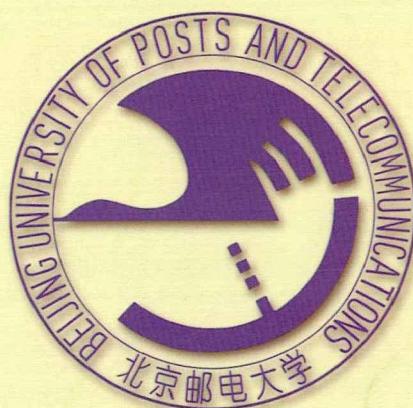


北京郵電大學

硕士学位论文



题目：基于单目视觉的 3D 目标检测算法研究

学 号：2018110189

姓 名：曹波

专 业：信息与通信工程

导 师：刘勇

学 院：信息与通信工程学院

2021 年 6 月 2 日

中国 · 北京

密级： 保密期限：



题目：基于单目视觉的 3D 目标检测算法研究

学 号： 2018110189

姓 名： 曹波

专 业： 信息与通信工程

导 师： 刘勇

学 院： 信息与通信工程学院

2021 年 6 月 2 日

# **BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS**

## **Thesis for Master Degree**



**Topic:** Research on 3D Object Detection Algorithms  
Based on Monocular Vision

**Student No.:** 2018110189

**Author:** Cao Bo

**Major:** Information and  
Communication Engineering

**Advisor:** Liu Yong

**Institute:** School of Information and  
Communication Engineering

**June 2nd, 2021**

### 独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名: 曹波 日期: 2021年6月2日

### 关于论文使用授权的说明

本人完全了解并同意北京邮电大学有关保留、使用学位论文的规定，即：北京邮电大学拥有以下关于学位论文的无偿使用权，具体包括：学校有权保留并向国家有关部门或机构送交学位论文，有权允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，有权允许采用影印、缩印或其它复制手段保存、汇编学位论文，将学位论文的全部或部分内容编入有关数据库进行检索。（保密的学位论文在解密后遵守此规定）

本人签名: 曹波 日期: 2021年6月2日  
导师签名: 孙海波 日期: 2021年6月2日

# 基于单目视觉的 3D 目标检测算法研究

## 摘要

目标检测是计算机视觉领域基本任务之一。目前存在的 2D 目标检测算法可以给出目标在 RGB 图像中的矩形边界框和相应类别。但 2D 目标检测具有局限性，无法满足生活在三维世界中人们的特定需求。3D 目标检测会给出目标的 3D 边界框，在某些应用场景如自动驾驶、机器人领域、三维场景重建等发挥着重要的作用。单目相机相较于其它传感器具有价廉、易获取等特点，所以仅仅基于单目视觉完成 3D 目标检测具有潜在的商业价值和深远的研究意义。本文的主要研究内容和贡献如下：

本文将 3D 目标检测任务拆分成几个子任务，包括 2D 目标检测模块、维度预测模块、旋转角预测模块、几何约束模块。每一个子任务负责估计 3D 目标检测涉及到的特定参数。从而将一个复杂的问题分解成几个较简单的子问题，联合各子问题的解便得到原问题的解。

(1) 本文提出了一种基于交并比 (Intersection over Union, IoU) 的维度预测损失函数。本文将计算矩形边界框 IoU 算法扩展到三维空间，提出了针对维度预测场景下的计算 3D 边界框 IoU 的算法。与传统的损失函数将预测的每个维度分量单独计算误差相比，基于 IoU 的损失函数考虑边界框所有属性的内在联系，且具有尺度不变性特点。

(2) 本文提出了一种间接预测目标旋转角的策略。基于单目视觉实现 3D 目标检测仅仅提供了 RGB 图像信息，而目标旋转角与目标在 RGB 图像中的外观无直接的联系。为此本文选择预测局部旋转角并通过几何关系计算出最终需要的目标旋转角。

(3) 针对基于单目视觉完成 3D 目标检测任务时目标深度信息获取困难的问题，本文依据几何约束原理，通过最小二乘法计算出目标相对于观测者的位置坐标。同时本文提出一种优化网络，对目标位置坐标进行修正。结合 2D 目标检测，维度和旋转角预测结果完成最终的 3D 目标检测任务。实验阶段以多个指标对本文提出的 3D 目标检测算法性能进行评估，结果表明与其他检测算法相比本文提出的算法有更高的检测精确度。

**关键词：**3D 目标检测 自动驾驶 深度学习 几何约束

# **Research on 3D Object Detection Algorithms Based on Monocular Vision**

## **ABSTRACT**

Object detection is one of the basic tasks in the field of computer vision. The existing 2D object detection algorithms can give the rectangular bounding box and the corresponding category of the object in the RGB image. However, 2D object detection has limitations and cannot meet the specific needs of people living in a three-dimensional world. 3D object detection will give the object's 3D bounding box, which plays an important role in certain application scenarios such as autonomous driving, robotics, and 3D scene reconstruction. Compared with other sensors, the monocular camera has the characteristics of low price, easy acquisition, so it has potential commercial value and far-reaching research significance to complete 3D object detection based on monocular vision. The main research content and contributions of this paper are as follows:

This paper divides the 3D object detection task into several subtasks, including 2D object detection module, dimension prediction module, orientation prediction module, and geometric constraint module. Each subtask is responsible for estimating specific parameters involved in the 3D object detection task. Decompose a complex problem into several simpler sub-problems, and combine the solutions of the sub-problems to obtain the solution of the original problem.

(1) This paper proposes a dimension prediction loss function based on Intersection over Union (IoU). In this paper, the algorithm for calculating the rectangular bounding box IoU is extended to three-dimensional space, and an algorithm for calculating the 3D bounding box IoU in the dimension prediction scene is proposed. Compared with the traditional loss function that calculates the error of each dimensional component of the prediction separately, the loss function based on IoU considers the intrinsic relationship of all the attributes of the bounding box, and has the

characteristics of scale invariance.

(2) This paper proposes a strategy to indirectly predict the orientation of the object. The realization of 3D object detection based on monocular vision only provides RGB image information, and the orientation is not directly related to the appearance of the object in the RGB image. For this reason, this article chooses to predict the local orientation and calculates the final required global orientation through geometric relations.

(3) Aiming at the problem of difficulty in obtaining object depth information when completing 3D object detection task based on monocular vision, this paper uses the principle of geometric constraints to calculate the position coordinates of the object relative to the observer by the least square method. At the same time, this paper proposes an optimized network to modify the object position coordinates. Combining 2D object detection, dimension and orientation prediction results to complete the final 3D object detection task. In the experiment stage, this paper uses multiple indicators to evaluate the performance of the proposed 3D object detection algorithms. The results show that the algorithm proposed in this paper has higher detection accuracy compared with other detection algorithms.

**KEY WORDS:** 3D object detection    autonomous driving    deep learning  
geometry constraints

# 目 录

第一章 绪论.....	1
1.1 课题研究背景与意义.....	1
1.2 3D 目标检测研究现状.....	2
1.2.1 基于单目视觉的 3D 目标检测研究现状.....	2
1.2.2 基于点云数据的 3D 目标检测研究现状.....	4
1.2.3 基于多传感器融合的 3D 目标检测研究现状.....	4
1.3 基于单目视觉的 3D 目标检测技术难点与本文主要贡献.....	5
1.4 本文结构安排.....	7
第二章 相机成像模型与 2D 目标检测.....	9
2.1 相机成像模型理论.....	9
2.1.1 参考坐标系.....	10
2.1.2 坐标系之间的转换.....	10
2.2 2D 目标检测.....	14
2.2.1 基于传统方法的 2D 目标检测.....	14
2.2.2 基于深度学习的 2D 目标检测.....	15
2.3 本章小结.....	20
第三章 3D 目标检测相关参数估计算法研究.....	21
3.1 目标维度估计算法研究.....	21
3.1.1 基于维度均值策略的目标维度估计算法.....	21
3.1.2 基于交并比的维度预测损失函数.....	22
3.2 目标旋转角估计算法研究.....	24
3.2.1 旋转角分类预测策略.....	24
3.2.2 局部旋转角与全局旋转角.....	27
3.2.3 角度转换.....	27
3.3 参数估计网络与多尺度 2D 目标检测.....	28
3.3.1 参数估计网络.....	28
3.3.2 多尺度 2D 目标检测.....	30
3.4 本章小结.....	31
第四章 基于几何约束的 3D 目标检测与参数优化.....	33
4.1 几何约束理论.....	33
4.1.1 最小二乘法.....	33
4.1.2 成像模型与几何约束.....	35

4.1.3 几何约束优化.....	38
4.2 参数优化.....	40
4.2.1 几何约束存在的问题.....	40
4.2.2 位置坐标修正.....	40
4.3 本章小结.....	42
<b>第五章 实验结果与分析.....</b>	<b>43</b>
5.1 数据集.....	43
5.2 实验.....	44
5.2.1 评估指标.....	44
5.2.2 实验结果与分析.....	45
5.3 本章小节.....	54
<b>第六章 总结与展望.....</b>	<b>55</b>
6.1 工作总结.....	55
6.2 未来展望.....	56
参考文献.....	57
致谢.....	61
攻读硕士学位期间发表的学术论文目录.....	62

# 第一章 绪论

## 1.1 课题研究背景与意义

随着科学技术的不断进步，人类迈入了 5G 时代<sup>[1]</sup>。5G 网络给自动驾驶场景下海量数据的传输提供了通信的保障。其低延时的传输特性，使得大量数据能快速准确的被自动驾驶车辆的控制系统获取，并加以综合分析，从而控制下一步的路径规划和行驶速度，提高驾驶安全性和交通效率。自动驾驶技术快速发展。2017 年百度发布阿波罗 (Apollo)<sup>[2]</sup> 无人驾驶计划。2020 年百度推出 Apollo Go，提供出租车服务功能。近几年，自动驾驶领域不仅在理论上快速突破，也实现了多个商业领域的落地，证明了自动驾驶的可行性和良好的应用前景。

自动驾驶的安全性高度依赖于车辆对周围环境精确的感知。而能快速且准确的检测到周围车辆是后续自动驾驶进行相关控制决策的基础。本文把在 RGB 图像上绘制目标矩形边界框的检测技术称为 2D 目标检测，相应的边界框称之为 2D 边界框。给出目标在三维空间中立方体边界框的检测技术称为 3D 目标检测，相应的边界框称之为 3D 边界框。3D 边界框可以投影到成像平面绘制出来。3D 目标检测与 2D 目标检测具有很大的区别。2D 目标检测需要给出目标的 2D 边界框和相应的类别。2D 边界框可以表示为  $B^{2d} = (x^{2d}, y^{2d}, w^{2d}, h^{2d})$ 。其中  $x^{2d}$  和  $y^{2d}$  分别为 2D 边界框左上顶点的横坐标和纵坐标， $w^{2d}$  和  $h^{2d}$  分别为 2D 边界框宽和高。即 2D 目标检测仅仅给出了目标在 RGB 图像中的位置。3D 边界框可以表示为  $B^{3d} = (w, h, l, x, y, z, \theta_{yaw}, \theta_{pitch}, \theta_{roll})$ 。其中  $w$ 、 $h$ 、 $l$  分别为目标维度宽、高、长； $x$ 、 $y$ 、 $z$  分别为目标中心相对于观测者的空间位置坐标； $\theta_{yaw}$ 、 $\theta_{pitch}$ 、 $\theta_{roll}$  分别为目标相对于观测者的偏航角、俯仰角、翻滚角。在自动驾驶领域，仅仅考虑偏航角，因为总是假设地面是水平的，这也是目前所有 3D 目标检测工作默认的假设。所以 3D 边界框可以简化为  $B^{3d} = (w, h, l, x, y, z, \theta_{yaw})$ ，总共包括 7 个自由度，3D 目标检测的任务就是给出 3D 边界框的这 7 个参数和目标相应的类别。下图 1-1 给出了 2D 边界框和 3D 边界框在成像平面示意图：

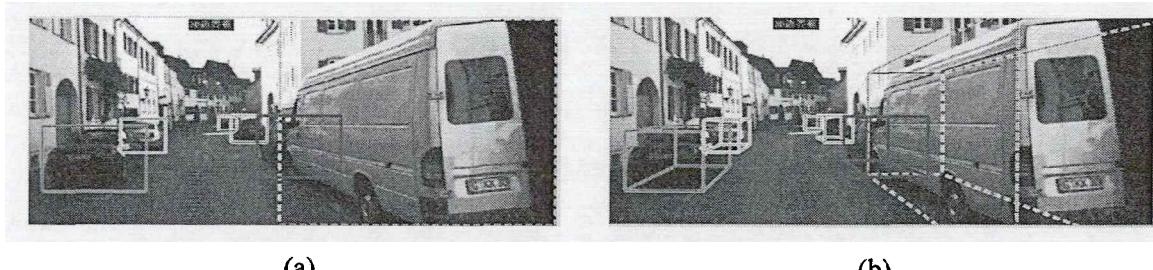


图 1-1 2D 边界框与 3D 边界框示意图。(a) 2D 边界框在成像平面示意图，(b) 3D 边界框在成像平面示意图。

目前存在的 2D 目标检测技术无法满足生活在三维空间人们的特定需求。在某些应用场景如自动驾驶、机器人领域<sup>[3]</sup>、三维场景重建<sup>[4]</sup>等，需要知道目标的三维空间信息。3D 目标检测输出的结果为目標的维度( $w, h, l$ )，目標相对于观测者的旋转角 $\theta_{yaw}$ ，目標相对于观测者的空间位置坐标( $x, y, z$ )和目標类别。相比于 2D 目标检测，3D 目标检测能让观测者得到目標实例更丰富的几何信息。在自动驾驶领域，这些信息有利于系统对车辆进行路径规划和控制，实现安全的自动驾驶；在机器人领域，利用目標在三维空间的信息，可以指导机器人进行目标抓取和障碍物躲避，实现机器人的功能化和智能化。在场景重建领域，这些信息可以帮助人们重建目標在三维空间中的立体图。基于此 3D 目标检测具有深刻的意义和理论研究价值。下表 1-1 给出了 2D 目标检测和 3D 目标检测各自要达成的目标和相应的优缺点：

表 1-1 2D 目标检测与 3D 目标检测对比表。

	任务	优点	缺点
<b>2D 目标检测</b>	在 RGB 图像平面给出目標的 2D 边界框和相应的类别。	具有大量公开数据集和成熟的检测算法；在绝大多数情况下仅仅需要提供 RGB 图像便能够得到精确的检测效果。	无法给出目標的物理尺寸和目標在三维空间的位置坐标，应用的领域有限。
<b>3D 目标检测</b>	给出目標的 3D 边界框和相应的类别，3D 边界框可以投影到 RGB 图像平面并绘制出来。	3D 边界框提供了目標的物理尺寸和空间位置坐标，这些信息有助于观测者更好的感知周围环境。	为了得到空间位置坐标需要对目標的深度进行估计；需要获取额外的物理尺寸增加了检测模型的复杂度；相关数据集稀缺。

由于很难从 RGB 图像中直接获取目標与相机之间的距离，即相应的深度（Depth）信息，即也无法进一步获取到目標在三维空间中的位置坐标，使得基于单目视觉完成 3D 目标检测成为一个高难度的挑战。相较于雷达和深度相机的昂贵，单目相机易于获取和装备在交通工具上，利用廉价的单目相机实现 3D 检测具有潜在的商业价值，同时也有着深远的研究意义。

## 1.2 3D 目标检测研究现状

### 1.2.1 基于单目视觉的 3D 目标检测研究现状

基于单目视觉的 3D 目标检测最大的挑战是如何得到目標空间位置坐标。Chen 等提出了 Mono3D<sup>[5]</sup>，在目標可能出现的空间区域穷举 3D 候选边界框，然后利用复杂的特征，如语义分割、上下文以及位置先验等来对候选框进行过滤，

最后将剩下来的候选框通过一个分类器，得到最终的 3D 边界框。然而此方法的不足之处在于三维空间比二维空间范围大得多，所以需要花费很多的计算资源来穷举这些 3D 候选框。Mono3D 建立在以前的工作 3DOP<sup>[6]</sup>上。与 Mono3D 不同的是，3DOP 使用了深度图（Depth Map）来产生候选框，但二者有着相似的检测流程。Pham 等<sup>[7]</sup>在 3DOP 的基础上对每一个类别利用单目 RGB 图像和深度图分别产生候选框，相比较于 Mono3D 和 3DOP 检测效果有了显著的提升。

为了利用目前成熟的 2D 目标检测算法，Mousavian 等提出 Deep3DBox<sup>[8]</sup>，通过一个 MultiBin 网络来回归目标的旋转角和维度，并根据 2D 边界框与 3D 边界框之间的几何约束（Geometry Constraints）来计算目标的空间位置坐标。Brazil 等提出了 M3D-RPN<sup>[9]</sup>来实现单目 3D 检测。意识到 2D 与 3D 检测最主要的区别在于是否需要给出目标深度信息，为了将 2D 检测与 3D 检测结合起来，形成一个统一的 2D/3D 检测框架，提出了 3D 锚（Anchor）的概念。同时 M3D-RPN 提出深度感知卷积（Depth Aware Convolution）来帮助修正相关参数。与传统的卷积运算空间不变性相比，深度感知卷积利用不共享的卷积核来提取图像基于空间认知的特征。同时提出了一个实用的算法对回归的参数进行优化从而得到精确的 3D 边界框。Liu 等提出 FQNet<sup>[10]</sup>来评价预测 3D 边界框与真值 3D 边界框之间的吻合度。将得到的初步预测框作为种子在三维空间中产生大量的候选框样本，然后通过 FQNet 网络给每个候选框打分，选择得分最高的候选框作为最终的预测边界框。Li 等提出 GS3D<sup>[11]</sup>，利用目前先进的 2D 检测器来获得 2D 边界框。将 2D 边界框与场景的先验认知综合考虑得到一个初步的粗糙 3D 边界框，并称为 Guidance。利用 Guidance 来指导生成最终的 3D 边界框。通过目标在 RGB 图像上可见表面的视觉特征来探索对象的 3D 结构信息。利用目标的该视觉特征消除了仅使用 2D 边界框优化 3D 边界框所带来的表示模糊性问题，从而使得最终的实验效果有了较明显的提升。

探索从 RGB 图像中直接获取深度信息也成为热门研究话题。Xu 等提出了 MF3D<sup>[12]</sup>。对于输入的 RGB 图像，通过一个全卷积网络（Fully Convolutional Networks, FCN）<sup>[13]</sup>来估计每个像素的视差，利用相机标定文件得到近似的深度和点云（Point Cloud）数据实现 3D 检测。Qin 等提出了 MonoGRNet<sup>[14]</sup>，给出了一种实例级别的深度估计网络（Instance Depth Estimation, IDE）用来直接估计目标的绝对深度信息，然后通过后续的精细化处理来提高最终 3D 检测的精度。然而本质上由于 RGB 图像深度信息的缺乏，这些算法估计出来的目标深度都不够准确，降低了整个检测系统的性能。

基于计算机辅助设计（Computer Aided Design, CAD）模型的单目视觉 3D 目标检测算法也日益发展。Chabot 等提出了 DeepMANTA<sup>[15]</sup>，利用一个多任务网

络来估计车辆的位置，并采用一组关键点来描述车辆的边界。首先通过两级精细化网络获得 2D 边界框和车辆的局部定位。然后根据推导出的形状与 CAD 模型进行匹配，得到最终三维姿态，完成 3D 目标检测任务。然而此方法需要提前设计每个目标的 CAD 模型，加深了检测算法的复杂度。

### 1.2.2 基于点云数据的 3D 目标检测研究现状

由于雷达能直接从点云数据恢复出目标在三维空间中的状态信息，所以基于雷达的 3D 目标检测也吸引着许多研究人员的兴趣。Li 等提出了一种利用圆柱投影匹配和 FCN 的 VeloFCN<sup>[16]</sup> 来预测 3D 边界框。Simon 等提出了 Complex-YOLO<sup>[17]</sup>，对成熟的 2D 目标检测算法进行扩展，实现在点云数据上给出目标的 3D 边界框。同时提出欧拉区域建议网络来估计目标的位姿，添加虚函数和实函数到回归网络形成封闭的复数空间，避免了在单个角度估计中带来的奇异性问题。Engelcke 等提出了 Vote3Deep<sup>[18]</sup> 来解决体素网络的稀疏性问题。然而论文假设对所有的检测目标具有固定的尺寸，导致算法的局限性。Zhou 等提出 VoxelNet<sup>[19]</sup>。通过 3D 卷积层来提取抽象特征。然而由于体素的稀疏性和 3D 卷积，使得整个网络在检测阶段具有比较高的耗时。Li 等提出 3DFCN<sup>[20]</sup>，在 VeloFCN 的基础上开展工作，将点云离散化为 4 维的张量，分别是长度、宽度、高度和通道。扩展基于全卷积网络的 2D 检测技术到 3D 来实现 3D 目标检测。相较于 VeloFCN，该方法在检测精度上实现了 20% 的提升。基于点云数据的 3D 目标检测往往能得到比较好的效果，然而点云数据需要通过特定的传感器如雷达等才能获得，导致了该算法的局限性。

### 1.2.3 基于多传感器融合的 3D 目标检测研究现状

点云数据不提供在目标检测和分类任务中有着重要作用的纹理信息。单目视觉无法提供目标深度信息，而深度信息对于目标的空间位置估计和维度回归很有必要。另外点云的密度随距离增加而逐渐降低，而相机仍能捕捉远处小目标。鉴于此，为了提高整体性能，一些方法尝试将不同的传感器结合起来，通过不同的策略完成 3D 目标检测。Chen 等提出了 MV3D<sup>[21]</sup>，融合来自不同视图的逐个候选框的特征去预测旋转的 3D 边界框。Ku 等提出了 AVOD<sup>[22]</sup>，输入 RGB 图像得到全分辨率的特征图，然后通过区域裁剪和调整大小将上述的特征图进行融合，最后得到 3D 边界框从而完成 3D 目标检测任务。基于多传感器融合逐渐成为未来 3D 目标检测热门研究方向之一，但同时将多个传感器融合在一起实现 3D 检测也加深了系统的复杂性。

### 1.3 基于单目视觉的 3D 目标检测技术难点与本文主要贡献

基于单目视觉实现 3D 目标检测主要难点在以下几个方面：

(1) 目标空间位置坐标获取困难。单目相机只能获取 RGB 图像，目标的三维空间信息丢失。因为无法像深度相机或雷达等传感器一样得到目标的深度信息从而计算出目标在三维空间中的位置坐标，所以提升了基于单目视觉的 3D 目标检测难度。如何从 RGB 图像高效、准确得到目标的位置坐标是一个急需解决的难题。

(2) 目标的旋转角  $\theta_{yaw}$  与目标在 RGB 图像上呈现的外观无直接联系。由于基于单目视觉实现 3D 检测能获取到的可用信息均来自 RGB 图像。而目标在图像上呈现的外观与  $\theta_{yaw}$  相关性不强，即  $\theta_{yaw}$  并不随目标外观的变化而变化，所以很难通过提取目标的外观特征直接预测旋转角  $\theta_{yaw}$ 。

(3) 目标维度的三个分量具有很强的内在联系。维度的三个分量共同决定了目标在整个三维空间的形状，所以应该将维度各个分量视为一个整体来看待而非单独考虑。目前主流的损失函数仅仅将每个分量作为独立的个体来分别计算误差，从而造成了维度预测不精确以及异常检测等问题。所以需要将维度的三个分量作为一个整体考虑，加强内在联系，进而提高维度预测精确度。

针对以上基于单目视觉实现 3D 目标检测存在的难点问题，本文提出了一种基于 3D 边界框在成像平面上的投影会严格约束在 2D 边界框之内这一理论前提的 3D 目标检测框架，同时将上述理论前提称之为几何约束<sup>[8]</sup>。整个检测框架由 4 个模块组成，如下图 1-2 所示：

