

MonoSPDC: 一种基于全局-通道注意力优化的单目 3D 目标检测

苏卫星¹, 郭增新², 刘芳¹, 王浩宇¹

(1. 天津工业大学 天津市自主智能技术与系统重点实验室, 天津 300387; 2. 天津工业大学 人工智能学院, 天津 300387)

摘要: 针对单目 3D 目标检测算法在自动驾驶场景中面临卷积结构网络特征表达能力有限、Transformer 结构的模型计算复杂度高、激光雷达依赖性较强等关键挑战, 提出了一种基于全局-通道注意力优化的单目 3D 目标检测算法——MonoSPDC。首先, 设计全局-通道特征重组模块, 通过融合全局空间关联性与通道维度注意力机制, 增强 HeatMap 的特征表征能力; 其次, 构建多角度联合预测结构, 将三维角点几何约束与深度回归结果动态融合, 有效降低深度估计误差。在 KITTI 数据集上的实验表明, 该方法在三维平均精度 (AP_{3D}) 指标达到 24.87%, 较 MonoDTR 和 MonoDETR 分别提升 13% 和 5%, 检测速度提高 16% 以上。在 Waymo 数据集上, AP_{3D} 达到 67.55%, 较 CaDDN 和 MonoCon 分别提升 5% 和 10%。此外, MonoSPDC 无需依赖激光雷达辅助训练, 降低了模型复杂性与数据需求, 为低成本自动驾驶系统的三维感知提供了高精度、轻量化的解决方案。

关键词: 3D 目标检测; 深度学习; 注意力机制; 自动驾驶; 上下文感知

中图分类号: TP391.4 **文献标志码:** A **文章编号:** 1671-5497(2026)06-1706-12

DOI: 10.13229/j.cnki.jdxbgxb.20241284

MonoSPDC: a novel monocular 3D object detection using global-channel attention optimization

SU Wei-xing¹, GUO Zeng-xin², LIU Fang¹, WANG Hao-yu¹

(1. The Tianjin Key Laboratory of Autonomous Intelligence Technology and Systems, Tiangong University, Tianjin 300387, China; 2. School of Artificial Intelligence, Tiangong University, Tianjin 300387, China)

Abstract: Monocular 3D object detection in autonomous driving scenarios faces critical challenges, including limited feature representation capabilities of convolutional networks, high computational complexity of Transformer-based models, and heavy reliance on LiDAR assistance. To address these issues, this paper proposes a novel monocular 3D object detection algorithm named MonoSPDC, which

收稿日期: 2024-11-29.

基金项目: 国家重点研发计划项目 (2021YFB2501800).

作者简介: 苏卫星 (1980-), 男, 教授, 博士. 研究方向: 制造物联与智能工厂技术, 工业大数据, 智能网联与自动驾驶, 人工智能与边缘计算. E-mail: suweixing@tjpu.com

通信作者: 刘芳 (1983-), 女, 教授, 博士. 研究方向: 运载储能系统管理技术, 智能驾驶与智能辅助驾驶技术, 复杂系统建模与辨识, 数据分析与数据挖掘. E-mail: 15900201597@163.com

integrates global-channel attention optimization and multi-view joint prediction. First, a Global-Channel Feature Reconstruction Module is designed to enhance feature representation in heatmaps by fusing global spatial correlations with channel-wise attention mechanisms. Second, a Multi-View Joint Prediction Structure is developed to dynamically integrate 3D corner geometric constraints with depth regression results, significantly reducing depth estimation errors. Experimental results on the KITTI dataset demonstrate that the proposed method achieves a 3D average precision (AP_{3D}) of 24.87%, outperforming the state-of-the-art methods MonoDTR and MonoDETR by 13% and 5%, respectively, while improving detection speed by over 16%. On the Waymo dataset, it attains an AP_{3D} of 67.55%, surpassing CaDDN and MonoCon by 5% and 10%. Notably, MonoSPDC eliminates the dependency on LiDAR-assisted training, substantially reducing model complexity and data requirements, thereby offering a high-precision, lightweight, and cost-effective solution for 3D perception in autonomous driving systems.

Key words: 3D object detection; deep learning; attention mechanism; autonomous driving; context awareness

0 引言

近年来,随着人工智能技术的飞速发展,自动驾驶和移动机器人等领域对环境感知技术提出了越来越高的要求。在这些涉及复杂三维运动的场景中,环境感知已成为实现高级自动化功能的关键环节。目标检测,尤其是3D目标检测技术,因其能够提供更丰富的空间信息而变得尤为重要^[1]。

在单目3D目标检测的早期研究中,研究人员通常借鉴2D目标检测的经验。例如,Fcos3D^[2]和Deep-3D-box^[3]将2D检测任务中的回归参数直接修改为3D回归参数,但这种方法忽视了3D检测中的深度估计问题,导致精度较低。随后,研究者们尝试引入独立模块进行深度估计。Reading等^[4]提出了一种基于网格的深度估计方法,通过生成鸟瞰图(BEV)来实现基于图像的3D目标检测。尽管加入深度估计模块提升了精度,但也显著增加了参数量,导致检测速度明显下降。

随着Detection transformer^[5](DETR)研究的持续升温,DETR3D^[6]首次将Transformer结构引入到3D目标检测任务中。进一步发展后,MonoDTR^[7]在Transformer的基础上引入了深度-察觉模块和增强模块,将深度位置提示融入到Transformer中,并使用激光雷达进行辅助训练。尽管MonoDTR提升了检测精度,但其额外的深度辅助训练大幅增加了训练成本,导致实际应用中面临较大的成本挑战。在近期的研究中,MonoDGP^[8]通过引入透视不变的几何误差修正投影公式,并采用分离的深度引导解码器与2D解

码器结构,结合区域分割头优化输入令牌,以提升单目3D检测的精度。但其对几何误差机制的依赖可能限制了模型在复杂遮挡场景下的泛化能力,且分离式解码器和区域分割头的引入增加了计算复杂度,可能影响实时性。

针对上述问题,本文提出了一种名为MonoSPDC的单目3D目标检测算法。该算法采用无锚点(Anchor-Free)^[9]结构,仅需待检测目标的中心点深度值信息进行训练,从而减少了激光雷达辅助训练和标注的成本。MonoSPDC算法通过引入全局-通道特征重组模块(FGCR)和多角度联合预测结构(MVPI),在提升检测精度的同时,降低了计算和训练成本。实验结果表明,MonoSPDC在KITTI^[10]数据集和Waymo^[11]数据集上均取得了优异的性能。最后,在实验的结尾添加了实车部署的验证,测试结果基本能够达到自动驾驶场景的检测需求。本文的贡献总结如下:

(1)提出了一个单目3D目标检测算法-MonoSPDC。该算法能够弥补传统卷积结构算法精度不足的问题,以及基于Transformer结构算法的训练数据依赖和计算冗余等问题。在KITTI数据集和Waymo数据集上对本文算法进行了验证,实验结果表明,所提的MonoSPDC在计算精度与速度上均取得了较为优异的成绩。

