DOI: 10.19650/j.cnki.cjsi.J2107801

基于 Stereo RCNN 的锚引导 3D 目标检测算法*

曹杰程[1],陶重犇[1,2]
(1. 苏州科技大学电子与信息工程学院 苏州 215009; 2. 清华大学苏州汽车研究院 苏州 215134)

摘 要:针对当前基于锚的双目 3D 目标检测算法存在的锚点数量选取较多,从而影响在线计算速度的问题,提出了一种基于 Stereo RCNN 的锚引导 3D 目标检测算法 FGAS RCNN。在第 1 阶段中,输入左右图像分别生成相应的概率图以生成稀疏锚点及稀疏锚框,再通过将左右锚作为一个整体生成 2D 预选框。第 2 阶段的关键点生成网络利用稀疏锚点信息生成关键点热图,并结合立体回归器融合生成 3D 预选框。针对原始图像在卷积后会出现像素级信息丢失的问题,通过 Mask Branch 生成的实例分割掩模结合实例级视差估计进行像素级优化。实验表明,在没有任何深度和位置先验信息输入的情况下,此方法依旧可以在减少计算量的同时保持较高的召回率。具体来说,此方法在以 0.7 为阈值的 3D 目标检测上平均精度为 44.07%。相比于 Stereo RCNN,本文方法在平均精度上提高了 4.5%。与此同时,此方法的整体运行时间较 Stereo RCNN 缩短了 0.09 s。

关键词: 3D 目标检测;立体视觉;关键点检测;稀疏锚点;实例分割
中图分类号: TP391.41 TH741 **文献标识码**: A **国家标准学科分类代码**: 520.6040

An anchor-guided 3D target detection algorithm based on stereo RCNN

Cao Jiecheng[1], Tao Chongben[1,2]
(*1. School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215009, China; 2. Suzhou Automotive Research Institute, Tsinghua University, Suzhou 215134, China*)

Abstract: The current binocular 3D detection algorithm has the problem of slow online calculation speed due to a large number of anchor points to be selected. To address this issue, an anchor-guided 3D target detection algorithm is proposed, which is based on the stereo RCNN. This method is named as the FGAS RCNN. In the first stage, a probability map is generated for the left and right input images to generate sparse anchor points and corresponding sparse anchor boxes. The left and right anchors are used as the whole entirety to generate a 2D preselection box. The second stage is based on the key-point generation network of the pyramid feature network. The key-point heatmaps are generated by the information of these sparse anchor points. A 3D bounding box can be generated by combining the stereo regressor with these key-point heatmaps. The original image will lose pixel-level information after convolution. The instance segmentation mask generated by Mask Branch can be used to solve this problem. The 3D bounding box center depth precision can be improved by the instance segmentation mask and the instance-level disparity estimation. Experimental results show that the proposed method can reduce the amount of calculation while maintaining a high recall rate without any depth and position prior information input. Specifically, the mean average precision is 44.07% on 3D target detection with a threshold of 0.7. Compared with the stereo RCNN, the proposed method improves the average precision by 4.5%. Meanwhile, the overall running time of our method is 0.09 s shorter than Stereo RCNN.

Keywords: 3D target detection; stereo vision; key point detection; sparse anchor point; instance segmentation

0 引 言

作为自动驾驶领域基础之一的目标检测是近几年较为热门的话题。以 Faster R-CNN[1], Mask R-CNN[2] 系列为引导的 2D 目标检测无论是在精度还是检测速度上均达到了较高的水准。但是仅仅只有 2D 目标检测依旧不能满足自动驾驶领域的需求,因此 3D 目标检测应运而

收稿日期:2021-04-22 Received Date: 2021-04-22
*基金项目:国家自然科学基金(61801323)、苏州市民生科技项目(SS2019029)、中国博士后科学基金(2021M691848)项目资助

生。3D 目标检测可以获取 2D 目标检测无法提供的物体的空间位置和距离等对于自动驾驶至关重要的信息。但是现阶段功能强大的 3D 检测器都严重依赖于 LiDAR 提供的数据信息,而高精度 LiDAR 的价格十分昂贵。相比之下,立体相机的性价比和实用性都很高,这使得其在现阶段许多复杂应用场景中引起了越来越多的关注。

现阶段主流的 3D 目标检测方法主要分成 4 类:基于单目图像的方法[3-4]、基于立体图像的方法[5-6]、基于图像和点云融合的方法[7-8]以及基于原始点云的方法[9-10]。并且这些不同方法之间各有其优劣之处。

本文是一种基于立体图像的方法,并且设计了一种使用稀疏锚进行左右感兴趣区域(region of interest, ROI)提议的方法称为 FGAS-RCNN。本文的网络架构分为 3 个模块。首先使用引导立体 RPN 模块(guided stereo region proposal network, GS RPN)对特征图输出前景位置的概率图,并生成稀疏锚点以预测对象形状。然后根据预测的锚点位置和锚框形状输出相应的左右 ROI 提案。而关键点检测网络模块通过多级锚方案以生成关键点热图,并预测包括 3D 边界框顶点和中心点在内的 9 个关键透视点,然后通过这些关键点得到的 3D 框约束来校正得到的粗略 3D 框。最后的中心深度校正模块主要利用对左右特征图使用 ROIAlign 生成的 ROI 及掩模生成实例分割掩模,结合视差计算每个像素的深度值以进行 3D 框中心深度优化。

本文进行了如下的创新:

1)针对现存锚点选取方法计算量较大且速度较慢的问题,提出了一个基于自适应锚框的引导立体 RPN 方法。将稀疏锚点与 GS RPN 结合的方法可以有效节省计算成本,提高计算效率;

2)针对立体 3D 目标检测方法中 3D 边界框精度较低的问题,提出了一个基于金字塔网络的关键点生成网络。通过引入融合了高级特征的多级锚方案的特征图,对 3D 边界框 9 个关键点约束进行提取,并减少输入的负样本数量,提高正样本占比;

3)针对卷积操作后会导致原始图像像素级信息丢失的问题,提出了一个基于像素级实例视差的 3D 边界框中心深度校正方法。

1 FGAS RCNN 网络架构

与诸如 Stereo RCNN[5]之类的均匀锚点检测器相比,FGAS RCNN 能根据对象大小和位置调整锚点的分布及大小。本文使用 ResNet-50 作为骨干网。FGAS RCNN 的框架如图 1 所示,本节将着重介绍图中的 3 个新颖模块。

