基于时空注意力Transformer的自动驾驶运动规划方法

袁 丁,李 源,孟羽倩,张 弘,杨一帆*
(北京航空航天大学宇航学院,北京 102206)

摘 要: 驾驶场景中的静态智能体、动态智能体、道路结构及各元素间的交互通常是复杂且随时空快速变化的. 因此,自动驾驶车辆的运动预测是一项十分具有挑战性的任务,其中一个尚未解决的难题就是如何高效表征和融合多模态场景信息,包括路况信息、不同智能体状态及其历史交互信息. 现有方法大多依靠独立设计的模块并行处理多个模态的数据,但这种方式会造成系统灵活度较差、调整困难,且独立组件往往会引起较高的计算冗余,系统计算效率较低. 此外,由自动驾驶场景的时间信息和空间信息解码获得保障安全驾驶的动作指令本身就是一项十分具有挑战性的任务. 本文提出基于时空注意力Transformer的自动驾驶运动规划方法,由分阶段多模态场景编码器和时空融合解码器组成,能够逐过程构建多模态运动场景描述,同时在时空融合下预测自车的未来安全运动. 本文在大规模自动驾驶数据集nuScenes上搭建了全新的比较基线,取得了较为领先的结果.

关键词: 自动驾驶运动预测;分阶段多模态编码器;时空融合解码器;Transformer;全新基线
基金项目: 国家自然科学基金(No.62002005,No.61972015)
中图分类号: TP183 **文献标识码:** A **文章编号:** 0372-2112(2025)07-2418-10
电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20241022

A Motion Planning Method for Autonomous Driving Based on Spatiotemporal Attention Transformer

YUAN Ding, LI Yuan, MENG Yu-qian, ZHANG Hong, YANG Yi-fan*
(*School of Astronautics*, *Beihang University*, *Beijing* 102206, *China*)

Abstract: The static and dynamic agents, road structures, and interactions among various elements in driving scenarios are typically complex and rapidly change across time and space. Consequently, motion prediction for autonomous vehicles remains a challenging task, especially with the open problem of efficiently representing and integrating multi-modal scene information, including road conditions, various agent states, and historical interaction information. Current approaches often rely on independently designed modules to process each modality in parallel. However, this approach tends to result in limited system flexibility, challenging adjustments, and, frequently, high computational redundancy, which reduces overall system efficiency. Furthermore, decoding the spatiotemporal information from autonomous driving scenarios to generate safe driving commands is inherently challenging. This paper proposes an autonomous driving motion planning method based on a spatiotemporal attention Transformer, comprising a phased multi-modal scene encoder and a spatiotemporal fusion decoder. This model progressively constructs a multi-modal scene representation and predicts the future safe trajectory of the autonomous vehicle under spatiotemporal fusion. The proposed approach establishes a new baseline on the large-scale nuScenes autonomous driving dataset, achieving competitive results.

Key words: autonomous driving motion prediction; phased multimodal encoder; spatiotemporal fusion decoder; transformer; new baseline
Foundation Item(s): National Natural Science Foundation of China (No.62002005, No.61972015)

收稿日期:2024-11-11;录用日期:2025-07-17;责任编辑:朱梅玉
*通讯作者:杨一帆

1　引言

随着深度学习的快速发展以及传感器质量精度的提升,目前的自动驾驶汽车感知算法已经能够通过获取自车过去状态、当前路况等多模态信息,实现对当前车辆所处场景的建模. 然而,由于驾驶环境复杂多变,如何理解此类场景,提示自动驾驶车辆注意到不同交通参与者的动作与潜在的意图,对车辆未来动作作出正确预测并给予准确的路径规划,仍是一项具有挑战性的任务.

当前运动规划研究存在的主要缺陷包括过度依赖预测模块输出的未来位置作为输入,将交通规则编码为硬约束,对长时场景的鲁棒性有限等[1,2]. 这是规划质量瓶颈很大程度上受限于上游运动预测的准确性与对交互意图的捕捉能力导致的. 同时,精确建模多智能体间的意图、决策过程及其互相影响极其复杂,涉及高维状态空间,现有算法和计算框架尚难以高效可靠地解决. 因此,探索能够有效融合不确定性预测、显式建模并遵守规则的运动规划方法,是当前研究的迫切需求.

最近,一些学者提出利用Transformer网络解决运动规划问题[3]. 虽然这些方法提供了简化的模型架构,但需依赖于领域先验知识和模态的针对性调整[4]. 文献[5]提出了一系列交叉关注层,顺序性处理每一种模态,且允许根据任务自定义模态的处理顺序,但枚举所有可能性组合是不切实际的. 文献[6]提出使用独立的编码器将每种模态嵌入展平为向量并输入预测器,同时开放网络类型及其容量. 虽然这些方法允许多种自由度,但也显著增加了搜索空间. 若无有效的网络搜索架构或大量的人工输入和手工设计,鉴于已探索的建模选项数量有限,所选择的模型可能无法达到最优.

另外,标准的Transformer结构最初是为自然语言处理任务设计的,用以处理序列数据中的长程依赖性[7]. 但这种模型假设输入序列中的每个位置是对等的,因此在处理时空数据时,缺乏显式的时空结构建模能力. 一些方法[8~10]选择先将时空信号转化为长序列后再进一步处理,但由于标准Transformer的计算复杂度与输入序列长度的二次方成正比,这类方法反而会引起更大的计算资源需求和更高昂的时间成本[11].

受以上算法的启发,本文提出了一种基于时空注意力Transformer的自动驾驶运动规划方法,重点关注多模态输入的信息表征,避免复杂的独立组件设计. 同时,构建了分阶段的多模态场景编码器,以逐过程描述多模态的运动场景,在尽可能不牺牲预测质量的前提下,处理同时存在于时间和空间维度中的多模态特征. 此外,针对自车动作规划问题,设计了时空融合解码器,利用自车及其他智能体在时空维度的变化,精准安全地规划自动驾驶车辆的未来运动. 最后,为进一步证明该方法的有效性,本文还在大规模自动驾驶数据集nuScenes上搭建了一个全新的比较基线. 实验结果表明,本文提出的模型在各个性能指标上均优于标准的Transformer编解码器模型以及其他经典方法,可同时实现准确的场景理解并给出未来的安全运动预测.

综上所述,本文提出的基于时空注意力Transformer的自动驾驶运动规划方法的主要贡献包括以下几方面.