(2)在本文MonoSPDC算法中,提出一种全局-通道特征重组模块(FGCR)。FGCR通过在通道和全局两个维度上实施注意力操作,提升了检测精度。

(3)在本文MonoSPDC算法中,设计了一种

多角度联合预测结构(MVPI)。该结构能够将全局的深度估计和基于角点的深度估计进行不确定性融合,有效提高深度估计的精度。

1 算法实现

1.1 总体网络结构

本文提出的MonoSPDC算法,采用单阶段无锚点检测框架^[9],包含4个核心组件(见图1):主干网络(Backbone)、全局-通道特征重组模块(FGCR)、多角度联合预测结构(MVPI)和回归头(Regression head),以实现高效的单目3D目标检测。

首先,本算法选用了Deep Layer Aggregation with 34 layers(DLA-34)^[12]作为特征提取的主干网络,并引入了Deformable Convolution v2(DC-Nv2)^[13]以增强其对空间变换的适应性。DLA-34是一种先进的主干网络架构,它通过深度层级聚合的方式构建。DLA-34的主要特点是它能够并行地聚合来自不同深度的特征,这有助于网络捕捉多尺度的特征,从而提高对图像中不同大小目标的检测能力。主干网络接受一张二维彩色图像 $I \in R^{3 \times H \times W}$ 作为输入,输出为原始图像1/4尺寸的高维特征图 $X \in R^{C \times H_i \times W_i}$,其中 $H_i = (1/4)H$ and $W_i = (1/4)W$, C 表示特征图的通道数。

其次,本文提出全局-通道特征重组模块(FGCR),该模块将主干网络生成的高维特征图

X 进行特征重组注意力优化,以得到特征信息更加精确的特征图 $F_{\text{rec}} \in R^{C \times H_i \times W_i}$,该部分将在1.2节被详细介绍。

进一步地,本文设计的多角度联合预测结构(MVPI)负责从优化后的特征图 F_{rec} 中估计深度信息。该结构首先独立地从 F_{rec} 中回归角点信息(3D框的8个顶点)和生成深度特征图,再分别通过角点信息和深度特征图估计出深度值 D_C 和 D_M 。最后,通过一种基于不确定性融合的策略,将 D_C 和 D_M 融合,得到最终的深度估计值 D_{final} ,MVPI将在1.3节被详细介绍。

最后,回归头承担着生成HeatMap、回归2D/3D信息(包括位置、偏航角、偏移量和目标框尺寸等)的任务。为了提高截断物体偏移量的回归精度和HeatMap的生成精度,引入了MonoFlex^[14]中的EdgeFusion模块。该模块能根据物体中心点的位置(物体内部或物体外部)采取不同的处理策略,从而增强边缘截断物体的检测精度,回归头部分将在1.4节中详细介绍。

本文中的技术路线图与章节之间的联系如图2所示。

1.2 全局-通道特征重组模块FGCR

本文所提出的FGCR模块如图3所示。该模块分为全局特征重组(FGR)和通道特征重组(ECR)两部分, X 表示主干网络输出的高维特征图。

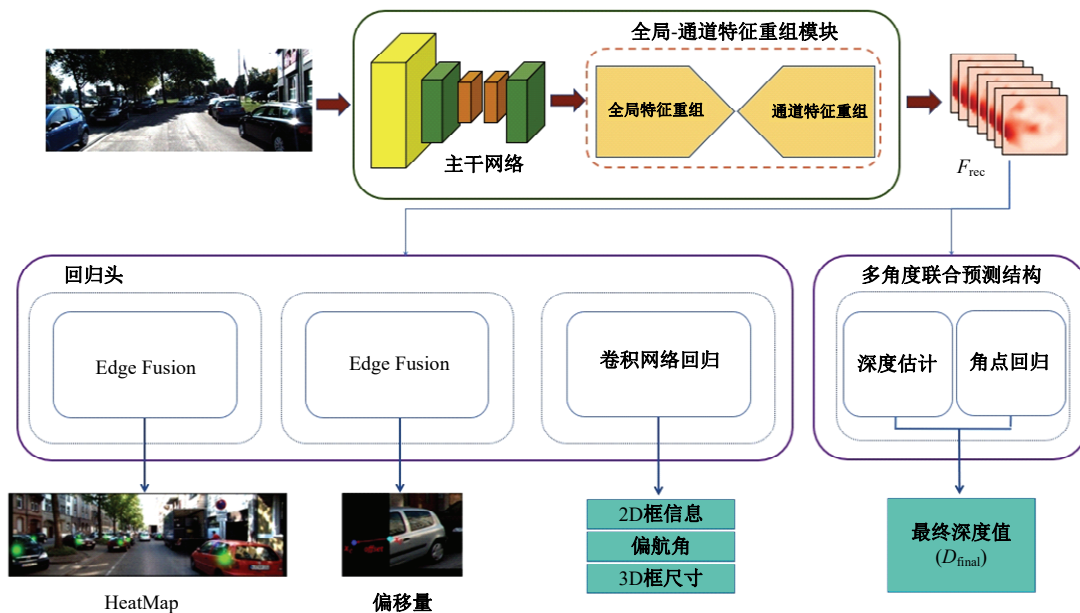


图1 MonoSPDC网络结构整体示意图,包括主干、所提出的FGCR、所提出的MVPI及回归头

Fig. 1 Framework of MonoSPDC, which consists of the backbone, the proposed FGCR, the proposed MVPI and the detection head

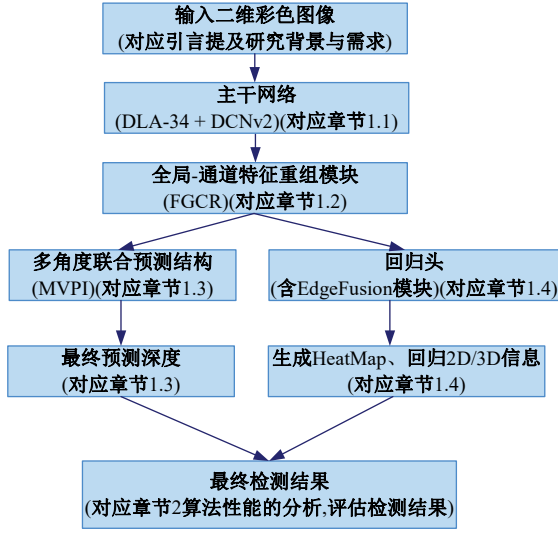


图 2 总体技术路线图

Fig. 2 Overall technology roadmap

1.2.1 全局特征重组 FGR

FGR结构如图3中左侧框内所示。该模块的主要功能是在全局范围内强化主干网络输出的高维特征图 X 的前景特征,同时抑制其背景特征。其基本操作过程如下:

(1)首先,确定 X 全局特征的关键性权重分布

FGR以backbone输出的高维特征图 $X \in R^{C \times H_i \times W_i}$ (C 为 X 的通道数)为输入,对 X 进行Group Normalization(GN)^[15]操作,公式如下:

$$F_{gn} = \text{Gamma} \frac{X - \mu}{\sqrt{\sigma^2 + \epsilon}} + \text{Bate} \quad (1)$$