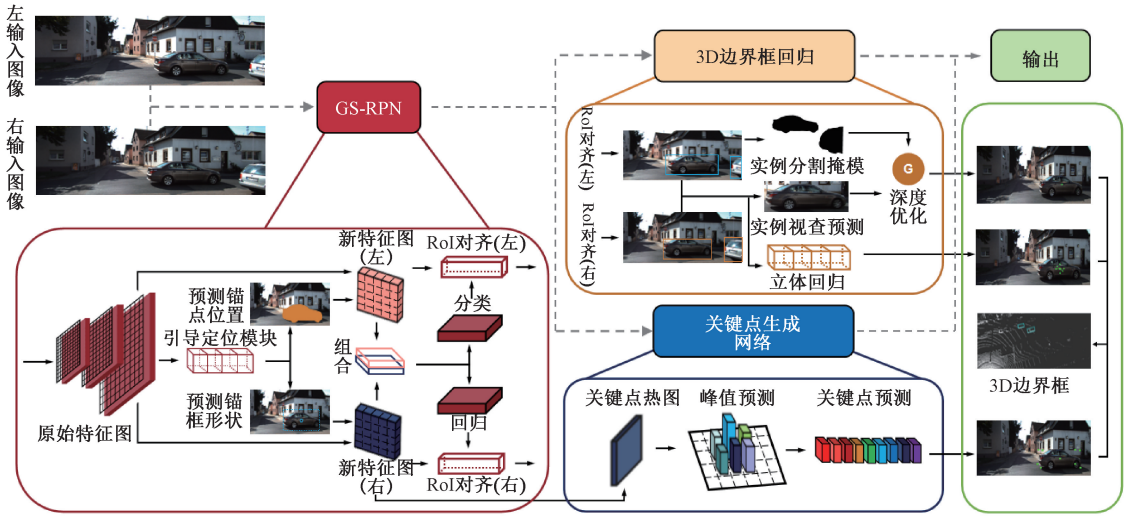


图 1 FGAS RCNN 算法框架图

Fig. 1 Framework of the FGAS RCNN algorithm

1.1 引导立体 RPN 模块

区域候选网络(region proposal network, RPN)通过在特征提取后利用 3×3 卷积减少通道和两个全连接层对输入位置回归对象种类和框偏移量。而本文方法的特殊之处在于仅对观察到的物体分配锚点,而不是均匀的分布在图像上,并且可以根据目标的几何形状来调整特征。本文的方案如图 2 所示,该方法包含 2 个分支,即锚点定位分支和锚框预测分支。

在锚点预测分支中,将一个 1×1 卷积和 Sigmod 函数应用到输入的特征图 F_l 中,生成一个与特征图相同大小的概率图 P 以获取客观性得分。式(1)代表了检测对象在该位置上的概率。

$$P(x_a, y_a \mid F_l) = P\left(\left(x_a+\frac{1}{2}\right)S, \left(y_a+\frac{1}{2}\right)S \mid I\right) \tag{1}$$

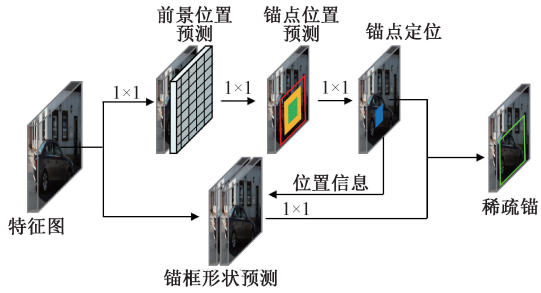


图 2　锚点生成方案

Fig. 2　Anchor point generation scheme

式中:(x_a, y_a)对应于输入左特征图 F_l 上的坐标,S 代表了特征图的步长。

只选择概率高于预设阈值的点作为可能存在对象的活动区可以在保证召回率的同时极大缩小可能存在对象的区域。而锚框预测分支会根据 F_l 和 P 预测概率高于阈值区域的最佳形状,即该形状与最接近的地面真值(ground-truth, GT)有最高的覆盖率。本文将能与最近 GT 框有最大交并比(intersection over union, IoU)的 w_a, h_a 作为预测的锚框尺寸。本文方法在每个位置仅预测一个最佳形状的锚框,而不是一组预定义的锚框。

由于本文中所有锚框都随位置改变而变化,所以采用了如图 3 所示的多级锚方案。该方案依照特征金字塔(feature pyramid network, FPN)体系结构在多尺度特征图中采集锚点,且锚点可以在所有尺度特征图中共享。为了将不同特征与其相应范围对应,通过基础锚点形状和一个 3×3 的可变卷积层从锚框预测分支输出中预测一个偏移量。然后将具有偏移量的可变形卷积应用到原始特征图 F_l 中获得新特征图 F_l'。

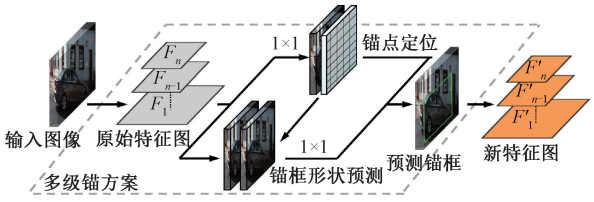


图 3　多级锚方案

Fig. 3　Multi-stage anchor scheme

在每个尺度上连接左右输出的新特征图,并将串联后的特征馈入 Stereo RPN 网络得到精确的检测框。如图 4 所示,与常规的对象目标不同,本文将串联的左右特征图当成对象分类的目标。将左右特征图串联后输入到 GS RPN 网络中,并通过 IoU 匹配对应的检测框。在进行 ROI 采样时,只有在同一个锚点位置的左右两侧检测框同时与对应并集 GT Box 的 IoU 都大于 0.5 才会被作为前景。并且如果两侧检测框与同一个并集 GT Box 的 IoU 都大于 0.8 时,则认为这两个检测框预测同一个目标对象。