(1)设计更为有效的多模态信息表征输入,并构建分阶段融合的多模态编码器,以解决独立组件设计的效率困难.

(2)构建时空注意力Transformer解码器,以避免时空环境的快速变化导致的信息遗漏,精准预测自车未来运动.

(3)在nuScenes上搭建全新的基线模型,开辟在大规模自动驾驶数据集上多模态运动规划任务的新基线.

(4)实验表明,本文所提模型在所有比较指标上均优于标准Transformer编解码器模型,验证了算法的有效性.

2　相关工作

2.1　多模态自动驾驶任务

由于成像设备和各种传感器的快速发展,如今的自动驾驶汽车上通常会配备各种传感器,用以提供当前车辆所处场景的状态信息. 例如,RGB图像传感器可以自然呈现人类感知世界的方式,可提供丰富的语义视觉信息;LiDAR(Light Detection And Ranging)利用激光进行探测,可获取三维点云并生成精确的数字三维模型;立体相机通过多个图像传感器模拟人类的双目视觉,可感知精确的深度信息. 此外,车辆状态(如车速表和惯性测量单元,分别提供车辆的速度信息和加速度信息)以及高级导航命令也是指导端到端系统的重要输入. 然而,不同传感器具有不同的视角和数据分布,它们之间的巨大差异给数据的有效融合互补带来了极大的挑战. 目前,多传感器融合研究主要集中在感知领域,例如目标检测、跟踪和语义分割等,端到端轨迹预测算法的相关研究仍处于探索阶段.

最近的研究采用Transformer来模拟特征对之间的交互. 利用两个独立的卷积编码器分别处理图像和LiDAR输入,并将每个特征与Transformer编码器互连,从而实现不同阶段的特征融合. 其中,注意力机制在聚合不同传感器输入的上下文信息,以及实现更安全的端到端驾驶性能方面表现出了极大的有效性.

然而,如何融合多模态的输入仍是一项值得深究

的任务，目前常见的融合方法包括前层融合和后层融合[12,13]. 前层融合是指在进行编码之前利用拼接或投影的方法将多模态参数合并为一层，后接由单个自注意力编码器组成的场景编码器. 这种做法的好处是使网络能以自适应的形式为不同的模态划分权重，但是原始数据噪声易导致对齐误差与信息损失. 在后层融合中，每种模态都有自己专用的编码器[14]，输出决策级结果后再融合，这也是目前运动预测模型的常用方法. 然而，不同模态的分离计算和网络的单独设置均需大量搜索的人工设计[15,16]. 同时，新兴趋势表明，分阶段融合正成为突破方向. E2E-MFD 通过梯度矩阵任务对齐技术同步优化图像融合与目标检测任务，实现单阶段目标检测任务[17]；FusionAD 则在 BEV（Bird's-Eye-View）空间构建多模态时序融合模块，引入模态自注意力细化目标跟踪任务[18]. 这些工作证明，分阶段方法能平衡计算效率与特征交互深度，避免传统方法的结构局限. 因此，本文提出分阶段的融合方法，在聚合多模态向量的同时避免复杂的手工设计.

2.2 Transformer 架构

Transformer 主要分为编码器和解码器两部分，通过多层迭代实现各种下游学习任务[7]. 编码器和解码器各由多个模块叠加而成，这种多层迭代的方式可以对输入数据实现深度特征提取和最终输出生成. 为区别于时空编解码器的特点，在本文中对基础的编码器和解码器进行详细描述.

编码器的核心组件是多头自注意力机制（Multi-Head Self-Attention）和前馈网络（Feed Forward Network）. 其输入通常表示为矩阵 $\boldsymbol{X}\in\mathbb{R}^{n\times d}$，其中 n 是输入序列的长度，d 表示嵌入维度.

自注意力机制的目的是捕捉输入序列中不同位置之间的依赖关系. 对于输入矩阵 $\boldsymbol{X}$，首先通过线性变换获得查询（Query）、键（Key）、值（Value）3 个矩阵 $\boldsymbol{Q}=\boldsymbol{X}\boldsymbol{W}^{Q}$，$\boldsymbol{K}=\boldsymbol{X}\boldsymbol{W}^{K}$，$\boldsymbol{V}=\boldsymbol{X}\boldsymbol{W}^{V}$，其中 $\{\boldsymbol{W}_i^{Q},\boldsymbol{W}_i^{K},\boldsymbol{W}_i^{V}\}\in\mathbb{R}^{d\times d}$ 是学习到的权重矩阵. 自注意力的输出由以下公式计算：

$$\mathrm{SelfAtt}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=\mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{1}$$

其中，d_k 表示键的维度.

为了捕捉输入序列中不同子空间的特征，多头注意力机制将自注意力计算扩展到多个头上，每个头独立地执行注意力操作. 设有 H 个头，多头注意力的输出为：

$$\mathrm{MultiAtt}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=\mathrm{Concat}\left(\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_H\right)\boldsymbol{W}^{O} \tag{2}$$

其中，$\boldsymbol{h}_i=\mathrm{SelfAtt}\left(\boldsymbol{Q}\boldsymbol{W}_i^{Q},\boldsymbol{K}\boldsymbol{W}_i^{K},\boldsymbol{V}\boldsymbol{W}_i^{V}\right)$，$\{\boldsymbol{W}_i^{Q},\boldsymbol{W}_i^{K},\boldsymbol{W}_i^{V}\}\in\mathbb{R}^{d\times d_k}$ 是权重矩阵，$\boldsymbol{W}^{O}\in\mathbb{R}^{H\times d_k\times d}$ 为多头映射矩阵.

多头注意力输出可通过前馈网络 FFN（Feed-Forward Network）进行传播. 前馈 FFN 通常设计为由 ReLU 激活层连接的全连接网络，计算公式可表示为

$$\mathrm{FFN}\left(\boldsymbol{x}\right)=\mathrm{ReLU}\left(\boldsymbol{x}\boldsymbol{W}_1+\boldsymbol{b}_1\right)\boldsymbol{W}_2+\boldsymbol{b}_2 \tag{3}$$

其中，$\boldsymbol{W}_1\in\mathbb{R}^{d\times d_f}$，$\boldsymbol{W}_2\in\mathbb{R}^{d_f\times d}$，$\boldsymbol{b}_1\in\mathbb{R}^{d_f}$，$\boldsymbol{b}_2\in\mathbb{R}^{d}$ 分别为网络映射权值和偏置.