式中: $F_{gn} \in R^{C \times H_i \times W_i}$ 为经过GN计算后的中间特征图; X 为主干网络的输出; μ 、 σ 分别为 X 的均值和方差; ϵ 是为了保障除法稳定性而加入的一个极小的正常数; $\text{Gamma} \in R^{C \times 1 \times 1}$, $\text{Bate} \in R^{C \times 1 \times 1}$ 分别为GN操作的缩放因子和偏移因子,Gamma和Bate都为可学习参数,它们根据全局特征的信息,学习

X 中每个特征点的重要性。

利用Gamma、Bate的这一特点可以确定 X 上每个特征点的关键性权重分布。具体计算过程如下:

$$W_{\text{gamma}} = \left\{ w_j^g = \frac{\text{Gamma}_j}{\sum_{j=1}^C \text{Gamma}_j}, i, j = 1, 2, \dots, C \right\} \quad (2)$$

$$W_{\text{bate}} = \left\{ w_i^b = \frac{\text{Bate}_i}{\sum_{j=1}^C \text{Bate}_j}, i, j = 1, 2, \dots, C \right\} \quad (3)$$

$$W_{\text{att}} = \text{Sigmoid}(F_{gn} * W_{\text{gamma}} + \partial W_{\text{bate}}) \quad (4)$$

式(2)(3)对Gamma、Bate进行归一化处理,以得到 $[0, 1]$ 之间的数值; $W_{\text{att}} \in R^{C \times H_i \times W_i}$ 为关键性权重分布,代表 X 上每个特征点的关键程度。

利用 W_{att} 可以判断 X 上每个特征点的前景(关键程度高)和背景(关键程度低)的概率,以实现注意力机制。相比Transformer中的Self-attention^[16]操作,该操作节约了大量的计算成本。 ∂ 为偏移因子 W_{bate} 的权重因子,用以控制偏移因子的重要程度,其同样为可学习参数。

(2)基于关键性权重分布 W_{bate} ,拆分 F_{gn} :

$$F_{\text{true}} = \begin{cases} F_{gn}, & W_{\text{att}} > \alpha \\ 0, & W_{\text{att}} < \alpha \end{cases} \quad (5)$$

$$F_{\text{false}} = \begin{cases} 0, & W_{\text{att}} > \alpha \\ F_{gn}, & W_{\text{att}} < \alpha \end{cases} \quad (6)$$

从式(5)(6)可以看出,设一个和 F_{gn} 相同形状的空特征图 F_{true} ,对于 F_{gn} 的每一个特征点而言,当其 W_{att} 大于阈值 α 时(α 建议为0.5),取 F_{gn} 的特征点原始值赋于 $F_{\text{true}} \in R^{C \times H_i \times W_i}$ 的对应位置;当 W_{att}

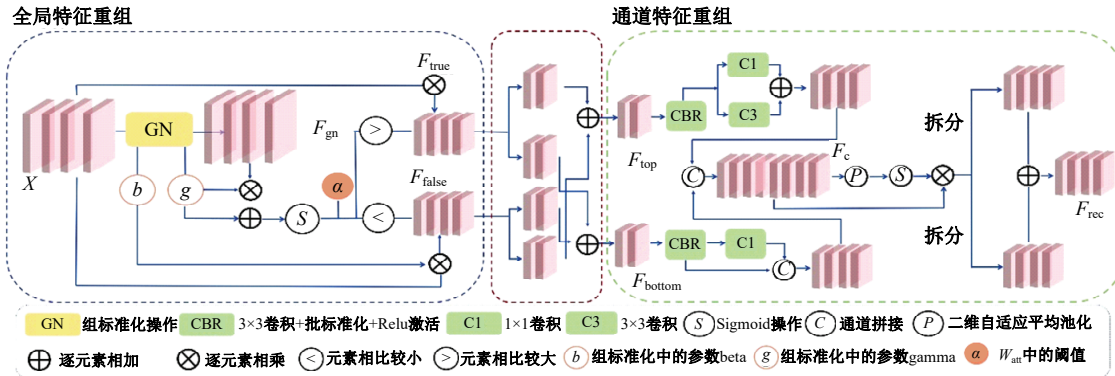


图 3 FCGR 整体结构,

Fig. 3 Overall structure of FCGR module

小于阈值 α 时,取 0 值赋于 F_{true} 的对应位置。至此,得到了注意力强化的前景特征图 $F_{\text{true} \circ}$

类似 F_{true} ,式(6)用于得到代表 F_{g_s} 不关键特

$$\begin{cases} F_{\text{true}}^{\text{before}} = F_{\text{true}}, & i = 1 - 0.5C, j = 1 - H_i, k = 1 - W_i \\ F_{\text{true}}^{\text{after}} = F_{\text{true}}, & i = 0.5C + 1, \dots, C, j = 1 - H_i, k = 1 - W_i \\ F_{\text{false}}^{\text{before}} = F_{\text{false}}, & i = 1 - 0.5C, j = 1 - H_i, k = 1 - W_i \\ F_{\text{false}}^{\text{after}} = F_{\text{false}}, & i = 0.5C + 1, \dots, C, j = 1 - H_i, k = 1 - W_i \end{cases} \quad (7)$$

式中, $F_{\text{true}}^{\text{before}} \in R^{C \times H_i \times W_i}$ 、 $F_{\text{true}}^{\text{after}} \in R^{C \times H_i \times W_i}$ 、 $F_{\text{false}}^{\text{before}} \in R^{C \times H_i \times W_i}$ 、 $F_{\text{false}}^{\text{after}} \in R^{C \times H_i \times W_i}$ 为 FGR 模块最终输出的 4 个特征图。

在 FGR 中,利用关键性权重分布来区分前景和背景特征,通过将背景特征的权重设置为零,有效地忽略背景差异,仅对特征中心点给予高度关注,降低背景的干扰,显著提高了 HeatMap 的生成准确度。

1.2.2 通道特征重组 ECR

在 FGR 的输出进入 ECR 前,如图 3 中间框内所示,FGR 模块的输出 $F_{\text{true}}^{\text{before}}$ 、 $F_{\text{false}}^{\text{before}}$ 、 $F_{\text{true}}^{\text{after}}$ 以及 $F_{\text{false}}^{\text{after}}$ 首先进行错位相加操作。得到了两个特征图 F_{top} 、 F_{bottom} 。 F_{top} 和 F_{bottom} 为 ECR 模块的输入。这种将增强后的前景特征和弱化后的背景特征进行错位相加的方式,能够将误检的前景或背景进行融合,有效地增强检测的鲁棒性。

ECR 模块如图 3 中右侧框内所示。该部分在通道方向上进一步强化高维特征图 X 的前景特征,以生成特征更加分明的特征图 $F_{\text{rec} \circ}$ 。 F_{rec} 将被用于后续的 MVPI 模块以及回归头模块,以实现最终目标检测精度的提高。其基本操作过程如下:

(1) F_{top} 、 F_{bottom} 的通道混合编码-解码过程

当 F_{top} 、 F_{bottom} 进入 ECR 后,分别经过不共享的 CBR (Conv 3X3+BN+Relu) 模块,对 F_{top} 、 F_{bottom} 进行特征优化操作。这样可以得到两组分辨率不变、通道数按比例减少的通道编码特征图。