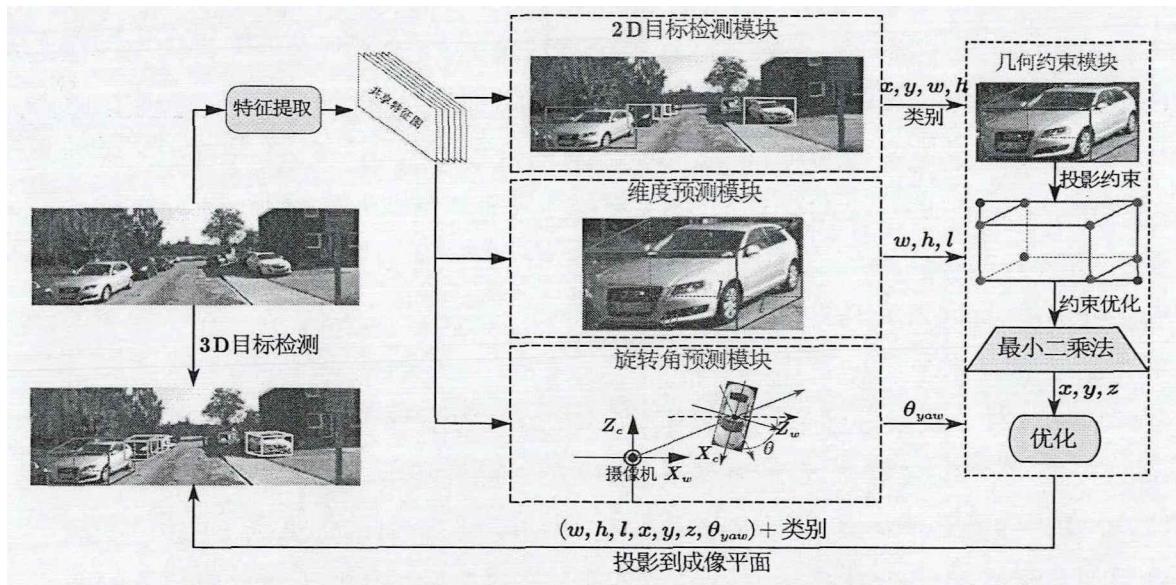


图 1-2 本文提出的基于单目视觉的 3D 目标检测框架示意图。

(1) 2D 目标检测模块。2D 目标检测模块负责预测目标的 2D 边界框和类别，且 2D 边界框会作为后续几何约束模块的输入数据。

(2) 维度预测模块。维度预测模块负责预测目标在三维空间中的维度，即 3D 边界框  $B^{3d}$  的  $w$ 、 $h$ 、 $l$  这 3 个参数。

(3) 旋转角预测模块。旋转角预测模块负责预测目标旋转角  $\theta_{yaw}$ 。由于目标的外观与  $\theta_{yaw}$  不具有直接的联系，所以本模块提出不直接预测  $\theta_{yaw}$ ，而是预测局部旋转角  $\theta_{alpha}$ 。在本文第三章中详细给出了  $\theta_{yaw}$  与  $\theta_{alpha}$  的区别和联系。

(4) 几何约束模块。本模块负责给出目标空间位置坐标  $(x, y, z)$ 。综合 2D 目标检测模块、维度预测模块、旋转角预测模块获取到的信息，通过几何约束这一理论前提，运用最小二乘法来解出  $(x, y, z)$ 。至此得到了目标的类别和 3D 边界框  $B^{3d} = (w, h, l, x, y, z, \theta_{yaw})$ ，完成了基于单目视觉的 3D 目标检测任务。

基于上面的内容，本文的贡献点主要分为以下几部分：

(1) 为了解决基于单目视觉实现 3D 目标检测的难题，本文将 3D 目标检测任务分成几个子任务，包括 2D 目标检测模块、维度预测模块、旋转角预测模块、几何约束模块。每一个子任务负责预测 3D 目标检测涉及到的特定参数，全部的子任务的完成意味着整个 3D 目标检测任务的完成。

(2) 本文提出了一种基于交并比（Intersection over Union，IoU）的维度预测损失函数。IoU 是 2D 目标检测评估算法性能阶段用来计算预测 2D 边界框与真值 2D 边界框重合度的指标。为了解决目标维度分量高、宽、长之间具有内在联系，需要将其作为一个整体而不是单独考虑的问题，本文将 IoU 引入到损失函数中，将应用于 2D 边界框的 IoU 扩展到三维空间，提出了计算 3D 边界框 IoU 的算法。实验证明本文提出适用于维度预测的 IoU 损失函数更具有优越性，为完成最终的 3D 目标检测任务提供良好的支撑。

(3) 本文提出了一种预测目标局部旋转角  $\theta_{alpha}$  的策略。由于目标在图像中的外观与  $\theta_{yaw}$  无直接的联系，但基于单目视觉实现 3D 目标检测仅仅提供了 RGB 图像信息，所以本文提出不直接预测  $\theta_{yaw}$ ，而是预测  $\theta_{alpha}$ 。 $\theta_{alpha}$  与目标的外观紧密联系。并且为了解决具有对称性目标旋转角预测模糊性的问题，本文将连续变量旋转角的回归问题转变成离散变量的分类问题，使得旋转角的预测更加精确。 $\theta_{yaw}$  可由  $\theta_{alpha}$  通过几何计算给出。实验证明本文提出的旋转角估计算法在相关评估指标上优于其它算法。

(4) 为了获取目标空间位置坐标  $(x, y, z)$ ，本文基于 3D 边界框在成像平面上的投影会严格约束在 2D 边界框之内这一前提理论，结合最小二乘法计算出  $(x, y, z)$ 。并且基于  $\theta_{alpha}$  预测的结果，将 4096 种约束可能降低为 64 种，减少了计算时间，提高整个系统检测效率。同时本文提出一种优化网络，对目标的位置坐标进行进一步的修正处理，提高最终 3D 目标检测的精确率。实验阶段本文以多个指标对提出的 3D 目标检测算法性能进行评估，结果表明与其他检测算法相

比，本文提出的算法在相关指标上具有更好的效果，证明了本文提出的基于单目视觉的 3D 目标检测算法的优越性。

## 1.4 本文结构安排

本文主要研究基于单目视觉的 3D 目标检测技术，根据内容可以分为以下六个章节：

第一章主要介绍相关研究背景和意义。分别介绍了 2D 目标检测与 3D 目标检测以及二者的区别。最后说明了本文的研究内容，主要贡献以及文章结构。

第二章介绍了本文提出的 3D 目标检测算法依赖的理论基础知识。主要介绍了相机的成像模型理论和 2D 目标检测技术。

第三章主要介绍本文提出的维度和旋转角估计算法。同时介绍了本文采用的多尺度 2D 目标检测算法。2D 检测是本文实现 3D 检测的基础。

第四章介绍本文依据的几何约束理论，在此基础上通过最小二乘法计算出目标空间位置坐标，同时提出了一种优化网络对计算出来的位置坐标进行修正，完成最终的 3D 目标检测任务。

第五章为本文的实验部分。首先介绍了本文进行实验的数据集。为了验证本文提出的 3D 目标检测框架的优越性，作者以多个指标对提出的 3D 目标检测算法性能进行评估，实验结果表明与其它算法相比，本文提出的算法有更高的检测精确度。

第六章对本文的工作进行总结，并对未来进行展望。



## 第二章 相机成像模型与 2D 目标检测

相机成像模型是图像处理方向基础知识之一，2D 目标检测是本文提出的 3D 目标检测算法的基础。本章在 2.1 节介绍了相机的小孔成像模型理论及相关成像坐标系，在 2.2 节介绍了基于传统方法和基于深度学习的 2D 目标检测技术。2.3 节对本章内容做相关总结。

### 2.1 相机成像模型理论

相机成像模型描述了三维空间中的某点与其在成像平面上相对应的投影点之间的数学关系。以针孔相机模型为例，其不考虑畸变，在计算机视觉领域的绝大多数应用场景中能满足精度需求。如下图 2-1 所示：

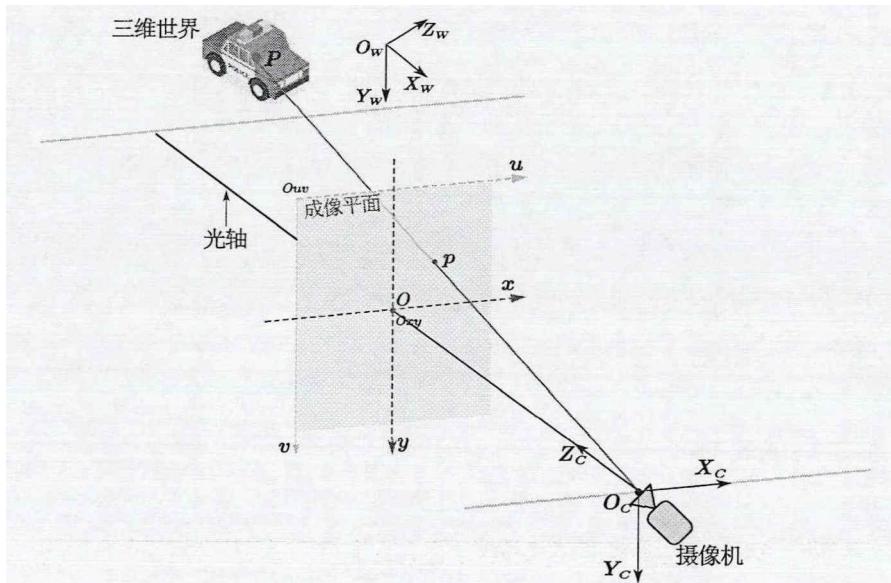


图 2-1 针孔相机模型示意图。

为了定量的描绘出整个投影过程涉及到的点的坐标转换，针孔相机模型引入了不同的坐标系统。以坐标轴  $X_w$ 、 $Y_w$ 、 $Z_w$  建立世界坐标系。设  $P$  点在该坐标系下的坐标为  $P_{world}(X_w, Y_w, Z_w)$ 。为了描述投影点  $p$  在成像平面的位置，以坐标轴  $x$ 、 $y$  建立图像坐标系， $p$  点相应的坐标为  $p_{image}(x, y)$ 。以坐标轴  $u$ 、 $v$  建立像素坐标系， $p$  点相应的坐标为  $p_{pixel}(u, v)$ 。以坐标轴  $X_c$ 、 $Y_c$ 、 $Z_c$  建立相机坐标系， $P$  点相应的坐标为  $P_{camera}(X_c, Y_c, Z_c)$ 。不同的坐标系之间相互联系且共同协作描述了整个投影过程中涉及到的点的坐标数学转换过程。坐标系建立原则和转换关系如下几小节所示。

## 2.1.1 参考坐标系

### (1) 世界坐标系

本质上世界坐标系没有固定的建立原则,或者说如何能更方便的描述应用场景便如何建立。本文研究的内容是3D目标检测,且应用场景为自动驾驶的交通场景。所以为了研究方便起见,本文建立世界坐标系的原则为设目标的中心为世界坐标系的原点 $O_w$ ,以目标前进的方向为 $X_w$ 的正方向, $Y_w$ 正方向垂直向下,且世界坐标系的建立满足右手定则。

### (2) 相机坐标系

相机坐标系遵循固定的建立原则。以相机的光心为坐标原点。 $Z_c$ 轴与相机的光轴重合,指向成像平面的方向为 $Z_c$ 轴的正方向。 $X_c$ 、 $Y_c$ 、 $Z_c$ 轴之间满足右手定则,如图2-1所示。

### (3) 图像坐标系

图像坐标系以图像的中心点 $O$ 为坐标原点 $o_{xy}$ , $x$ 和 $y$ 轴分别与 $X_c$ 和 $Y_c$ 轴平行。图像坐标系描述了图像中某点在图像中所处的位置,图像坐标系的坐标度量单位为毫米。

### (4) 像素坐标系

如图2-1所示,坐标轴 $u$ 、 $v$ 构成了像素坐标系。原点 $o_{uv}$ 在图像的左上角。 $u$ 轴和 $v$ 轴分别与相机坐标系的 $x$ 轴和 $y$ 轴平行。像素坐标系刻画了图像中每个像素的位置坐标,即该像素位于第几行和第几列。

## 2.1.2 坐标系之间的转换

### 2.1.2.1 世界坐标系与相机坐标系

世界坐标系与相机坐标系都是三维空间坐标系。二者通过平移矩阵 $T$ 和旋转矩阵 $R$ 建立坐标转换联系,如下图2-2所示:

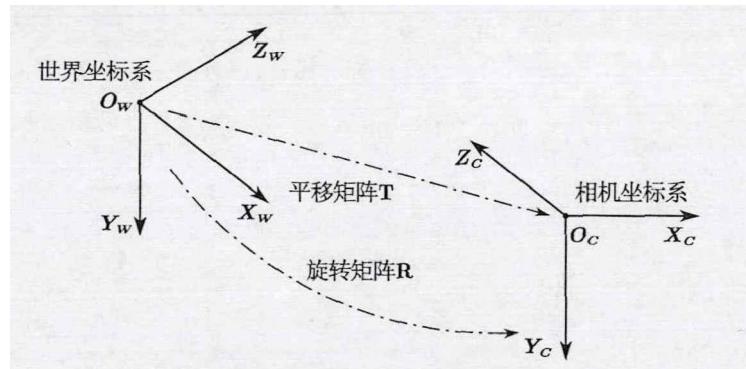


图2-2 世界坐标系与相机坐标系转换关系示意图。

设  $P$  在世界坐标系下的坐标为  $P_{world}(X_W, Y_W, Z_W)$ , 在相机坐标系下的坐标为  $P_{camera}(X_C, Y_C, Z_C)$ 。坐标  $P_{world}$  与  $P_{camera}$  通过  $R$  和  $T$  矩阵建立坐标转换关系。点  $P$  绕  $X$  轴旋转  $\theta_{pitch}$  的示意图如下所示:

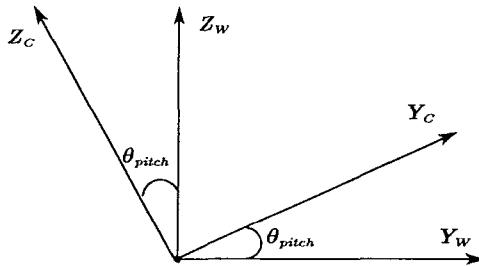


图 2-3 世界坐标系中的点绕坐标轴旋转示意图。

由图 2-3 可知,  $P$  点在世界坐标系下的坐标  $P_{world}$  与在相机坐标系下的坐标  $P_{camera}$  之间的坐标转换关系如下所示:

$$\begin{cases} X_C = X_W \\ Y_C = Y_W \cdot \cos \theta_{pitch} + Z_W \cdot \sin \theta_{pitch} \\ Z_C = -Y_W \cdot \sin \theta_{pitch} + Z_W \cdot \cos \theta_{pitch} \end{cases} \quad (2-1)$$

将上述公式用矩阵的形式表示如下所示:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{pitch} & \sin \theta_{pitch} \\ 0 & -\sin \theta_{pitch} & \cos \theta_{pitch} \end{bmatrix} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = R_X \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} \quad (2-2)$$

同理可以推导出世界坐标系中的点绕  $Y$  轴旋转  $\theta_{yaw}$  时相应的坐标转换关系如下所示:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \begin{bmatrix} \cos \theta_{yaw} & 0 & -\sin \theta_{yaw} \\ 0 & 1 & 0 \\ \sin \theta_{yaw} & 0 & \cos \theta_{yaw} \end{bmatrix} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = R_Y \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} \quad (2-3)$$

世界坐标系中的点绕  $Z$  轴旋转  $\theta_{roll}$  时相应的坐标转换关系如下所示:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \begin{bmatrix} \cos \theta_{roll} & \sin \theta_{roll} & 0 \\ -\sin \theta_{roll} & \cos \theta_{roll} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} = R_Z \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \end{bmatrix} \quad (2-4)$$

根据上述的公式 (2-2) - (2-4), 结合图 2-2 可知  $P_{world}(X_W, Y_W, Z_W)$  与  $P_{camera}(X_C, Y_C, Z_C)$  可以通过平移矩阵和旋转矩阵完成坐标转换, 如下所示:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} \quad (2-5)$$

上述公式 (2-5) 中, 对于矩阵  $R$ , 有  $R = R_X \cdot R_Y \cdot R_Z$ 。对于矩阵  $T$ , 有  $T = [T_x \ T_y \ T_z]^T$ , 其中  $T_x$ 、 $T_y$ 、 $T_z$  表示这两个坐标系原点之间的相对位置平移量。

### 2.1.2.2 相机坐标系与图像坐标系

本小节分析相机坐标系与图像坐标系之间的转换关系。如下图 2-4 所示:

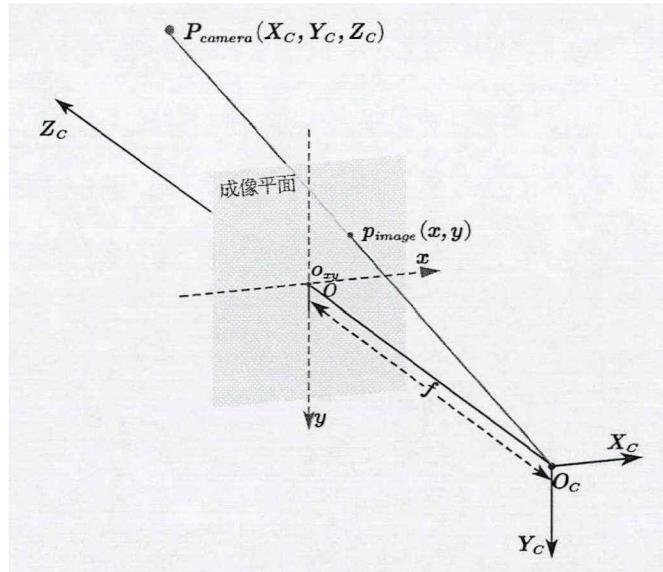


图 2-4 相机坐标系与图像坐标系关系示意图。

在图 2-4 中, 点  $P$  在相机坐标系下的坐标为  $P_{camera}(X_C, Y_C, Z_C)$ , 在图像坐标系下的坐标为  $p_{image}(x, y)$ 。其中  $f$  为相机的焦距。由几何比例关系, 坐标  $P_{camera}(X_C, Y_C, Z_C)$  与  $p_{image}(x, y)$  满足下面公式:

$$\begin{cases} \frac{x}{X_C} = \frac{f}{Z_C} \\ \frac{y}{Y_C} = \frac{f}{Z_C} \end{cases} \quad (2-6)$$

将上述公式 (2-6) 与公式 (2-5) 联立化简得:

$$\begin{aligned} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} &= \frac{1}{Z_C} \cdot \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} \\ &= \frac{1}{Z_C} \cdot \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \end{aligned} \quad (2-7)$$

上述公式(2-7)给出了 $P_{world}(X_w, Y_w, Z_w)$ 与 $p_{image}(x, y)$ 之间的数学转换关系。通过公式(2-7), 将三维空间中的点的坐标与其在成像平面的投影点的坐标建立了数学映射。给出一点 $P$ 的坐标, 便能计算相应的投影点在图像坐标系下的坐标。

### 2.1.2.3 图像坐标系与像素坐标系

本小节给出了图像坐标系与像素坐标系之间的转换关系。如下图 2-5 所示:

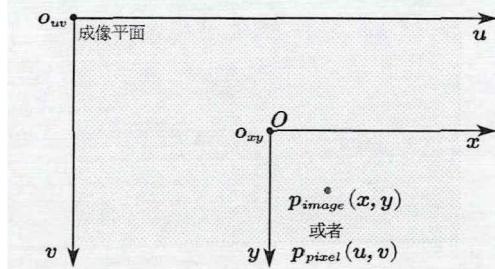


图 2-5 图像坐标系与像素坐标系关系示意图。

点 $p$ 在图像坐标系下的坐标为 $p_{image}(x, y)$ , 在像素坐标系下的坐标为 $p_{pixel}(u, v)$ 。 $p_{image}(x, y)$ 与 $p_{pixel}(u, v)$ 之间的转换关系如下所示:

$$\begin{cases} dx = \frac{x}{u - u_0} \\ dy = \frac{y}{v - v_0} \end{cases} \quad (2-8)$$

其中 $(u_0, v_0)$ 为图像中心 $O$ 的像素坐标。 $dx$ 、 $dy$ 分别表示一个矩形像素点在 $x$ 轴和 $y$ 轴上的长度。将上述公式用矩阵的形式表示得:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2-9)$$

联合公式(2-5)、(2-7)和(2-9), 可以得到 $P_{world}(X_w, Y_w, Z_w)$ 与 $p_{pixel}(u, v)$ 关系如下所示:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{Z_c} \cdot \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-10)$$

公式(2-10)也称之为投影坐标约束, 揭示了针孔相机模型的基本原理。投影过程涉及到的坐标转换关系可由公式(2-10)诠释。其中参数 $f_x$ 、 $f_y$ 分别是沿着 $u$ 轴和 $v$ 轴的等效焦距, 如下所示:

$$f_x = \frac{f}{dx}, f_y = \frac{f}{dy} \quad (2-11)$$

对公式 (2-10) 进行分析, 令:

$$\left\{ \begin{array}{l} K = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ M = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \end{array} \right. \quad (2-12)$$

将公式 (2-12) 带入到公式 (2-10) 中, 化简得:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{Z_c} \cdot K \cdot M \cdot \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-13)$$

公式 (2-12) 中, 矩阵  $K$  涉及到的参数有  $f_x$ 、 $f_y$ 、 $u_0$ 、 $v_0$ 。 $K$  仅仅与相机的内部参数有关, 所以将矩阵  $K$  称为内参矩阵。矩阵  $M$  由  $R$  和  $T$  共同决定, 是相机的外部参数, 所以也称矩阵  $M$  为外参矩阵。公式 (2-13) 中,  $Z_c$  为深度。公式 (2-13) 揭示了整个投影过程中涉及到的坐标转换关系。

## 2.2 2D 目标检测

在过去的 20 年间, 可以将 2D 目标检测的历史大致分为两个阶段, 早期的目标检测算法输入的目标特征大多是手工构建的。由于当时无法对图像进行有效的表示以及计算能力的匮乏, 人们只能通过设计复杂的算法来提取特征, 采用一系列的计算加速技巧来充分利用有限的硬件计算资源。随着技术的进步, 基于深度学习技术来完成 2D 目标检测任务逐渐成为主流。本节主要介绍不同的目标检测算法。

### 2.2.1 基于传统方法的 2D 目标检测

基于传统方法的 2D 目标检测技术主要通过手工构建复杂特征来完成相应的目标检测任务。2001 年 Viola 等提出了 Viola-Jones 检测器(Viola-Jones Detector), 在没有任何约束的情况下首次实现了实时人脸检测<sup>[23][24]</sup>。在保证同样的检测精度条件下, 运算速度提升了数十倍乃至数百倍。通过结合积分图像<sup>[25][26][27]</sup>等技术, 检测速度得到了巨大的提升。

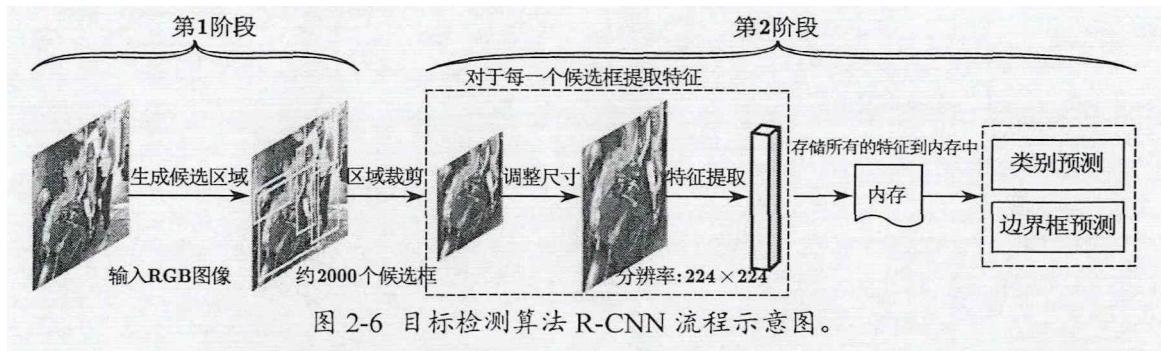
2005 年, Dalal 等提出了 HOG<sup>[28]</sup>算法, 用来平衡平移、缩放、光照等情况下特征的不变性和非线性。HOG 起初被设计用来处理行人检测任务, 但也可以用来检测各种类别。HOG 是许多目标检测算法的基础<sup>[29][30][31]</sup>。时间来到 2008 年, Felzenszwalb 等提出了 DPM<sup>[29]</sup>算法。DPM 将目标对象建模拆解成几个部分的组合。如对于人类来说, DPM 将人类视为头部、身体、腿、胳膊的组合。DPM 作为一种检测算法被多次使用<sup>[32][33]</sup>。

## 2.2.2 基于深度学习的 2D 目标检测

2012 年, Krizhevsky 等提出了 AlexNet<sup>[34]</sup>, 深度学习重新引起了广泛的关注。利用卷积神经网络 (Convolutional Neural Networks, CNN) 完成目标检测任务可以追溯到 2014 年, Girshick 等首次提出了 R-CNN (Region-CNN)<sup>[35]</sup>, 并在 PASCAL VOC 2012<sup>[36]</sup>数据集中取得了令人瞩目的表现, 由此证明了深度学习技术的优越性。所以也将 2014 年作为 2D 目标检测历史的分水岭。2D 目标检测算法可以被分为两种类型。两阶段检测器在第一阶段生成目标候选区域 (Region Proposals)。第二阶段完成目标的类别预测和边界框预测。由于生成目标候选区域的任务需要额外训练, 所以称之为两阶段检测算法。单阶段检测算法直接完成目标的类别和边界框预测任务, 不需要额外的生成目标候选区域, 故称之为单阶段检测算法。二者都有各自的优点和不足。下面几节详细介绍了两种不同的目标检测算法。