每一层的输出在进入下一个模块之前都会进行层归一化，以稳定训练过程，提高模型的收敛性.

解码器的结构与编码器类似，但除了自注意力和前馈网络外，还增加了一个编码器-解码器注意力层（Cross-Attention Layer）. 该层类似于自注意力机制，但查询矩阵来自解码器的输入，而键和值矩阵来自编码器的输出. 本文的时空编解码器将基于上述机制，在每个时间步中，解码器接收编码器的输出和解码器本身的输入，并输出一个新的预测，直到生成完整的序列.

2.3 自动驾驶运动规划

运动规划作为自动驾驶决策系统的核心环节，其目标是在满足车辆动力学约束、遵守交通规则并确保安全的前提下，为自车生成一条从当前位置到目标位置的可执行轨迹. 近年来，该领域的研究逐渐从传统几何方法演变为更注重交互和学习的混合方法. 基于优化的方法将规划问题建模为非线性优化问题，直接最小化目标函数（如舒适性、效率、安全性）并满足动力学模型、环境约束（障碍物避碰）、交通规则等. 预测控制（Model Predictive Control，MPC）框架是此类方法的典型代表，通过滚动时域优化生成局部最优轨迹[19,20]. 这类方法能生成高质量轨迹，但对初始解敏感，计算负担重，且依赖于精确的环境模型和约束表达. 随着深度强化学习和模仿学习被广泛应用于运动规划，深度强化学习（Deep Reinforcement Learning，DRL）通过与仿真环境交互学习规划策略，能处理高维状态和复杂交互，但训练困难、样本效率低且可解释性差[21]. IL 通过学习人类驾驶员数据获得规划能力，能生成类人轨迹，但对数据质量要求高，泛化能力存疑[22].

另外，当前主流的运动规划器严重依赖于运动预测模块提供的精确、多模态的未来场景信息. 预测的误差或不确定性会直接传导至规划结果，导致规划失效[23]. 同时，精确、高效地建模多智能体间的意图、决策逻辑及其相互影响（博弈、协作）极其困难. 现有方法（如简化博弈模型、有限场景树）往往在模型复杂度与计算实时性之间进行妥协，难以在开放道路环境中实现鲁棒、类人的交互规划[24].

3 基于时空注意力的 Transformer

3.1 多模态驾驶场景描述

为了准确、有效地描述自动驾驶场景，本算法使用一组紧凑的可学习向量描述驾驶环境中的多模态元

素,包括道路结构,车辆、行人等智能体的状态及其交互,自动驾驶车辆的历史动作,以及交通信号灯的状态.该向量不仅包含了驾驶场景中最重要的语义属性,还可以创建自车的动作标签.该向量的设计方法如下.

车辆状态向量(Vector car):$\boldsymbol{V}_{\mathrm{c}}\in[T_{\mathrm{c}},N_{\mathrm{c}},D_{\mathrm{c}}]$,描述在历史时间与当前时间步$T_{\mathrm{c}}$内自车附近$N_{\mathrm{c}}$台车辆的状态.其中$D_{\mathrm{c}}\in\mathbb{R}^{\mathrm{c}}$描述了车辆是动态或静态、速度、加速度、在自车坐标系中的位置与朝向、在自车坐标系中的俯仰角度、尺寸、类别以及角点的位置.

行人状态向量(Vector ped):$\boldsymbol{V}_{\mathrm{p}}\in[T_{\mathrm{p}},N_{\mathrm{p}},D_{\mathrm{p}}]$,描述在历史时间与当前时间步$T_{\mathrm{p}}$内自车附近$N_{\mathrm{p}}$名行人的状态.其中$D_{\mathrm{p}}\in\mathbb{R}^{\mathrm{p}}$描述了行人是否正在穿越马路、速度、在自车坐标系中的位置与方向、尺寸以及类别.

自车状态向量(Vector ego):$\boldsymbol{V}_{\mathrm{e}}\in[T_{\mathrm{e}},N_{\mathrm{e}},D_{\mathrm{e}}]$,描述在历史时间与当前时间步$T_{\mathrm{e}}$内自动驾驶车辆的$N_{\mathrm{e}}$个状态.其中$D_{\mathrm{e}}\in\mathbb{R}^{\mathrm{e}}$描述了自动驾驶车辆的速度、在自车坐标系中的位置与方向、历史动作、尺寸以及类别.

信号灯状态向量(Vector tl):$\boldsymbol{V}_{\mathrm{tl}}\in[T_{\mathrm{tl}},N_{\mathrm{tl}},D_{\mathrm{tl}}]$,描述在历史时间与当前时间步$T_{\mathrm{tl}}$内交通信号灯的$N_{\mathrm{tl}}$个状态.其中$D_{\mathrm{tl}}\in\mathbb{R}^{\mathrm{tl}}$描述了自车前方的组合交通信号灯的距离以及颜色状态.

地图点向量(Vector map):$\boldsymbol{V}_{\mathrm{m}}\in[T_{\mathrm{m}},N_{\mathrm{m}},D_{\mathrm{m}}]$描述在历史时间与当前时间步$T_{\mathrm{m}}$内驾驶场景中$N_{\mathrm{m}}$个地图点(车道线、道路边缘、人行横道)的向量建模.其中$D_{\mathrm{m}}\in\mathbb{R}^{\mathrm{m}}$描述了地图元素在自车坐标系中的位置与方向、道路宽度以及速度限制.需要注意的是,由于道路的静态特性,道路元素描述在输入的批次中是没有时间维度的.

3.2　分阶段场景编码器

如图1所示,本文设计的模型主要由两个组件组成:分阶段场景编码器和时空融合动作解码器.