对于 F_{top} 而言,其通道编码特征图分别通过 C1 和 C3 两个模块进行解码操作后,再将解码结果相加,以得到 F_{top} 的编码-解码操作结果。通过不同尺寸卷积核解码的方式能够有效地将通道特征和全局特征进行关联。相加操作能够在通道维度上获得更加稳定的通道重组效果。

对于 F_{bottom} 而言,其通道编码特征图则仅通过 C1 模块,单纯地实现通道维度的解码。将通过 C1 的解码结果和 F_{bottom} 的通道编码特征图在通道

维的特征图 $F_{\text{false}} \in R^{C \times H_i \times W_i}$ 。

进一步,将 F_{true} 、 F_{false} 在通道维度上被平均分成前后两部分,具体公式如下:

维度上进行拼接,以得到 F_{bottom} 的编码-解码操作结果。

最后,将上述 F_{top} 和 F_{bottom} 的编码-解码结果在通道维度上进行拼接,以得到通道解码特征图 $F_c \in R^{C_j \times H_i \times W_i}$, $C_j = 2C$ 。通过拼接的方式,无损地获得了两种解码方式下的特征图 F_c 。

(2) F_c 的通道注意力机制操作过程

首先,将上述拼接的通道解码特征图 F_c 进行 Pooling-Softmax 操作以得到 F_c 的通道重要性权重分布 $W_c \in R^{C_j \times H_i \times W_i}$,公式如下:

$$W_c = \text{softmax}(\text{Pooling}(F_c)) \quad (8)$$

式(8)中,平均池化操作可以在每个通道维度上,计算该特征图的数值大小。 softmax 操作可以进一步将这一数值转化为概率分布,以得到 F_c 的通道重要性权重分布 W_c 。上述过程即实现了基于注意力机制的 F_c 在通道维度上的重要性评判。其次,依据 F_c 的通道重要性权重分布 W_c ,对其实施注意力机制操作,公式如下:

$$F_{\text{att}} = F_c \bullet W_c \quad (9)$$

式中: $F_{\text{att}} \in R^{C_j \times H_i \times W_i}$ 为 F_c 通道维度上注意力操作后的特征图。

最后, F_{att} 在通道维度上进行 split-add 操作,即得到 FGCR 模块最终输出的具有更加分明特征的特征图 $F_{\text{rec} \circ}$ 。

在 FGR 中,利用通道重要性权重分布 W_c 在通道维度上执行注意力优化,以加强通道方向上的特征信息,并为后续将通道特征转换为深度特征提供基础。

综上,主干网络输出的高维特征图 X 在经过 FGCR 模块的两个子模块(FGR、ECR)操作后所得到的 F_{rec} ,即为全局-通道特征重组特征图。

后续,将 F_{rec} 同步分发给 MVPI 以及回归头两个模块做进一步处理。

1.3 多角度联合预测结构 MVPI

如前所述,MVPI 模块将 FGCR 的输出 F_{rec} 进行两个独立的操作:①通过回归的方式得到角点信

息,再利用角点信息计算出深度估计值 D_C ;②通过特征转换的方式得到深度特征图,再利用深度特征图回归深度值 D_M 。在得到以上两个深度估计值后,MVPI模块采用基于不确定性融合的方法以获得最终的深度估计值 D_{final} (见图 4)。

1.3.1 基于角点信息的深度值 D_C

本文借鉴了 Monoflex 的操作,首先,把 F_{rec} 中的每个特征点认定为一个目标的中心点,进而利用特征点通道维度的特征值回归该目标的角点值。最后,通过几何运算得到目标基于角点信息的深度估计值 D_C 。

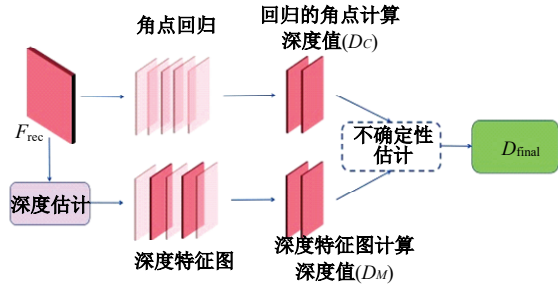


图 4 MVPI 的整体结构

Fig. 4 Overall structure of MVPI

1.3.2 基于深度特征图的深度值 D_M

类似于 LSS^[17] 的思想,将图像的深度特征在通道维度上进行表示。LSS 为一张特征图的每一个特征点生成所有可能深度的概率,即将通道维度上每个数值表示为某一深度区间的概率值。这样做可以将深度估计转换为分类问题,提高深度估计鲁棒性,在一些深度值具有歧义性的像素点上表现更佳。但这样的做法会生成一个巨大的视锥点云,导致大量的冗余数据,这些冗余数据占用大量内存资源、降低计算效率。

与 LSS 不同的是,本文对整张特征图(F_{rec})只生成一个深度概率分布,并将其命名为深度权重分布。这种做法能够保留算法鲁棒性的同时,减少大量的冗余计算以及内存资源的有效占用,具体过程如下:

(1) F_{rec} 的深度注意力机制操作过程

F_{rec} 的深度注意力机制的网络结构如图 3 上半侧框内所示。其中,深度权重分布生成的完整过程如式(10)所示。

$$W_{\text{depth}} = \text{softmax} \left(C_1 \left(CBR_1 \left(\text{pool} \left(CBR_3(F_{\text{rec}}) \right) \right) \right) \right) \quad (10)$$

式中: F_{rec} 为 FGCR 的输出。

从式(10)中可以看出,首先对 F_{rec} 进行了 CBR_3 操作。这一过程将 F_{rec} 在通道维度上进行深度信息的映射,以得到深度编码特征图。这种方式使 F_{rec} 每个特征点在通道维度上的数值可以代表深度值分布的概率。

其次,如式(10)所示,在深度编码特征图上进一步进行了全局平均池化——CBR_1-C1 操作。其中,全局平均池化操作将每个深度范围内的所有特征融合成 1×1 的大小;CBR_1-C1 操作使深度特征加强。

最后,进行 softmax 操作,以生成深度权重分布 $W_{\text{depth}} \in R^{D \times 1 \times 1}$, D 为深度值的范围。注意力机制下的通道深度特征图 F_{CD} 计算如下:

$$F_{\text{CD}} = F_{\text{rec}} \cdot W_{\text{depth}} \quad (11)$$

式中: F_{CD} 代表了深度层面存在目标的情况,但通道深度特征图 F_{CD} 只是从深度层面估计目标的存在性,进一步,还需对其进行全局定位与匹配。

(2) 目标的定位与匹配

进一步对 F_{CD} 进行全局特征点的匹配,以将深度值定位到某一位置。此过程具体结构如图 5 下半部分虚线框,可以看出,首先,将上半侧框内经过 CBR_1 操作后的中间特征图 F_{gad} 取出,并对其进行 C1 操作以改变特征图通道的长度来匹配 C3 的权重长度。需要说明的是,选择 F_{rec} 的深度注意力机制操作过程中生成的中间特征图 F_{gad} 的原因在于,其含深度编码信息和未被完全同化的全局特征信息,这有利于获得全局特征和深度之间的关联,以达到定位的作用。