### 2.2.2.1 两阶段目标检测算法

2014 年, Girshick 等提出 R-CNN。R-CNN 也是第一个两阶段目标检测器。R-CNN 检测算法由四部分组成。首先通过选择搜索算法<sup>[37]</sup>生成与类别无关的候选框。之后通过 CNN 提取特征向量。通过分类器进行分类, 判断该特征向量对应的目标的类别。最后通过一个边界框回归模型对粗糙的边界框进行位置修正, 得到最终精确的目标边界框。R-CNN 网络架构如下图 2-6 所示:



R-CNN 目标检测算法表明将强大的深度学习技术应用到目标检测领域是可行的。但 R-CNN 也存在着明显的缺点如重复计算、分布处理、检测速度慢等。

为了解决 R-CNN 存在的问题，2015 年，Girshick 等在 R-CNN 工作的基础上提出了 Fast R-CNN (Fast Region-CNN) [38]。首先从输入图像中提取整体特征图，通过映射关系得到局部特征图。然后得到特征向量实现目标检测。与 R-CNN 算法将每一个目标候选框送入到 CNN 模型中相比，Fast R-CNN 只进行一次特征提取。同时将边界框预测任务整合到整个检测网络中，与分类任务共享卷积特征，将分类损失和边界框定位回归损失结合在一起统一训练。而 R-CNN 算法是分开进行的。但 Fast R-CNN 仍然通过选择搜索算法实现提取目标候选框，这一过程也存在着很大的时间消耗。Fast R-CNN 算法简要流程如下图 2-7 所示：

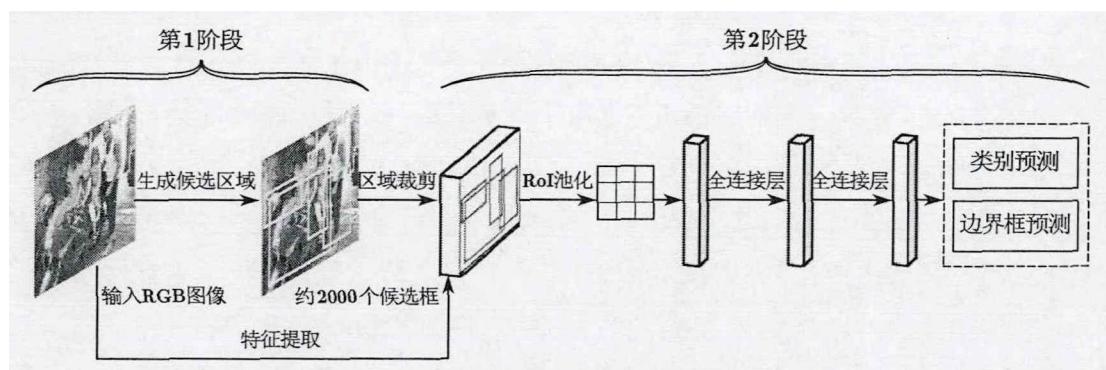


图 2-7 目标检测算法 Fast R-CNN 流程示意图。

在 Fast R-CNN 提出后不久，Girshick 等提出了 Faster R-CNN (Faster Region-CNN) [39]，优化了目标候选框提取过程。R-CNN 和 Fast R-CNN 都是通过选择搜索算法来获取相应的目标候选框，但是这种算法耗时严重。Faster R-CNN 通过使用 RPN 网络来得到目标候选框。遍历特征图，为每一个点都配置 9 种不同的候选框。候选框共有 3 种面积  $\{128^2, 256^2, 512^2\}$  和 3 种宽和高的比例  $\{1:1, 1:2, 2:1\}$ ，两两组合形成 9 种不同的目标候选框。然后通过 CNN 判断每个候选框是否有目标。有目标的候选框记为正样本，没有目标的候选框记为负样本。对正样本进行边界框精细化处理。最后通过非极大值抑制 [40] 算法得到预测目标边界框。如下图 2-8 所示：

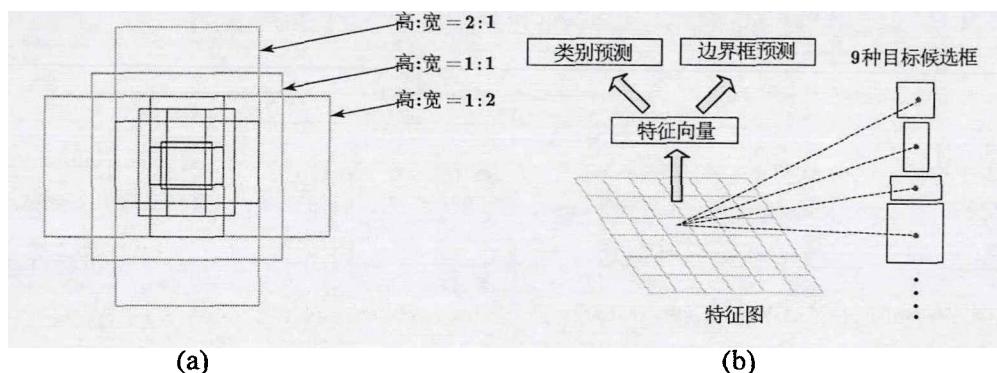


图 2-8 目标候选框与 RPN 网络示意图。(a) Faster R-CNN 算法中 9 种不同的目标候选框，图中绿色矩形框面积为  $512^2$  像素，红色矩形框面积为  $256^2$  像素，蓝色矩形框面积为  $128^2$  像素，(b) RPN 网络结构图，对于特征图的每一点，均会生成 9 种目标候选框。

Faster R-CNN 不需要额外训练，但由于需要生成目标候选区域，所以仍然称之为两阶段检测器。Faster R-CNN 目标检测简要流程如下图 2-9 所示：

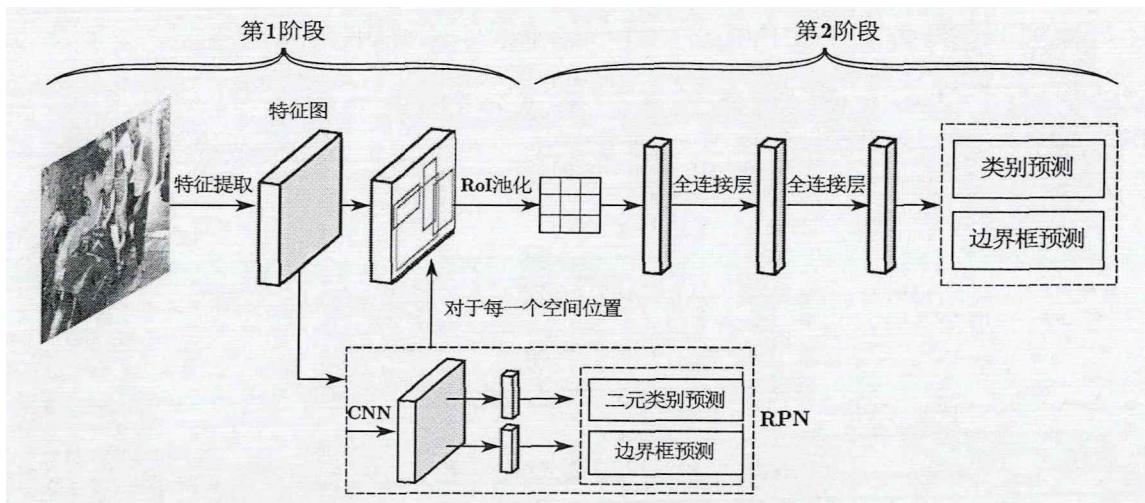


图 2-9 目标检测算法 Faster R-CNN 流程示意图。

### 2.2.2.2 单阶段目标检测算法

2015 年，Joseph 等提出了第一个单阶段目标检测器 YOLO (You Only Look Once)<sup>[41]</sup>。区别于两阶段的目标检测算法需要生成目标候选框，YOLO 直接预测目标的边界框和类别。由于边界框值的范围比较大，导致神经网络在一开始训练时不稳定，所以 YOLO 选择预测偏移量而不是坐标值。在 YOLO 的第 2 个版本 YOLO v2<sup>[42]</sup>中，通过在训练集中使用 K 均值聚类算法 (K-Means Clustering Algorithm) 预先设定边界框的尺寸，相比较于 Faster R-CNN 算法中手动设置边界框，使得网络更加稳定快速收敛。YOLO v2 预测偏移值  $(t_x, t_y, t_w, t_h)$ 。设模型预测的边界框为  $(b_x, b_y, b_w, b_h)$ ， $c_x$  和  $c_y$  为当前网格左上点的坐标， $p_w$  和  $p_h$  为预先设计的边界框的宽和高。参数之间的关系如下所示：

$$\begin{cases} t_x = \log((b_x - c_x)/(1 - (b_x - c_x))) \\ t_y = \log((b_y - c_y)/(1 - (b_y - c_y))) \\ t_w = \log(b_w/p_w) \\ t_h = \log(b_h/p_h) \end{cases} \quad (2-14)$$

由于不同的目标具有不同形状的边界框，比如对于车，其边界框近似为矮胖的矩形，对于行人，其边界框为高瘦的矩形。基于训练集数据预先设定边界框的大小，再以其为基准进行预测，回归坐标偏移量。由于这些值的范围很小，非常有利于检测网络收敛。

对于小目标检测，YOLO v2 精度仍然不高。小目标的像素信息随着卷积层不断加深而丢失，导致对小目标的检测效果比较差。为此，Joseph 等提出了 YOLO

v3<sup>[43]</sup>, 引入了多尺度预测, 使整个模型对于大中小目标均具有良好的处理效果。同时将 YOLO v2 中的特征提取网络从 19 层的 Darknet 修改为 53 层的 Darknet, 使得模型的预测能力大大增强。YOLO v3 网络结构如下图 2-10 所示:

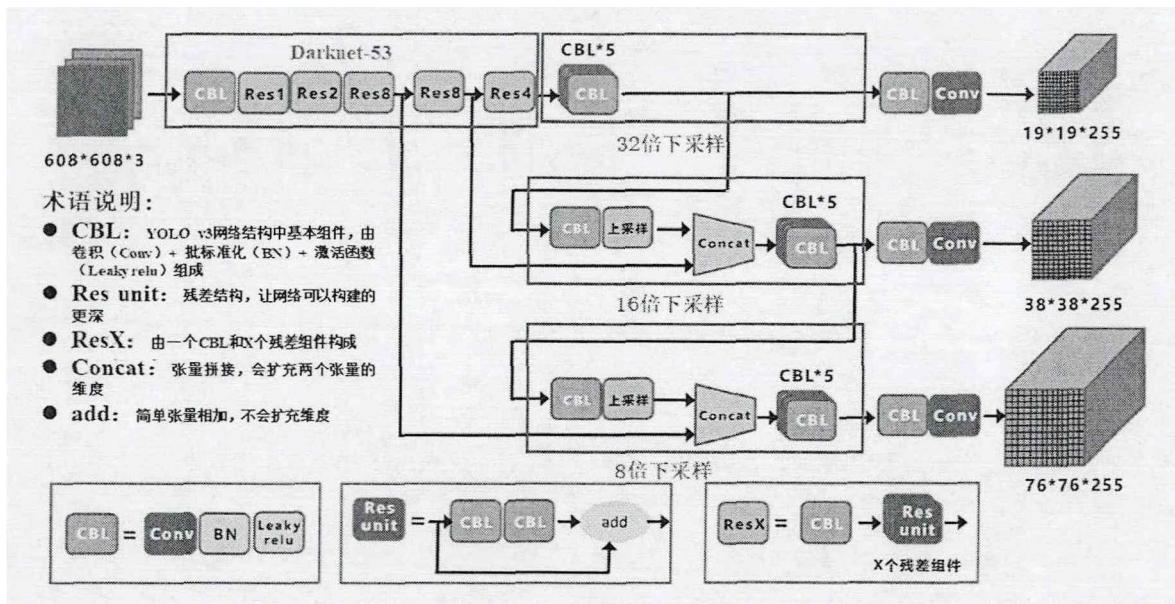


图 2-10 YOLO v3 网络结构示意图。

YOLO v3 经过 Darknet-53 抽取图像特征信息, 然后分成 3 个分支, 分别为 32 倍、16 倍、8 倍下采样, 对应的特征图中每个点的感受野由大到小, 分别去预测大、中、小目标, 从而预测更加准确, 相关性能指标也得到了提升。下表 2-1 给出了在 PASCAL VOC 和 MS COCO 数据集下, 单阶段和两阶段 2D 目标检测算法相关指标数据的对比结果:

表 2-1 不同目标检测算法评估指标对比结果。

算法	基础网络	mAP		AP MS COCO
		VOC 2007	VOC 2010	
R-CNN <sup>[35]</sup>	VGG-16	66.0	62.4	N/A
Fast R-CNN <sup>[38]</sup>	VGG-16	70.0	68.4	19.7
Faster R-CNN <sup>[39]</sup>	VGG-16	73.2	70.4	21.9
Faster R-CNN <sup>[39]</sup>	ResNet-101	76.4	73.8	N/A
YOLO <sup>[41]</sup>	VGG-16	66.4	57.9	N/A
YOLO v2 <sup>[42]</sup>	Darknet-19	78.6	73.5	21.6
YOLO v3 <sup>[43]</sup>	Darknet-53	N/A	N/A	33.0

从上表可以看出, 这些算法在公开的数据集上都能达到比较好的检测效果。同时可以看出, 选用不同的特征提取网络, 检测结果也不相同。如 Faster R-CNN

采用 ResNet-101 作为特征提取网络相比较于 VGG-16 在 mAP 指标上分别提升了 3.2% 和 3.4%。从表中数据可以看出，随着 YOLO 版本的迭代，检测性能也越来越好。YOLO v3 由于采用了多尺度采样和 Darknet-53 网络，相比较于 YOLO v2，在 AP 指标上有了显著的提升，达到了 11.4%。本文至此介绍了两类典型的 2D 目标检测算法。

### 2.2.2.3 2D 目标检测算法里程碑

本小节介绍其它 2D 检测算法。He 等在 Faster R-CNN 的基础上提出了 Mask R-CNN<sup>[44]</sup>，不仅可以完成目标检测，也可以实现实例分割。Mask R-CNN 通过将残差网络<sup>[45]</sup>和特征金字塔网络<sup>[46]</sup>结合，实现了更卓越的检测精确度和处理速度。Liu 等提出了单阶段的目标检测算法 SSD<sup>[47]</sup>，沿用了 YOLO 中直接回归边界框和分类概率的方法，同时又参考了 Faster R-CNN 中 RPN 生成目标候选区域的做法来提升识别准确度。Cheng 等在 SSD 的基础上提出了 DSSD<sup>[48]</sup>。SSD 在处理小目标时提取的浅层特征表征能力不够强，容易造成误检和漏检的现象。DSSD 使用了更好的基础特征提取网络和反向卷积，通过跳跃连接来解决该问题。Duan 等提出一种不需要提前设置锚的两阶段目标检测算法 CPNDet<sup>[49]</sup>。由于基于锚的检测算法如 Faster R-CNN 是通过实验和经验设置锚的尺寸，导致算法不够灵活也不能够对特殊形状的物体进行精确检测，所以作者通过角点提取目标候选框，然后进行边界框回归和类别预测。Li 等提出 TridentNet<sup>[50]</sup>，提出了一种尺度感知的训练方案，取得了精确的检测效果。Zhou 等提出 CenterNet<sup>[51]</sup>，通过将目标视作点模型不仅可以实现 2D 目标检测，还可以实现姿态估计、3D 目标检测等。下图 2-11 给出了基于时间轴的目标检测算法发展历史。

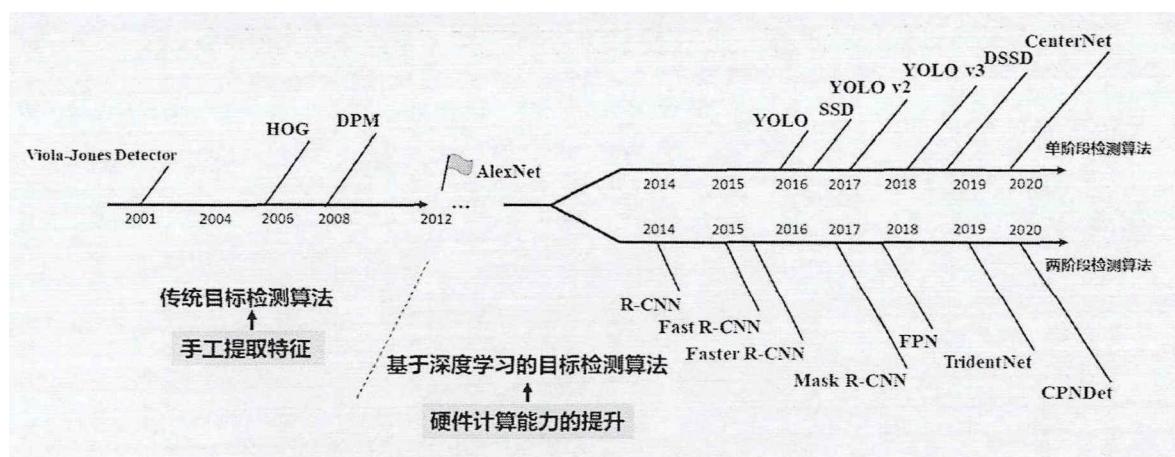


图 2-11 目标检测算法发展时间轴示意图。

2D 目标检测算法蓬勃发展。从图 2-11 来看，目标检测算法经历了从手工提取特征到基于深度学习的转变，2012 年 AlexNet 的提出是这个转变的分水岭。至

此以后，不论是单阶段还是两阶段目标检测算法都不断的被提出和改进，在检测精度和速度上都有了明显的提升。2D 目标检测算法越来越成熟，且不断的被应用到各个领域，解决了人们的特定需求。至此，本节完成了对 2D 目标检测原理和相关发展历史的介绍。

### 2.3 本章小结

本章主要介绍了与本文研究的 3D 目标检测方向相关的基础理论知识。相机的小孔成像模型一直是计算机视觉领域最基础也是最重要的知识之一。本文研究的基于单目视觉 3D 目标检测算法以 2D 目标检测为基础，因此重点介绍了 2D 检测算法原理和发展历史。至此本章完成了对本文提出的 3D 目标检测算法基础理论和相关技术的介绍。

## 第三章 3D 目标检测相关参数估计算法研究

本章主要研究 3D 目标检测任务中相关参数估计问题。3.1 节介绍了本文提出的基于维度均值策略的目标维度预测模块，在此基础上提出了一种基于交并比（Intersection over Union, IoU）的损失函数。3.2 节介绍了目标旋转角预测模块。本文提出不直接预测全局旋转角，而是预测局部旋转角，并将连续变量的目标旋转角回归问题转换成离散变量的分类问题。3.3 节介绍了参数估计网络以及本文采用的 2D 目标检测算法。3.4 节对本章内容做简单的总结。

### 3.1 目标维度估计算法研究

#### 3.1.1 基于维度均值策略的目标维度估计算法

对于同一类别的不同目标实例，其三维形状在一个很小的范围内波动，并且维度呈现出基于类别聚集分布的特征<sup>[8][10][52]</sup>。根据此先验条件，本文提出了一种基于训练集数据先验统计知识的维度预测策略。首先统计训练集数据中每个类别的所有目标实例平均维度，作为维度预测的基于类别先验值。即对于每一个类别，均有一个平均维度信息，记为  $\overline{dim} = (\bar{w}_i, \bar{h}_i, \bar{l}_i)$ ,  $i = 1, 2, \dots, K$ ，其中  $K$  为训练集中目标的类别数目。 $\bar{w}_i$ 、 $\bar{h}_i$ 、 $\bar{l}_i$  分别为每个类别所有目标的平均宽、高、长。为了使网络更好收敛，维度预测模块不直接预测目标的绝对维度，而是预测该目标的维度  $(w, h, l)$  与所属类别的平均维度  $(\bar{w}_i, \bar{h}_i, \bar{l}_i)$  之间的残差  $(\Delta w, \Delta h, \Delta l)$ 。其中  $\Delta w$ 、 $\Delta h$ 、 $\Delta l$  定义如下公式 (3-1) 所示：

$$\begin{cases} \Delta w = \ln(w/\bar{w}_i) \\ \Delta h = \ln(h/\bar{h}_i) \\ \Delta l = \ln(l/\bar{l}_i) \end{cases} \quad (3-1)$$

在维度预测网络的训练阶段，当前目标属于哪个类别，便预测该目标维度与相应的类别平均维度  $\overline{dim}$  之间的残差。通过相应的损失函数来惩罚网络，从而调整神经网络权重参数实现误差优化。预测阶段，网络的输出参数是一个三维向量  $(\Delta w, \Delta h, \Delta l)$ 。由于预先统计了训练集数据每个类别下所有目标实例的平均维度  $\overline{dim}$ ，并相应的记录下来，所以根据 2D 目标检测输出的类别得到对应的平均维度  $\overline{dim}$ ，然后计算出目标最终预测的维度。由平均维度和神经网络输出的残差计算出目标实际维度由公式 (3-1) 给出，实现了基于均值维度策略的目标维度预

测。整个维度预测网络充分利用了具有大量样本的训练集数据统计特征，所以维度预测网络更容易收敛，预测的结果更加精确。

### 3.1.2 基于交并比的维度预测损失函数

对于维度预测网络，本文提出了一种基于 IoU 的损失函数。IoU 是目标检测任务中用来评价检测算法性能优劣的一种尺度。IoU 刻画了两个矩形框的重叠程度，且具有尺度不变性，是两个矩形框的交集与并集的比值。下图 3-1 形象的刻画了 IoU 的物理意义和计算过程：

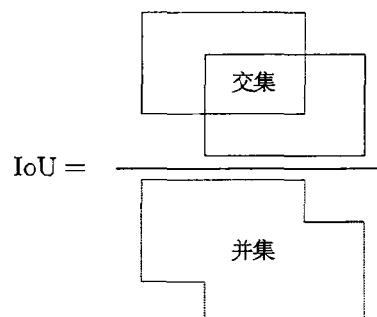


图 3-1 IoU 计算示意图。

传统的最小绝对误差（Least Absolute Error, LAE）<sup>[53][54]</sup>和最小平方误差（Least Square Error, LSE）<sup>[53][54]</sup>损失函数对每一个预测参数独立计算其相对于真值的误差，最后将所有的误差值求和作为最终的总误差，通过反向传播（Back Propagation）机制调整神经网络中权重和偏置，对误差进行优化。最终误差达到一个阈值，认为网络预测达到最优，此时停止迭代，完成神经网络的训练过程。但是本质上维度的三个分量  $w$ 、 $h$ 、 $l$  是紧密相关的，构成了一个整体的 3D 边界框，具有很强的内在联系。如果采用 LAE 或者 LSE 损失函数，在极端情况下，会出现维度的两个分量预测精度很高，与真值相比误差很低，另外一个维度分量预测精度很差，但计算出来的总误差呈现比较低的数值。最终导致整个网络输出的维度在某两个维度分量上预测准确率比较高，但是在另一个维度上预测准确度比较低，继而最终维度预测的结果不准确，降低了整个 3D 目标检测系统的性能。

为了预防此类情况发生，提高预测的精确度，本文深刻分析了维度三个值  $w$ 、 $h$ 、 $l$  的内在联系。由于维度预测的三个参数  $w$ 、 $h$ 、 $l$  是一个整体，三者共同构成了目标在三维空间的几何形状，具有很强的内在关系。所以本文将 2D 目标检测任务中评估度量尺度 IoU 引入到损失函数中。但传统的 IoU 评价的是二维平面中两个矩形边界框之间的重叠程度，而目标的维度是三维空间中的信息，具有  $w$ 、 $h$ 、 $l$  这三个参数。所以本文进一步将二维空间中的 IoU 算法扩展到三维空间，提出了一种基于 IoU 的维度预测损失函数。为了区分方便，本文将适用于 2D 边

界框的 IoU 算法记为  $\text{IoU}_{2D}$ ，将本文提出的评价 3D 边界框的 IoU 记为  $\text{IoU}_{3D}$ 。  
 $\text{IoU}_{3D}$  是衡量三维空间中两个立方体的重叠程度的算法，将维度的三个参数  $w$ 、 $h$ 、 $l$  作为一个整体来考虑。而不是和 LAE 或者 LSE 损失函数一样，将每个参数分开考虑。从而让最终的维度预测结果更加准确可靠。 $\text{IoU}_{2D}$  和  $\text{IoU}_{3D}$  如下图 3-2 所示：