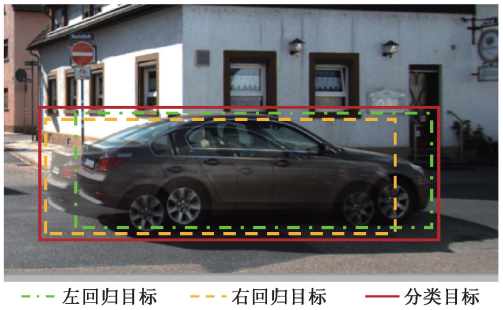


图 4　目标分配

Fig. 4　Target allocation

传统 RPN 回归器一般只有 4 个输出,而 GS RPN 的回归器一般有 6 个输出。本文采用了式(2)中的 6 个坐标参数化来进行 2D 边界框的回归。然后通过对左右侧 ROI 区域使用非极大值抑制(non-maximum suppression, NMS),再次进行检测框的筛选。

$$\begin{cases} t_{xl} = \dfrac{(x - x_a)}{w_a},\ t_{wl} = \log\left(\dfrac{w}{w_a}\right) \\ t_{xr} = \dfrac{(x' - x'_a)}{w'_a},\ t_{wr} = \log\left(\dfrac{w'}{w'_a}\right) \\ t_y = \dfrac{(y - y_a)}{h_a} = \dfrac{(y' - y'_a)}{h'_a} \\ t_h = \log\left(\dfrac{h}{h_a}\right) = \log\left(\dfrac{h'}{h'_a}\right) \end{cases} \tag{2}$$

式中:x, y, w, h 代表了预测框的水平和垂直坐标,宽和高;x, x', x_a 分别代表了左预测框,右预测框和锚框的水平坐标。

本文在训练中将每个 ROI 采样定义为一个多任务损失:$L = L_{cls} + L_{reg} + L_{ga}$。除了常规的分类损失 L_{cls} 和回归损失 L_{reg} 之外,还引入引导锚损失 L_{ga}。分类损失和回归损失的定义类似于文献[1]。引导锚模块通过逐像素的 S 形函数对每个像素输出其为目标对象的概率,并通过对一些常用(w, h)进行采样来模拟所有数值的遍历,所以引导锚损失 L_{ga} 如式(3)所示。

$$L_{ga} = -\alpha(1 - P_i)^{\gamma}\log(P_i) + L_1\left(1 - \min\left(\frac{w}{w_g}, \frac{w_g}{w}\right)\right) + L_1\left(1 - \min\left(\frac{h}{h_g}, \frac{h_g}{h}\right)\right) \tag{3}$$

式中:i 代表的是某个锚点对应的索引;P_i 代表了该锚点是目标对象的概率;加权因子 $\alpha \in [0,1]$,聚焦参数 $\gamma \in [0,5]$,在本文中选择 $\gamma = 2, \alpha = 0.25$;$(w, h)$ 和 (w_g, h_g) 分别表示预测锚框和对应 GT 框的宽和高。

1.2 关键点生成网络

本文的关键点生成网络仅对右侧图像进行关键点预测,并且仅将通过多级锚方案生成的新特征图作为输入。本文从 3D 边界框的顶点和中心点生成透视点,然后将输出的中心点热图,顶点热图,顶点坐标和视点角度作为基本模块从而进行 3D 框回归和校正。

为了避免由尺度过小导致的关键点重叠问题,采取了如图 5 所示的方法。由于图像中的关键点不存在大小上的差异,所以将多级锚方案生成的每个尺度特征图通过 3 个双线性插值进行 3 次上采样,并在每次后加入一个 1×1 卷积层减小通道。在上采样之前要级联相应的特征图,将得到的 F 个多尺度特征图 $f(1 \leqslant f \leqslant F)$ 调整到最大尺度大小,并通过 Softmax 函数运算生成软权重 ∂。通过这些生成的软权重可以直观的看出每个尺度的重要程度。然后通过线性加权就可以获取尺度空间得分图 S_{scale}。

$$S_{scale} = \sum_{\hat{f}} \hat{f} \odot \partial_{\hat{f}} \tag{4}$$

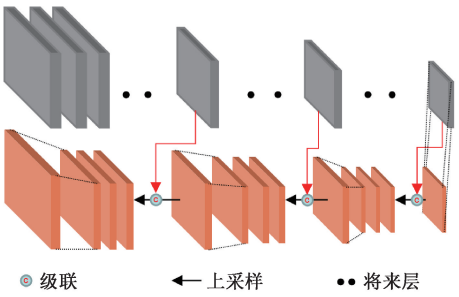


图 5 上采样流程

Fig. 5 Up-sampling process

如图 6 所示,检测头主要有 3 个组件构成。通过将锚点位置作为可能存在的关键点位置可以有效避免在截断情况下对象的 3D 投影点超出图像边界的情况。将 2D 边界框中心点热图定义为 $M_m \in [0,1]^{\frac{H}{s}\times\frac{W}{s}\times C}$,其中,$H$, W 代表输入图像的宽和高,C 代表对象类别的数量,S 代表了步长。检测头的另一个组成部分是 3D 边界框顶点和中心点投影的 9 个透视点的热图 $M_v \in [0,1]^{\frac{H}{s}\times\frac{W}{s}\times 9}$。

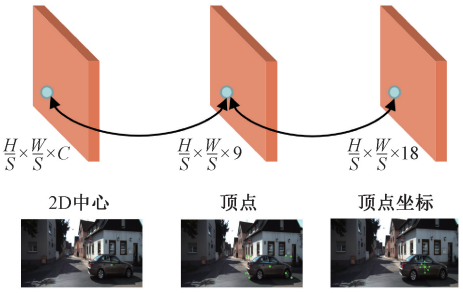


图 6 多任务检测头组成

Fig. 6 Composition of multi-task head

从 2D 边界框中心回归的局部偏移量为 $V_C \in R^{\frac{H}{s}\times\frac{W}{s}\times 18}$,本文将最接近 V_C 坐标的 9 个关键点认为是同一个对象的一组关键点坐标。并通过这 9 个关键点的 18 个约束来恢复对象的 3D 边界框。

虽然本文通过多级锚方案消除了大量的负样本,但是关键点生成网络的训练目标依旧是为了解决正负样本与焦点损失的不平衡问题。

$$L_{key}^{n} = -\frac{1}{N}\sum_{k=1}^{n}\sum_{x=1}^{\frac{H}{S}}\sum_{y=1}^{\frac{W}{s}} \begin{cases} (1-p_t)^{\alpha}\log(p_t), & Y_{kxy}=1 \\ (1-Y_{kxy})^{\beta}(1-p_t)\log(p_t), & Y_{kxy}=-1 \end{cases} \tag{5}$$