其次,将上述经过 C1 操作后的结果进行 view 操作,并将操作后的结果填充到 C3 操作的卷积核权重 W 中。

最后,使用被填充的 C3 对 F_{CD} 进行卷积操作,以得到全局定位后的深度特征图 D_M 。

1.3.3 不确定性融合

本算法在 1.3.1 基于角点信息的深度估计和 1.3.2 基于全局定位深度特征图 D_M 的深度估计的同时,分别建立了一张可学习不确定性特征图 $U_M \in R^{1 \times H_i \times W_i}$ 和 $U_C \in R^{1 \times H_i \times W_i}$ 。该特征图的数值为深度估计结果的可信度。

基于上述特点, U_M 与 U_C 进行基于可信度的融合操作,并确定最终深度估计结果 D_{final} 。具体过程如下:

$$U_{\text{sum}} = \text{softmax}(U_M) + \text{softmax}(U_C) \quad (12)$$

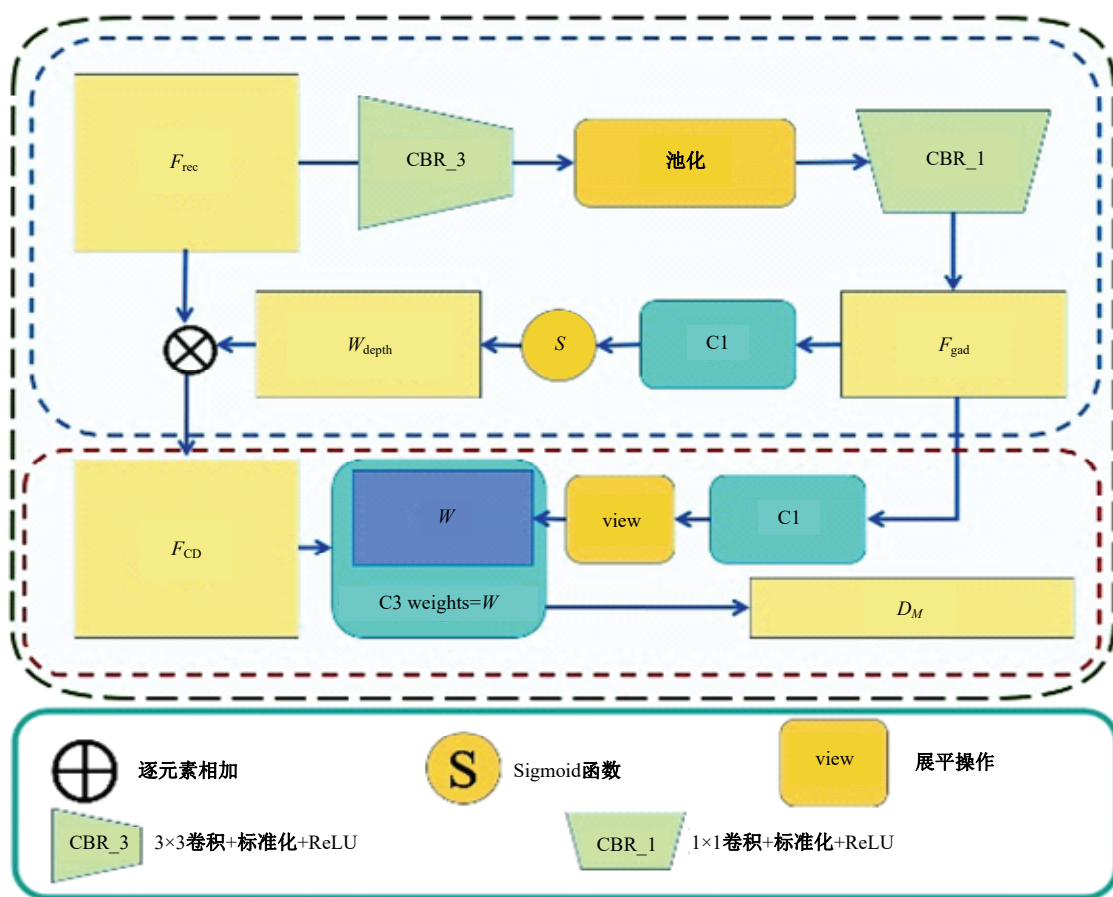


图 5 基于深度特征图的深度估计网络结构图

Fig. 5 Network structure diagram of depth estimation based on depth feature map

$$D_{\text{final}} = \frac{D_M \cdot \text{softmax}(U_M) + D_C \cdot \text{softmax}(U_C)}{U_{\text{sum}}} \quad (13)$$

式(12)及(13)中的 softmax 操作目的在于将 U_M 以及 U_C 统一到相同数量级。式(13)中的“ $\cdot$ ”以及“/”表示元素级的对应运算。上述过程实现将 D_M 和 D_C 在每个点上进行加权操作。这种加权操作能给置信度高的部分分配更多的权重,更具有鲁棒性。

1.4 回归头

对于结果的回归,本文采用了 MonoFlex 中的回归头结构和损失函数。简单来说,基于 FGCR 模块输出的全局-通道特征重组特征图 F_{rec} ,使用直接回归的方法得到 2D 目标框、偏航角、3D 框尺寸、HeatMap 以及 Offsets。其中 Offsets 指 2D 目标框中心与 3D 目标框中心的偏移量。在回归 HeatMap 和 Offsets 之前,使用 Edge Fusion 模块加强检测边缘物体。回归头和 Edge Fusion 模块在 MonoFlex 中能看到更多细节。

不同的是,本算法在目标深度估计方面,采用

1.3 节中介绍的 MVPI 实现。

2 算法性能分析

2.1 数据集以及实现方法

本文提出的算法将在 KITTI 3D Object Detection Benchmark^[10]和 Waymo 数据集上进行训练、验证和测试。KITTI 数据集共包括 7 481 个训练样本和 7 518 个测试样本。本文按照文献[18]中的方式,将 7 518 个测试样本分为 3 749 个测试样本和 3 769 个验证样本两部分。在 Waymo 数据集中,使用训练和验证集的一个子集来形成最终的训练集(3 000 个样本)和验证集(2 716 个样本),仅包括前置摄像头图像,并将图像缩放到 960×640 的大小。与此同时,本文对 MonoSPDC 也进行了消融实验以验证各个模块的有效性。

本文采取的评测指标为 $AP_{3D|R40}$ 和 AP_{BEV} ,并从物体边界框的高度情况、遮挡情况和截断情况三方面将检测目标分为简单、中等和困难 3 个等级来评估。检测的类别包括 Car, Pedestrian, 和 Cyclist 3 个类别。为了便于观察,本文将检测效

果通过鸟瞰图(BEV)进行展示。

本算法在单个 RTX 3090 GPU 上训练测试,优化器采用 AdamW^[19], epoch 设置为 100, batch 设置为 8, 初始学习率为 3e-4, 分别在 80epoch 和 90epoch 时, 学习率分别缩小 10 倍。