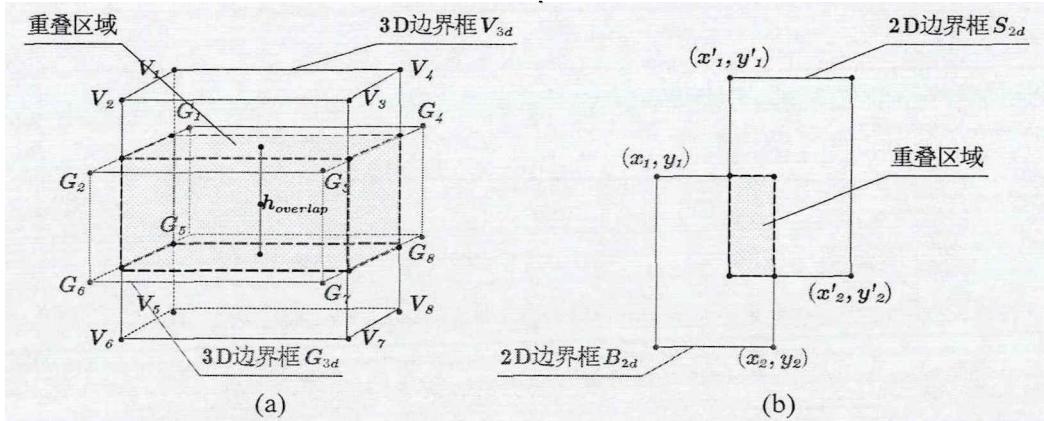


图 3-2  $\text{IoU}_{2D}$  与  $\text{IoU}_{3D}$  示意图。(a) 图中阴影部分为三维空间两个边界框  $V_{3d}$  和  $G_{3d}$  重叠部分。(b) 图中阴影部分为二维平面两个边界框  $B_{2d}$  和  $S_{2d}$  重叠部分。

为了计算  $\text{IoU}_{3D}$ ，本文在  $\text{IoU}_{2D}$  算法的基础上进行了扩展。本文提出的待计算  $\text{IoU}_{3D}$  的两个 3D 边界框是中心重合的，如图 3-2 中 (a) 图所示。设两个 3D 边界框  $V_{3d}$  和  $G_{3d}$  的底面面积分别为  $S_{V_{3d}}$  和  $S_{G_{3d}}$ ，高分别为  $h_{V_{3d}}$  和  $h_{G_{3d}}$ 。首先计算两个 3D 边界框下底面之间的重合区域面积记为  $S_{overlap}$ ，然后考虑  $h$  维度，计算高  $h$  部分的重叠长度  $h_{overlap}$ ，计算出 3D 边界框之间重合部分的体积，最终计算出  $\text{IoU}_{3D}$ 。计算  $\text{IoU}_{3D}$  的公式如下所示：

$$\text{IoU}_{3D} = \frac{S_{overlap} \cdot h_{overlap}}{S_{V_{3d}} \cdot h_{V_{3d}} + S_{G_{3d}} \cdot h_{G_{3d}} - S_{overlap} \cdot h_{overlap}} \quad (3-2)$$

其中  $0 \leq \text{IoU}_{3D} \leq 1$ 。当值为 0 时，表示 2 个 3D 边界框完全不重叠，当值为 1 时，表示两个 3D 边界框完全重叠。下面公式给出了 LAE 损失函数、LSE 损失函数、3D IoU 损失函数的计算过程：

$$\left\{ \begin{array}{l} loss_{LAE} = \sum_{i=1}^n |y_i - f(x_i)| \\ loss_{LSE} = \sum_{i=1}^n (y_i - f(x_i))^2 \\ loss_{\text{IoU}_{3D}} = 1 - \text{IoU}_{3D} \end{array} \right. \quad (3-3)$$

其中  $loss_{LAE}$ 、 $loss_{LSE}$ 、 $loss_{\text{IoU}_{3D}}$  分别为 LAE、LSE 和本文提出的维度预测损失函数。如公式 (3-3) 所示，在  $loss_{LAE}$  和  $loss_{LSE}$  中， $y_i$  为目标值， $f(x_i)$  为网络预测的输出值。 $loss_{LAE}$  本质上是将真值与预测值之间差值的绝对值之和最小

化，而 $loss_{LSE}$ 是为了将真值与预测值之间差值的平方之和最小化。 $loss_{LSE}$ 相比 $loss_{LAE}$ 对于异常样本具有更好鲁棒性。然而二者相比较于 $loss_{IoU_{3D}}$ ，均是将维度的三个分量宽、高、长单独考虑，没有将其作为一个整体。本文提出的维度预测损失函数的范围为 $0 \leq loss_{IoU_{3D}} \leq 1$ 。计算三维空间两个立方体IoU<sub>3D</sub>如下算法1所示：

---

#### 算法1：计算中心重合边界框的IoU<sub>3D</sub>

---

**输入：**两个3D边界框的维度， $B_{3d} = (w, h, l)$ ,  $B'_{3d} = (w', h', l')$

**输出：**两个3D边界框的IoU<sub>3D</sub>

---

**流程：**

- 1: 对于 $B_{3d}$ 和 $B'_{3d}$ ，计算 $w_{\max} = \max(w, w')$ ,  $l_{\max} = \max(l, l')$ ;
  - 2: 计算两个3D边界框对应的2D边界框， $B_{2d} = (-w/2, -l/2, w/2, l/2)$ ,  $B'_{2d} = (-w'/2, -l'/2, w'/2, l'/2)$ ，4个参数分别代表2D边界框的左下顶点和右上顶点坐标；
  - 3: 根据1的计算结果将两个2D边界框平移到第一象限，使其四个顶点坐标均为正数，平移后的坐标记为 $B_{2d} = (x_1, y_1, x_2, y_2)$ ,  $B'_{2d} = (x'_1, y'_1, x'_2, y'_2)$ ；
  - 4: 计算出两个3D边界框重合的体积 $V_{overlap}$ ；
  - 5: 计算两个3D边界框体积： $V_B = (w \times h \times l)$ ,  $V_{B'} = (w' \times h' \times l')$ ；
  - 6: 计算IoU<sub>3D</sub>；
- 

根据维度预测网络预测出的值通过公式(3-1)可以计算出目标的维度。然后通过算法1可以计算出三维空间两个立方体边界框的IoU<sub>3D</sub>。最终通过公式(3-3)得到维度预测损失函数 $L_{dim} = loss_{IoU_{3D}}$ 。综上所述完成了维度参数的估计。

## 3.2 目标旋转角估计算法研究

### 3.2.1 旋转角分类预测策略

目标的旋转角是连续变量。对于连续变量的预测本质上是一个回归<sup>[53]</sup>问题，回归预测会给出一个具体的连续值。而分类<sup>[54]</sup>主要解决离散型变量的归属问题，分类预测会给出当前变量所属的类别。回归与分类在本质上都是对于一系列样本数据 $(x, y)$ ，建立 $f(x) \rightarrow y$ 的一种映射。但回归与分类在某种情况下有所重叠。给连续型变量赋予分类类别概率的属性，即每个连续变量均归属于某个类别。在神经网络的训练过程中，调整相关权重参数，通过损失函数反向传播，降低该连续变量与类别之间的误差。在预测阶段该变量属于哪个类别，便由该类别来负责预测。从而将连续变量的回归问题转换成离散变量的分类问题。

虽然旋转角预测问题是一个连续变量的回归问题，然而本文指出采用回归的方式解决旋转角的预测问题在预测时可能会造成角度模糊性。对于一个具有几何对称的目标，从不同的角度看去，具有非常高的几何相似度，如对于车的前后两个表面或者左右两个表面。这样卷积神经网络在提取图片特征来回归相应的旋转角时，会出现角度预测模糊现象<sup>[55]</sup>。即不同的旋转角下的同一目标可能会提取到相同的特征。从而出现旋转角预测完全相反的情况。本文对这一现象进行了分析，下图 3-3 给出了具有几何对称特性目标的不同侧面的几何相似度关系：

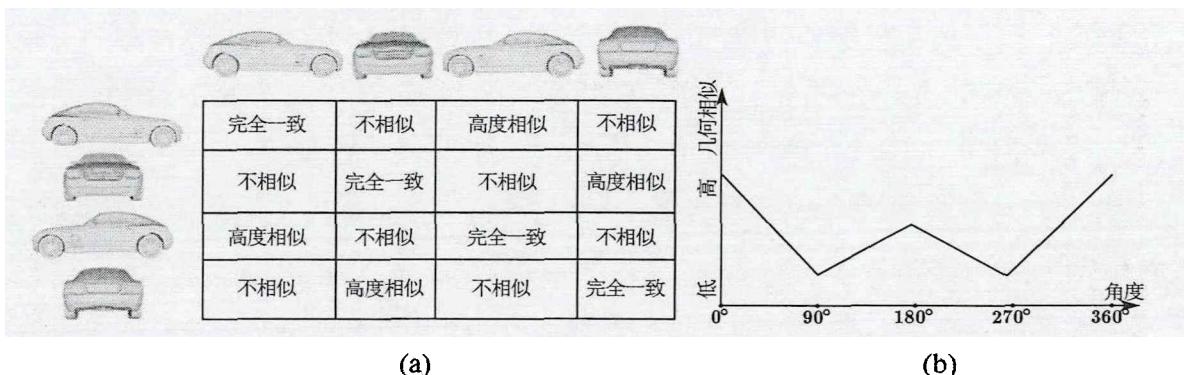


图 3-3 具有几何对称性的目标不同侧面关系示意图。(a) 图为目标不同侧面的几何相似程度，有完全一致，高度相似，不相似 3 种等级。(b) 图简易描述了随观察角度的变化几何相似程度变化曲线。

为了解决旋转角回归时可能产生的角度预测模糊的问题，本文选择将连续变量的角度回归问题转换成离散变量的分类问题。将旋转角的范围  $[0, 2\pi]$  离散为  $N_b$  个等长的区间  $\Theta_i$ ，其中  $i = 0, 1, 2, \dots, N_b - 1$ ， $\Theta_i$  的范围如下所示：

$$\Theta_i = \left\{ \theta \in [0, 2\pi) \mid \frac{2\pi}{N_b} \cdot i \leq \theta < \frac{2\pi}{N_b} \cdot (i + 1) \right\} \quad (3-4)$$

在旋转角预测网络训练阶段，训练集数据目标的真值旋转角  $\theta_{i_0}$  在哪个区间  $\Theta_i$  范围内，便由该区间来负责预测该旋转角。这样做即使对于相似表面抽取到了类似的特征，但由于角度  $\theta_{i_0}$  属于不同的区间，所以也就在一定程度上避免了角度预测时出现的模糊问题。并且本文不直接预测  $\theta_{i_0}$ ，而是预测  $\theta_{i_0}$  与区间  $\Theta_i$  的中心角度  $\Theta_{i\_c}$  的残差值  $\Delta\theta_{i_0}$ 。其中  $\Theta_{i\_c}$  计算公式如下所示：

$$\Theta_{i\_c} = \frac{\left( \frac{2\pi}{N_b} \cdot i + \frac{2\pi}{N_b} \cdot (i + 1) \right)}{2} \quad (3-5)$$

为了更好的表达残差角  $\Delta\theta_{i_0}$ ，本文提出将预测  $\Delta\theta_{i_0}$  转换成预测  $\sin(\Delta\theta_{i_0})$  和  $\cos(\Delta\theta_{i_0})$ 。根据  $\sin(\Delta\theta_{i_0})$  和  $\cos(\Delta\theta_{i_0})$ ，可以得到  $\Delta\theta_{i_0}$  的计算公式如下所示：

$$\Delta\theta_{i_0} = \arctan \left( \frac{\sin(\Delta\theta_{i_0})}{\cos(\Delta\theta_{i_0})} \right) \quad (3-6)$$

训练阶段网络输出  $N_b$  个 3 维向量  $[c_i, \sin(\Delta\theta_i), \cos(\Delta\theta_i)]$ ,  $i = 0, 1, \dots, N_b - 1$ ,  $c_i$  为每个区间  $\Theta_i$  的置信度, 当前真值角度落在哪个  $\Theta_i$ , 那么期望神经网络对该  $\Theta_i$  输出  $c_i$  为 1, 其余区间为 0。旋转角预测网络的预测阶段, 选取具有最大  $c_i$  的那个区间的输出作为对输入角度的预测, 然后通过公式 (3-6) 计算出残差  $\Delta\theta_{i_0}$ 。根据残差  $\Delta\theta_{i_0}$  和相应的区间  $\Theta_i$ , 得到最终的预测旋转角。

整个网络的损失函数  $L_\theta = L_{conf} + wL_{pre}$  由两部分组成,  $w$  为超参数。 $L_{conf}$  为交叉熵损失 (Cross Entropy Loss) [56]。对于每一个区间  $\Theta_i$  需要计算其预测的置信度损失。负责预测目标旋转角的区间  $\Theta_i$  输出  $c_i$  期望为 1, 其它不负责预测的区间  $\Theta_i$  输出  $c_i$  为 0。是一个二分类问题, 采用交叉熵损失能达到比较好的效果。

将  $p$  和  $q$  定义为两个概率分布, 用  $q$  来表示  $p$  的交叉熵为:

$$H(p, q) = - \sum_x p(x) \log q(x) \quad (3-7)$$

交叉熵表示的是两个分布之间的距离。其中  $p$  代表真值数据,  $q$  表示预测结果。 $p$  和  $q$  的分布越接近则预测结果越准确。在神经网络中, 利用 Softmax<sup>[53]</sup> 函数将前向传播的结果转换为概率分布。所以交叉熵损失函数由两部分构成。一部分为 Softmax 函数, 另一部分为交叉熵。假设原始神经网络的输出是  $y_1, y_2, y_3, \dots, y_n$ , 则经过 Softmax 函数的输出结果为:

$$q_i = \text{Softmax}(y_i) = \frac{e^{y_i}}{\sum_{j=1}^n e^{y_j}} \quad (3-8)$$

用交叉熵作为置信度损失函数, 则有:

$$L_{conf} = - \sum_{i=1}^n p_i \log(q_i) \quad (3-9)$$

对于预测的旋转角损失函数  $L_{pre}$ , 由两部分组成。其中残差损失  $L_{res}$  采用最小平方误差损失函数:

$$L_{res} = (\sin(\Delta\theta_{i_0}) - \sin(\widehat{\Delta\theta}_{i_0}))^2 + (\cos(\Delta\theta_{i_0}) - \cos(\widehat{\Delta\theta}_{i_0}))^2 \quad (3-10)$$

其中  $\Delta\theta_{i_0}$  是真值旋转角  $\theta_{i_0}$  与区间  $\Theta_i$  中心角度的残差值。 $\widehat{\Delta\theta}_{i_0}$  为旋转角预测网络输出的残差值。同时本文增加了一个额外的损失函数  $L_{cnt}$  来保证  $\sin^2(\widehat{\Delta\theta}_{i_0}) + \cos^2(\widehat{\Delta\theta}_{i_0}) = 1$ 。 $L_{cnt}$  定义如下所示:

$$L_{cnt} = (1 - (\sin^2(\widehat{\Delta\theta}_{i_0}) + \cos^2(\widehat{\Delta\theta}_{i_0})))^2 \quad (3-11)$$

综上所述, 损失函数  $L_{pre}$  如下所示:

$$L_{pre} = L_{res} + L_{cnt} \quad (3-12)$$

### 3.2.2 局部旋转角与全局旋转角

目标的旋转角分为局部旋转角  $\theta_{alpha}$  和全局旋转角  $\theta_{yaw}$ 。全局旋转角  $\theta_{yaw}$  刻画了世界坐标系与相机坐标系之间的几何旋转关系，直接反映了目标在相机坐标系下的位姿。同时  $\theta_{yaw}$  也是 3D 目标检测中需要求解的参数之一。而局部旋转角  $\theta_{alpha}$  直接反映了目标在 RGB 图像中呈现的外观。如下图 3-4 所示：

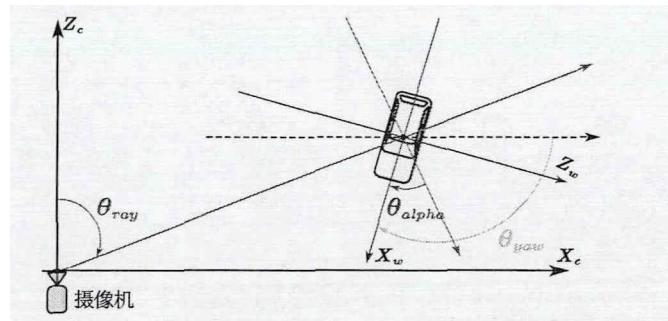


图 3-4 局部旋转和全局旋转角关系示意图。红色箭头示意的角为局部旋转角，橘色箭头示意的角为全局旋转角。

由于目标的外观与局部旋转角密切相关，所以本文选择预测目标的局部旋转角而不是全局旋转角。如下图 3-5 中 (a) 图所示，当目标从摄像机视野的左边移动到右边时，目标虽然处在不同的位置但是目标的全局旋转角没有发生变化，然而目标在图像中的外观和局部旋转角却发生了变化；同理，如下图 3-5 中 (b) 图所示，目标在摄像机视野中绕圈运动，在不同的位置目标的全局旋转角发生了变化，但是目标的局部旋转角和目标的外观均没有发生变化。即目标的外观与局部旋转角密切相关。局部旋转角和全局旋转角与目标外观的关系如下图所示：

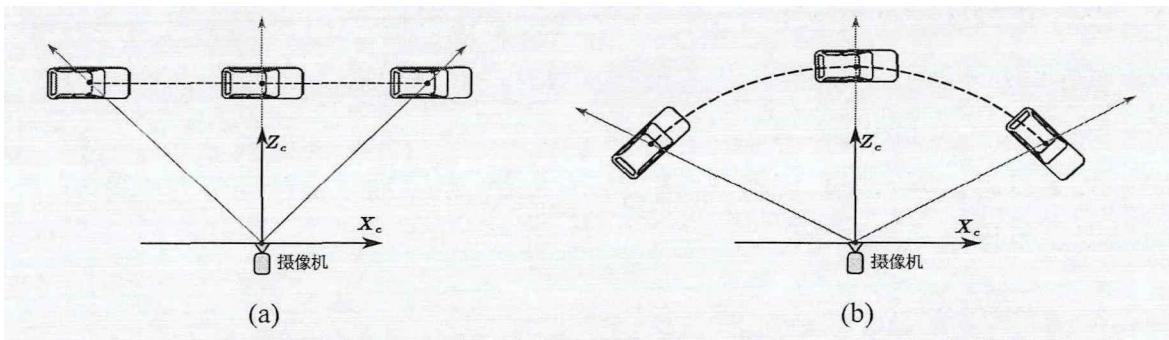


图 3-5 局部和全局旋转角与目标外观关系示意图。(a) 汽车直行, (b) 汽车绕圈运动。

### 3.2.3 角度转换

由于本文提出的旋转角预测模块没有直接预测目标的全局旋转角  $\theta_{yaw}$ ，而是局部旋转角  $\theta_{alpha}$ 。所以为了由  $\theta_{alpha}$  求解出  $\theta_{yaw}$ ，本文分析了  $\theta_{alpha}$  与  $\theta_{yaw}$  之间的几何关系，由上图 3-4 所示，可以得到如下结论：

$$\begin{cases} \theta_{yaw} = \theta_{ray} + \theta_{alpha} \\ \theta_{ray} = \arctan\left(\frac{x}{z}\right) \end{cases} \quad (3-13)$$

其中参数  $x$  和参数  $z$  分别是目标中心在相机坐标系下的坐标  $(x, y, z)$  在  $X_c$  轴和  $Z_c$  轴上的分量值。由第二章介绍的投影过程中点的坐标转换关系且仅仅考虑内参，可以得到：

$$z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3-14)$$

其中  $(u, v)$  为目标中心在成像平面的投影点的像素坐标。 $f_x$  和  $f_y$  为等效焦距。 $(u_0, v_0)$  是中点坐标。并且摄像机的内参在 3D 目标检测任务中均是默认已知的。化简 (3-15) 式得：

$$\begin{cases} zu = xf_x + u_0 z \\ zv = yf_y + v_0 z \end{cases} \quad (3-15)$$

由 (3-15) 式解得：

$$\frac{x}{z} = \frac{u - u_0}{f_x} \quad (3-16)$$

本文假设目标中心投影到成像平面的投影点为 2D 边界框的中心，由于已经根据 2D 检测算法得到目标的 2D 边界框  $B^{2d} = (x^{2d}, y^{2d}, w^{2d}, h^{2d})$ 。从而能解出 2D 边界框左上顶点坐标  $(X_{min}, Y_{min})$  和右下顶点的坐标  $(X_{max}, Y_{max})$ 。根据假设关系，即有  $(u, v) = (X_{max} - X_{min}, Y_{max} - Y_{min})$ 。此时  $u$ ,  $u_0$ ,  $f_x$  都是已知的，由 (3-13) 式可以解出  $\theta_{ray}$ ，继而解出全局旋转角  $\theta_{yaw}$ 。从而通过估计局部旋转角  $\theta_{alpha}$  间接得到目标全局旋转角  $\theta_{yaw}$ ，完成对旋转角参数的估计。

### 3.3 参数估计网络与多尺度 2D 目标检测

#### 3.3.1 参数估计网络

综合上面两小节介绍的维度和旋转角估计算法，本小节给出 3D 目标检测相关参数估计网络的结构和相关实验参数细节说明。维度和旋转角参数估计网络结构如下图 3-6 所示：

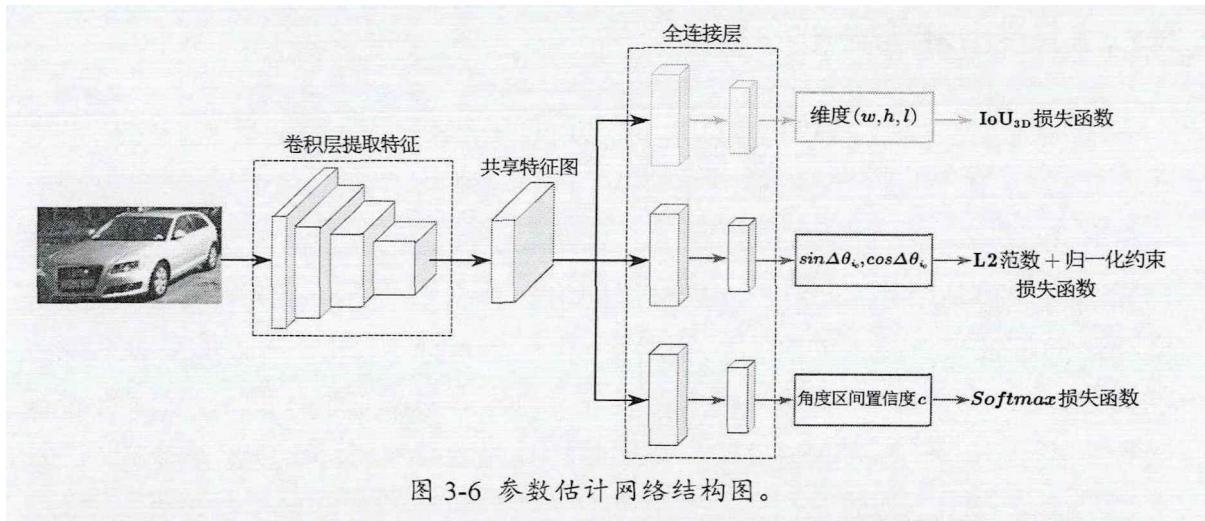


图 3-6 参数估计网络结构图。

如上图所示，通过训练集给定的标签数据，将图像中的相应目标区域进行裁剪调整尺寸统一为 $224 \times 224$  像素大小输入到网络中。本文采用 VGG-19<sup>[57]</sup> 网络提取目标相应的特征，并舍弃其全连接层。共享特征分为三个分支，每个分支接全连接层。维度预测分支输出三个参数宽、高、长，通过本文提出的IoU<sub>3D</sub>损失函数实现维度的精确预测。旋转角预测分支输出区间个数 $N_b$ 组参数，本实验取 $N_b = 2$ 。每组参数由 $\sin(\Delta\theta_{i_0})$ 和 $\cos(\Delta\theta_{i_0})$ 构成。置信度分支输出 $N_b$ 个参数 $c_i$ ，对应每个区间的置信度。网络的训练阶段，当目标旋转角落在哪个区间，那么对应该区间的 $c_i$ 为 1，其它区间的 $c_i$ 为 0。预测阶段，通过 2D 目标检测技术得到目标的 2D 边界框，然后将其送入到已经训练好的参数估计网络中，通过维度预测分支得到目标的维度，通过区间置信度分支输出的 $N_b$ 个区间的置信度，选取置信度最大的那个区间，从对应的旋转角分支中得到相应的 $\sin(\Delta\theta_{i_0})$ 和 $\cos(\Delta\theta_{i_0})$ ，根据公式 (3-6) 得到相应的残差角，继续结合区间信息得到最终的局部旋转角 $\theta_{alpha}$ ，即得到了全局旋转角 $\theta_{yaw}$ 。整个网络的损失函数为多任务损失函数，如下所示：

$$L = \alpha \times L_{dim} + L_\theta \quad (3-17)$$

其中 $L_{dim}$  为维度损失函数， $L_\theta$  为旋转角损失函数， $\alpha$  平衡二者设置的超参数，本文设置 $\alpha = 0.6$ ，对于损失函数 $L_\theta = L_{conf} + wL_{pre}$ ，设置超参数 $w = 0.4$ 。网络采用随机梯度下降 (Stochastic Gradient Descent, SGD) 优化网络，通过反向传播调整网络权重。设置网络的学习率 $lr = 0.0001$ ，批处理数据量 $batch\_size = 8$ ，网络进行 20000 次迭代，根据验证集选择最好的某次迭代模型。本文添加了颜色失真，并随机对图像进行镜像翻转，使得网络更具有鲁棒性。

