疲劳驾驶检测方法研究进展[J].汽车实用技术,2023,48(21):179-186.

[5] 马庆禄,张杰.基于激光雷达的无人车动态避障方法[J].计算机仿真,2023,40(05):197-201.

[6] MinJoong K ,SungHun Y ,TongHyun K , et al.On the Development of Autonomous Vehicle Safety Distance by an RSS Model Based on a Variable Focus Function Camera[J].Sensors,2021,21(20):6733-6733.

[7] Obando-Ceron J S, Romero-Cano V, Monteiro S. Probabilistic multi-modal depth estimation based on camera–LiDAR sensor fusion[J]. Machine Vision and Applications, 2023, 34(5): 79.

[8] 李超群. 激光雷达和摄像头融合的目标检测与跟踪方法研究[D].吉林大学,2023.

[9] John L P ,Kim G H ,Jang M , et al.Object Detection and Localization on Map using Multi-Camera and Lidar PointCloud[J].INTERNATIONAL CONFERENCE ON FUTURE INFORMATION COMMUNICATION ENGINEERING,2022,13(1):267-275.

[10] Wenqi Z ,Han X ,Yunfan C , et al.PIFNet: 3D Object Detection Using Joint Image and Point Cloud Features for Autonomous Driving[J].Applied Sciences,2022,12(7):3686-3686.

[11] J D A ,C V C ,Anand B M , et al.Fully convolutional neural networks for LIDAR–camera fusion for pedestrian detection in autonomous vehicle[J].Multimedia Tools and Applications,2023,82(16):25107-25130.

[12] 胡方超. 基于三维点云分析的智能汽车目标检测方法研究[D].重庆邮电大学,2021.

[13] Rawlley O, Gupta S. Artificial intelligence‐empowered vision‐based self driver assistance system for internet of autonomous vehicles[J]. Transactions on Emerging Telecommunications Technologies, 2023, 34(2): e4683.

[14] 付强,刘海峰,刘志远.夜间交通灯识别方法研究[C]//中国自动化学会过程控制专业委员会.第 27 届中国过程控制会议（CPCC2016）摘要集.哈尔滨工业大

学航天学院;,2016:1.

[15] 刘曰,彭海娟,路锦文.智能驾驶中车辆检测方法综述[J].汽车实用技术,2017(11):22-26.

[16] 黄成.基于决策树分类的数字图像数据挖掘探究[J].现代计算机(专业版),2010(12):18-21.

[17] 张建明,张玲增,刘志强.一种结合多特征的前方车辆检测与跟踪方法[J].计算机工程与应用,2011,47(05):220-223+241.

[18] 郭君斌,王建强,易世春,李克强.基于单目相机的夜间前方车辆检测方法[J].汽车工程,2014,36(05):573-579+585.

[19] A. D ,D. P ,Laxman S , et al.Machine learning-driven pedestrian detection and classification for electric vehicles: integrating Bayesian component network analysis and reinforcement region-based convolutional neural networks[J].Signal, Image and Video Processing,2023,17(8):4475-4483.

[20] 王战古.不良天气条件下车辆检测方法研究[D].吉林大学,2022.

[21] 叶丽燕,赵建民,朱信忠,等.基于 Adaboost的轿车尾部检测方法[J].计算机仿真,2011,28(01):327-330.

[22] 余小角,郭景,徐凯,等.一种基于类Haar特征和AdaBoost算法的前车检测方法[J].微型机与应用,2017,36(13):22-25.

[23] Chen X, Liu L, Deng Y, et al. Vehicle detection based on visual attention mechanism and adaboost cascade classifier in intelligent transportation systems[J]. Optical and Quantum Electronics, 2019, 51: 1-18.

[24] Ren S， He K， Girshick R , et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[C]// NIPS. 2016.

[25] 邹斌,张聪.基于Faster-RCNN的密集人群检测算法[J/OL].计算机应用:1-7[2022-128].

[26] 邵延华,张铎,楚红雨,张晓强,饶云波.基于深度学习的YOLO目标检测综述[J].电子与信息学报,2022,44(10):3697-3708.