2.2 主要结果

表 1 为本文所提的 MonoSPDC 检测算法在 KITTI 测试集上对于 Car 类别的检测以及对比结果。采用 IoU=0.7 作为汽车类别的阈值。Improvement 表示与上述所有算法的最好的结果的比较结果。

从表 1 中可见, 本文在 KITTI 数据集的测试集上对 Car 类别使用多种不同优秀的算法进行全面的定量分析。MonoSPDC 在简单(Easy)、中等(Mod)和困难(Hard)任务测试中, AP_{3D} 和 AP_{BEV}

两项指标均优于目前主流算法, 这证明了 MonoSPDC 网络设计的有效性。本算法的检测速度位列第 4 位, 优于绝大多数算法的检测检测速度, 在检测速度上, MonoSPDC 与 SMOKE^[20]、MonoEF^[21]、MonoCon^[22]三个算法相比, 仅相差 1 ms、1 ms、5 ms, 而在检测精度方面, MonoSPDC 相比 SMOKE、MonoEF、MonoCon 分别提升了 10.84、3.58、2.37。

表 2 为本文所提的 MonoSPDC 在 KITTI 数据集的测试集上, 对于 Pedestrian 和 Cyclist 类别的检测以及对比结果。采用 IoU=0.5 作为 Pedestrian 和 Cyclist 类别的阈值, metric = R40。

从表 2 中可知, 在 KITTI 数据集的测试集, 对 Pedestrian 和 Cyclist 类别与多种不同的优秀算法进行了全面的定量比较。

表 1 本文 MonoSPDC 在 KITTI 的测试集上对于 Car 类别的检测比较结果

Table 1 Detection comparison results of the proposed MonoSPDC on KITTI' test set for the Car category

方法	时间/ms	AP_{3D} (Car)			AP_{BEV} (Car)		
		Easy	Mod	Hard	Easy	Mod	Hard
SMOKE ^[20]	30	14.03	9.76	7.84	20.83	14.49	12.75
PatchNet ^[23]	400	15.68	11.12	10.17	22.97	16.86	14.97
MonoDLE ^[24]	40	17.23	12.26	10.29	24.79	18.89	16.00
CaDDN ^[4]	630	19.17	13.41	11.46	27.94	18.91	17.19
MonoEF ^[21]	30	21.29	13.87	11.71	29.03	19.70	17.26
MonoFlex ^[14]	35	19.94	13.89	12.07	28.23	19.75	16.89
GPUNet ^[25]	34	20.11	14.20	11.77	—	—	—
MonoDTR ^[7]	37	21.99	15.39	12.73	28.59	20.38	17.14
MonoDETR ^[26]	43	23.65	15.92	12.99	32.08	21.44	17.85
MonoDistill ^[27]	40	22.97	16.03	13.60	31.87	22.59	19.72
MonoCon ^[22]	26	22.50	16.46	13.95	31.12	22.10	19.00
Ours	31	24.87	16.78	14.19	32.94	23.17	19.90
Improvement	—	+1.22	+0.32	+0.24	+0.86	+0.58	0.18

表 2 本文 MonoSPDC 在 KITTI 的测试集上对于 Pedestrian 以及 Cyclist 类别的检测及比较结果

Table 2 Detection and comparison results of the proposed MonoSPDC on KITTI's test set for Pedestrian and Cyclist categories

方法	时间/ms	AP_{3D} (Pedestrian)			AP_{3D} (Cyclist)		
		Easy	Mod	Hard	Easy	Mod	Hard
MonoDLE ^[24]	40	9.64	6.55	5.44	4.59	2.66	2.45
CaDDN ^[4]	630	12.87	8.14	6.76	7.00	3.41	3.30
MonoEF ^[21]	30	4.27	2.79	2.21	1.80	0.92	0.71
MonoFlex ^[14]	35	9.43	6.31	5.26	4.17	2.35	2.04
GPUNet ^[25]	34	14.72	9.53	7.87	4.18	2.65	2.09
MonoDTR ^[7]	37	15.33	10.18	8.61	5.05	3.27	3.19
MonoDistill ^[27]	40	12.79	8.17	7.45	5.53	2.81	2.40
MonoCon ^[22]	26	13.10	8.41	6.94	2.80	1.92	1.55
Ours	31	16.06	10.35	7.84	7.64	3.55	2.99
Improvement	—	+0.73	+0.17	-0.77	+0.64	+0.14	-0.31

在 Pedestrian 类别中,基于 Transformer 的 MonoDTR 算法在 Hard 任务中取得了最佳优秀的成绩。本文所提 MonoSPDC 在 Easy 和 Mod 任务中取得了最佳的成绩。MonoDTR 算法在 Hard 任务中的优秀的成绩得益于其基于激光雷达额外深度值信息的辅助训练,但这也使 MonoDTR 算法的训练成本远高于 MonoSPDC。相比于 MonoDTR 算法需要额外信息的辅助训练, MonoSPDC 在自动驾驶等更关注计算速度和数据采集成本的应用领域中更具优势,这也证明了本文算法 MonoSPDC 的应用成本低,部署简洁的特性。

在 Cyclist 类别中, CaDDN 算法在 Hard 任务中取得了最优秀的成绩, MonoSPDC 在 Easy 和 Mod 任务中取得最佳成绩。与 MonoDTR 算法相同, CaDDN 算法也使用了额外的深度值信息来辅助训练。此外, CaDDN 算法的网络结构十分复杂,耗时达到了 630 ms。这一数值是 MonoSPDC 的 21.3 倍。如此耗时的算法并不适用于关注计算速度的应用领域。

除此之外, MonoSPDC 算法还在 Waymo 数据集的汽车类别与 CaDDN 和 MonoCon 的官方代码进行了对比分析,结果如表 3 所示。MonoSPDC 在汽车类别上的检测精度实现了明显提升,分别提高了 6.07、3.31。

综上,本文算法 MonoSPDC 在多个数据集进行验证,表现了很强的泛化性,无论在精度还是检测速度以及实际应用的可行性上都处于绝对的优势。

2.3 定量分析

基于 KITTI 数据集,本文对所提出的 Mono-

表 3 所提 MonoSPDC 在 Waymo 的验证集上对于 Car 类别的检测比较结果

Table 3 Detection comparison results of the proposed MonoSPDC on Waymo's val set for the Car category

Method	AP_{3D} (Car)		AP_{3D} (Car)	
	Easy	Mod.	Easy	Mod.
CaDDN ^[4]	61.48	54.33	67.29	57.09
MonoCon ^[22]	64.24	56.17	69.77	59.64
Ours	67.55	59.17	70.98	62.66
IMPROVEMENT	+3.31	+3.00	+1.21	+3.02

SPDC 算法中的各模块进行消融实验,消融实验数据如表 4 所示。表 4 中显示了在消融实验中,本文中的两个模块 (FGCR, MVPI) 以及 MonoFlex^[13] 中的 EdgeFusion 模块都提升了 AP_{3D} 和 AP_{BEV} 检测指标。其中, EdgeFusion 模块通过将物体分为 inside 和 outside 的方法提升了边缘截断目标的召回率,在 MonoFlex 中有详细的实验过程。