### 3.3.2 多尺度 2D 目标检测

本文在第二章中详细介绍了两阶段 2D 目标检测算法 Faster R-CNN<sup>[39]</sup>。由于本文提出的 3D 目标检测框架依赖于 2D 目标检测的结果，所以 2D 目标检测的精确与否十分重要。自动驾驶场景下具有很多小目标交通车辆。这些目标在 RGB 图像上仅仅占几个像素。对于卷积神经网络来说，浅层网络输出的特征与深层网络输出的特征具有很大的区别。深层网络由于感受野比较大，所以语义信息表征能力强，但是输出特征图的分辨率比较低，几何信息表征能力弱。浅层网络的感受野比较小，输出的特征图虽然具有很强的几何细节信息，特征图分辨率高，但是语义信息表征能力弱。在自动驾驶场景下，相机获得的图像中既包含大目标，又包含小目标，是一个多尺度目标的集合。而 Faster R-CNN 没有很好的考虑到此应用场景。Faster R-CNN 提出的 RPN 网络预先设定 9 种固定尺寸的锚框作为先验边界框，然后对固定尺寸的特征图的每一个点进行枚举边界框，最后通过精细化处理得到最终的边界框。然而这会对那些仅仅占据很少像素点的小目标造成漏检的现象。

为此本文采用 Cai 等提出的 MS-CNN<sup>[58]</sup>用于多尺度目标检测。MS-CNN 由两个子网络组成，区域建议子网络和检测子网络。两个子网络都是端到端的，也是权重共享的。区域建议子网络输出不同的分支，每个分支都连接了不同的检测层，负责一定的尺度范围目标预测。所以本质上 MS-CNN 网络是 Faster R-CNN 的多尺度目标检测版本。这也是一些基于 2D 目标检测算法完成 3D 目标检测工作<sup>[8]</sup>所采用的算法。结合实际场景和为了后续算法性能对比公平性，本文也采用 MS-CNN 作为提出的基于单目视觉完成 3D 目标检测依赖的 2D 目标检测算法。本文提出的参数估计和 2D 目标检测处理流程如下图 3-7 所示：

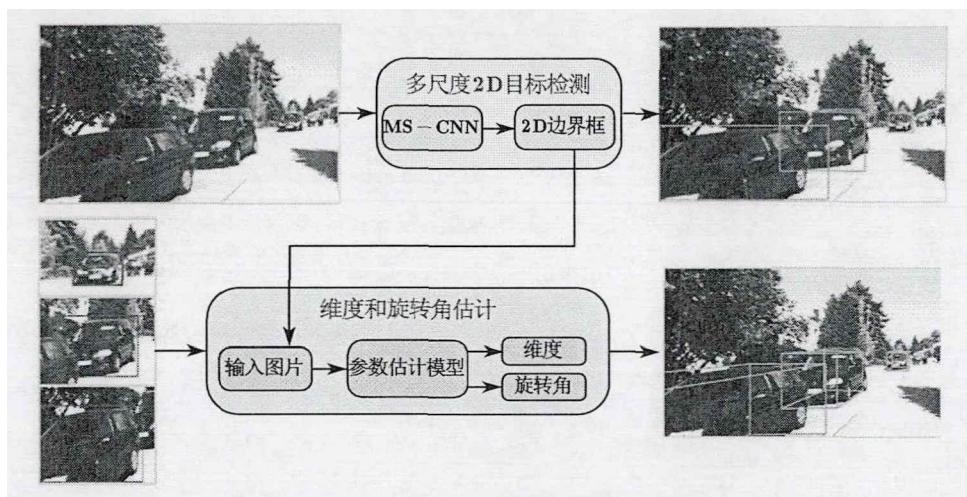


图 3-7 结合 2D 目标检测的参数预测网络处理流程图。

如上图 3-7，在预测阶段，参数估计网络的处理步骤如下所示：

(1) 通过多尺度 2D 目标检测算法 MS-CNN, 得到目标的 2D 边界框和目标相应的类别。在此过程中判断当前 2D 边界框是否合理, 即 2D 边界框的坐标都应该是正值。对于错误的 2D 边界框不做任何处理。

(2) 根据 2D 边界框对 RGB 图像上的相应目标进行裁剪统一尺寸, 并对数据进行预处理。得到单个目标区域送入到维度和旋转角估计模块。通过神经网络提取目标特征, 全连接层的不同分支输出相应的不同参数。

(3) 由步骤 (2) 得到目标的维度和局部旋转角  $\theta_{alpha}$ 。由于输出的维度是相对于类别均值维度  $\overline{dim}$  的残差  $(\Delta w, \Delta h, \Delta l)$ , 所以根据 MS-CNN 检测算法得到的目标类别和该类别预先统计的平均维度  $\overline{dim}$  推断出该预测目标绝对维度  $(w, h, l)$ 。由于神经网络输出的旋转角是局部旋转角  $\theta_{alpha}$ , 所以根据 3.2 小节中的  $\theta_{yaw}$  和  $\theta_{alpha}$  之间的几何关系, 可以计算出所需的全局旋转角  $\theta_{yaw}$ 。至此得到了目标的维度和旋转角, 完成对 3D 目标检测部分参数的估计。

## 3.4 本章小结

本章主要研究基于单目视觉的 3D 目标检测相关参数估计算法。详细介绍了本文提出的维度和旋转角估计算法。同时针对自动驾驶场景下出现的多尺度目标检测问题, 采用 MS-CNN 多尺度 2D 目标检测技术得到目标的精确 2D 边界框, 为后面完成整个 3D 目标检测提供精确的数据支撑。最后给出了结合 MS-CNN 目标检测算法的参数估计模型处理过程图。



## 第四章 基于几何约束的 3D 目标检测与参数优化

本章主要介绍本文提出的基于几何约束的 3D 目标检测框架。基于第三章中 3D 目标检测相关参数估计的结果，完成对最后目标的位置坐标估计。在 4.1 节中详细介绍了本文提出的几何约束理论，并结合最小二乘法（Least Squares Method）计算出位置坐标。在 4.2 节中分析了利用几何约束理论计算坐标存在的问题，并在此基础上采用一种优化网络对位置坐标进行误差修正。4.3 节对本章内容进行总结。

### 4.1 几何约束理论

#### 4.1.1 最小二乘法

最小二乘法是一种数学优化算法，广泛应用于统计学领域。该算法的核心思想是寻找一种函数模型，使得模型数据与观测到的样本数据之间的误差平方和最小。本小节以  $n$  元线性回归模型为例，推导最小二乘法的矩阵形式的解<sup>[59]</sup>。对于从总体中获取到的  $m$  组样本观察值  $(x_{i1}, x_{i2}, \dots, x_{in}; y_i), i = 1, 2, 3, \dots, m$ ，将其写成方程组的形式，如下所示：

$$\left\{ \begin{array}{l} \theta_1 x_{11} + \theta_2 x_{12} + \dots + \theta_n x_{1n} = y_1 \\ \theta_1 x_{21} + \theta_2 x_{22} + \dots + \theta_n x_{2n} = y_2 \\ \dots \\ \theta_1 x_{m1} + \theta_2 x_{m2} + \dots + \theta_n x_{mn} = y_m \end{array} \right. \quad (4-1)$$

上述的 (4-1) 式可能无解，即任意的一组参数  $\theta(\theta_1, \theta_2, \dots, \theta_n)$  都无法使得上述方程组成立。然而最小二乘法需要寻找一组参数  $\theta$ ，使得对于下面的误差平方和具有最小的值，并且称这样的解  $\theta$  为最小二乘解：

$$\sum_{i=1}^m (\theta_1 x_{i1} + \theta_2 x_{i2} + \dots + \theta_n x_{in} - y_i)^2 \quad (4-2)$$

对于 (4-1) 式，为了推导方便，将其写成矩阵的形式。设自变量矩阵为  $A$ ，函数值矩阵为  $b$ ，系数矩阵为  $x$ ，令  $y = Ax$ ，如下所示：

$$A = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix} \quad (4-3)$$

$$b = [y_1 \ y_2 \ \dots \ y_m]^T \quad (4-4)$$

$$x = [\theta_1 \ \theta_2 \ \dots \ \theta_n]^T \quad (4-5)$$

$$y = Ax = \left[ \sum_{i=1}^n \theta_i x_{1i} \ \sum_{i=1}^n \theta_i x_{2i} \ \dots \ \sum_{i=1}^n \theta_i x_{ni} \right]^T \quad (4-6)$$

将上述公式 (4-3) - (4-6) 带入到 (4-2) 式中化简得:

$$|y - b|^2 = |Ax - b|^2 = \sum_{i=1}^m (\theta_1 x_{i1} + \theta_2 x_{i2} + \dots + \theta_n x_{in} - y_i)^2 \quad (4-7)$$

即最小二乘法本质上是在给定矩阵方程  $Ax = b$  的情况下, 需要找到  $x$  使得  $y$  与  $b$  之间的距离最短。化简 (4-6) 式得:

$$y = \theta_1 \begin{bmatrix} x_{11} \\ x_{21} \\ \dots \\ x_{m1} \end{bmatrix} + \theta_2 \begin{bmatrix} x_{12} \\ x_{22} \\ \dots \\ x_{m2} \end{bmatrix} + \dots + \theta_n \begin{bmatrix} x_{1n} \\ x_{2n} \\ \dots \\ x_{mn} \end{bmatrix} \quad (4-8)$$

把矩阵  $A$  的各列向量分别记为  $\alpha_1, \alpha_2, \dots, \alpha_n$ 。由它们生成的子空间记为  $L(\alpha_1, \alpha_2, \dots, \alpha_n)$ , 那么  $y$  为该子空间中的一个向量。最小二乘法需要找到解  $x$  使得 (4-2) 式最小, 本质上是在子空间  $L(\alpha_1, \alpha_2, \dots, \alpha_n)$  中找到向量  $y$  使得  $b$  到该向量的距离比到子空间  $L(\alpha_1, \alpha_2, \dots, \alpha_n)$  中其它向量的距离都短。

设  $y = Ax = \theta_1 \alpha_1 + \theta_2 \alpha_2 + \dots + \theta_n \alpha_n$  为要求解的向量, 有:

$$c = b - y = b - Ax \quad (4-9)$$

由 (4-9) 式,  $c$  必定垂直于子空间  $L(\alpha_1, \alpha_2, \dots, \alpha_n)$ , 为此只需且必须:

$$(c, \alpha_1) = (c, \alpha_2) = \dots = (c, \alpha_n) = 0 \quad (4-10)$$

从而根据向量内积定义可得:

$$\alpha_1^T c = 0, \alpha_2^T c = 0, \dots, \alpha_n^T c = 0 \quad (4-11)$$

而  $\alpha_1^T, \alpha_2^T, \dots, \alpha_n^T$  按行正好排成矩阵  $A^T$ , 即有:

$$A^T c = A^T (b - y) = A^T (b - Ax) = 0 \quad (4-12)$$

化简 (4-12) 式得最小二乘法的全局最优解  $x$  为:

$$x = (A^T A)^{-1} A^T b \quad (4-13)$$

其中  $|A^T A| \neq 0$ , 即对于线性条件下的最小二乘法, 给定形如  $Ax = b$  形式的矩阵方程, 通过公式 (4-13) 便可以得到具有最小误差平方和的全局最优解  $x$ 。

本文中，通过几何约束理论得到的一组位置约束方程便具有此形式，从而可以通过最小二乘法解出目标相对于观测者的位置坐标。

#### 4.1.2 成像模型与几何约束

3D 边界框经过相机在成像平面上的投影会严格约束在 2D 边界框之内，这意味着对于 2D 边界框的每一条边，都至少会有 3D 边界框 8 个顶点中的某一个顶点投影到这条边上<sup>[8]</sup>。更进一步来说，对于 3D 边界框的任何一个顶点，其投影到成像平面对应的投影点像素坐标的横坐标  $u$  满足  $X_{\min} \leq u \leq X_{\max}$ ，纵坐标  $v$  满足  $Y_{\min} \leq v \leq Y_{\max}$ 。其中  $(X_{\min}, Y_{\min})$  为 2D 边界框的左上顶点的像素坐标， $(X_{\max}, Y_{\max})$  为 2D 边界框右下顶点的像素坐标。2D 边界框与 3D 边界框之间的几何约束如下图 4-1 所示：

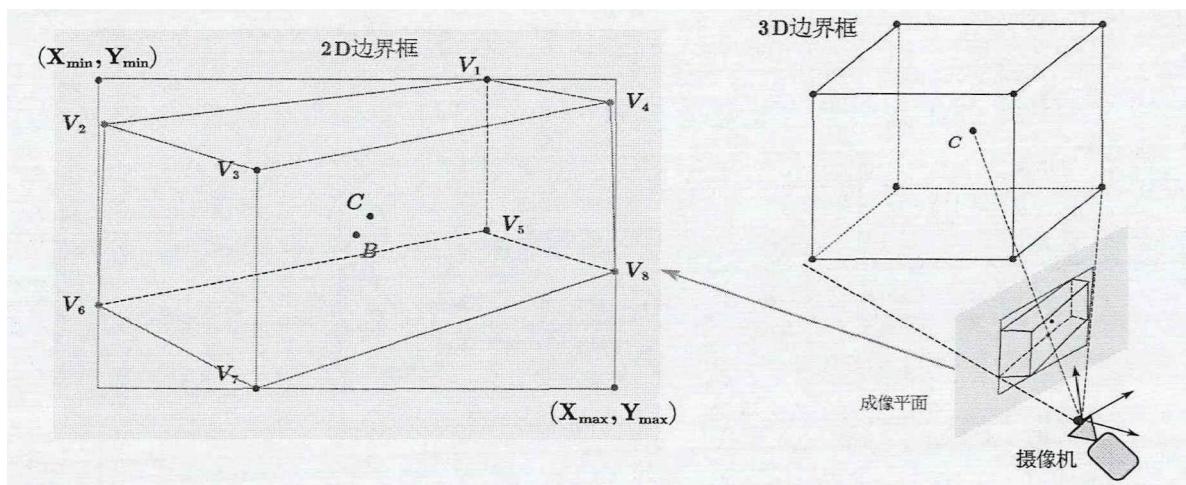


图 4-1 2D 边界框与 3D 边界框投影约束示意图。红色矩形为 2D 边界框，中心点为 B。蓝色立方体为 3D 边界框，中心点为 C。

由第二章介绍的相机模型理论可知，三维空间中的一点通过坐标系统与成像平面相对应的像素点构成了坐标一一映射关系。如下公式 (4-14) 所示：

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (4-14)$$

$P$  点的世界坐标为  $(X_w, Y_w, Z_w)$ ，像素坐标为  $(u, v)$ 。 $P$  点在相机坐标系下的坐标为  $(X_c, Y_c, Z_c)$ ， $Z_c$  为目标深度。 $dx$  和  $dy$  是图像中单个矩形像素的宽和高。 $(u_0, v_0)$  是图像中点的像素坐标，为了研究方便起见，不考虑相机发生畸变，本文认为图像的主点即图像的中点，即有  $(u_0, v_0) = (w/2, h/2)$ ，其中  $w$  为图像的宽， $h$  是图像的高。 $f$  是相机的焦距。 $R$  与  $T$  分别是旋转矩阵和平移矩阵。

将上述(4-14)式进行矩阵变换，经过化简得：

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} I & RX_{3d} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} T_x \\ T_y \\ T_z \\ 1 \end{bmatrix} \quad (4-15)$$

其中 $I$ 为3行3列的单位矩阵， $K$ 、 $X_{3d}$ 、 $T$ 、 $f_x$ 、 $f_y$ 的值如下所示：

$$K = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, X_{3d} = \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix}, T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, f_x = \frac{f}{dx}, f_y = \frac{f}{dy} \quad (4-16)$$

为了表示方便，省略运算时不需要的细节部分，令：

$$M = \begin{bmatrix} m_0 & m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 & m_7 \\ m_8 & m_9 & m_{10} & m_{11} \end{bmatrix} = K \begin{bmatrix} I & RX_{3d} \\ 0^T & 1 \end{bmatrix} \quad (4-17)$$

矩阵 $M$ 的未知参数为旋转矩阵 $R$ 中的 $\theta_{yaw}$ 和 $P$ 点在世界坐标系下的坐标 $X_{3d}$ ，其它未知量可通过相机内参得到。继续化简(4-15)式得：

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 & m_7 \\ m_8 & m_9 & m_{10} & m_{11} \end{bmatrix} \begin{bmatrix} T_x \\ T_y \\ T_z \\ 1 \end{bmatrix} \quad (4-18)$$

化简(4-18)式得：

$$\begin{cases} (m_8 T_x + m_9 T_y + m_{10} T_z + m_{11}) u = m_0 T_x + m_1 T_y + m_2 T_z + m_3 \\ (m_8 T_x + m_9 T_y + m_{10} T_z + m_{11}) v = m_4 T_x + m_5 T_y + m_6 T_z + m_7 \end{cases} \quad (4-19)$$

其中 $(u, v)$ 为点 $P$ 投影到图像平面的投影点在像素坐标系下的像素坐标。 $T_x$ 、 $T_y$ 、 $T_z$ 是相机坐标系与世界坐标系之间的相对平移量，令：

$$\left\{ \begin{array}{l} \Upsilon_0 = [m_0 \ m_1 \ m_2] \\ \Upsilon_1 = [m_4 \ m_5 \ m_6] \\ \Upsilon_2 = [m_8 \ m_9 \ m_{10}] \\ \lambda_0 = m_3 \\ \lambda_1 = m_7 \\ \lambda_2 = m_{11} \end{array} \right. \quad (4-20)$$

其中 $\Upsilon_0$ 、 $\Upsilon_1$ 、 $\Upsilon_2$ 为1行3列的矩阵。将(4-20)带入到(4-19)得：

$$\begin{cases} (\Upsilon_0 - u\Upsilon_2) [T_x \ T_y \ T_z]^T = u\lambda_2 - \lambda_0 \\ (\Upsilon_1 - v\Upsilon_2) [T_x \ T_y \ T_z]^T = v\lambda_2 - \lambda_1 \end{cases} \quad (4-21)$$

在第三章中已经完成对目标维度参数的预测。本文设置世界坐标系的坐标原点为目标的中心，目标前进的方向为 $X_w$ 的正方向，竖直向下为 $Y_w$ 的正方向，坐标系满足右手定则，如下图 4-2 所示：

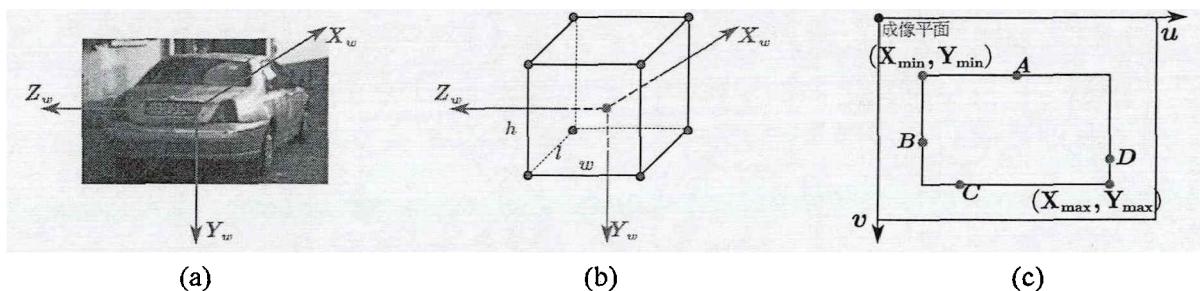


图 4-2 (a) 图为本文采用的世界坐标系建立规则。 $X_w$  轴正方向为目标前进的方向， $Y_w$  轴正方向竖直向下，且为右手坐标系。(b) 图立方体为为目标的 3D 边界框。(c) 图为成像平面示意图。矩形框为目标的 2D 边界框。

结合目标的维度分量 $w$ 、 $h$ 、 $l$ ，3D 边界框的 8 个顶点的坐标为：

$$\left\{ \begin{array}{l} x = \left[ \frac{l}{2} \quad \frac{l}{2} \quad -\frac{l}{2} \quad -\frac{l}{2} \quad \frac{l}{2} \quad \frac{l}{2} \quad -\frac{l}{2} \quad -\frac{l}{2} \right] \\ y = \left[ \frac{h}{2} \quad \frac{h}{2} \quad \frac{h}{2} \quad \frac{h}{2} \quad -\frac{h}{2} \quad -\frac{h}{2} \quad -\frac{h}{2} \quad -\frac{h}{2} \right] \\ z = \left[ \frac{w}{2} \quad -\frac{w}{2} \quad -\frac{w}{2} \quad \frac{w}{2} \quad \frac{w}{2} \quad \frac{w}{2} \quad -\frac{w}{2} \quad -\frac{w}{2} \right] \end{array} \right. \quad (4-22)$$

由于 2D 边界框 4 条边中的每一条边都至少会被 3D 边界框的 8 个顶点中的任一个顶点投影在上面，即可以得到 $8^4 = 4096$  种可能的投影约束情况。考虑其中任意的一种约束，假设 8 个顶点中的任意四个点分别投影到 2D 边界框的上、左、下、右四条边，对应的投影点记为 A、B、C、D，如上图 4-2 的 (c) 图所示。那么有以下结论：

- (1) A 点投影到图像投影点像素坐标的纵坐标 $v = Y_{\min}$ ，横坐标 $u$ 未知；
- (2) B 点投影到图像投影点像素坐标的横坐标 $u = X_{\min}$ ，纵坐标 $v$ 未知；
- (3) C 点投影到图像投影点像素坐标的纵坐标 $v = Y_{\max}$ ，横坐标 $u$ 未知；
- (4) D 点投影到图像投影点像素坐标的横坐标 $u = X_{\max}$ ，纵坐标 $v$ 未知；

将上述关于 A、B、C、D 这 4 个点的结论与 (4-21) 式结合，可以得出以下公式：

$$\left\{ \begin{array}{l} (Y_1 - Y_{\min}) [T_x \quad T_y \quad T_z]^T = Y_{\min} \lambda_2 - \lambda_1 \\ (Y_0 - X_{\min}) [T_x \quad T_y \quad T_z]^T = X_{\min} \lambda_2 - \lambda_0 \\ (Y_1 - Y_{\max}) [T_x \quad T_y \quad T_z]^T = Y_{\max} \lambda_2 - \lambda_1 \\ (Y_0 - X_{\max}) [T_x \quad T_y \quad T_z]^T = X_{\max} \lambda_2 - \lambda_0 \end{array} \right. \quad (4-23)$$

根据(4-22)式,可得3D边界框8个顶点在世界坐标系下的坐标,从而对于公式(4-17), $X_{3d}$ 是已知的。同时 $\theta_{yaw}$ 可由第三章旋转角预测模块得到,旋转矩阵 $R$ 也是已知的。即对于公式(4-23)式, $\Upsilon_0$ 、 $\Upsilon_1$ 、 $\Upsilon_2$ 、 $\lambda_0$ 、 $\lambda_1$ 、 $\lambda_2$ 均为已知值,未知的参数仅为 $T_x$ 、 $T_y$ 、 $T_z$ 。由世界坐标系的设置规则可知,世界坐标系与相机坐标系之间的旋转矩阵 $T = [T_x \ T_y \ T_z]^T$ 即相应的为目标中心在相机坐标系下的坐标,即有:

$$\begin{cases} T_x = X_c \\ T_y = Y_c \\ T_z = Z_c \end{cases} \quad (4-24)$$

公式(4-23)为形如 $Ax = b$ 的形式,其中:

$$\begin{cases} A = (\Upsilon_i - \eta\Upsilon_2), i \in \{0, 1\}, \eta \in \{X_{\min}, X_{\max}, Y_{\min}, Y_{\max}\} \\ x = [T_x \ T_y \ T_z]^T \\ b = \xi\lambda_2 - \lambda_j, \xi \in \{X_{\min}, X_{\max}, Y_{\min}, Y_{\max}\}, j \in \{0, 1\} \end{cases} \quad (4-25)$$

