arXiv:2603.02035v1 [cs.RO] 2 Mar 2026

LAD-Drive：以动作感知扩散 Transformer 桥接语言与轨迹

Fabian Schmidt[1, 2, *], Karol Fedurko[1, *], Markus Enzweiler[1], Abhinav Valada[2]

摘要—尽管多模态大语言模型（MLLMs）为自动驾驶提供了先进的推理能力，但将其离散语义知识转化为连续轨迹仍是一项根本性挑战。现有方法往往依赖单模态规划头，这从本质上限制了其表征多模态驾驶行为的能力。此外，大多数生成式方法常常以独热编码动作为条件，从而丢弃了对复杂场景至关重要的细微导航不确定性。为克服这些局限，我们提出 LAD-Drive，一种在结构上将高层意图与低层空间规划解耦的生成式框架。LAD-Drive 采用动作解码器推断概率元动作分布，从而建立显式信念状态，以保留通常被独热编码丢失的细微意图。该分布与车辆运动学状态融合后，作为动作感知扩散解码器的条件；该解码器利用截断去噪过程，将学习到的运动锚点细化为安全且运动学可行的轨迹。在 LangAuto 基准上的大量评估表明，LAD-Drive 取得了最先进的结果，驾驶评分最高超出有竞争力的基线 59%，同时显著减少了路线偏离和碰撞。我们将在 https://github.com/iis-esslingen/lad-drive 公开发布代码与模型。

I. 引言

端到端自动驾驶领域随着预训练大语言模型（大语言模型）的融入发生了显著转变。早期框架利用大语言模型丰富的世界知识与推理能力，以应对复杂的长尾交通场景。具体而言，开创性方法 [1]–[3] 将连续轨迹规划重构为自回归文本生成任务，直接在离散语言空间中预测空间航点。然而，这种直接转换存在离散语言词元与连续动作空间之间严重的模态不匹配。离散文本分词固有的数值不敏感性不仅经常生成物理上不可行且格式错误的输出，进而触发下游解析

失败 [4], , 而且还会产生不稳定的预测，严重损害长期规划 [5]。

为避免这些问题，许多方法采用专用回归头，直接将视觉语言特征映射为

* 同等贡献。[1] 智能系统研究所，埃斯林根应用科学大学，德国。[2] 计算机科学系，弗赖堡大学，德国。本研究由BMFTR资助，资助代码为13FH544KB2（ADRIVE-GPT）。论文内容由作者独立负责。作者感谢设于埃斯林根大学、并由MWK在德国研究联合会“Großger¨ate der L¨ander”计划下共同资助的DACHS数据分析集群，为本研究提供了所需的计算资源。

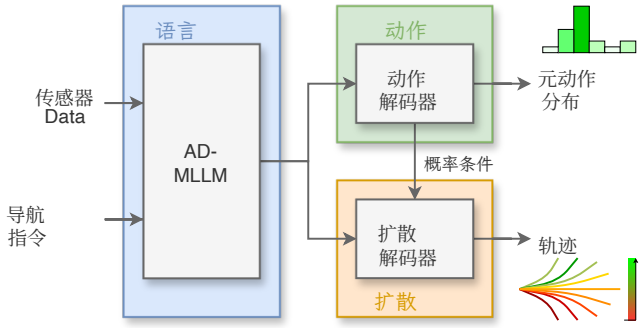


图 1. LAD-Drive 通过三个模块将高层语义推理与低层空间规划解耦。首先，语言模块利用多模态大语言模型（AD-MLLM），将传感器数据和导航指令合成为上下文化隐藏状态。其次，动作解码器直接根据这些状态推断离散元动作分布。最后，扩散解码器利用这一概率条件，迭代地将带噪先验细化为运动学一致的多模态轨迹。

连续轨迹 [6]–[8]。虽然计算效率较高，但这些耦合架构面临两项关键局限。从架构层面看，强迫单个网络头同时处理高层语义意图（做什么）和低层空间规划（如何执行），会造成严重的纠缠，可能超出模型容量。从数学层面看，回归目标（如 L1 或 L2 损失）从根本上无法有效处理驾驶行为的多模态特性。它们必然会对彼此不同但同样合理的行为取平均（例如，为避开障碍物而左转或右转）。这种模式平均可能导致物理上不可行的预测，使模型在复杂场景中的泛化能力较差 [9]。

为捕捉多模态驾驶行为，近期研究探索了两种主要的架构范式 [10]。第一种方法将轨迹预测重新定义为分类任务，通过把连续动作空间离散化为由专门动作词元组成的有限词表 [11], [12]。虽然这种方法通过迫使模型输出类别分布来避免模式平均，但其固有地引入了量化误差，从而损害平滑的低层规划 [5]。第二种范式转向生成式建模。通过利用扩散模型显式学习未来运动的底层分布，该方法能够灵活地进行轨迹预测，并有效捕捉人类驾驶的复杂多模态性 [9]。

尽管扩散模型擅长生成多样且运动学上合理的路径 [13], [14], , 但如何有效地以大语言模型的语义推理对这些规划器进行条件化，仍是一个关键瓶颈。正如我们的分析所示，将