实验 7 不使用 FGCR 模块和实验 8 使用 FGCR 模块的效果图如图 6 所示,可以看出使用 FGCR 模块相比于不使用 FGCR 模块,生成的 HeatMap 过滤掉了杂质(如方框内所示),使 HeatMap 的生成更加可靠,提高了物体的精确率。

实验 5 不使用 MVPI 结构和实验 8 使用 MVPI 结构的效果图如图 7 所示(彩图参见电子版,下同),其中蓝色框为真值,绿色框为检测结果,由图可知, MVPI 结构则提高了对物体深度值估计的准确性,使得物体的定位更加精准。

通过上述消融实验,充分证明了本文所提出的 FGCR 模块和 MVPI 结构以及引用的 EdgeFusion 模块的有效性。

表 4 本文所提出的两个模块 FCGR, MVPI 以及采用的 MonoFlex 中的 EdgeFusion 模块进行消融实验的结果

Table 4 Results of the ablation experiments were performed on the two modules FCGR, MVPI and the EdgeFusion module in MonoFlex that were used in this paper.

实验	BackBone	EdgeFusion	FGCR	MVPI	$AP_{3D}@IoU=0.7$			$AP_{BEV}@IoU=0.7$		
					Easy	Mod	Hard	Easy	Mod	Hard
1	✓				17.21	10.32	8.13	23.33	18.02	13.44
2	✓	✓			19.94	13.89	12.07	28.23	19.70	16.89
3	✓		✓		20.03	13.54	10.55	26.76	20.09	16.88
4	✓			✓	21.98	14.12	11.89	28.79	20.13	18.11
5	✓	✓	✓		22.03	15.10	12.86	29.76	22.06	18.50
6	✓		✓	✓	22.97	14.79	13.13	30.14	21.76	18.55
7	✓	✓		✓	23.06	15.97	13.44	30.56	21.99	19.01
8	✓	✓	✓	✓	23.87	16.78	14.19	31.94	23.17	19.90

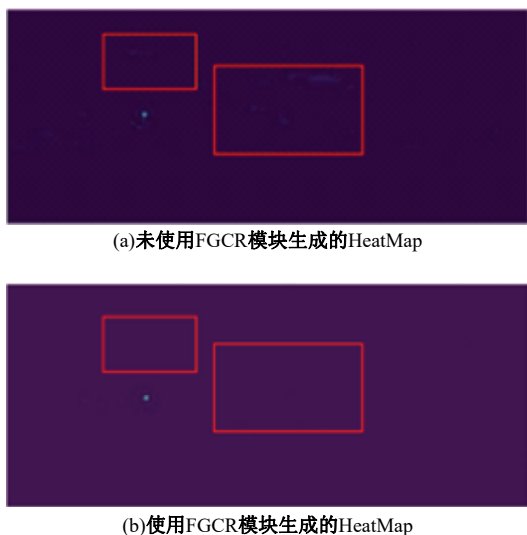


图 6 未使用 FGCR 模块与使用 FGCR 模块生成的 HeatMap 效果对比图

Fig. 6 Comparison of HeatMap effects generated with and without the FGCR

2.4 定性分析

本文在 KITTI 数据集上提供定性示例,如图 8 和图 9 所示,绿色框表示检测结果,红色框表示数据集真值。

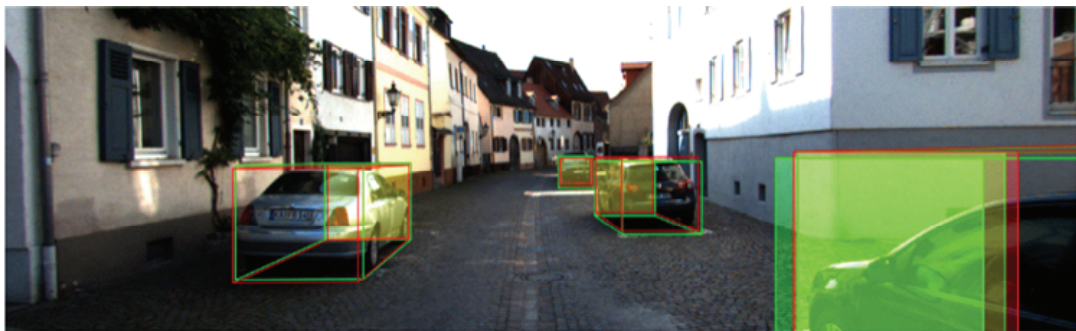


图 8 MonoSPDC 算法目标检测效果图

Fig. 8 Object detection rendering of the MonoSPDC algorithm

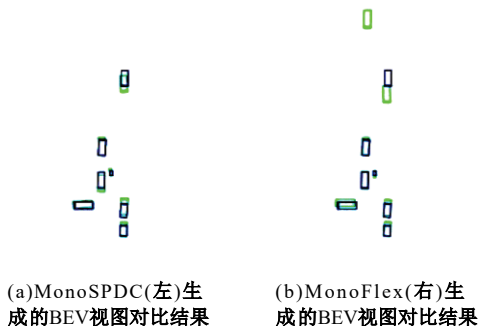
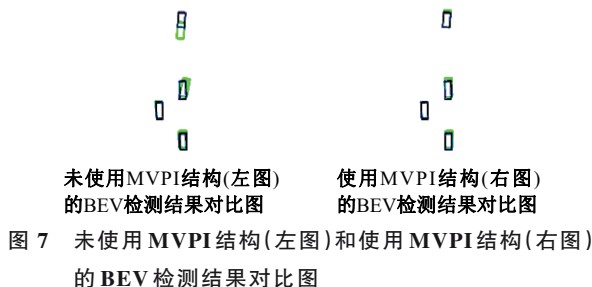


图 9 MonoSPDC 和 MonoFlex 生成的 BEV 视图对比结果

Fig. 9 Comparison results of BEV views generated by MonoSPDC(a) and MonoFlex(b)



未使用 MVPI 结构(左图)的 BEV 检测结果对比图

使用 MVPI 结构(右图)的 BEV 检测结果对比图

图 7 未使用 MVPI 结构(左图)和使用 MVPI 结构(右图)的 BEV 检测结果对比图

Fig. 7 Comparison of BEV detection results without (left image) and with (right image) the MVPI structure

图 8 为 MonoSPDC 的目标检测效果图,可见, MonoSPDC 对出现的所有目标进行了标注,没有漏检和误检。这证明本算法的检测精度是非常可靠的。从 BEV 角度看, MonoSPDC 与 MonoFlex 算法的检测效果如图 9 所示。

在图 9 中,与 MonoFlex 相比, MonoSPDC 表现出优越的精度和鲁棒性。由于缺乏对深度预测模块的关注, MonoFlex 的深度预测准确率相对较低。与 MonoFlex 相比, MonoSPDC 在 BEV 视图的生成方面 also 具有很强的置信度,取得了良好的检测效果。