对于每一种可能的投影约束情况,均有一组观察值 $A$ 和 $b$ 。根据最小二乘法计算出相应的( $T_x, T_y, T_z$ )和拟合误差,取拟合误差最小的那个解( $T_x, T_y, T_z$ )作为最终目标在相机坐标系下的坐标( $T_x, T_y, T_z$ ),即3D边界框 $B^{3d}$ 中的( $x, y, z$ )参数。至此得到了目标的类别和相应的3D边界框 $B^{3d} = (w, h, l, x, y, z, \theta_{yaw})$ ,初步完成了基于单目视觉的3D目标检测任务。

#### 4.1.3 几何约束优化

本节提出了一种基于目标可视化表面的投影约束优化算法。基于此算法将可能的4096种投影约束降低到64种,提高了3D目标检测后续处理效率。本文以驾驶员为基准,定义目标前后左右上下6个表面。即驾驶员面对的方向为目标前面,驾驶员左手侧为目标左面,依此类推。本文提出的优化算法基于局部旋转角 $\theta_{alpha}$ 和目标上下左右前后6个表面可视化关系。由 $\theta_{alpha}$ 的定义可知:

- (1) 当 $0 < \theta_{alpha} < \pi$ 时,目标的前侧在成像平面是可见的;
- (2) 当 $-\pi < \theta_{alpha} < 0$ 时,目标的后侧在成像平面是可见的;
- (3) 当 $-\pi/2 < \theta_{alpha} < \pi/2$ 时,目标的右侧在成像平面是可见的;
- (4) 当 $-\pi < \theta_{alpha} < -\pi/2$ 或者 $\pi/2 < \theta_{alpha} < \pi$ 时,目标的左侧在成像平面是可见的;

可见目标的上侧总是可见的，下侧总是不可见的。下图 4-3 给出了  $\theta_{alpha}$  分别为  $2.0rad$ 、 $0.5rad$ 、 $-1.3rad$ 、 $-1.8rad$  时目标状态示意图，包括目标的 2D 边界框、3D 边界框和相应的目标可视化表面：

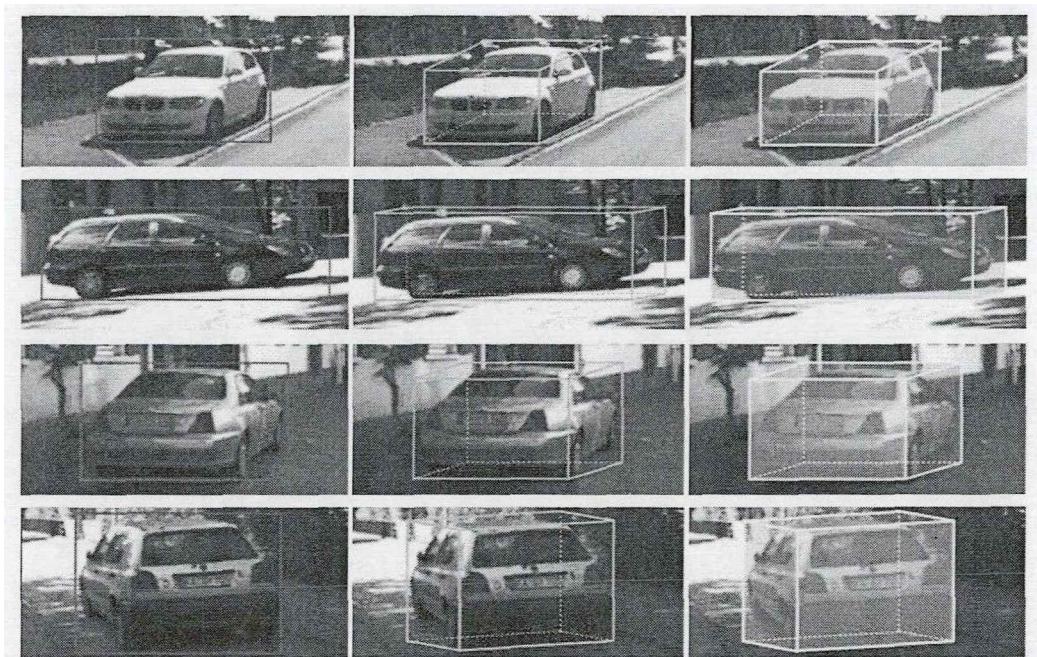


图 4-3 目标状态示意图。图中第 1 排，第 2 排，第 3 排，第 4 排分别是局部旋转角为  $2.0rad$ ,  $0.5rad$ ,  $-1.3rad$ ,  $-1.8rad$  时目标的 2D 边界框，3D 边界框，目标可视化表面。

由于 3D 目标检测任务只考虑  $\theta_{yaw}$ ，其它两个旋转角  $\theta_{roll}$  和  $\theta_{pitch}$  均为 0，所以对于 2D 边界框的上下两条边来说，只可能是 3D 边界框上顶面的 4 个顶点会投影在 2D 边界框的上边，3D 边界框下底面的 4 个顶点会投影在 2D 边界框的下边。所以总共有 16 种可能的约束。根据局部旋转角  $\theta_{alpha}$  的不同，2D 边界框的左右两条边的约束具有不同的可能情况，如下所示：

(1) 当  $-\pi/2 < \theta_{alpha} < \pi/2$  时，对于 2D 边界框的左右两条边，若有  $-\pi/2 < \theta_{alpha} < 0$ ，则左边投影点只可能是 3D 边界框的左侧与后侧相交垂直边的两个端点，右边投影点只可能是前侧与右侧相交垂直边的两个端点，如上图 4-3 第 3 排所示；当  $0 < \theta_{alpha} < \pi/2$  时，左边投影点只可能是右侧和后侧相交垂直边的两个端点，右边投影点只可能是左侧和前侧相交垂直边的两个端点，如图 4-3 第 2 排所示。所以共 2 种可能的情况。

(2) 当  $-\pi < \theta_{alpha} < -\pi/2$  或者  $\pi/2 < \theta_{alpha} < \pi$  时，对于 2D 边界框的左右两条边，当  $-\pi < \theta_{alpha} < -\pi/2$  时，左边投影点只可能是 3D 边界框左侧和前侧相交垂直边的两个端点，右边投影点只可能是右侧和后侧相交垂直边的两个端点，如上图 4-3 第 4 排所示；当  $\pi/2 < \theta_{alpha} < \pi$  时，左边投影点只可能是右侧和前侧相交垂直边的两个端点，右边投影点只可能是左侧和后侧相交垂直边的两个端点，如上图 4-3 第 1 排所示。所以共 2 种可能的情况。

综上所述，2D 边界框的上下边共有  $4 \times 4 = 16$  种约束可能，左右两边根据  $\theta_{alpha}$  的值共有  $2 \times 2 = 4$  种约束可能。所以总共有  $16 \times 4 = 64$  种可能的情况。通过分析  $\theta_{alpha}$  与目标可视化表面之间的关系，将可能的 4096 种投影约束降低至 64 种。每一种约束都对应着一个由 4 个方程构成的方程组，如公式（4-23）所示。然后通过最小二乘法求解出未知参数  $(T_x, T_y, T_z)$ ，取具有最小误差的那个  $(T_x, T_y, T_z)$  作为目标中心最终的在相机坐标系下的坐标  $(x, y, z)$ 。相比较于解决 4096 种情况对应的 4096 个方程组，本文根据已知的  $\theta_{alpha}$  降低需要求解的方程组个数，从而提升了整个 3D 目标检测的后处理效率。

## 4.2 参数优化

### 4.2.1 几何约束存在的问题

在 4.1 节中基于 2D 边界框与 3D 边界框具有几何约束特性计算出坐标  $(x, y, z)$ 。该算法处理高效，且不需要格外的训练过程，满足了自动驾驶场景下需要实时获取目标空间位置的需求。然而通过最小二乘法求解目标位置坐标时，基于假设 3D 边界框的中心在成像平面的投影点与 2D 边界框的中心重合，然而当目标在 RGB 图像上被截断或者被其它目标遮挡时，并不满足该假设。在维度或者旋转角估计不精确的情况下，即使在成像平面上仅仅几个像素点的偏移在三维空间会带来几米甚至十几米的位置误差，同时 2D 边界框预测是否精确也影响着最后求解的坐标结果。目标被精确检测到是后续将 3D 目标检测技术应用到自动驾驶领域的重要前提，为此需要对几何约束算法进一步优化，从而提升最终的检测性能。

### 4.2.2 位置坐标修正

为了解决几何约束算法存在的问题，本文采用了一种优化网络 ShiftNet<sup>[52]</sup>，对目标空间位置坐标进行误差修正。ShiftNet 本质上是一个多层感知机（Multilayer Perceptron, MLP）<sup>[60]</sup>模型。本文在第三章中已知目标的维度  $(w, h, l)$ 、局部旋转角  $\theta_{alpha}$ 、全局旋转角  $\theta_{yaw}$ 、目标的 2D 边界框等信息。在本章 4.1 节，通过几何约束算法获取到了目标的空间位置坐标。同时在 3D 目标检测任务中，训练集数据的相机内参是已知的。本文将所有已知的信息输入到 ShiftNet 网络中，网络输出的是修正后的目标位置  $(x, y, z)$ 。MLP 模型结构如下图 4-4 所示：

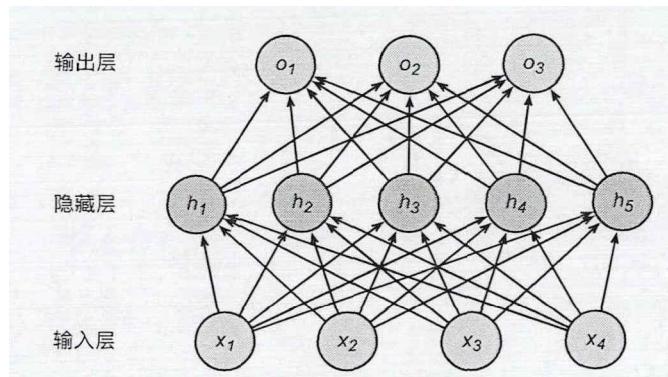


图 4-4 多层感知机模型。

多层感知机由输入层、隐藏层和输出层构成。ShiftNet 网络结构包含两个隐藏层，每个隐藏层包括 4096 个神经元。输入层输入已知的 2D 边界框和 3D 边界框相关数据，输出层为修正后的目标位置坐标。为了描述方便起见设初始位置坐标矩阵为  $t = [x \ y \ z]^T$ ，精细化处理后的位置坐标矩阵为  $t' = [x' \ y' \ z']^T$ 。为了更好的初始化网络权重，使用训练集标签文件给定的真值数据预训练网络，从而使得网络能更好的匹配 2D 到 3D 信息。令  $\Delta t = t - t'$ ，由第二章可知旋转矩阵为：

$$R = R_x \cdot R_y \cdot R_z \quad (4-26)$$

只考虑  $\theta_{yaw}$ ，所以  $R = R_y$ ，令  $\Delta t_{yaw} = R \cdot \Delta t$  有：

$$\begin{aligned} \Delta t_{yaw} &= R \cdot \Delta t = R_y \cdot \Delta t \\ &= \begin{bmatrix} \cos \theta_{yaw} & 0 & -\sin \theta_{yaw} \\ 0 & 1 & 0 \\ \sin \theta_{yaw} & 0 & \cos \theta_{yaw} \end{bmatrix} \cdot \begin{bmatrix} x - x' \\ y - y' \\ z - z' \end{bmatrix} \end{aligned} \quad (4-27)$$

令  $\Delta t_{yaw} = [\Delta x_{yaw} \ \Delta y_{yaw} \ \Delta z_{yaw}]^T$ ，ShiftNet 网络的损失函数为：

$$\mathcal{L} = w \times h \times |\Delta x| + w \times l \times |\Delta y| + h \times l \times |\Delta z| \quad (4-28)$$

基于 ShiftNet 网络，完成对目标位置坐标优化和最终的 3D 目标检测任务。

本文将整个 3D 目标检测任务分成几个子任务，维度预测和局部旋转角预测模块得到 3D 边界框的维度  $(w, h, l)$  和局部旋转角  $\theta_{alpha}$ 。同时由  $\theta_{alpha}$  通过几何关系计算出 3D 目标检测所需要的全局旋转角  $\theta_{yaw}$ 。本文意识到自动驾驶场景下存在的多尺度目标，同时为了和其它算法公平比较，验证本文提出的算法优于其它算法<sup>[8]</sup>，本文采用 MS-CNN<sup>[58]</sup>得到目标的 2D 边界框，为后续的几何约束提供精确的 2D 边界框。同时为了解决几何约束带来的误差问题，本文采用了 ShiftNet 网络对目标位置坐标优化，提高整个 3D 目标检测的精确度。下图 4-5 给出本文提出的算法处理流程图：

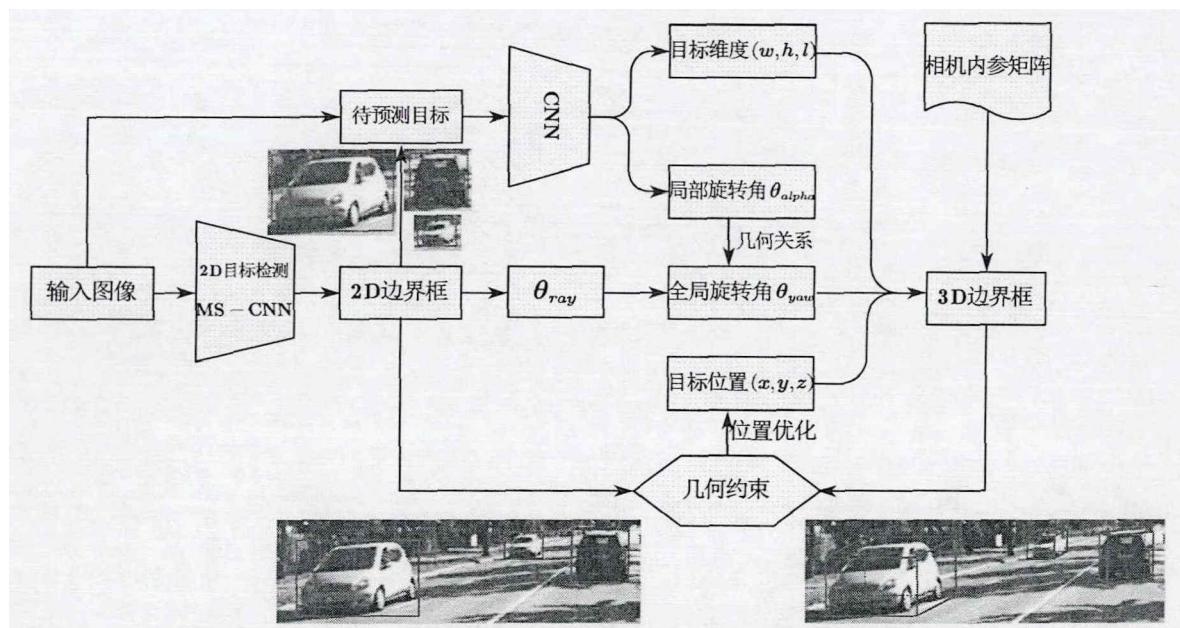


图 4-5 基于单目视觉的 3D 目标检测系统流程处理图。

如上图所示，首先通过 MS-CNN 算法得到输入图像的 2D 目标检测结果，即目标相应的 2D 边界框和目标类别。在这个过程中可能会出现有些目标没有被检测到，对于这些目标当然后续也不会进行 3D 目标检测。同时会判断当前 2D 边界框是否合理，正确的 2D 边界框的坐标都是正数，对于错误的 2D 边界框当然也是不做任何处理的。然后在图像平面对目标进行裁剪和预处理，通过 CNN 提取目标特征。CNN 不同分支输出不同的参数。这样便得到目标的维度 $(w, h, l)$ 和局部旋转角 $\theta_{alpha}$ 。通过 2D 边界框且假设目标中心投影到成像平面的投影点为 2D 边界框的中心计算出角 $\theta_{ray}$ 。通过 $\theta_{alpha}$  和 $\theta_{ray}$  计算出需要的全局旋转角参数 $\theta_{yaw}$ 。通过几何约束理论结合最小二乘法计算出目标初步的 3D 边界框。由于该边界框存在误差，所以采用 ShiftNet 优化算法对目标空间位置坐标优化，得到精细化处理后的 3D 边界框，至此得到目标最终的 3D 边界框和目标类别，实现了基于单目视觉的 3D 目标检测任务。

### 4.3 本章小结

本章主要介绍几何约束理论，基于该理论结合最小二乘法计算出目标的空间位置坐标，完成 3D 目标检测任务。为了提高计算效率，基于局部旋转角 $\theta_{alpha}$  将解 4096 种约束可能的方程组优化到 64 种，同时采用 ShiftNet 网络对目标的空间位置坐标进行优化，提高了最终的 3D 目标检测算法的效率和精确度。

## 第五章 实验结果与分析

本章主要对 3D 目标检测任务涉及到的相关算法进行实验评估。整个评估过程主要分为两部分。首先是对本文提出的参数预测算法进行评估，其次是对 3D 目标检测整体框架性能进行评估。通过与其它算法进行对比，证明本文提出的算法具有更好的有效性、精确性和鲁棒性。本文在公开的数据集上进行实验，使得评估结果更具有客观性、公平性。

### 5.1 数据集

KITTI<sup>[61][62]</sup>是应用于自动驾驶场景下最流行的公开实验数据集。该数据集由德国卡尔斯鲁厄理工学院 (Karlsruhe Institute of Technology, KIT) 等联合创办，用于评测 3D 目标检测等视觉任务算法性能。数据集中的目标具有不同程度的遮挡 (Occluded) 与截断 (Truncated)，是一个综合的应用于自动驾驶场景下的数据集。本文提出的基于单目视觉的 3D 目标检测框架基于 KITTI 数据集评估相关指标性能。

KITTI 数据集中的目标被分为汽车 (Car) 等 9 类。并且每一张图像均对应着一个标签 (label) 文件。标签中提供了目标相应的 2D 边界框、3D 边界框、局部旋转角、全局旋转角、维度、目标中心在相机坐标系下的位置坐标等标注信息。同时标签中还提供了目标相应的遮挡和截断程度。从 0 到 1 的小数表示截断程度越来越严重。对于遮挡，0 表示完全可见，1 表示部分遮挡，2 表示大部分遮挡，3 表示完全不可见。根据遮挡或截断程度以及目标在成像平面所占像素大小等因素不同，目标被检测到的难易程度划分为简单 (Easy)、适中 (Moderate)、困难 (Hard) 3 个等级，在评估检测算法性能阶段，不同的难易等级会有相应的评价结果。KITTI 数据集分为训练集和测试集两部分，训练集包括 7481 张 RGB 图像，每张图像都提供了相应的标签文件和拍摄该图像时相机的内外参数。测试集包括 7518 张 RGB 图像，但没有提供相应的标签文件和相机内外参数。同时对于 3D 目标检测任务，KITTI 数据集从不同的角度给出了不同的评价指标。从而使得基于 KITTI 数据集的目标检测算法能够充分的被评估其性能。本文提出的算法在 KITTI 数据集上进行实验并验证其检测性能，同时与其它同样使用该数据集的检测算法从多个评估指标角度进行对比。KITTI 数据集相关统计数据如下图 5-1 所示：

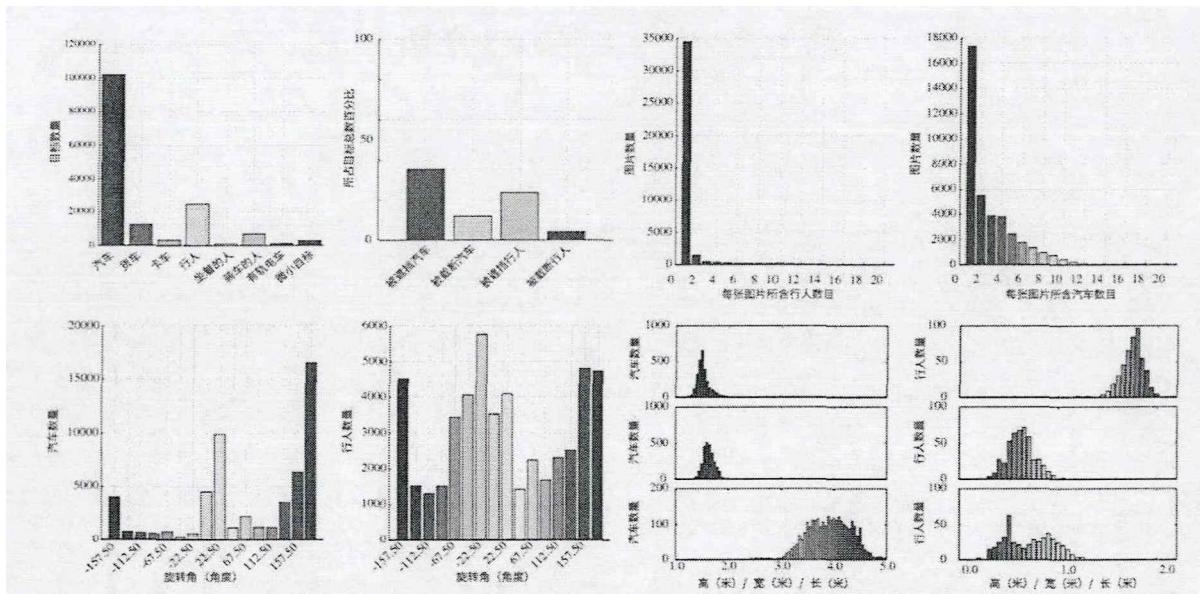


图 5-1 KITTI 数据集统计数据示意图。第一排左起分别为不同种类目标数目示意图；被截断和遮挡的汽车以及行人占相应类别目标总数百分比示意图；每张图片所含行人数目分布图；每张图片所含汽车数目分布图。第二排左起分别为汽车旋转角分布示意图；行人旋转角分布示意图；汽车维度分布示意图；行人维度分布示意图。

从上图给出的统计数据可以看出，KITTI 数据集对于汽车类别提供了足够的样本数据，而其它类别提供的样本数量太少。与目前所有采用 KITTI 作为实验数据集的相关工作一样，本文提出的算法主要基于汽车类别评估相关指标性能。本文在训练集上进行实验。为了实验的公平性，本文遵循 3DOP<sup>[6]</sup>将训练集数据分成 train/val1 两部分，遵循 SubCNN<sup>[63]</sup>将训练集数据分成 train/val2 两部分。其中 train 用来训练神经网络，val1 或者 val2 用来评估相关算法性能。这两种数据集分割方式均使得图像不来自同一个视频序列，从而保证了数据的随机性。

## 5.2 实验

### 5.2.1 评估指标

对于目标维度，本文遵循 FQNet<sup>[10]</sup>提出的计算平均维度误差  $E_{dim}$  作为维度预测算法的评估标准。 $E_{dim}$  计算公式如下所示：

$$E_{dim} = \frac{1}{N} \sum_{i=1}^N \sqrt{\Delta w_i^2 + \Delta h_i^2 + \Delta l_i^2} \quad (5-1)$$

其中  $N$  为样本数目， $\Delta w$ 、 $\Delta h$ 、 $\Delta l$  分别为预测的维度分量宽  $w_{pre}$ 、高  $h_{pre}$ 、长  $l_{pre}$  与真实维度对应分量  $w$ 、 $h$ 、 $l$  之间的差值。

对于目标旋转角，本文采用 KITTI 数据集官方提供的平均旋转角相似度 (Average Orientation Similarity, AOS) 作为评估标准，AOS 计算公式如下所示：

$$AOS = \frac{1}{11} \sum_{r \in \{0, 0.1, \dots, 1\}} \max_{\tilde{r}: \tilde{r} \geq r} s(\tilde{r}) \quad (5-2)$$

$$r = \frac{TP}{TP + FN} \quad (5-3)$$

$$s(r) = \frac{1}{D(r)} \sum_{i \in D(r)} \frac{1 + \cos \Delta_\theta^{(i)}}{2} \delta_i \quad (5-4)$$

其中  $r$  为 PASCAL VOC<sup>[36]</sup> 数据集定义的召回率 (Recall)，如果一个预测的 2D 边界框与任意一个真值 2D 边界框的 IoU 超过 0.5，那么认为该边界框是有效的。 $D(r)$  表示召回率为  $r$  时所有正样本的集合。 $\Delta_\theta^{(i)}$  表示对于检测目标  $i$ ，估计的旋转角与真值旋转角之间的差值。为了惩罚单个目标被多次检测到，如果目标  $i$  已被分配给某个真值且二者之间的 IoU 大于 0.5，设置  $\delta_i = 1$ ，否则为 0。