[27] 胡昭华,王莹.改进YOLOv5 的交通标志检测算法[J/OL].计算机工程与应用:1-15[2022-11-28].

[28] Jin X ,Yang H ,He X , et al.Robust LiDAR-Based Vehicle Detection for On-Road Autonomous Driving[J]. Remote Sensing, 2023, 15(12): 3160.

[29] Zhang F ,Knoll A .Vehicle Detection Based on Probability Hypothesis Density Filter[J].Sensors,2016,16(4):510- 510.

[30] Hyeong T K ,Hyeong T P .Fast 3D Detection of Vehicles Using Density Image of

Lidar[J].Journal of Institute of Control, Robotics and Systems,2018,24(12):1161-1169.

[31] Heng W ,Xiaodong Z .Real-time vehicle detection and tracking using 3D LiDAR[J].Asian Journal of Control,2021,24(3):1459-1469.

[32] 柏琳娟.基于深度学习的点云 3D目标检测方法研究[D].重庆邮电大学,2022.

[33] 张维.基于深度学习的点云车辆检测方法研究[D].电子科技大学,2022.

[34] Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1-9.

[35] Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. Advances in Neural Information Processing Systems (NeurIPS), 30.

[36] Lang, A. H., Vora, S., Caesar, H., & Zhou, L. (2019). Pointpillars: Fast Encoders for Object Detection From Point Clouds. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),12697-12705.

[37] 宫铭钱. 基于激光雷达和相机信息融合的车辆识别与跟踪研究[D].西南大学,2022.

[38] 岑佳婧.跨传感器车辆目标检测及重识别方法研究[D].华中科技大学,2022.

[39] 许艳伟.基于激光雷达的无人驾驶系统三维车辆检测[J].自动化应用,2023,64(05):179-182.

[40] Lele W ,Yingping H .Fast vehicle detection based on colored point cloud with bird’s eye view representation[J].Scientific Reports,2023,13(1):7447-7447.

[41] Wentao C ,Wei T ,Xiang X , et al.RGB Image- and Lidar-Based 3D Object Detection Under Multiple Lighting Scenarios[J].Automotive Innovation,2022,5(3):251-259.

[42] 黄峥华. 基于 3D激光点云与 2D图像融合的目标检测算法研究[D].哈尔滨工业大学,2022.

[43] Yuli W ,Hui L ,Nan C .Vehicle Detection for Unmanned Systems Based on Multimodal Feature Fusion[J].Applied Sciences,2022,12(12):6198-6198.

[44] Tang Q, Liang J, Zhu F. A comparative review on multi-modal sensors fusion based on deep learning[J]. Signal Processing, 2023: 109165.

[45] 张煜. 基于激光点云和相机融合的智能车前方障碍物检测方法研究[D].南京林业大学,2023.

[46] Oh S ,Kang H .Object Detection and Classification by Decision-Level Fusion for

Intelligent Vehicle Systems[J].Sensors,2017,17(1):207-207.

[47] Danapal G, Santos G A, da Costa J P C L, et al. Sensor fusion of camera and LiDAR raw data for vehicle detection[C]//2020 Workshop on Communication Networks and Power Systems (WCNPS). IEEE, 2020: 1-6.

[48] Schlosser J, Chow C K, Kira Z. Fusing LIDAR and images for pedestrian detection using convolutional neural networks[C]//2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016: 2198-2205.

[49] Wang H, Lou X, Cai Y, et al. Real-time vehicle detection algorithm based on vision and lidar point cloud fusion[J]. Journal of Sensors, 2019.

[50] Li J, Li R, Li J, et al. Dual-view 3d object recognition and detection via lidar point cloud and camera image[J]. Robotics and Autonomous Systems, 2022, 150: 103999.

[51] Liu H, Wu C, Wang H. Real time object detection using LiDAR and camera fusion for autonomous driving[J]. Scientific Reports, 2023, 13(1): 8056.