本文分别在白天和夜间两个场景在实际道路进行实时测试,使用行车记录仪作为图像采集设备,图 10 为算法实际应用的可视化效果。

如图 10 所示,本文使用了白天场景和夜晚场景对本文算法进行了实际应用验证。通过图 10 可知, MonoSPDC 在白天、夜晚两个场景的实际表现都比较可靠,即对于车辆的检测并没有遗漏、误检等,对于目标框的定位也较为清晰准确。同时,在检测实时性方面,通过模型的量化部署操作后,在搭载英伟达芯片的嵌入式设备运行速度达到了 64 帧/s,满足实际场景需求,证明了 MonoSPDC 在实际应用中的潜力。



(a) 日间场景的检测结果



(b) 夜间场景的检测结果

图 10 MonoSPDC 在实际场景中的检测效果图

Fig. 10 MonoSPDC in the actual scene of the detection effect

3 结束语

本文为解决现有单目 3D 目标检测算法中,基于卷积结构的算法检测精度有待提高、基于 Transformer 结构的算法结构复杂、需要额外的辅助训练等问题,提出了一种名为 MonoSPDC 的纯卷积结构、无需辅助训练的单目 3D 目标检测算法。本文提出的全局-通道特征重组模块(FGCR)加强了 HeatMap 的生成效果,对 3D 目标检测的召回率有较好的改善效果。同时,对深度预测模块进行重构,设计了一种利用不确定性融合的多角度联合预测结构(MVPI),提高了深度预测的准确性。MonoSPDC 采用纯卷积的结构,并且不需要激光雷达的辅助训练,相比基于 Transformer 的结构有着很大的速度以及部署优势,同时在检测精度上仍然保持着 SOTA 水平。

尽管 MonoSPDC 在单目 3D 目标检测中取得了较好的性能,但本文算法并没有添加图像增强模块,导致雨天雾天等场景,其精度仍有不足。此外, MonoSPDC 的网络结构虽然相对于 Transformer 结构的算法更加轻量,检测速度在实车部署验证时虽然已经达到实时的效果,但在 100 Km/h 及以上的高速场景中 0.5~1.0 m 的采集间隔仍然有很大的空间进行提升。最后,本文

实验主要聚焦于自动驾驶场景(如 KITTI、Waymo 数据集),尚未验证算法在机器人导航、增强现实(AR)等其他依赖单目 3D 感知的领域中的适用性。MonoSPDC 算法在其他非结构化环境(如室内动态障碍物、AR 虚实交互)中的泛化能力仍需进一步探索。

在未来的研究中,基于 MonoSPDC 算法,将继续设计添加图像增强模块以适用于多种复杂场景。对于检测速度方面,继续优化耗时的模块,使算法更加轻量,增加高速驾驶场景的可靠性和降低硬件依赖的成本。除此之外,在室内移动机器人场景,AR 虚实交互等场景,验证并优化其对不同场景下的动态障碍物(如行人、家具)的检测能力,并探索轻量化部署方案(如嵌入式设备适配)。

参考文献:

- [1] Qian R, Lai X, Li X. 3D object detection for autonomous driving: a survey[J]. Pattern Recognition, 2022, 130: 108796.
- [2] Wang T, Zhu X, Pang J, et al. FCOS3D: fully convolutional one-stage monocular 3D object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, Canada, 2021: 913-922.
- [3] Mousavian A, Anguelov D, Flynn J, et al. 3D bounding box estimation using deep learning and geometry[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, USA, 2017: 7074-7082.
- [4] Reading C, Harakeh A, Chae J, et al. Categorical depth distribution network for monocular 3D object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 2021: 8555-8564.
- [5] Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers[C]//European Conference on Computer Vision, Online, 2020: 213-229.
- [6] Wang Y, Guizilini V C, Zhang T, et al. DETR3D: 3D object detection from multi-view images via 3D-to-2D queries[C]//Conference on Robot Learning, Baltimore, USA, 2022: 180-191.
- [7] Huang K C, Wu T H, Su H T, et al. MonoDTR: monocular 3D object detection with depth-aware transformer[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,

- New Orleans, USA, 2022: 4012-4021.
- [8] Pu F, Wang Y, Deng J, et al. MonoDGP: monocular 3D object detection with decoupled-query and geometry-error priors[J/OL]. [2024-10-25]. <https://arxiv.org/abs/2410.19590>.
 - [9] Duan K, Bai S, Xie L, et al. CenterNet: keypoint triplets for object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), 2019: 6569-6578.
 - [10] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite [C]//IEEE Conference on Computer Vision and Pattern Recognition, Providence, USA, 2012: 3354-3361.
 - [11] Sun P, Kretschmar H, Dotiwalla X, et al. Scalability in perception for autonomous driving: Waymo open dataset[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, USA, 2020: 2446-2454.
 - [12] Yu F, Wang D, Shelhamer E, et al. Deep layer aggregation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, USA, 2018: 2403-2412.
 - [13] Zhu X, Hu H, Lin S, et al. Deformable ConvNets v2: more deformable, better results[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, USA, 2019: 9308-9316.
 - [14] Zhang Y, Lu J, Zhou J. Objects are different: flexible monocular 3D object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 2021: 3289-3298.
 - [15] Wu Y, He K. Group normalization[C]//Proceedings of the European Conference on Computer Vision, Munich, Germany, 2018: 3-19.
 - [16] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. *Advances in Neural Information Processing Systems*, 2017, 30: 1-15.
 - [17] Philion J, Fidler S. Lift, splat, shoot: encoding images from arbitrary camera rigs by implicitly unprojecting to 3D[C]//Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, 2020: 23-28.
 - [18] Wang T, Zhu X, Pang J, et al. Probabilistic and geometric depth: detecting objects in perspective[C]//Conference on Robot Learning, PMLR, Baltimore, USA, 2022: 1475-1485.
 - [19] Loshchilov I, Hutter F. Decoupled weight decay regularization[C]//International Conference on Learning Representations, New Orleans, USA, 2019: 1-14.
 - [20] Liu Z, Wu Z, Tóth R. SMOKE: single-stage monocular 3D object detection via keypoint estimation [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, USA, 2020: 996-997.
 - [21] Zhou Y, He Y, Zhu H, et al. Monocular 3D object detection: an extrinsic parameter free approach[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 2021: 7556-7566.
 - [22] Liu X, Xue N, Wu T. Learning auxiliary monocular contexts helps monocular 3D object detection[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, 36(2): 1810-1818.
 - [23] Philion J, Fidler S. Lift, splat, shoot: encoding images from arbitrary camera rigs by implicitly unprojecting to 3D[C]//European Conference on Computer Vision, Glasgow, UK, 2020: 194-210.
 - [24] Ma X, Zhang Y, Xu D, et al. Delving into localization errors for monocular 3D object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 2021: 4721-4730.
 - [25] Wang L, Yu C, Salián S, et al. Searching the deployable convolution neural networks for GPUs[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, USA, 2022: 12227-12236.
 - [26] Zhang R, Qiu H, Wang T, et al. MonoDETR: depth-guided transformer for monocular 3D object detection[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2023: 9155-9166.
 - [27] Chong Z, Ma X, Zhang H, et al. MonoDistill: learning spatial features for monocular 3D object detection [C]//International Conference on Learning Representations, Online, 2022: 1-16.