评估 3D 目标检测任务整体性能，采用 KITTI 数据集定义的两种指标：

(1) 评估 3D 边界框鸟瞰图 (Bird's Eye View, BEV) 的平均精度 (Average Precision, AP) 即  $AP_{BEV}$ 。BEV 意味着从俯视角度评估目标检测性能。因此首先将所有预测 3D 边界框和真值 3D 边界框投影到地平面得到两个矩形边界框，然后依据 2D 目标检测评估标准，计算相应的精确率 (Precision) 和召回率，绘制 Precision-Recall 曲线，曲线与坐标轴围成的面积即为相应的  $AP_{BEV}$ 。由于 BEV 仅仅考虑平面两个矩形边界框的重合度，而没有考虑高这个维度分量，所以  $AP_{BEV}$  评估指标具有局限性，无法充分说明 3D 检测算法性能。

(2) 评估 3D 目标检测 (3D Object Detection) 平均精度即  $AP_{3D}$ 。该指标是直接在三维空间中评估算法性能，更符合 3D 目标检测任务的要求。与 2D 目标检测计算两个矩形框 IoU 不同，在计算 Precision 和 Recall 时考虑预测 3D 边界框与真值 3D 边界框之间的 IoU。由于额外增加了一个维度，与 BEV 指标相比评价更严格，因此在同样的条件下  $AP_{3D}$  指标数据也相应的低于  $AP_{BEV}$ 。所以  $AP_{3D}$  指标更能体现 3D 目标检测算法的性能优劣。

## 5.2.2 实验结果与分析

### 5.2.2.1 维度与旋转角预测任务实验结果与分析

为了验证本文在第三章提出的  $loss_{IoU_{3D}}$  相比较于  $loss_{LAE}$  和  $loss_{LSE}$  具有更好的维度预测效果，分别以这 3 个损失函数作为维度预测模块的损失函数进行实验，并分别计算出平均维度误差  $E_{dim}$ ，同时与其它算法做对比，证明本文提出的算法的优越性。实验结果如下表 5-1 所示：

表 5-1 平均维度误差在 train/val1 和 train/val2 数据集对比结果。

算法	train/val1	train/val2
3DOP <sup>[6]</sup>	0.3527	N/A
Mono3D <sup>[5]</sup>	0.4251	N/A
Deep3DBox <sup>[8]</sup>	N/A	0.1934
本文提出+ $loss_{LAE}$	0.2013	0.1989
本文提出+ $loss_{LSE}$	0.1923	0.1901
本文提出+ $loss_{IoU_{3D}}$	0.1810	0.1902

从表 5-1 可知，本文提出的 $loss_{IoU_{3D}}$ 相比较于 3DOP 和 Mono3D 在数据集 train/val1 上  $E_{dim}$  分别降低了 0.1717 和 0.2441。由于 3DOP 和 Mono3D 没有提供 train/val2 实验数据，所以无法与其比较。在 train/val2 数据集上相比较于 Deep3DBox 算法  $E_{dim}$  降低了 0.0032。相比较于  $loss_{LAE}$  和  $loss_{LSE}$ ，采用  $loss_{IoU_{3D}}$  在数据集 train/val1 上  $E_{dim}$  分别降低了 0.0203 和 0.0113。在数据集 train/val2 上误差数据相差不大，相比较于  $loss_{LAE}$  降低 0.0087。虽然相比较于  $loss_{LSE}$  误差增加了 0.0001，但可以忽略不计。综上证明了本文提出的维度预测算法的优越性。

表 5-2 2D 目标检测算法 AP 指标数据对比结果。

算法	AP					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	94.49	N/A	89.65	N/A	80.97	N/A
Mono3D <sup>[5]</sup>	95.75	N/A	90.01	N/A	80.66	N/A
Deep3DBox <sup>[8]</sup>	N/A	98.84	N/A	97.20	N/A	81.17
SubCNN <sup>[63]</sup>	N/A	95.77	N/A	86.64	N/A	74.07
DeepMANTA <sup>[15]</sup>	97.58	97.90	90.89	91.01	82.72	83.14
本文提出	97.21	98.87	96.69	97.25	81.03	81.25

表 5-3 本文提出的旋转角估计算法 AOS 指标数据对比结果。

算法	AOS					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	92.98	N/A	87.34	N/A	78.24	N/A
Mono3D <sup>[5]</sup>	93.70	N/A	87.61	N/A	78.00	N/A
Deep3DBox <sup>[8]</sup>	N/A	98.59	N/A	96.69	N/A	80.51
SubCNN <sup>[63]</sup>	N/A	94.55	N/A	85.03	N/A	72.21
DeepMANTA <sup>[15]</sup>	97.44	97.60	90.66	90.66	82.35	82.66
本文提出	97.89	98.65	97.10	96.77	81.32	80.65

为了说明本文提出的旋转角估计算法的优越性，上表 5-2 和 5-3 分别给出了 2D 目标检测评估指标 AP 和旋转角评估指标 AOS 数据。与维度估计算法一样，本文分别给出了 train/val1 和 train/val2 数据集上的实验结果。

从与其它算法对比结果来看，本文提出的旋转角估计算法在 AOS 评估指标上有了明显的提升。在数据集为 train/val1 时，相比较于 3DOP 和 Mono3D，本文提出的算法在所有的难度等级下 AOS 指标均达到最优，与 DeepMANTA 相比，本文提出的算法在适中难度下具有明显的优势，提升了 6.44%。当数据集为 train/val2 时，以适中难度为例，本文提出的算法均高于 DeepMANTA、SubCNN 和 Deep3DBox，分别提升了 6.11%、11.74% 和 0.08%。由于 KITTI 数据集中绝大多数目标处于适中难度等级下，所以在该难度下评价算法性能更加具有说服力。通过与其它 3D 目标检测算法对比并对 AOS 指标分析证明本文提出的旋转角估计算法具有优越性。

本文与 Deep3DBox 采用 MS-CNN<sup>[58]</sup>检测算法得到精确的 2D 边界框，进而完成 3D 检测。Mono3D 首先在目标可能出现的空间区域穷举 3D 候选边界框，然后对候选框进行过滤得到最终的边界框。3DOP 在 Mono3D 的基础上额外利用了深度信息完成 3D 检测。DeepMANTA 采用 CAD 模型来实现 3D 检测。这些算法虽然不依赖于额外的 2D 目标检测模块，但从表 5-2 来看，其 AP 指标比较低。

图 5-2 给出了本文采用 MS-CNN 检测算法得到的 2D 目标检测结果示意图：

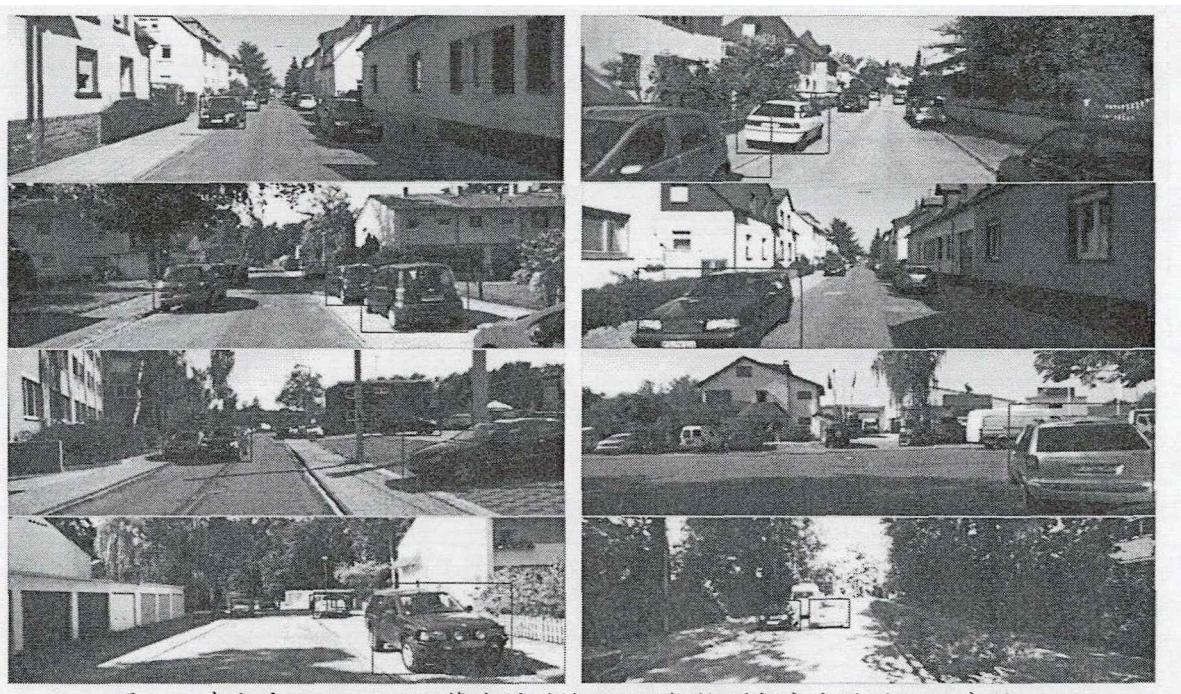


图 5-2 本文采用 MS-CNN 算法得到的 2D 目标检测部分实验结果示意图。

从 2D 目标检测的结果来看，MS-CNN 目标检测算法在 KITTI 数据集上取得了优越的表现。精确的 2D 边界框为本文完成 3D 目标检测提供了良好的基础。

同时为了与其它 3D 检测算法 Deep3DBox 公平对比，本文选择 MS-CNN 作为本文提出的 3D 目标检测算法的 2D 目标检测器。

### 5.2.2.2 3D 目标检测任务实验结果与分析

本小节以  $AP_{3D}$  和  $AP_{BEV}$  两个指标对本文提出的基于单目视觉的 3D 目标检测框架进行评估，并和其它 3D 目标检测算法性能做对比。为了充分说明本文提出的检测算法优越性，分别设置真值 3D 边界框与预测 3D 边界框之间的 IoU 阈值为 0.5 和 0.7，低于此阈值则认为预测 3D 边界框为负样本，不参加相应指标的计算过程。同时本文根据目标被检测难易程度给出对应的 Precision-Recall 曲线， $AP_{3D}$  和  $AP_{BEV}$  分别是相应曲线与坐标轴围成的面积。本文从多个评估指标验证提出的算法优于其它算法，使得实验结果更具有公正性和说服力。

(1) 3D 目标检测平均精度  $AP_{3D}$  指标实验结果如下表 5-4 和表 5-5 所示。同时本文也给出了相应的 Precision-Recall 曲线，如下图 5-3 和图 5-4 所示。

以下为 IoU 阈值为 0.7 时对应的  $AP_{3D}$  指标数据和相应的 Precision-Recall 曲线：

表 5-4 IoU 阈值为 0.7 时  $AP_{3D}$  指标在 train/val1 和 train/val2 数据集对比结果。

算法	$AP_{3D}(IoU=0.7)$					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	6.55	N/A	5.07	N/A	4.10	N/A
Mono3D <sup>[5]</sup>	2.53	N/A	2.31	N/A	2.31	N/A
Deep3DBox <sup>[8]</sup>	N/A	5.85	N/A	4.10	N/A	3.84
OFT-Net <sup>[64]</sup>	4.07	2.50	3.27	3.28	3.29	2.27
FQNet <sup>[10]</sup>	5.98	5.45	5.50	5.11	4.75	4.45
本文提出	8.16	8.88	6.52	8.67	6.10	8.73

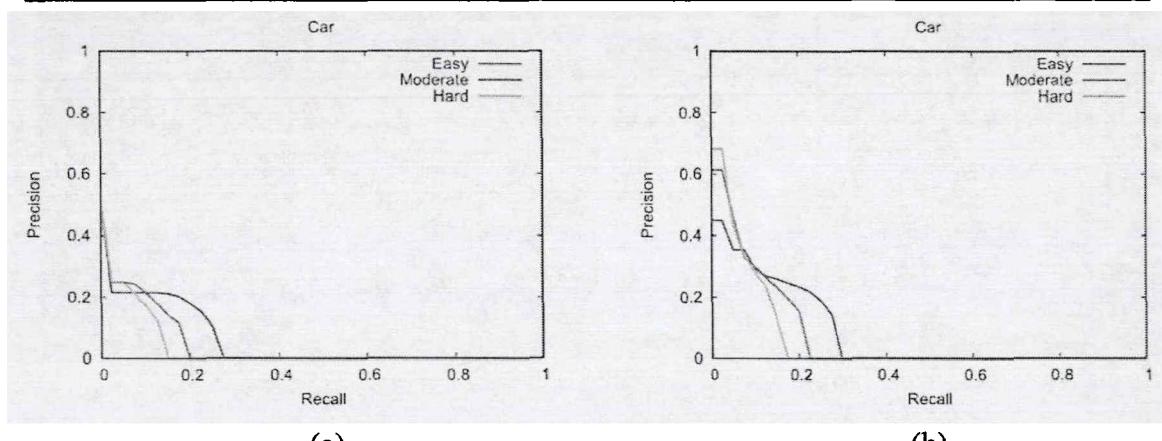


图 5-3 IoU 阈值为 0.7 时 3D 检测指标在不同数据集上的 Precision-Recall 曲线，(a) 数据集为 train/val1 时 Precision-Recall 曲线。(b) 数据集为 train/val2 时 Precision-Recall 曲线。

以下为 IoU 阈值为 0.5 时对应的 AP<sub>3D</sub> 指标数据和相应的 Precision-Recall 曲线：

表 5-5 IoU 阈值为 0.5 时 AP<sub>3D</sub> 指标在 train/val1 和 train/val2 数据集对比结果。

算法	AP <sub>3D</sub> (IoU=0.5)					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	46.04	N/A	34.63	N/A	30.09	N/A
Mono3D <sup>[5]</sup>	25.19	N/A	18.20	N/A	15.22	N/A
Deep3DBox <sup>[8]</sup>	N/A	27.04	N/A	20.55	N/A	15.88
OFT-Net <sup>[64]</sup>	N/A	N/A	N/A	N/A	N/A	N/A
FQNet <sup>[10]</sup>	28.16	28.98	21.02	20.71	19.91	18.59
本文提出	28.17	30.04	21.57	24.26	20.45	21.03

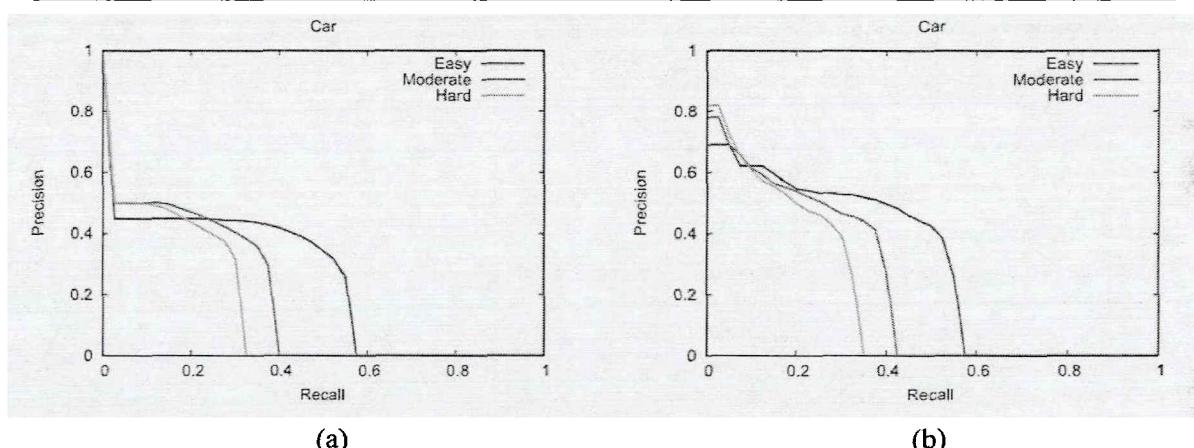


图 5-4 IoU 阈值为 0.5 时 3D 检测指标在不同数据集上的 Precision-Recall 曲线，(a) 数据集为 train/val1 时 Precision-Recall 曲线。(b) 数据集为 train/val2 时 Precision-Recall 曲线。

对于指标 AP<sub>3D</sub>, 本文提出的算法与其它算法相比有明显的提升。在设置 IoU 阈值为 0.7 时, 优于所列出的所有算法。但 IoU 阈值为 0.5 时, AP<sub>3D</sub> 低于依赖深度图的 3DOP, 但好于其它所有基于单目视觉的目标检测算法。总体来看, 所有算法的 AP<sub>3D</sub> 数值还是比较低的。以 IoU 阈值为 0.7 为例, 在验证集为 val1 且难度为适中时, 本文提出的算法相比较于 3DOP、Mono3D、OFT-Net 和 FQNet 分别提升了 1.45%、4.21%、3.25% 和 1.02%, 但整体 AP<sub>3D</sub> 仅仅 6.52%。同样的在简单模式下分别提升了 1.61%、5.63%、4.09% 和 2.18%。由于 Deep3DBox 缺少 train/val1 数据, 所以无法比较。在数据集为 train/val2 时, 本文提出的算法相比较于 Deep3DBox 在适中难度下提升了 4.57%。可见是一个比较大的提升。在 IoU 阈值为 0.5 时, 虽然 3DOP 算法在适中难度下 AP<sub>3D</sub> 高于本文提出算法 13.06%, 但在绝大多数情况下, IoU 阈值取更严格的 0.7 时更具有实际参考意义, 并且 3DOP 算法额外采用了深度图信息, 即不严格属于仅仅依靠单目视觉实现 3D 目标检测算法。所以本文提出的算法仍具有很大的优越性。

(2) 鸟瞰图平均精度 AP<sub>BEV</sub> 指标数据和相应的 Precision-Recall 曲线如下所示。

以下为 IoU 阈值为 0.7 时对应的 AP<sub>BEV</sub> 指标数据和相应的 Precision-Recall 曲线：

表 5-6 IoU 为 0.7 时指标 AP<sub>BEV</sub> 在 train/val1 和 train/val2 数据集对比结果。

算法	AP <sub>BEV</sub> (IoU=0.7)					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	12.63	N/A	9.49	N/A	7.59	N/A
Mono3D <sup>[5]</sup>	5.22	N/A	5.19	N/A	4.13	N/A
Deep3DBox <sup>[8]</sup>	N/A	9.99	N/A	7.71	N/A	5.30
OFT-Net <sup>[64]</sup>	11.06	9.50	8.79	7.99	8.91	7.51
FQNet <sup>[10]</sup>	9.50	10.45	8.02	8.59	7.71	7.43
本文提出	11.88	13.40	9.39	11.93	7.23	11.33

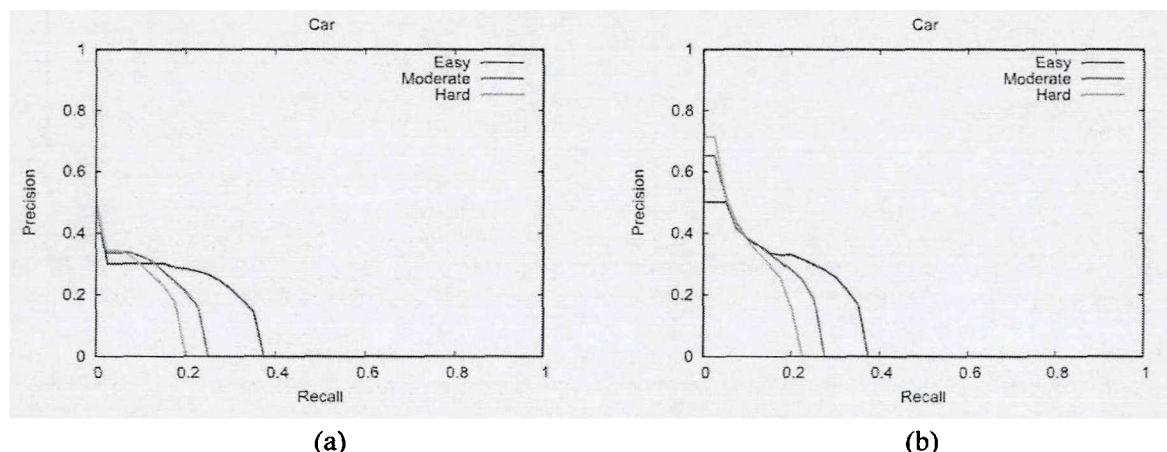


图 5-5 IoU 阈值为 0.7 时 BEV 指标在不同数据集上的 Precision-Recall 曲线，(a) 数据集为 train/val1 时 Precision-Recall 曲线。(b) 数据集为 train/val2 时 Precision-Recall 曲线。

以下为 IoU 阈值为 0.5 时对应的 AP<sub>BEV</sub> 指标数据和相应的 Precision-Recall 曲线：

表 5-7 IoU 为 0.5 时指标 AP<sub>BEV</sub> 在 train/val1 和 train/val2 数据集对比结果。

算法	AP <sub>BEV</sub> (IoU=0.5)					
	简单 (Easy)		适中 (Moderate)		困难 (Hard)	
	train/val1	train/val2	train/val1	train/val2	train/val1	train/val2
3DOP <sup>[6]</sup>	55.04	N/A	41.25	N/A	34.55	N/A
Mono3D <sup>[5]</sup>	30.50	N/A	22.39	N/A	19.16	N/A
Deep3DBox <sup>[8]</sup>	N/A	30.02	N/A	23.77	N/A	18.83
OFT-Net <sup>[64]</sup>	N/A	N/A	N/A	N/A	N/A	N/A
FQNet <sup>[10]</sup>	32.57	33.37	24.60	26.29	21.25	21.57
本文提出	33.44	35.54	26.73	27.62	22.55	22.89

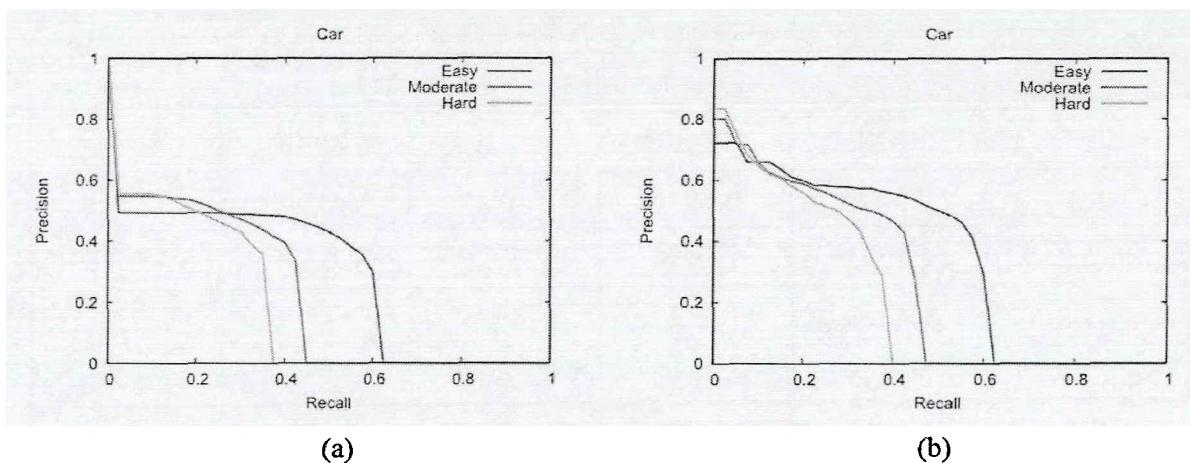


图 5-6 IoU 阈值为 0.5 时 BEV 指标在不同数据集上的 Precision-Recall 曲线，(a) 数据集为 train/val1 时 Precision-Recall 曲线。(b) 数据集为 train/val2 时 Precision-Recall 曲线。

对于指标 AP<sub>BEV</sub>，在设置 IoU 阈值为 0.7 时，本文提出算法在所有基于单目视觉的检测算法中性能达到了最优，略低于 3DOP。在适中难度下且 IoU 阈值为 0.7 时，当数据集为 train/val1，本文提出算法相比较于 Mono3D 和 FQNet 分别提升了 4.20% 和 1.37%，但较 3DOP 低 0.1%，当数据集为 train/val2，相比较于 Deep3DBox 和 OFT-Net 分别提升了 4.22% 和 3.94%，可见本文提出的算法在 BEV 指标下也有优秀的表现。值得注意的是在相同的 IoU 阈值条件下，指标 AP<sub>BEV</sub> 的数值高于 AP<sub>3D</sub>，这也表明对于 3D 目标检测任务，AP<sub>3D</sub> 指标更严格，由于是在三维空间中评估预测的 3D 边界框精确度，也更具有实际指导意义。