式中:N 代表了图像中中心点和顶点的数量;n 代表了不同的关键点通道(若 $n=c$ 代表在中心点处,$n=9$ 代表在顶点处);α, β 代表了用于平衡正负样本的权重的超参数:

$$p_t = \begin{cases} p, & Y_{kxy}=1 \\ 1-p, & Y_{kxy}=-1 \end{cases} \tag{6}$$

式中:p 代表了对象处于关键点处的估计概率。

1.3 立体回归和 3D 框估计

通过 GS-RPN 后,对左右特征图分别使用 Mask RCNN 中提出的 ROI Align。之后合并左右 ROI 特征,并输入到两个全连接层以提取语义信息。通过立体回归器,可以得到包括对象类别,立体边界框,物体尺寸和视点角度的 4 个输出。本文使用汽车前进方向以及 ROI 视角夹角作为视点回归角度,并选择[sinβ, cosβ]作为回归量避免不连续性。然后通过将回归角度与 3D 位置之间的关系去耦获得车辆方向。将立体框与物体尺寸结合的方法可以更好的获得深度信息。

除了视点角度和深度信息,本文还通过投影在 2 框中间的 9 个透视关键点为 3D 框估计提供额外的 18 个约束。对于一个输入图像 I,本文的关键点检测网络会给出 9 个关键点来表示一组 N 个对象。相应的 3D 边界框可以通过视点角度 θ_i,3D 框中心坐标 $M_i = [x_i, y_i, z_i]$ 和回归尺寸 $D_i = [w_i, h_i, l_i]$。然后通过给定透视关键点信息可以推导出 3D 框顶点与 2D 框顶点间的对应关系,并通过高斯-牛顿法求解由投影变换得到的 3D-2D 关系公式。

本文将最小化 3D 关键点和 2D 关键点的重投影问题化为非线性最小二乘优化问题。

$$\mathrm{R}^{*}, M^{*}, D^{*} = \operatorname{argmax}_{\theta,M,D} \sum \frac{1}{S_i} K \begin{bmatrix} R & M \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \times \operatorname{diag}(D_i)\operatorname{Cor}\left| \widehat{kp_i} - kp_i \right| + w_d(\hat{D}_i - D_i) + w_r \log(\theta_i^{-1}\hat{\theta}_i) \tag{7}$$

式中:关键点表示为 $\widehat{kp_i}$,其维度和方向分别为 $\hat{D}_i$,$\hat{\theta}_i$;K 是

给定的相机固有矩阵,R 代表 3D 框的旋转角度 $R_i(\theta)$;本文取 $w_d=1$,$w_r=1$。

1.4　中心深度校正

本文从左右边界框提供的视差信息中恢复了大致的深度信息,但是由于之前的处理过程丢失了大量像素级信息,所以使用大量像素级测量来解决 3D 框中心深度的校正问题。

为了排除背景及其他对象的像素对校正的影响,如图 7 所示,本文对左侧图像使用 ROI Align 从特征图中提取对象特征时生成了实例分割掩模。然后通过 FGAS RCNN 中提供的 2D 边界框和实例分割掩模可以从完整图像中裁剪并在水平方向上对齐左右 ROIs。

$$D_i(p)=u_p^L-u_p^R-(b_L-b_R) \tag{8}$$

式中:b_L 和 b_R 分别代表了左 2 边界框的左,右边框归一化坐标 u_p^L,u_p^R;$D_i(p)$ 代表预测的实例视差值。

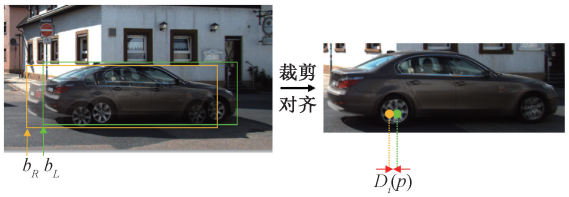


图 7　裁剪和对齐过程

Fig. 7　Cropping and alignment process

通过计算所有掩模区域内的像素的视差并结合基线 B 与相机内参 f 可以计算掩模内每个像素的 3D 位置和深度值,3D 位置的计算公式为:

$$x_p=\frac{(u_p-u_c)}{f_u},\ y_p=\frac{(v_p-v_c)}{f_v} \tag{9}$$

式中:$(u_c,\ v_c)$ 代表相机中心像素位置;$(f_u,\ f_v)$ 分别是相机的水平与垂直焦距。

而其深度计算公式为:

$$Z_p=\frac{Bf}{D_i(p)+|x_{pr}-x_{pl}|}=\frac{Bf_u}{D_i(p)+(b_L-b_R)} \tag{10}$$

式中:x_{pl},x_{pr} 分别代表了像素点 p 在左右边框中的水平坐标。

本文设定的总匹配样本为掩模区域内所有像素的平方差总和为:

$$E=\sum_{p=0}^{N}\left\|I_L(u_P,v_P)-I_R\left(u_P-\frac{B}{Z+\Delta Z_P},v_P\right)\right\| \tag{11}$$

式中:ΔZ_p 表示的是掩模内的像素 P 与 3D 框中心的深度差值,I_L 和 I_R 分别表示的是左右图像中的 3 通道 RGB 矢量。

通过最小化总匹配成本就能得到优化后的中心深入深度,本文使用的是枚举法来产生最优深度 Z。总的过程是先在前文预估的 3D 框深度值的基础上以 0.5 m 为间隔选择 40 个粗略深度,然后以 0.05 m 为间隔对粗略深度再次枚举得到最优深度。通过对掩模区域内所有像素固定对齐深度,可以校正整个 3D 框。并且因为掩模区域内的每个像素均贡献了深度估计值,所以可以避免立体深度估计中的不连续和不适定问题。

本文将提出方法的总的多任务损失定义为:

$$L=\omega_{ga}^P L_{ga}^P+\omega_{cls}^P L_{cls}^P+\omega_{reg}^P L_{reg}^P+\omega_{cls}^r L_{cls}^r+\omega_{sreg}^r L_{sreg}^r+\omega_c^r L_{key}^c+\omega_{ver}^r L_{ver}^8 \tag{12}$$

式中:p,r 作为上标分别代表 RPN 和 RCNN。下标 ga,$sreg$,key,ver 分别代表引导锚模块,立体回归器,3D 框中心点和顶点的损失。

2　实验与分析

在具有挑战性的 Kitti[11] 和 NuScenes[12] 3D 对象检测基准上评估本文提出的方法,并与最新方法进行了比较。然后又进行了消融研究以分析提出方法不同组成部分的有效性。最后,还提供了实际的车载实验平台及实验场地的相关介绍。使用 Ubuntu18.04,搭载 i7-9700k CPU 和双 2080Ti GPU,PyTorch 来运行本文的网络。