[52] 阜远远,王建平,张太盛,等.基于YOLOv5s的车辆实时检测与跟踪研究[J].安徽工程大学学报,2023,38(01):26-32.

[53] 颜学坤,楚建安.基于YOLOv5 改进算法的印花图案疵点检测[J].电子测量技术,2022,45(04):59-65.

[54] 邓天民,谭思奇,蒲龙忠.基于改进YOLOv5s的交通信号灯识别方法[J].计算机工程,2022,48(09):55-62.

[55] 邱天衡,王玲,王鹏,等.基于改进YOLOv5 的目标检测算法研究[J].计算机工程与应用,2022,58(13):63-73.

[56] 刘超阳,曲金帅,范菁,等.基于改进YOLOv5 算法的车辆目标检测[J].云南民族大学学报(自然科学版),2022,31(06):749-754.

[57] Liu Z J, Li Y M, Berwo M A, et al. Vehicle detection based on improved yolov5s algorithm[C]//2022 3rd International Conference on Information Science, Parallel and Distributed Systems (ISPDS). IEEE, 2022: 295-300.

[58] Shen X ,Liu T ,Wang Y .Vehicle detection based on improved YOLOv5s using coordinate attention and decoupled head[J].Journal of Electronic Imaging,2023,32(6):063023-063023.

[59] 黎戈. YOLOv5 目标检测算法多阶段改进[D].兰州大学,2021.

[60] 李阿娟. YOLOv5 算法改进及其现实应用[D].中北大学,2021.

[61] Wu Y, Wang Y, Zhang S, et al. Deep 3D object detection networks using LiDAR data: A review[J]. IEEE Sensors Journal, 2020, 21(2): 1152-1171.

[62] Meng'Ao L, Dongxue M, Songyuan G, et al. Research and improvement of DBSCAN cluster algorithm[C]//2015 7th International Conference on Information Technology in Medicine and Education (ITME). IEEE, 2015: 537-540.

[63] Royo S, Ballesta-Garcia M. An overview of lidar imaging systems for autonomous vehicles[J]. Applied sciences, 2019, 9(19): 4093.

[64] Fernandes D, Silva A, Névoa R, et al. Point-cloud based 3D object detection and classification methods for self-driving applications: A survey and taxonomy[J]. Information Fusion, 2021, 68: 161-191.

[65] Lang A H, Vora S, Caesar H, et al. Pointpillars: Fast encoders for object detection from point clouds[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 12697-12705.

[66] Hao Xu, Xiang Dong, Wenxuan Wu, Biao Yu, & Hui Zhu. (2023). A Two-Stage Pillar Feature-Encoding Network for Pillar-Based 3D Object Detection. World Electric Vehicle Journal, 14(146), 146.

[67] Xiang Song, Weiqin Zhan, Xiaoyu Che, Huilin Jiang, Biao Yang, "Scale-Aware Attention-Based PillarsNet (SAPN) Based 3D Object Detection for Point Cloud", Mathematical Problems in Engineering, vol. 2020, Article ID 3927365, 12 pages, 2020.

[68] Duan X, Jiang H, Tian D, et al. V2I based environment perception for autonomous vehicles at intersections[J]. China Communications, 2021, 18(7): 1-12.

[69] 常昕, 陈晓冬,张佳琛,等. 基于激光雷达和相机信息融合的目标检测及跟踪[J]. 光电工程, 2019, 46(7): 180420-1-180420-11.

[70] Zhang Y J. Camera calibration[M]//3-D Computer Vision: Principles, Algorithms and Applications. Singapore: Springer Nature Singapore, 2023: 37-65.