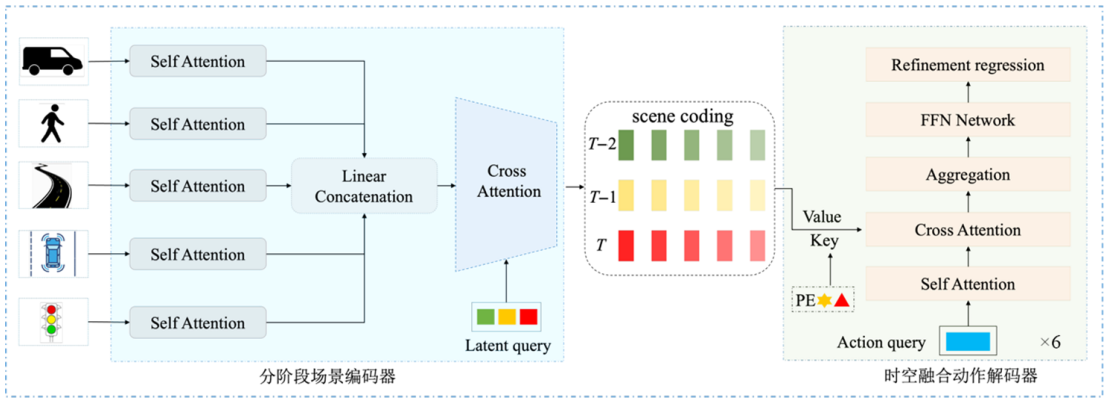

图1　时空注意力Transformer模型网络结构

分阶段场景编码器主要由多个概括驾驶场景的注意力编码器组成,通过融合不同模态的信息生成对整体环境的特征表示.图1左侧展示了多模态输入及场景编码的生成过程.该场景编码充当解码器的查询库,以生成覆盖输出空间的k个可能动作多模态的描述.

不同的模态输入在编码时可能会学习到不同的空间特征,而驾驶场景需要统一的坐标空间和特征空间,因此空间特征的多样性不利于聚焦到自动驾驶车辆本身.为解决这一问题,本文通过调整智能体的运动姿态,旋转智能体的位置到以自动驾驶车辆为中心的坐标系中,同时将历史时间步对齐到当前时间步,在沿时空维度连接所有模态之前将其转换到公共坐标空间,从而实现特征和坐标空间的统一.

另外,带有位置嵌入的自注意力本质上是具有置换不变性的,也即编码器可以被视为集合编码器而非顺序编码器.然而,在自动驾驶场景中,某些特定模态间可能存在顺序要求,例如在不同历史时间步上的速度等.由此可见,打破置换不变性并考虑使用顺序信息有利于智能体的充分学习.基于此,本文通过给所有模态添加可学习的位置嵌入,并设置初始值为0,让模型自动学习是否有必要在模态中加入顺序要求,实现顺序信息的自适应引入.

本文设计的分阶段融合机制结合了前层融合和后层融合的优势,能够在跨模态交互编码的同时,避免不同模态特定编码器复杂的手工设计.具体来说,首先将多模态输入数据送入独立的自注意力编码器中进行初步的特征提取和模态内信息整合,这些编码器保持相同的Transformer架构,由输入序列长度二次方的自注意力模块及前馈网络构成,但不共享参数,以保留模态特性.然后,将经过自注意力编码器编码后的特征序列沿着特征维度进行拼接.最终,将拼接特征向量送入一个由M个时间编码器块和M个空间编码器块组成的子编码器块中进行核心交互.关键点在于,其中的自注意力模块同时在空间维度和时间维度上作用于拼接后的联合特征序列.这意味着序列中的任意时刻下每一个位置的特征向量都可以通过注意力机制关注到序列中

其他任何位置的特征向量，从而实现了模态间的自由交互和信息融合.

同时，为降低长序列带来的每个块的自注意力和前馈网络的计算成本，本文在编码器块中引入了潜在查询. 具体来说，本文不对输入特征序列的所有元素进行密集的自注意力计算，而是将输入$\boldsymbol{x}\in\mathbb{R}^{T\times M\times D}$映射到一个低维的潜在空间$\boldsymbol{z}\in\mathbb{R}^{T\times M'\times D'}$. 首先，生成一个可学习的潜在查询矩阵$\boldsymbol{Q}_{\text{latent}}$；然后，在自注意力层中，使用这个固定的、较小的潜在查询作为Query，而Key和Value仍然由输入序列通过线性变换得到. 其中，潜在查询的维度由缩放因子$K=\dfrac{M'}{M}(0<K\leqslant 1)$决定. 缩放因子为输入序列长度的百分比，代表潜在查询数量占原始序列长度的百分比. 为了保证计算过程的一致性以及维持后续解码器的特征维度对齐，分阶段融合过程的所有子编码器中缩放因子K恒定为0.25.

3.3 时空融合动作解码器

时空融合动作解码器的核心组件之一是时空注意力模块. 输入可学习的初始查询和时空位置编码，然后与场景编码之间进行交叉注意力操作提取出最关键的时空特征，用于最终的动作规划生成. 该模块在Transformer注意力机制的基础上进行了特定的优化设计，以适应自动驾驶场景中复杂且动态变化的时空环境.

具体来说，模块的输入包括可学习的初始查询(Query)和来自编码器的场景编码(Key，Value)，其中Query是一组嵌入向量，表示解码器在当前时刻对不同特征的关注程度，Key和Value均为多阶段融合编码器学习到的场景抽象表示.

同时，为了进一步增强时空注意力模块的性能，使模型能够更好地理解和利用输入数据的时空结构，本文还添加了时空位置编码(Spatio-Temporal Positional Encoding)，为自动驾驶车辆的未来动作提供了更精细的规划. 其中，时间位置编码用于标识不同时间步之间的相对位置，与传统序列位置编码类似，采用正弦、余弦函数生成：

$$\mathrm{PE}_{(t,2i)}=\sin\left(\frac{t}{10\,000^{2i/d}}\right) \tag{4}$$

$$\mathrm{PE}_{(t,2i+1)}=\cos\left(\frac{t}{10\,000^{2i/d}}\right) \tag{5}$$

其中，t表示时间步，i为编码的维度索引，d是编码的总维度.

基于自动驾驶车辆的运动特性，空间位置编码描述了车辆的空间位置信息，同样采用了正弦、余弦函数生成：

$$\mathrm{PE}_{(t,2i)}=\sin\left(\frac{\boldsymbol{p}}{10\,000^{2i/d}}\right) \tag{6}$$

$$\mathrm{PE}_{(t,2i+1)}=\cos\left(\frac{\boldsymbol{p}}{10\,000^{2i/d}}\right) \tag{7}$$

其中，$\boldsymbol{p}(x,y,z)$是自动驾驶车辆在世界坐标系下的位置，i为编码的维度索引，d是编码的总维度.

在得到时空位置编码后，分别将其添加到Key和Query上，使模型能够同时根据特征图和自车时空位置计算注意力权重. 不同时间帧和空间区域上的权重学习是相互独立的. 因此，本文使用多头注意力在不同的时空维度上并行计算，以捕捉特定时间及位置上的重要特征，如接近路口时的横向车辆运动状态和直道上前方车辆的速度变化等. 此外，变化的权重能够帮助模型在不同时刻和不同位置上聚焦到最有助于驾驶的特征. 这种适应性对于处理复杂的动态自动驾驶环境也尤为重要.