2.1　Kitti 上的 3D 目标检测实验

Kitti 对象检测基准包含 7 481 个训练图像和 7 518 个测试图像。本文将训练图像大致分为拥有 3 712 个图像的训练集和 3 769 个图像的测试集。遵循 Kitti 的设置,对象将根据 2D 边界框大小,遮挡和截断程度被分成:容易,中等和困难 3 个级别。

本文使用平均精度(average precision, AP)进行 3D 检测(AP_{3d})和鸟瞰检测(AP_{bev})来评估 3D 检测和定位的性能,如表 1 和 2 所示,将提出的方法以 0.7 和 0.5 作为 IoU 阈值与之前的基于图像的汽车类别 3D 检测最新方法进行了比较。在训练时,本文的方法在指标上均优于 Stereo RCNN。具体来说,这种优势来自于关键点检测网络提供的大量约束条件。

通过图 8 可以直观的看出随着物体距离的增加,视差与深度的误差呈现相反的趋势。并且 3D 检测性能与物体距离也呈现反比例趋势。为了解决原始图像中像素级信息丢失问题,本文方法希望使用像素级视差来进行优化。普通的像素级视差估计问题存在着过度平滑的问题,所以本文使用了实例级视差估计作为实现亚像素匹配的信息。

表 3 是在 Kitti 测试集上进行的测试,将 0.7 作为 IoU 阈值,并与先前的立体方法进行了比较。与 Stereo RCNN 相比,本文的方法在所有指标上均实现了提升。具体而言,与最新的 OC-Stereo 方法相比,本文的方法在简单水平的鸟瞰图平均精度上提高了 7.59%,在简单和中等的 3D 框平均精度上分别提升了约 2.92% 和

表 1 利用 Kitti 验证集评估的鸟瞰图和 3D 框的平均精度比较(IoU 阈值为 0.5)

Table 1 Comparison of the average precision of the bird′s-eye view and the 3D box evaluated using the Kitti validation set (IoU threshold 0.5)

方法	传感器	$AP_{bev}^{0.5}$/%			$AP_{3d}^{0.5}$/%			时间/s
		简单	中等	困难	简单	中等	困难	
Mono3D[13]	Mono	30.50	22.39	19.16	25.19	18.20	15.52	0.100
MF3D[14]	Mono	55.02	36.73	31.27	47.88	29.48	26.44	0.350
RTM3D[15]	Mono	56.90	44.69	41.75	52.59	40.96	34.95	0.055
VeloFCN[16]	LiDAR	79.68	63.82	**62.80**	67.92	57.57	52.56	0.030
Stereo-RCNN[5]	Stereo	87.13	74.11	58.93	85.84	66.28	57.24	0.410
本文方法	Stereo	**90.25**	78.77	62.31	89.81	72.46	64.62	0.320

表 2 利用 Kitti 验证集评估的鸟瞰图和 3D 框的平均精度比较(IoU 阈值为 0.7)

Table 2 Comparison of the average precision of the bird′s-eye view and the 3D box evaluated using the Kitti validation set (IoU threshold 0.7)

方法	传感器	$AP_{bev}^{0.7}$/%			$AP_{3d}^{0.7}$/%			时间/s
		简单	中等	困难	简单	中等	困难	
Mono3D[13]	Mono	5.22	5.19	4.13	2.53	2.31	2.31	0.100
MF3D[14]	Mono	22.03	13.63	11.60	10.53	5.69	5.39	0.350
RTM3D[15]	Mono	24.74	22.03	18.05	19.47	16.29	15.57	0.055
VeloFCN[16]	LiDAR	40.14	32.08	30.47	15.20	13.66	15.98	0.030
Stereo-RCNN[5]	Stereo	68.50	48.30	41.47	54.11	36.69	31.07	0.410
本文方法	Stereo	**71.16**	**54.26**	**47.15**	**57.92**	**43.24**	**37.09**	0.320

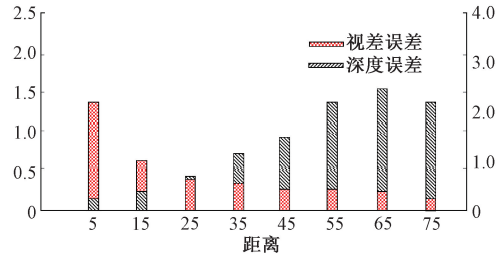


图 8 视差误差(pixel),深度误差(m)与物体距离(m)的关系

Fig. 8 The relationship among parallax error (pixel), depth error (m) and object distance (m)

2.68%。从表 3 中可以观察到本文的方法几乎超过了所有先前的立体方法。具体来说,本文方法用于 2D 检测和分割的时间为 0.11 s,用于关键点生成网络的时间为 0.08 s,用于 3D 边界框回归和中心优化的时间为 0.13 s。

本文方法的可视化结果如图 9 所示。

2.2 NuScenes 上的 3D 目标检测实验

NuScenes 数据集是最新的大规模自动驾驶数据集。为了加大数据集的挑战性,它收集了来自波士顿和新加坡的 1 000 个驾驶场景。与 Kitti 数据集相比,NuScenes 数据集通过 6 个多视图相机集 32 线 Lidar 收集数据,并提供了 7 倍多的对象注释。该数据集包含 28 130 个训练

表 3 利用 Kitti 测试集评估的 3D 对象检测结果及运行时间对比

Table 3 Comparison of 3D object detection results and running time evaluated using Kitti test set

方法	$AP_{bev}^{0.7}$/%			$AP_{3d}^{0.7}$/%			时间/s
	简单	中等	困难	简单	中等	困难	
Stereo RCNN[5]	61.67	43.87	36.44	49.23	34.05	28.39	0.41
PL;AVOD[17]	66.83	47.20	40.30	55.40	37.17	31.37	0.40
PL;F-PointNet[17]	72.80	51.80	44.00	59.40	39.80	**33.50**	0.67
OC-Stereo[18]	66.97	54.16	**46.70**	55.11	38.80	31.86	0.35
本文方法	**74.56**	**58.31**	46.24	**58.02**	**41.48**	32.53	**0.32**