攻读硕士学位期间发表的学术论文及专利

[1]阜远远,王建平,张太盛等.基于YOLOv5s的车辆实时检测与跟踪研究[J].安徽工程大学学报,2023,38(01):26-32.（已见刊）

[2]阜远远,王建平,周晋伟等.基于改进pointpillars对汽车点云三维检测研究[J].安徽工程大学学报（已录用）

[3]周晋伟,王建平,阜远远等. 基于改进YOLOv5s的交通标志检测[J].安徽工程大学学报（已录用）

[4]周晋伟,王建平,阜远远等. 基于全景图像的车位检测方法研究[J]. 重庆科技学院学报（已录用)

致谢

时光总是像留不住的烟火，匆匆闪过。自步入安徽工程大学成为一名研究生以来，三年的时间过去了。这两年，我在科研学习方面得到了长足的进步；我在日常生活方面得到了必要的提高；我在思想觉悟方面得到了应有的提升。即将结束所热爱的研究生生涯，去寻找属于自己的一片天地，纵有万般不舍，也要挥手告别。常言道：回顾过去，展望未来。在这里我要向那些在我生活和学习方面给予帮助的人说声谢谢，没有你们的帮助，我无法想象我的研究生生活将会多么困难，因为你们的存在，我收获了绚丽多彩的研究生生活。

感谢我的导师王建平导师，从拟定课题到完成课题，中间过程可谓十分坎坷，感谢您的耐心指导。在论文的反复修改中，我真切的感受到您不仅是以为科学严谨的导师，更是一位平易近人的朋友，每每遇到学术问题，您总能以朋友和蔼的语气对我做出精确的点拨。这种亦师亦友的师生之情，我将铭记于心，珍藏永生。除此之外，也要感谢在课堂上辛勤付出的授课老师们，你们的课程各具特色，让我能够在科研方面奠定坚实的基础。有一些相对不易理解的课程内容，您们不仅细心耐心的引导讲解，同时还不断鼓励我们坚持下去，进一步夯实了我们遇到困难不放弃，坚持到底，迎难而上的精神。感谢各位老师，我会牢记你们在我学业方面的帮助，祝愿您们工作顺利，平安幸福。

感谢我的父母数十年如一日的默默付出，因为有了你们的支持，让我有勇气坚持走完上学这段路程。能够一直学习是也许是件令人高兴的事，但是前提是有足够的经济条件支撑，作为一个农村家庭的孩子，我非常能理解你们外出务工的艰辛，即使再苦再累，你们都坚持供应我上学，在经济上提供了莫大的支持。同时你们又是作为心灵的栖息港而存在，遇到苦难和压力的时候，你们总是想尽办法与我一起面对，感谢你们给我提供的情感支持。从牙牙学语到现在能独当一面，父母的感情真的让我无以为报。当然，在此也要感谢我的弟弟刘腾飞，他在我伤心难过时，会给予我一些安慰，让我能够很快平复情绪，以最大的热情和动力投身科研学习。

感谢我的同门师兄弟和室友，你们在我的日常生活中给与了很多宽容和理解，无数个挑灯深研的夜晚我们一起度过，无数个篮球场肆意挥洒汗水的日子我们也一起度过，我们在科研方面的互相探讨增长了我的科研能力，在篮球方面的交流增强了我的体质，感谢你们三年的陪伴，我们下一段旅程再见。

感谢我的朋友们，他们构成了我人生中不可或缺的一环。尤其是知心好友刘焕焕，相识数十载，我们无话不谈，一起走过了许多个难忘的春夏秋冬。在过去的每个阶段，我们总能实现我们的各种约定，形成了独属于我们自己的默

契，现在无论悲伤还是快乐，我们都能感受的到，这是一种莫名的情感，我很感动也很珍惜。谢谢你们没有在我的人生中错失，给我莫大的动力继续前行，让我对自己的未来充满渴望与信心。

百感交集的时刻也要感谢一直默默努力的自己，无数个挑灯夜战，奋笔疾书的瞬间从我的脑海闪过，很庆幸自己选择了求学这条道路，遇到了很多友善的人和很多有趣的事，他们打开了我的思路，开阔了我的的眼界。

再美好的演出总归会有谢幕的那一刻，这一次，我将带着难舍的回忆与你们挥手道别，感谢你们的出现与帮助，我相信未来的自己一定不会辜负你们，我相信自己也能成为理想中的自己。

尚德教学 唯实惟新