本文的方法聚焦于将编码器中不同模态的集成信息解码为自动驾驶车辆的确定性动作规划输出. 与使用高斯混合模型表示智能体可能采取的动作不同，本算法直接使用线性层估计自动驾驶车辆在下一步的动作参数. 这些参数包括了车辆在每一个时间步的加速度响应(accelerate)、刹车比(brake)和转向角(steer)，可表示如下：

$$\boldsymbol{A}=\left\{a^t,b^t,s^t\right\},\quad t=1,2,\cdots,T \tag{8}$$

进一步地，为了保证系统输出的稳定性和安全性，本算法设置了加速限制、减速限制和半径限制，并在动作规划阶段严格遵守. 具体来说，为了让车辆在加速时不超过安全范围，设置了加速阈值为0.3~2.8 m/s^2；为了防止剧烈刹车、提高乘车的舒适性，设置了减速阈值为−2.8~−0.8 m/s^2；为了防止车辆在转弯时失去控制，设置了转弯半径阈值为8~18 m.

同时，本文根据自车的初始速度v、轨迹曲率κ和加速度a将真实轨迹数据进行聚类，并创建了候选轨迹集合. 训练时，可在输出状态的候选轨迹集合中检索自车轨迹.

此外，考虑到车辆在现实世界中行驶时应遵循动力学模型，算法的轨迹输出还应采用广泛用于自动驾驶汽车规划的自行车模型生成加速度和转向速率分布[25]. 也就是说，自车的二维路径可被描述为一条典型的欧拉螺旋线(Euler spiral). 在该曲线上，曲率κ与其路径距离ε成正比，即$\kappa(\varepsilon)=\pi\varepsilon$. 该曲率特性表明了轨迹点的角速度也是线性的. 因此，在二维平面的欧拉螺旋曲线s上，与曲线上任意点s_0距离为ε的自车路径$s(\varepsilon)$可计算如下：

$$s(\varepsilon)=s_0+a\left[C(\varepsilon')T_0+S(\varepsilon')N_0\right] \tag{9}$$

其中，$\varepsilon'=\dfrac{\varepsilon}{\beta}$，$\beta$是比例因子，取值区间为[1, 10]；$T_0$和$N_0$表示曲线在点$s_0$处的切线和法向量. 上式中，$S(\varepsilon')$和

$C(\varepsilon')$的定义如下:

$$S(\varepsilon')=\int_0^{\varepsilon'}\sin\left(\frac{\pi u^2}{2}\right)\mathrm{d}u \tag{10}$$

$$C(\varepsilon')=\int_0^{\varepsilon'}\cos\left(\frac{\pi u^2}{2}\right)\mathrm{d}u \tag{11}$$

最后,还需要考虑自车沿路径$s(\varepsilon)$上的纵向运动速度$v(t)=\dot{v}t+v_0$. 其中,v_0为自车在当前轨迹点的初速度. 由此,自车的轨迹具有连续曲率变化的特点,且在自车轨迹上定义了连续的速度,从而使自车轨迹在符合动力学模型约束的同时,具备更高的平滑性和可控性.

4　实验结果与分析

本文在新搭建的基线以及相应的数据集上对所提出的模型进行性能验证与评估. 本节主要针对基于时空融合Transformer的自动驾驶运动规划任务进行性能比较和消融实验,以此说明所提方法的有效性.

4.1　实验设置

4.1.1　数据集生成

为了验证算法的有效性与鲁棒性,本文选取自动驾驶领域中流行的大型公开数据集nuScenes[26]进行实验验证. 该数据集中包含来自真实世界的大量带标注的传感器数据、目标相应位置、车辆驾驶状态、连续场景下的车辆位姿等,可以用于多种自动驾驶任务的指标评估. 如3.1节所述,实验根据nuScenes的多类型标注生成多模态的驾驶场景描述子,其中训练集长度为13 576帧,测试集长度为2 429帧.

4.1.2　基线模型

本节以所提出的基于时空注意力Transformer的自动驾驶运动规划方法作为全新的基线. 同时,为验证方法的有效性和差异性,采用基于多层感知机(Multi-Layer Perceptron,MLP)的编解码器作为对比,将其中场景编码网络的分阶段编码器替换为MLP,并使用额外的时空信息作为附加输入,通过MLP解码器输出的自动驾驶车辆的动作预测信号. 为公平地比较模型性能,实验中对用以对比的基线模型进行了校准,使其可训练参数与评测数据集和本文方法完全一致.

4.1.3　参数设置

本节实验均在表1所示的实验环境中进行,模型基于PyTorch的Transformer框架搭建. 在训练过程中,采用AdamW优化器,学习率设置为0.000 1,并采取余弦退火下降方式,在2块GeForce RTX A5000上设置了batch size为24的分布式训练.

4.2　性能对比与分析

在4.1.3节所述的实验环境下,本节详细评测了4.1.2节的对比基线模型. 实验采用如下定量指标评估模型性能:(1)为衡量输出的自动驾驶车辆速度预测准确性,计算归一化后的自车加速度和刹车压力与真实标签间的平均绝对误差(Mean Absolute Error,MAE),记作E_{ahead};(2)为衡量输出的自动驾驶车辆方向预测准确性,计算归一化后的自车方向盘转角与真实标签间的MAE,记作E_{steer};(3)为衡量交通信号灯预测准确性,计算交通信号灯预测准确率,记作A_{tl};(4)为衡量自车动作与交通信号灯的关联性,计算自车与交通信号灯距离的MAE,记作E_{tl};(5)为衡量其他参与交通的智能体数量预测准确性,计算预测数量与真实标签之间的MAE,记作E_{agent}.