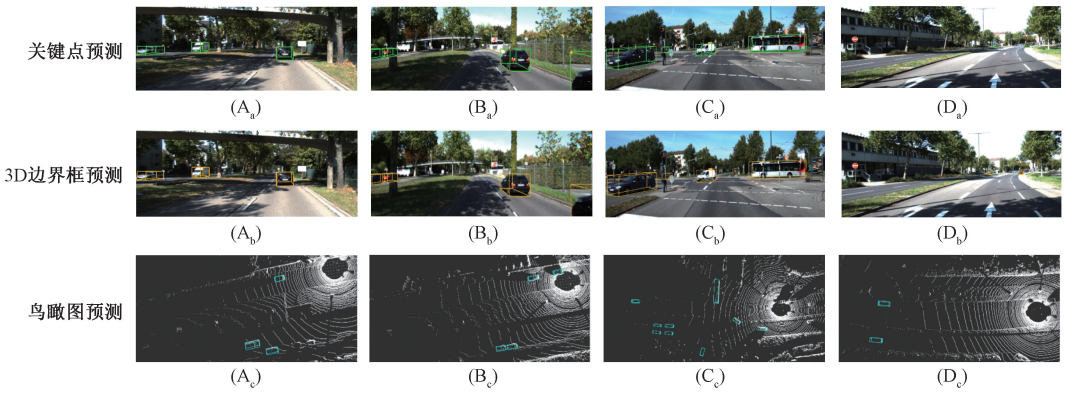


图 9　可视化结果

Fig. 9　Visualize the results

表 4　NuScenes 测试集上的 3D 检测 mAP

Table 4　3D detection mAP on the NuScenes test set

方法	汽车	公交	厢型车	mAP/%	NDS/%	时间/s
SECOND[19]	69. 16	34. 87	23. 73	26. 32	35. 31	0. 050
SARPNET[20]	59. 90	19. 40	18. 40	31. 66	49. 75	0. 070
PointPillars[21]	68. 47	28. 26	23. 42	30. 55	45. 37	0. 016
3D-CVF[22]	79. 69	54. 96	37. 94	42. 17	49. 78	0. 075
本文方法	**79. 54**	**55. 68**	**39. 54**	**44. 70**	**51. 26**	0. 320

样本和 6 019 个验证样本。与 Kitti 数据集不同,NuScenes 数据集一般使用 Nuscenes 检测分数(NuScenes detection score, NDS)作为衡量指标。

通过在 NuScenes 数据集测试了 FGAS RCNN 以验证本文方法的泛化性。表 4 提供了通过本文方法实现的关于车辆类别中特定 3 类的检测,平均精度(mean average precision, mAP)和 NDS。

相比于最新的 3D-CVF[22],本文方法在 mAP 和 NDS 分别提升了大约 2. 53% 和 1. 48%。并且在有关车辆类别的检测上,本文方法显著优于其他方法。

2. 3　消融实验

GS RPN 本文提出了以引导性的稀疏锚点来进行区域提案的方法。为了直观的了解模块性能,本文以每个图像 300 个样本的中等范围平均召回率(average recall, AR)和 2D 检测平均精度对 FGAS-RCNN,Stereo-RCNN 和 Faster-RCNN 进行比较。该测试均使用相同的骨干网和左右特征融合策略。

在表 5 中,本文使用了相同的骨干网络,超参数和增强方法,并展示了不同方法在经过 NMS 处理后的比较数据。从表内数据可以看出使用 GS RPN 在减少时间消耗的同时保持了较高的召回率。相比于之前的方法,本文方法在 2D 平均精度上全方位提升大约 1. 34%。

表 5　300 个样本的平均召回率(AR_{300}, %)和 2D 检测平均精度(AP_{2d}, %)比较

Table 5　Comparison of average recall rate of 300 samples (AR_{300}, %) and average precision of 2D detection (AP_{2d}, %)　%

方法	AR_{300}			AP_{2d}								
				左			右			立体		
	左	右	立体	简单	中等	困难	简单	中等	困难	简单	中等	困难
Faster RCNN[1]	86. 08	–	–	98. 57	89. 01	71. 54	–	–	–	–	–	–
Stereo RCNN[5]	85. 50	85. 56	74. 60	98. 73	88. 48	71. 26	98. 71	88. 50	71. 28	98. 53	88. 27	71. 14
本文方法	87. 30	87. 33	75. 81	99. 16	89. 27	72. 53	99. 32	88. 69	72. 51	98. 87	88. 37	72. 13

位置阈值 ϵ_L 能控制锚点分布的稀疏性。本文选择通过改变位置阈值的方式来比较每个图像生成锚点的平均数量,平均召回率和每秒帧率(frame per second, FPS),其结果如表 6 所示。

本文还研究使用本文的方法生成的提案的 IoU 分布情况,并与传统 RPN 方法进行了对比。如图 10 所示,通过观察 RPN,Stereo-RPN 和 GS-RPN 生成提案的 IoU 分布情况,可以明显观察到 GS-RPN 提供的高 IoU 提案数量更多。

表 6 不同位置阈值 ϵ_L 的实验结果

Table 6 Experimental results of different location threshold ϵ_L

位置阈值	锚点/图片	AR_{100}/%	AR_{300}/%	AR_{1000}/%	FPS/Hz
0	75 583(100.00%)	59.2	65.2	68.5	7.8
0.01	20 283(26.84%)	59.2	65.2	68.4	8.2
0.05	4 662(6.16%)	59.1	64.8	68.2	8.4
0.10	2 247(2.97%)	58.9	64.6	67.5	8.4

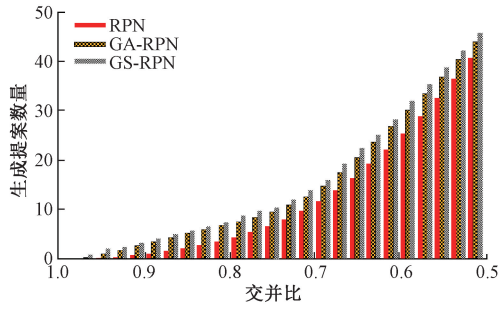


图 10 RPN,Stereo-RPN,GS-RPN 的 IoU 分布

Fig. 10 IoU distribution for RPN, Stereo-RPN, and GS-RPN

从表 7 的结果可以观察到,相比于 Stereo-RPN,GS-RPN 有着更多高 IoU 的积极提案,这使得本文方法有着更高的平均精度。并且由于本文方法可以在提案较少的情况下依旧保持较高的召回率,所以本文可以在仅对 300 提案进行训练后依旧可以提高最终的 mAP。