从上述的 AP<sub>3D</sub> 和 AP<sub>BEV</sub> 评估指标的实验结果来看，本文提出的算法优于其它算法。并且为了实验的客观和公平性，本文设置不同的 IoU 阈值来对比实验结果数据，并且对比算法的所有实验数据均来自算法论文当中的数据。本文对于 train/val1 数据集和 train/val2 数据均进行了实验，得到 AP<sub>3D</sub> 和 AP<sub>BEV</sub> 指标数据。而其它算法有的只给出了单个数据集上的实验数据，如 Deep3DBox 仅仅给出了 train/val2 数据集上的实验数据，Mono3D 和 3DOP 仅仅给出了 train/val1 数据集上的数据。所以本文提出的算法实验更加完整，更加具有说服力，在两个数据集上均有着很好的性能表现。

由于基于单目视觉的 3D 目标检测缺乏目标的深度信息，本文提出基于几何约束理论求解出目标空间位置坐标并通过 ShiftNet 对其进行误差修正。为了更好的说明本文提出的算法在预测目标深度时优于其它算法，下表 5-8 给出了随着目标离相机距离不断增加，不同检测算法预测的目标深度与真值深度之间的平均误差统计数据，由于不是所有的算法均给出了其实验结果，所以本文只选择 3DOP、Mono3D、Deep3DBox、SubCNN 这四种算法与本文所提出的算法对比：

表 5-8 不同算法平均误差结果。

算法	距离区间 (单位: 米)				
	[0-10]	[10-20]	[20-30]	[30-40]	[40-50]
3DOP <sup>[6]</sup>	0.5899	0.6015	1.0821	2.0348	3.5918
Mono3D <sup>[5]</sup>	1.1571	1.3984	2.7725	4.5575	5.0372
Deep3DBox <sup>[8]</sup>	1.4569	1.0379	1.8936	2.4746	2.9394
SubCNN <sup>[63]</sup>	1.5449	1.8513	2.6507	4.1296	6.1791
本文提出	1.3568	0.9878	1.9901	2.2076	2.8007

如表 5-8 所示, 将目标离相机的距离分为 5 个区间, 分别为 0-10 米、10-20 米、20-30 米、30-40 米、40-50 米。在每个区间内统计所有目标预测的深度的平均误差。为了更好的说明预测深度误差与目标离相机距离之间的关系, 将上表数据绘制成相应的折线图, 如下图 5-7 所示。横坐标为不同的距离区间, 纵坐标为平均距离误差, 即每个区间内所有目标预测的深度与该目标实际深度之间差值的统计平均值。

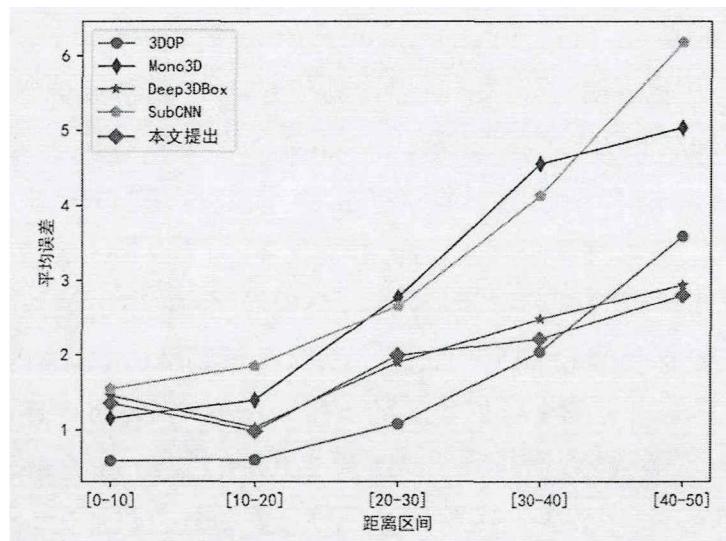


图 5-7 不同 3D 目标检测算法预测的平均误差对比图。

分析可知, 离相机较近的目标, 基于单目视觉的检测算法表现出相似的误差值, 而基于深度图的 3DOP 误差比较小, 如区间 0-10 米。随着目标离相机的距离不断增加, 所有算法预测的目标深度误差也随之增加。值得指出的是, 本文提出算法产生的误差在此过程中均低于 Mono3D, SubCNN。且在距离为 40-50 米的范围内, 与所有的算法相比, 具有最小的平均误差。虽然 3DOP 算法在目标距离小于 40 米时表现最好, 但是 3DOP 使用了深度信息, 所以并不是严格的基于单目视觉的 3D 目标检测算法。并且在较远距离 40-50 米时误差高于本文提出的

算法。通过以上分析证明本文提出的算法不论是处理近处还是远处微小的目标均有着良好的表现。

本文提出的基于单目视觉的 3D 目标检测算法实验结果如下图 5-8 所示：

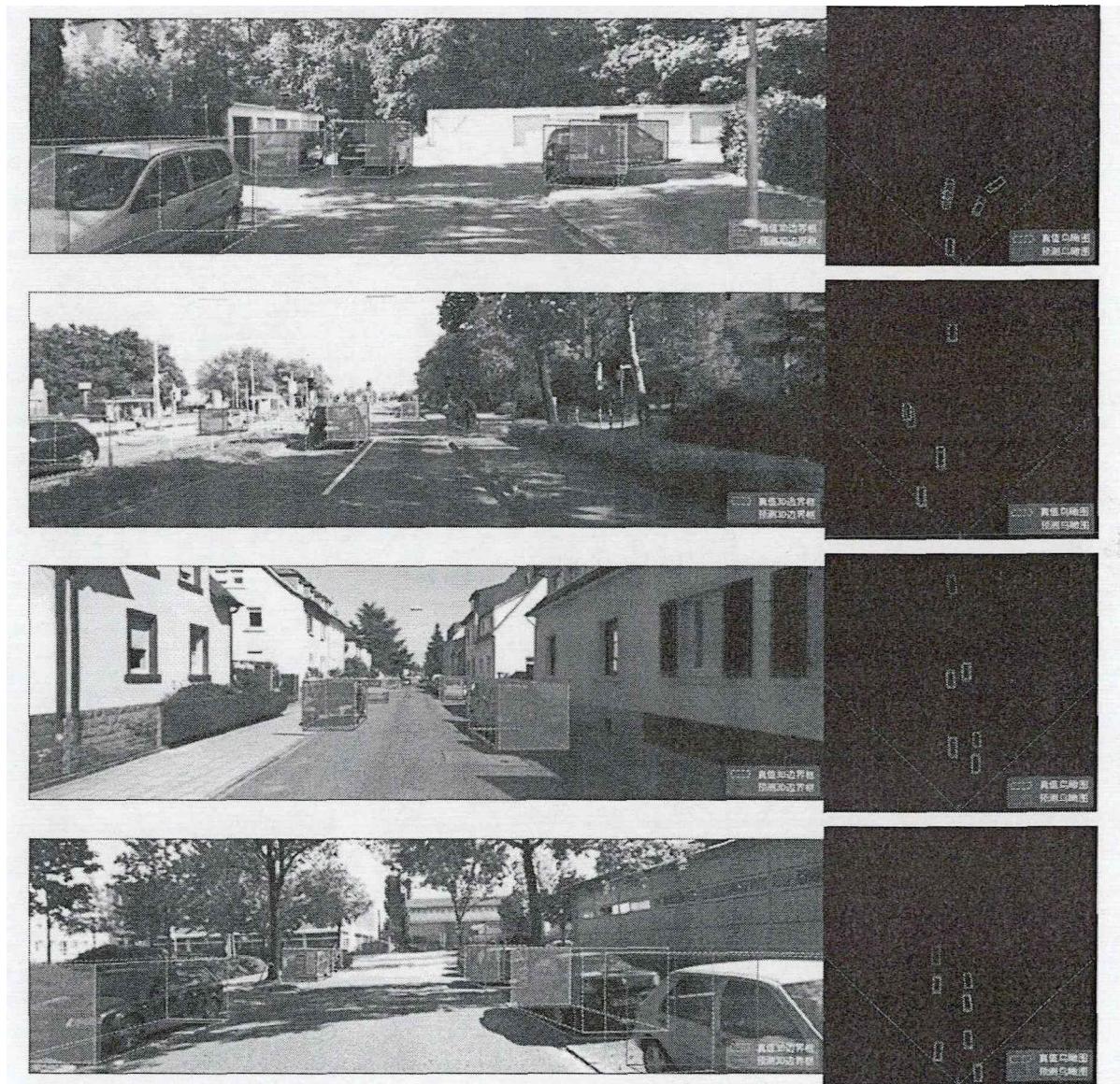


图 5-8 实验结果图。

为了更好的说明算法性能，本文将预测和真值 3D 边界框同时绘制在 RGB 图像上。如图 5-8 所示，每一排左图中橘色虚线 3D 边界框为真值边界框，绿色实线 3D 边界框为预测边界框。右图为对应的预测和真值 3D 边界框鸟瞰图。如实验结果图第 1 排所示，对于多车辆且具有遮挡和轻微截断的复杂交通场景下，本文提出的算法不论是在鸟瞰图还是 3D 边界框上都表现出了良好的检测效果。由第 2 排可见，对于小目标和严重截断的目标都能很准确的预测出来。证明本文提出的算法具有很好的鲁棒性，不会随着目标被遮挡或者截断而出现性能急剧下降。从第 3 排和第 4 排可以看出，对于路边停放的车辆，虽然出现了漏检和误

检，但对于那些被正确检测到的车辆，本文提出的算法仍然表现良好。综上所述，本文提出的 3D 目标检测算法对截断目标、小目标、复杂场景下的多目标均具有很高的检测精确度。

### 5.3 本章小节

本章主要对本文提出的 3D 目标检测框架进行实验与评估。从局部参数评估到整体框架评估，充分证明了本文提出的算法可行性。对评估整体 3D 目标检测性能，本文采取 AP<sub>3D</sub> 和 AP<sub>BEV</sub> 两个评估指标，并设置不同的 IoU 阈值，通过与其它 3D 检测算法对比，充分说明本文提出的 3D 目标检测框架具有很好的检测效果。本文提出的算法在数据集 train/val1 和 train/val2 均进行了实验，所以更加具有说服力，在两个数据集上均有着很好的性能表现。

## 第六章 总结与展望

### 6.1 工作总结

随着科学技术的不断进步，自动驾驶技术越来越成熟。自动驾驶被期望能提高驾驶安全性、实现交通流量控制，从而提升最终的交通效率。自动驾驶的安全性高度依赖于车辆对周围环境精确的感知。而能快速且准确的检测到周围车辆是后续自动驾驶进行相关控制决策的基础。3D 目标检测能提供目标在三维空间形状、姿态等信息，这为后续进行路径规划和控制提供重要参考依据。相比较于雷达和深度相机的昂贵，单目相机价格低廉，且能很方便的配备在交通车辆上。所以基于单目视觉实现 3D 目标检测有着很好的商业前景和研究意义。本文基于单目视觉提出了一个高效的 3D 目标检测框架，将 3D 目标检测任务拆分成几个子任务，每一个子任务负责预测 3D 目标检测涉及到的特定参数，全部子任务的完成意味着整个 3D 目标检测任务的完成。在 KITTI 数据集上的实验结果表明，本文提出的 3D 目标检测算法优于其它的算法，在相关评估指标上有了显著的提升。本文的工作总结如下所示：

(1) 目标维度的三个分量具有很强的内在联系，应该将其作为一个整体考虑。基于此本文提出了一种基于 IoU 的维度预测损失函数。本文将 2D 目标检测任务中用来计算预测 2D 边界框与真值 2D 边界框重合度的指标 IoU 引入到损失函数中，并将应用于 2D 边界框的 IoU 扩展到三维空间，提出了  $\text{IoU}_{3D}$ 。实验结果表明本文提出的适用于维度预测的 IoU 损失函数与其它损失函数相比在平均维度误差指标上有着更低的数值，为完成最终的 3D 目标检测任务提供精确的数据支撑。

(2) 目标旋转角  $\theta_{yaw}$  与目标外观无直接联系，而基于单目视觉完成 3D 目标检测仅仅提供了 RGB 图像信息。为了解决  $\theta_{yaw}$  预测问题，本文提出预测局部旋转角  $\theta_{alpha}$ ，通过  $\theta_{yaw}$  与  $\theta_{alpha}$  之间的几何关系计算出  $\theta_{yaw}$ 。同时本文考虑到旋转角预测过程中可能产生的角度模糊问题，为此选择将连续变量旋转角的回归问题转变成了离散变量的分类问题，使得旋转角的预测更加精确。实验结果表明本文提出的旋转角预测算法优于其它算法。

(3) 针对基于单目视觉难以获取目标空间位置坐标的问题，本文依据几何约束理论，结合最小二乘法计算出目标初步的位置坐标。几何约束意味着 3D 边界框在成像平面的投影会严格约束在 2D 边界框之内，更进一步说即 2D 边界框的每一条边，都至少会有 3D 边界框 8 个顶点中的某一个顶点投影到这条边上。同时本文根据预测出的  $\theta_{alpha}$  将可能的 4096 种约束降低为 64 种，大大提高了后

处理效率。为了解决目标遮挡、截断和几何约束过程中带来的误差等问题，本文采用了一种优化网络。将全部已知的 2D 边界框信息、3D 边界框信息输入到网络中，对位置坐标进行修正。在算法评估阶段，以多个评估指标评价本文提出的 3D 目标检测框架的性能，并与其它检测算法对比。实验结果表明本文提出的算法在相关指标上有了显著的提升，优于其它算法。

## 6.2 未来展望

本文完全基于单目相机结合几何约束理论提出了一种 3D 目标检测框架。并与其它基于单目视觉的算法相比，在多个评估指标上有了显著的提升。但如第五章实验所示，本文提出的算法虽优于其它单目视觉目标检测算法，但在相关指标上却低于基于深度图的 3D 目标检测算法。本质上是由于 RGB 图像三维空间信息丢失造成的。所以对于此，有以下可继续进一步探索的地方：

(1) 本文提出的目标检测框架由 4 个模块构成，而不是一个端到端的检测系统，是分布进行的。不同的子模块会带来额外的噪声干扰，影响整个 3D 目标检测框架的性能。如何将各个模块整合在一起，形成一个端到端的检测系统值得进一步深入研究。

(2) 由实验可知基于单目视觉的 3D 目标检测在相关指标上与其它传感器有着比较大的差距。现如今基于多传感器融合也成为 3D 目标检测领域热门研究方向之一。因此接下来可以考虑如何结合其它传感器的优势，提高基于单目视觉的 3D 目标检测算法性能。

## 参考文献

- [1] Agarwal A , Misra G , Agarwal K , et al. The 5th Generation Mobile Wireless Networks- Key Concepts, Network Architecture and Challenges[J]. American Journal of Electrical and Electronic Engineering, 2015, 3(2): 22-28.
- [2] 张婷. 基于百度 Apollo2.0 的无人驾驶策略分析[J]. 现代经济信息, 2019(2).
- [3] Guerry J , Boulch A , Saux B L , et al. SnapNet-R: Consistent 3D Multi-view Semantic Labeling for Robotics[C]// 2017 IEEE International Conference on Computer Vision Workshop (ICCVW). IEEE, 2017.
- [4] Chabra R , Straub J , Sweeny C , et al. StereoDRNet: Dilated Residual Stereo Net[J]. IEEE, 2019.
- [5] Chen X , Kundu K , Zhang Z , et al. Monocular 3D Object Detection for Autonomous Driving[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2016.
- [6] Xiaozhi Chen\*, Kaustav Kundu \*, Zhu Y . 3D Object Proposals for Accurate Object Class Detection[C]// International Conference on Neural Information Processing Systems. MIT Press, 2015.
- [7] Pham C C , Jeon J W . Robust Object Proposals Re-ranking for Object Detection in Autonomous Driving Using Convolutional Neural Networks[J]. Signal Processing Image Communication, 2017, 53:110-122.
- [8] Mousavian A , Anguelov D , Flynn J , et al. 3D Bounding Box Estimation Using Deep Learning and Geometry[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.
- [9] Brazil G , Liu X . M3d-rpn: Monocular 3d region proposal network for object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 9287-9296.
- [10] Liu L , Lu J , Xu C , et al. Deep fitting degree scoring network for monocular 3d object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 1057-1066.
- [11] Li B , Ouyang W , Sheng L , et al. GS3D: An Efficient 3D Object Detection Framework for Autonomous Driving[C]// 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2019.
- [12] Xu B , Chen Z . Multi-level fusion based 3d object detection from monocular images[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 2345-2353.
- [13] Shelhamer E , Long J , Darrell T . Fully Convolutional Networks for Semantic Segmentation[M]. IEEE Computer Society, 2017.
- [14] Qin Z , Wang J , Lu Y . MonoGRNet: A Geometric Reasoning Network for Monocular 3D Object Localization[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33:8851-8858.
- [15] Chabot F , Chaouch M , Rabarisoa J , et al. Deep MANTA: A Coarse-to-Fine Many-Task Network for Joint 2D and 3D Vehicle Analysis from Monocular

- Image[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.
- [16] Li B , Zhang T , Xia T . Vehicle Detection from 3D Lidar Using Fully Convolutional Network[J]. 2016.
- [17] Simony M, Milzy S, Amendey K, et al. Complex-yolo: An euler-region-proposal for real-time 3d object detection on point clouds[C]//Proceedings of the European Conference on Computer Vision (ECCV) Workshops. 2018: 0-0.
- [18] Engelcke M , Rao D , Wang D Z , et al. Vote3Deep: Fast Object Detection in 3D Point Clouds Using Efficient Convolutional Neural Networks[C]// 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017.
- [19] Zhou Y, Tuzel O. Voxelnet: End-to-end learning for point cloud based 3d object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 4490-4499.
- [20] Li B. 3d fully convolutional network for vehicle detection in point cloud[C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017: 1513-1518.
- [21] Chen X , Ma H , Wan J , et al. Multi-View 3D Object Detection Network for Autonomous Driving[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2017.
- [22] Ku J, Mozifian M, Lee J, et al. Joint 3d proposal generation and object detection from view aggregation[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018: 1-8.
- [23] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]//Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. IEEE, 2001, 1: I-I.
- [24] Viola P, Jones M J. Robust real-time face detection[J]. International journal of computer vision, 2004, 57(2): 137-154.
- [25] Papageorgiou C P, Oren M, Poggio T. A general framework for object detection[C]//Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271). IEEE, 1998: 555-562.
- [26] Papageorgiou C, Poggio T. A trainable system for object detection[J]. International journal of computer vision, 2000, 38(1): 15-33.
- [27] Mohan A, Papageorgiou C, Poggio T. Example-based object detection in images by components[J]. IEEE transactions on pattern analysis and machine intelligence, 2001, 23(4): 349-361.
- [28] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Ieee, 2005, 1: 886-893.
- [29] Felzenszwalb P, McAllester D, Ramanan D. A discriminatively trained, multiscale, deformable part model[C]//2008 IEEE conference on computer vision and pattern recognition. IEEE, 2008: 1-8.
- [30] Felzenszwalb P F, Girshick R B, McAllester D. Cascade object detection with deformable part models[C]//2010 IEEE Computer society conference on computer vision and pattern recognition. IEEE, 2010: 2241-2248.

- [31] Malisiewicz T, Gupta A, Efros A A. Ensemble of exemplar-svms for object detection and beyond[C]//2011 International conference on computer vision. IEEE, 2011: 89-96.
- [32] Felzenszwalb P F, Girshick R B, McAllester D, et al. Object detection with discriminatively trained part-based models[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 32(9): 1627-1645.
- [33] Girshick R, Felzenszwalb P, McAllester D. Object detection with grammar models[J]. Advances in Neural Information Processing Systems, 2011, 24: 442-450.
- [34] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.
- [35] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
- [36] Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.
- [37] Van de Sande K E A, Uijlings J R R, Gevers T, et al. Segmentation as selective search for object recognition[C]//2011 International Conference on Computer Vision. IEEE, 2011: 1879-1886.
- [38] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.
- [39] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. arXiv preprint arXiv:1506.01497, 2015.
- [40] Neubeck A, Van Gool L. Efficient non-maximum suppression[C]//18th International Conference on Pattern Recognition (ICPR'06). IEEE, 2006, 3: 850-855.
- [41] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
- [42] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.
- [43] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.
- [44] He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.
- [45] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
- [46] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

- [47] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]/European conference on computer vision. Springer, Cham, 2016: 21-37.
- [48] Fu C Y, Liu W, Ranga A, et al. Dssd: Deconvolutional single shot detector[J]. arXiv preprint arXiv:1701.06659, 2017.
- [49] Duan K, Xie L, Qi H, et al. Corner proposal network for anchor-free, two-stage object detection[J]. arXiv preprint arXiv:2007.13816, 2020.
- [50] Li Y , Chen Y , Wang N , et al. Scale-Aware Trident Networks for Object Detection[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2020.
- [51] Zhou X, Wang D, Krähenbühl P. Objects as points[J]. arXiv preprint arXiv:1904.07850, 2019.
- [52] Naiden A, Paunescu V, Kim G, et al. Shift r-cnn: Deep monocular 3d object detection with closed-form geometric constraints[C]//2019 IEEE International Conference on Image Processing (ICIP). IEEE, 2019: 61-65.
- [53] 李航. 统计学习方法[M]. 清华大学出版社, 2012.
- [54] 周志华. 机器学习 := Machine learning[M]. 清华大学出版社, 2016.
- [55] Li Z, Wang Y, Ji X. Monocular viewpoints estimation for generic objects in the wild[J]. IEEE Access, 2019, 7: 94321-94331.
- [56] Goodfellow I , Bengio Y , Courville A . Deep Learning[M]. The MIT Press, 2016.
- [57] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
- [58] Cai Z, Fan Q, Feris R S, et al. A unified multi-scale deep convolutional neural network for fast object detection[C]/European conference on computer vision. Springer, Cham, 2016: 354-370.
- [59] 张贤科, 许甫华. 高等代数学[M]. 清华大学出版社, 1998.
- [60] Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain[J]. Psychological review, 1958, 65(6): 386-408.
- [61] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the kitti vision benchmark suite[C]//2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2012: 3354-3361.
- [62] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The kitti dataset[J]. The International Journal of Robotics Research, 2013, 32(11): 1231-1237.
- [63] Choi W , Lin Y , Xiang Y , et al. Subcategory-aware convolutional neural networks for object detection[J]. 2018.
- [64] Roddick T , Kendall A , Cipolla R . Orthographic Feature Transform for Monocular 3D Object Detection[J]. 2018.

## 致谢

如果说人生是一段旅行，那么研究生这一段时光将是我永远难忘的风景。研究生三年的学习时光让我依依不舍，有欢笑，有泪水。实验室里遇到了可爱的师兄师姐、师弟师妹，遇到了和蔼的老师们，这一切都让我觉得自己如此幸运。

感谢我的父母将我抚养成人，默默的付出供我读书。在我幼稚时包容我的任性，在我失落时安慰我，在我叛逆时将我引入正确的道路上来。在我回家时给我烧一桌可口的饭菜。越长大越体谅父母的不容易。感谢父母为我付出的一切。

感谢我的研究生导师刘老师，感谢实验室的杜老师，感谢实验室的袁老师，感谢实验室所有的老师们。是他们教会了我如何科研，教会我如何读文献，教会我如何解决科研上遇到的困难。这次论文从选题到研究到撰稿，导师给予了最为耐心细致的指导。实验室的每次组会都让我收获很多。感谢老师教导我三年，指导我科研。感谢老师付出的一切。

感谢实验室可爱的师兄师姐们。初来实验室，面对陌生的环境是他们的热情打消了我的不安。感谢实验室的王师姐、刘师姐、毛师姐、苏师兄、吴师兄、苏师姐、高师姐在日常学习中给予我的帮助。

感谢实验室可爱的同届同学。最先认识的便是优秀的你们。与你们同窗三年，打打闹闹，那一段回忆会永远留在心中。感谢毛同学、郭同学、汪同学、宋同学。很高兴在实验室遇到你们，与你们一同学习进步。在这个毕业季，一起走向职场，迎接下一个挑战。

感谢实验室可爱的师弟师妹们。你们的学习态度深深的影响着我。感谢马师弟、尹师弟、韩师弟、杨师妹、王师妹、秦师弟、赵师弟、张师弟。你们都是实验室最可爱的人。虽然我要早一步离开实验室，但此处送上最美的祝福，期待与你江湖再见。

行文至此虽已画上句号，但世间的故事却还在继续前进。你和我不管身处光明或者黑暗，内心一定全是祝福与美好。

## 攻读硕士学位期间发表的学术论文目录

- [1] 曹波, 刘勇, 杜海清. 基于单目视觉的3D目标检测场景中旋转角估计算法研究[EB/OL]. 中国科技论文在线, 202012-16.