表1　实验环境与配置

	环境
OS	Ubantu18.04.1LTS
Software	Python & PyTorch
HDD	5 TB
CPU	Intel® Xeon® Silver 4214 CPU @ 2.20 GHz
GPU	2 * GeForce RTX A5000
算力	56 TFLOPS(FP32)

表2给出了本文方法与现有经典方法及基线模型的指标对比结果,其中Vanilla方法仅使用真值进行监督. 实验的最佳性能以粗体表示. 本文方法在各个指标上均表现出最佳性能. 具体地,相较于已发表的最佳结果(LLMDriver),本文方法在E_{ahead}上有20%的提升,在E_{steer}上有80%的提升,在A_{tl}上有4%的提升,在E_{tl}上有11%的提升,在E_{agent}上有16%的提升. 同时,相对于基线模型,本文方法在E_{ahead}上有70%的提升,在E_{steer}上有89%的提升,在A_{tl}上有32%的提升,在E_{tl}上有30%的提升,在E_{agent}上有16%的提升. 这证明在自动驾驶场景中,基于MLP的编解码器在处理传统的预测任务时具备一定的泛化性能,但是在动作规划方面具有较低的潜力,而本文所提出的方法开创性地引入时空注意力Transformer进行自动驾驶运动规划,在动作规划方面展现出了领先的优势.

此外,为进一步与其他的经典方法对比,实验还计算了1 s和2 s内预测轨迹与真实标签之间的欧式距离,分别记作L2(1 s)和L2(2 s),用以评估轨迹预测的准确性;同时利用预测的轨迹评估了与其他智能体碰撞的频率,以F_{col}(1 s)和F_{col}(2 s)表示. 为了公平比较,本文采用4.1.1节所述生成的数据集评测指标.

表3给出了本文方法与其他经典模型的指标对比结果,其中最佳性能以粗体表示. 如表3所示,本文方法在自车轨迹预测方面亦能获得最佳性能. 特别是相对于其他方法的最佳结果,本文方法在L2(1 s)上有5%的提升,在F_{col}(1 s)上有3%的提升,在F_{col}(2 s)上有5%的提升.

表2 动作规划结果实验对比

模型	发表信息	E_{ahead}↓	E_{steer}↓	A_{tl}↑	E_{tl}↓	E_{agent}↓	算力需求(FP32)/TFLOPS
Vanilla	—	0.163	0.197	0.800	0.598	0.815	7
NMP[27]	CVPR 2019	0.098	0.157	0.772	0.499	0.627	11
Perceiver-BC[28]	arXiv 2021	0.111	0.181	0.890	0.410	0.749	—
ST-P3[29]	ECCV 2022	0.059	0.062	0.907	0.381	0.396	108
LLMDriver[30]	ICRA 2024	0.066	0.094	0.758	0.475	0.568	36
基线模型	—	0.182	0.169	0.703	0.516	0.424	20
本文方法	—	**0.053**	**0.017**	**0.930**	**0.363**	**0.354**	72

表3 轨迹预测结果实验对比

模型	发表信息	L2(1 s)/m↓	L2(2 s)/m↓	F_{col}(1 s)↓	F_{col}(2 s)↓	算力需求(FP32)/TFLOPS
基线模型	—	1.57	2.49	1.03	1.25	10
Freespace[31]	CVPR 2021	0.56	**1.27**	0.65	0.86	20
MP3[32]	CVPR 2021	0.57	1.47	0.41	0.74	72
ST-P3[29]	ECCV 2022	1.33	2.10	0.33	0.66	108
UniAD[33]	CVPR 2023	0.68	1.34	0.31	0.62	48
OccNet[34]	ICCV 2023	1.31	2.18	0.35	0.65	83
本文方法	—	**0.53**	1.38	**0.30**	**0.59**	36

4.3 消融实验

为了分别验证本文所提出的分阶段融合场景编码器与时空解码器的有效性,本节分别使用前层融合场景编码器、后层融合场景编码器和标准解码器进行了对比消融实验. 其中,前层融合场景编码器在编码阶段仅使用线性投影对多模态输入向量进行对齐;后层融合场景编码器首先使用相同的编码器对多模态输入向量进行编码,然后使用线性投影将编码后的场景描述向量进行对齐;标准解码器由掩码多头自注意力模块、编解码器注意力模块和前馈网络构成,可训练参数仍保持与时空融合解码器一致[33]. 表4和表5展示了消融实验的评估结果,其中最佳性能以粗体表示. 当采用标准解码器时,相较于后层融合方式,前层融合方式在各指标上表现更差. 其原因在于,前层融合方式直接将原始多模态信息输入、拼接,而后层融合方式先进行独立编码、再投影拼接,使网络学习到更丰富的场景编码信息,从而获得较低的预测平均绝对误差和较高的准确率. 分阶段融合方式则结合了二者的优势,既保持了原始多模态输入的信息维度,也包含了丰富的独立模态编码信息. 因此,在指标上分阶段融合方式明显优于前两种方法.

表4 动作规划消融实验结果

前层融合场景编码器	后层融合场景编码器	标准解码器	分阶段融合场景编码器	时空解码器	E_{ahead}↓	E_{steer}↓	A_{tl}↑	E_{tl}↓	E_{agent}↓
√		√			0.157	0.147	0.790	0.440	0.418
	√	√			0.152	0.135	0.798	0.432	0.400
		√	√		0.139	0.108	0.801	0.416	0.383
√				√	0.087	0.049	0.887	0.399	0.370
	√			√	0.071	0.041	0.894	0.374	0.360
			√	√	**0.053**	**0.017**	**0.930**	**0.363**	**0.354**

表5 预测轨迹消融实验结果

前层融合场景编码器	后层融合场景编码器	标准解码器	分阶段融合场景编码器	时空解码器	L2(1 s)↓	L2(2 s)↓	F_{col}(1 s)↓	F_{col}(2 s)↓
√		√			0.67	1.68	0.43	0.76
	√	√			0.62	1.64	0.41	0.73
		√	√		0.60	1.61	0.38	0.69
√				√	0.61	1.59	0.37	0.65
	√			√	0.57	1.40	0.33	0.60
			√	√	**0.53**	**1.38**	**0.30**	**0.59**

此外,本文所提出的时空位置编码可对自车的动作规划起到记忆历史动作,并结合当前环境给出未来动作规划的重要作用. 因此,对比使用标准解码器和时空解码器的实验结果可以发现,标准解码器在对智能体数量的预测以及对自动驾驶车辆的动作规划输出上均表现逊色. 这也进一步验证了本文提出的时空解码器的有效性.

综上所述,与其他5种消融模型的最优结果相比,本文提出的分阶段场景编码器与时空融合解码器在标志性指标E_{ahead}上提升了25%,在E_{steer}上提升了60%,在A_{tl}上提升了3%,在E_{tl}上提升了3%,在E_{agent}上提升了2%,在L2(1 s)上提升了7%,在L2(2 s)上提升了1%,在F_{col}(1 s)上提升了9%,在F_{col}(2 s)上提升了2%.