图 11 中的曲线展示了本文方法与其他检测方法在 ROI 分类准确性上的比较情况,从图中可以看出本文方法在提升检测效率的同时依旧可以保持较高的性能。

关键点检测网络 本文提出关键点生成方法可以提供 18 个约束以校正 3D 框。为了验证这种方法的好处,本文评估了不使用关键点约束的粗略 3D 框性能及使用了关键点网络校正后的 3D 框性能。并且还加入了利用回归视角和 2D 框信息生成关键点约束的方法以进行比较。

如表 8 所示,由于关键点检测网络可以提供 2D 框以外的大量像素级约束,所以本文提出的方法显著优于其他关键点检测方法,其中当 IoU = 0.7 时,简单和困难难度下分别提升了 7.14% 和 8.54%。

表 7 不同提案和 IoU 阈值的实验结果

Table 7 Experimental results of different proposals and IoU threshold

方案	数量	IoU_{thr}	AP/%	AP_{50}/%	AP_{75}/%
Stereo-RPN[5]	1 000	0.5	36.7	58.7	39.4
	1 000	0.6	37.4	57.2	40.5
	300	0.5	36.3	57.6	39.0
	300	0.6	37.2	56.2	39.7
GS-RPN	1 000	0.5	37.7	60.1	39.8
	1 000	0.6	39.1	58.9	42.5
	300	0.5	37.9	59.7	40.6
	300	0.6	39.8	59.2	43.5

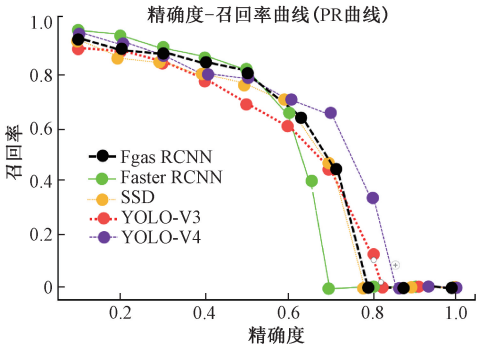


图 11 ROI 分类的 PR 曲线

Fig. 11 PR curve of ROI classification

表 8 在 Kitti 验证集上对有无关键点约束的 3D 检测评估对比

Table 8 Comparison of 3D detection evaluation with or without key point constraints on the Kitti validation set %

衡量标准	w/o Key-point			w/Key-point			w/Key-pointNet		
	简单	中等	困难	简单	中等	困难	简单	中等	困难
$AP_{3d}^{0.5}$	85.21	65.23	55.75	85.84	66.28	57.24	87.47	68.86	58.46
$AP_{3d}^{0.7}$	46.58	30.29	25.07	54.11	36.69	31.07	58.34	37.43	33.61

值得注意的是本文方法在单幅图像上具有更高的召回率和检测精度,并且可以同时在左右图像中产生高质量提案而无需增加额外计算。传统 RPN 是基于滑动窗口的方法,而 GS-RPN 是基于引导锚的方法。如图 12 所示,相比于 RPN,GS-RPN 的锚框更集中在前景目标上。

2.4 实际平台测试

为了验证提出方法的有效性,本文还在实际车载平台上进行了真实测试。如图 13 所示,该平台由多传感器构成。

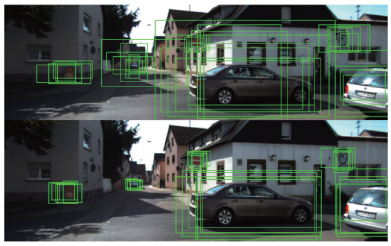


图 12　RPN 提案(上)与 GS-RPN 提案(下)

Fig. 12　RPN proposal (top) and GS-RPN proposal (bottom)

图 13　车载实验平台

Fig. 13　Vehicle-mounted experimental platform

除了基础的 16 线雷达以外,该平台还加入了额外 Tele-15 雷达和毫米波雷达。毫米波雷达主要负责前方车辆的速度和距离,而 Tele-15 雷达作为其额外的安全冗余增加车辆的安全性;16 线雷达负责短距离,大角度,低功率的探测任务。

为了更直观的让读者判断模型的优劣,本文在加入对比模型时考虑了传感器的因素。从表 9 中可以看出,本文方法在车辆类别的检测精度上均高于其他对比模型,分别达到了 86.25%,79.57% 和 73.60%,平均精度也达到了 79.81%。

表 9　针对汽车,厢型车,公交车的 3D 对象检测性能比较

Table 9　Performance comparison on 3D object detection for car, van and bus　%

方法	传感器	AP_{car}	AP_{bus}	AP_{van}	mAP
Stereo RCNN[5]	Stereo	72.78	69.38	61.70	67.95
PointPillars[21]	Mono+Lidar	82.67	77.81	68.83	76.44
Point-RCNN[23]	Lidar	84.76	78.47	70.14	77.79
F-Pointnet[7]	Lidar	81.48	74.54	69.52	75.18
本文方法	Stereo	86.25	79.57	73.60	79.81

在车载平台上的实验中,本文通过 3 个方面来验证本文方法在鸟瞰目标检测和 3D 目标检测中的性能。图 14 比较了在不同的 3D IoU 标准和不同距离下 FGAS RCNN 与 Stereo RCNN 的 3D 检测平均精度 AP_{3d}。图 14 由实际平台采集的真实数据构建,从图 14 中可以观察到本文提出的方法在相同距离和相同 3D IoU 的情况下性能均高于 Stereo RCNN。其中,在距离为 15 m,IoU=0.65 时,性能提升为 10.17%。本文方法相比于其他最新方法能够输出精确度较高的高质量预测。随着物体距离的增大,误报数量更多,但是相比于小型模型依据具有更少误报。在中远距离情况下,本文提出的方法可以达到 68.3% 的平均精度。