为评估多模态信息表征的有效性,表6展示了逐模块消融的实验对比结果. 结果表明,当缺失任一场景向量时,对应场景的关键性能指标均显著衰退.

表6　多模态信息表征消融实验结果

多模态信息表征	$E_{\text{ahead}}\downarrow$	$E_{\text{steer}}\downarrow$	$A_{\text{tl}}\uparrow$	$E_{\text{tl}}\downarrow$	$E_{\text{agent}}\downarrow$	F_{col}(1 s)$\downarrow$	F_{col}(2 s)$\downarrow$
缺失车辆状态向量	0.142	0.279	0.921	0.353	0.879	0.62	0.87
缺失行人状态向量	0.095	0.107	0.919	0.360	0.501	0.47	0.64
缺失自车状态向量	0.692	0.508	0.908	0.579	0.371	0.44	0.70
缺失信号灯状态向量	0.060	0.019	0.348	0.761	0.365	0.38	0.62
缺失地图点向量	0.064	0.027	0.833	0.497	0.355	0.37	0.60
完整多模态输入	0.053	0.017	0.930	0.363	0.354	0.30	0.59

当缺少任一维度向量输入时,例如,当输入多模态表征缺少自车状态向量时,自车运动平滑性对应的E_{ahead}和E_{steer}会明显上升;当输入多模态表征缺少信号灯状态向量时,交通违规场景对应的A_{tl}和E_{tl}显著增大;当输入多模态表征缺少行人状态向量时,碰撞率对应的F_{col}(1 s)和F_{col}(2 s)显著升高. 这充分证明了本文所提多模态信息表征的有效性和不可替代性,能够避免由于局部信息缺失引发的规划失效.

图2展示了基于鸟瞰图的场景建模可视化结果. 其中环视图像展示了6台车载摄像机捕捉到的不同视角(前方左侧、前方、前方右侧、后方左侧、后方、后方右侧)下的城市复杂路口场景. 上述定性结果证明,在当前场景下,自车能够以预测的油门响应、刹车响应以及方向盘转角通过城市复杂路口,预测的自车轨迹能够与轨迹真值高度重合,二者的L2距离为0.04 m. 这直观表明了本文方法能够在复杂城市环境中准确预测自车的运动响应并规划安全的自车轨迹,具有良好的适应性和鲁棒性.

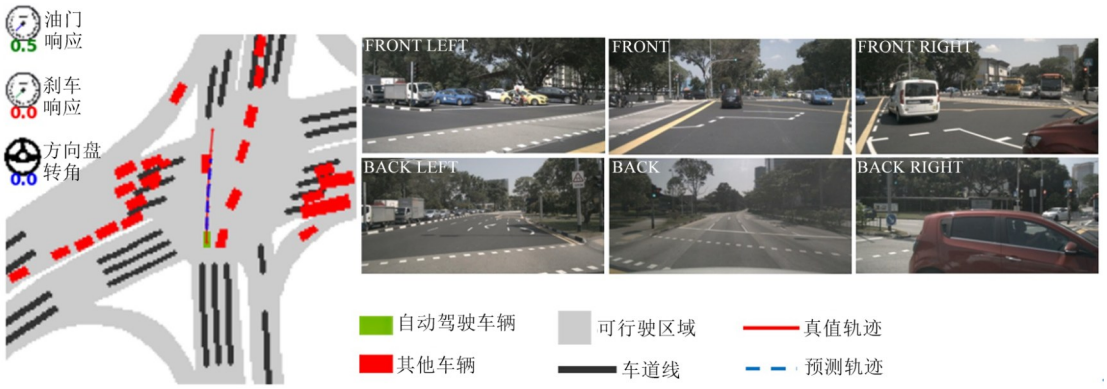

图2　定性实验结果的可视化展示

5　结论

多模态的驾驶场景及时空变化引起的模态间的复杂交互是端到端自动驾驶动作规划任务难以实现的主要挑战. 因此,本文提出基于时空注意力Transformer的自动驾驶运动规划方法. 首先,设计了分阶段的多模态场景编码器,逐过程构建多模态的运动场景描述;然后,设计了时空融合解码器,利用自车及其他智能体在时间与空间位置上的动态变化精准预测自动驾驶车辆的未来安全运动;最后,在大规模自动驾驶数据集nuScenes上搭建了全新的比较基线,并且与5种不同场景下的经典Transformer架构以及其他经典方法进行对比,均取得较为领先的实验结果. 但是,时空融合模型对数据质量和算力要求较高,实车部署难度较大. 同时,系统的实时性仍然面临一定的挑战. 因此,未来研究工作将聚焦于优化模型结构,探索轻量级的时空注

意力机制以降低计算复杂度,以及针对数据的长尾分布进行更加鲁棒的处理,最终实现完全意义上的端到端自动驾驶.

参考文献

[1] GIRGIS R, GOLEMO F, CODEVILLA F, et al. Latent variable sequential set transformers for joint multi-agent motion prediction[EB/OL]. (2022-02-11)[2024-11-10]. https://arXiv.org/abs/2104.00563.

[2] MERCAT J, GILLES T, EL ZOGHBY N, et al. Multi-head attention for multi-modal joint vehicle motion forecasting[C]//2020 IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2020: 9638-9644.

[3] NGIAM J, CAINE B, VASUDEVAN V, et al. Scene Transformer: A unified architecture for predicting multiple agent trajectories[EB/OL]. (2022-03-04)[2024-11-10]. https://arXiv.org/abs/2106.08417.

[4] YUAN Y, WENG X S, OU Y L, et al. AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting[C]//2021 IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE, 2022: 9793-9803.

[5] LIU Y C, ZHANG J H, FANG L J, et al. Multimodal motion prediction with stacked transformers[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2021: 7573-7582.

[6] VARADARAJAN B, HEFNY A, SRIVASTAVA A, et al. MultiPath++ : Efficient information fusion and trajectory aggregation for behavior prediction[C]//2022 International Conference on Robotics and Automation. Piscataway: IEEE, 2022: 7814-7821.

[7] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: ACM, 2017: 6000-6010.

[8] GIRDHAR R, JOÃO CARREIRA J, DOERSCH C, et al. Video action transformer network[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2020: 244-253.

[9] NEIMARK D, BAR O, ZOHAR M, et al. Video transformer network[C]//2021 IEEE/CVF International Conference on Computer Vision Workshops. Piscataway: IEEE, 2021: 3156-3165.