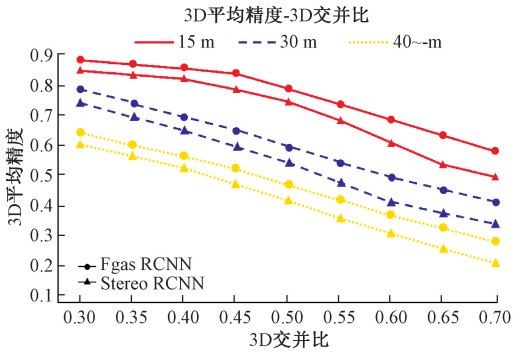


图 14　不同标准下的平均精度

Fig. 14　Average precision under different standards

本文依托车载平台进行了真实场景下的测试及数据采集任务,并将点云数据输入到模型中进行了实时 3D 目标检测,其可视化结果如图 15 所示。

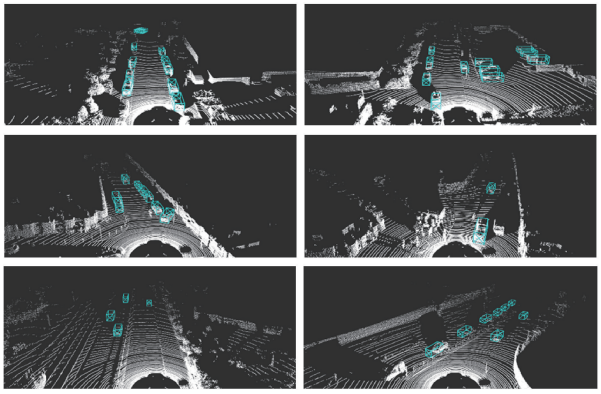


图 15　可视化检测结果

Fig. 15　Visual inspection results

3 结 论

本文提出了一种基于 Stereo RCNN 的锚引 3D 目标检测算法。本文方法充分利用输入图像的语义信息来指导稀疏锚点的生成,并通过预测锚点位置及其锚框形状生成非均匀锚点。通过关键点生成网络结合立体回归器生成 3D 边界框,随后通过使用实例视差和实例掩模计算特定区域内的逐像素深度来优化 3D 边界框精度。在公开数据集上的实验表明,本文方法在保持同类算法高精度的同时提升了计算效率。此外,通过在不同的数据集上的实验证明了本文方法的泛化性和可移植性,并且在不同环境下都具有较好的鲁棒性。

参考文献

[1] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39(6): 1137-1149.

[2] HE K, GKIOXARI G, DOLLÁR P, et al. Mask r-cnn [C]. Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.

[3] 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.

[4] 刘士兴,周启航,马登科,等. 基于单目视觉的电梯曳引轮磨损检测系统研制[J]. 电子测量与仪器学报, 2020,34(9):55-61.
LIU SH X, ZH Q H, MA D K, et al. Development of elevator traction sheave wear detection system based on monocular vision [J]. Journal of Electronic Measurement and Instrumentation, 2020, 34(9):55-61.

[5] LI P, CHEN X, SHEN S. Stereo r-cnn based 3d object detection for autonomous driving [C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 7644-7652.

[6] 董方新, 蔡军, 解杨敏. 立体视觉和三维激光系统的联合标定方法[J]. 仪器仪表学报, 2017, 38(10): 2589-2596.
DONG F X, CAI J, XIE Y M. Joint calibration method of stereo vision and 3D laser system [J]. Chinese Journal of Scientific Instrument,2017, 38(10): 2589-2596.

[7] QI C R, LIU W, WU C, et al. Frustum pointnets for 3d object detection from rgb-d data [C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 918-927.

[8] 郑少武, 李巍华, 胡坚耀. 基于激光点云与图像信息融合的交通环境车辆检测 [J]. 仪器仪表学报, 2019, 40(12):143-151.
ZHENG SH W, LI W H, HU J Y. Vehicle detection in traffic environment based on laser point cloud and image information fusion [J]. Chinese Journal of Scientific Instrument, 2019, 40(12):143-151.

[9] YANG B, LUO W, URTASUN R. Pixor: Real-time 3d object detection from point clouds [C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 7652-7660.

[10] 叶一飞,王建中. 基于点云的复杂环境下楼梯区域识别[J]. 电子测量与仪器学报, 2020, 34(4): 124-133.
YE Y F, WANG J ZH. Stair area recognition in complex environment based on point cloud [J]. Journal of Electronic Measurement and Instrumentation, 2020, 34(4):124-133.

[11] 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.

[12] CAESR H, BANKITI V, LANG A H, et al. Nuscenes: A multimodal dataset for autonomous driving [C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 11621-11631.

[13] YAN C, SALMAN E. Mono3D: Open source cell library for monolithic 3-D integrated circuits [J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2017, 65(3): 1075-1085.

[14] TUNG F, LITTLE J J. MF3D: Model-free 3D semantic scene parsing[C]. 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017:

4596-4603.

[15] LI P, ZHAO H, LIU P, et al. RTM3D: Real-time monocular 3D detection from object keypoints for autonomous driving[J]. ArXiv Preprint, 2020, arXiv: 2001,03343.

[16] LI B, ZHANG T, XIA T. Vehicle detection from 3d lidar using fully convolutional network[J]. ArXiv Preprint, 2016, arXiv:1608.07916.

[17] WANG Y, CHAO W L, GARG D, et al. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 8445-8453.

[18] PON A D, KU J, LI C, et al. Object-centric stereo matching for 3d object detection[C]. 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020: 8383-8389.

[19] YAN Y, MAO Y, LI B. Second: Sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[20] YE Y, CHEN H, ZHANG C, et al. Sarpnet: Shape attention regional proposal network for lidar-based 3d object detection[J]. Neurocomputing, 2020, 379: 53-63.

[21] 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.

[22] YOO J H, KIM Y, KIM J S, et al. 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection[J]. ArXiv Preprint, 2020, arXiv:2004,12636.

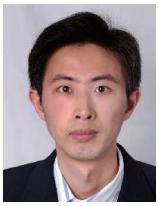
[23] SHI S, WANG X, LI H. Pointrcnn: 3d object proposal generation and detection from point cloud[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 770-779.

作者简介

曹杰程,2019 年于常州工学院获得学士学位,现为苏州科技大学电子与信息工程学院硕士研究生,主要研究方向为目标检测和机器视觉。

E-mail: caojc9527@163.com

Cao Jiecheng received his B.Sc. degree from Changzhou Institute of Technology in 2019. He is currently a master student in the College of Electrical and Information Engineering at Suzhou University of Science and Technology. His main research interests include object detection and instance segmentation.



陶重犇(通信作者),2014 年于江南大学获得博士学位,现为清华大学苏州汽车研究院博士后,主要研究方向三维语义建图和自主导航。

E-mail: chongbentao@usts.edu.cn

Tao Chongben (Corresponding author) received his Ph.D. degree from Jiangnan University in 2014. He is currently a postdoctoral fellow in the Suzhou Automotive Research Institute at Tsinghua University. His main research interests include 3D semantic mapping and autonomous navigation.