[10] ZOLFAGHARI M, SINGH K, BROX T. ECO: Efficient convolutional network for online video understanding[C]// Computer Vision - ECCV 2018. Cham: Springer, 2018: 713-730.

[11] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[EB/OL]. (2021-06-03)[2024-11-11]. https://arXiv.org/abs/2010.11929.

[12] BALTRUSAITIS T, AHUJA C, MORENCY L P. Multimodal machine learning: A survey and taxonomy[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(2): 423-443.

[13] NGIAM J, KHOSLA A, KIM M, et al. Multimodal deep learning[C]//ICML'11: Proceedings of the 28th International Conference on International Conference on Machine Learning. New York: ACM, 2011: 689-696.

[14] SRIVASTAVA N, SALAKHUTDINOV R R. Multimodal learning with deep boltzmann machines[J]. The Journal of Machine Learning Research, 2014, 15(1): 2949-2980.

[15] GADZICKI K, ASHARI R K, ZETZSCHE C. Multimodal convolutional neural networks for human activity recognition[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE, 2018: 1-6.

[16] WANG W, ZHANG M. Tensor deep learning model for heterogeneous data fusion in Internet of Things[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2020, 4(1): 32-41.

[17] ZHANG J Q, CAO M X, YANG X, et al. E2E-MFD: Towards end-to-end synchronous multimodal fusion detection[EB/OL]. (2024-05-23)[2024-11-10]. https://arXiv.org/abs/2403.09323.

[18] YE T J, JING W, HU C Y, et al. FusionAD: Multi-modality fusion for prediction and planning tasks of autonomous driving[EB/OL]. (2023-08-14)[2024-11-11]. https://arXiv.org/abs/2308.01006.

[19] GUO T. From theory to practice: Advancing multi-robot path planning algorithms and applications[EB/OL]. (2025-06-11)[2025-07-10]. https://arXiv.org/abs/2506.09914.

[20] BAEK S, MOON B, KIM S, et al. PIPE planner: Pathwise information gain with map predictions for indoor robot exploration[EB/OL]. (2025-03-10)[2025-07-10]. https://arXiv.org/abs/2503.07504.

[21] KENDALL A, HAWKE J, JANZ D, et al. Learning to drive in a day[C]//2019 International Conference on Robotics and Automation. Piscataway: IEEE, 2019: 8248-8254.

[22] LIANG X D, WANG T R, YANG L N, et al. CIRL: Controllable imitative reinforcement learning for vision-based self-driving[C]//Computer Vision - ECCV 2018. Cham: Springer, 2018: 604-620.

[23] WANG Z Z, MEGER D. Leveraging world model disentanglement in value-based multi-agent reinforcement learning[EB/OL]. (2023-09-08)[2024-11-10]. https://arXiv.org/abs/2309.04615.

[24] ZHANG D K, LIANG J M, GUO K, et al. CarPlanner: Consistent auto-regressive trajectory planning for large-scale reinforcement learning in autonomous driving[C]// 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2025: 17239-17248.
[25] POLACK P, ALTCHÉ F, D'ANDRÉA-NOVEL B, et al. The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles[C]// 2017 IEEE Intelligent Vehicles Symposium (IV). Piscataway: IEEE, 2017: 812-818.
[26] CAESAR H, BANKITI V, LANG A H, et al. nuScenes: A multimodal dataset for autonomous driving[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2020: 11618-11628.
[27] ZENG W Y, LUO W J, SUO S, et al. End-to-end interpretable neural motion planner[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2020: 8652-8661.
[28] JAEGLE A, BORGEAUD S, ALAYRAC J B, et al. Perceiver IO: A general architecture for structured inputs & outputs[EB/OL]. (2022-03-15) [2024-11-10]. https://arXiv.org/abs/2107.14795.
[29] HU S C, CHEN L, WU P H, et al. ST-P3: End-to-end vision-based autonomous driving viaSpatial-temporal feature learning[C]//Computer Vision - ECCV 2022. Cham: Springer, 2022: 533-549.
[30] CHEN L, SINAVSKI O, HÜNERMANN J, et al. Driving with LLMs: Fusing object-level vector modality for explainable autonomous driving[C]//2024 IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2024: 14093-14100.
[31] HU P Y, HUANG A, DOLAN J, et al. Safe local motion planning with self-supervised freespace forecasting[C]// 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2021: 12727-12736.
[32] CASAS S, SADAT A, URTASUN R. MP3: A unified model to map, perceive, predict and plan[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2021: 14398-14407.
[33] HU Y H, YANG J Z, CHEN L, et al. Planning-oriented autonomous driving[C]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2023: 17853-17862.
[34] TONG W W, SIMA C, WANG T, et al. Scene as occupancy[C]//2023 IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE, 2024: 8372-8381.

作者简介

袁　丁　女,1978年1月出生于河北省承德市. 现为北京航空航天大学宇航学院教授、博士生导师. 主要研究方向为视觉导航、视觉三维重建. 获国家科技进步奖二等奖(2/10)、教育部技术发明一等奖(3/6). 在国内外发表学术论文60余篇.
E-mail: dyuan@buaa.edu.cn

李　源　男,1999年9月出生于甘肃省合作市. 现为航天科技集团第五研究院第510研究所初级工程师. 主要研究方向为计算机视觉、自动驾驶、深空探测技术.
E-mail: gannanly@buaa.edu.cn

孟羽倩　女,2002年4月出生于辽宁省大连市. 现为北京航空航天大学宇航学院博士研究生. 主要研究方向为计算机视觉、图像分析与理解.
E-mail: Myqian@buaa.edu.cn

张　弘　女,1966年12月出生于河北省秦皇岛市. 现为北京航空航天大学宇航学院教授、博士生导师. 主要研究方向为图像理解、目标跟踪. 获国家科技进步奖二等奖(1/10)、教育部技术发明一等奖(1/6)等多项省部级奖励. 在国内外发表学术论文120余篇.
E-mail: dmrzhang@buaa.edu.cn

杨一帆　男,1986年11月出生于湖南省长沙市. 现为北京航空航天大学宇航学院副教授. 主要研究方向为目标识别与跟踪、高性能嵌入式智能硬件设计. 获国家科技进步奖二等奖(3/10)、教育部技术发明一等奖(4/6). 在国内外发表学术论文30余篇.
E-mail: yifanyang@buaa.edu.cn