原始的大语言模型隐藏状态直接输入规划器会引入语义噪声和因果混淆。相反，试图简化这种条件化会引入新的瓶颈。僵化的独热编码动作 [15]–[17] 默认采用赢者通吃范式，从而丢弃关键的不确定性，而不透明的隐式规划标记 [18] 则掩盖了智能体的语义意图，使其无法显式建模或解释底层的概率信念状态。

为弥合这一意图—行动鸿沟，我们提出 LAD-Drive（**语言与动作感知扩散驾驶**），这是一个端到端框架，可在结构上将语义意图与几何执行解耦。如图 1 所示，LAD-Drive 使用动作解码器 推断概率元动作分布，从而形成显式信念状态，保留关键的不确定性。为避免执行这一意图时产生因果混淆，扩散解码器采用动作感知设计，将生成过程同时建立在该信念状态和蒸馏后的大语言模型特征之上。通过在截断扩散调度中将聚类锚点用作轨迹先验，LAD-Drive 仅需两个去噪步骤，便能高效地优化出安全且满足运动学约束的路径。

总而言之，我们的核心贡献在于通过显式概率条件实现语义推理与空间规划的结构解耦，并设计了一种高效的动作感知扩散解码器，用于生成多模态轨迹。广泛的评估和系统的消融研究验证了这一范式，证实了双重条件策略的必要性，并表明 LAD-Drive 在 LangAuto 基准上达到了最先进的性能 [6]。我们已在 https://github. com/iis-esslingen/lad-drive 上公开代码和训练好的模型。

II. 相关工作

A. 端到端自动驾驶

端到端自动驾驶范式已从简单的回归基线显著演进为复杂的生成式框架。UniAD [19] 开创了统一方法，将感知、预测和规划整合到基于查询的 Transformer 架构中，从而实现有效的多任务交互。为更好地应对驾驶行为固有的不确定性，VADv2 [11] 引入了概率规划，将连续动作空间离散化为轨迹词汇表，以显式建模未来动作的分布。

在需要连续且多模态输出的背景下，近期研究越来越多地采用扩散模型，并特别关注优化推理效率和引导方式 [9], [20]–[23]。DiffusionPlanner [20] 将规划构建为条件生成任务，利用扩散 Transformer 结合分类器引导，对自车和周围智能体进行联合建模。为解决此类模型通常存在的推理延迟问题，DiffusionDrive [21] 提出了一种截断扩散策略，从学习得到的锚点而非随机噪声开始去噪过程。GoalFlow [22]则通过引入目标驱动流匹配进一步优化生成过程，以估计目标点为条件来生成轨迹，从而确保一致性和效率。DiffRefiner [9] 采用由粗到细范式，利用扩散细化器通过强制实现细粒度语义一致性，迭代改进初始轨迹候选。最近，BridgeDrive [23] 通过将规划构建为连接当前状态与目标状态的对称扩散桥接过程，以生成符合物理约束的轨迹，对以往基于锚点的方法在理论上的不对称性提出了质疑。

B. 基于大语言模型的自动驾驶

大语言模型在自动驾驶中的应用，已从简化环境中的高层推理 [24] 发展到复杂仿真中的闭环轨迹规划 [6]–[8], [15], [16], [25]–[28]。LMDrive [6],率先实现了向真实场景闭环控制的转变，将多模态传感器数据与自然语言指令相结合。后续研究则致力于优化这一基础架构。在此架构上，BEVDriver [7]改进了感知模块，生成更全面的 BEV表征，从而优化空间感知与语义对齐。SimLingo [8] 在纯视觉框架中实现语言-动作对齐，以确保语言推理与执行的机动动作保持一致。针对计算效率问题，AdaDrive [25] 采用自适应快慢系统来控制大语言模型激活，而 VLDrive [26]则通过使用轻量大语言模型和动态视觉剪枝来降低开销。同样，SToRM [27] 通过引入监督式令牌缩减机制，降低多模态大语言模型的高计算开销，从而提升驾驶性能。为弥合抽象推理与精确规划之间的差距，AD-H [15]和 Senna [16] 等分层框架利用多模态大语言模型推断离散的高层元动作，再以此为专用低层规划器提供条件，从而进行离散轨迹预测。

为克服基于回归的规划器在单模态方面的局限，近期研究转向能够捕捉多模态轨迹分布的生成模型 [4], [17], [18], [29]–[31]。如何有效地对这些模型进行条件控制，是一项主要挑战。ORION [18] 在大语言模型中采用隐式规划词元来引导生成式规划器，而 DiffVLA [17] 则利用多模态大语言模型生成独热编码的横向与纵向指令，供混合稀疏—稠密扩散规划器使用。SGDrive [30] 采用分层方法，以场景—智能体—目标的方式预测世界知识，并将其作为潜在条件。为突破自回归模型的序列限制，dVLM-AD [29] 采用具有双向注意力的离散扩散多模态大语言模型，通过迭代去噪提升可控性，并增强高层推理与低层规划之间的一致性。最后，确保生成的轨迹符合物理规律且安全，仍至关重要。DrivePI [31]将三维占据和流作为时空定位的辅助任务，而 ReCogDrive [4]则整合扩散组相对策略优化，以满足安全与舒适约束。

与依赖不透明的

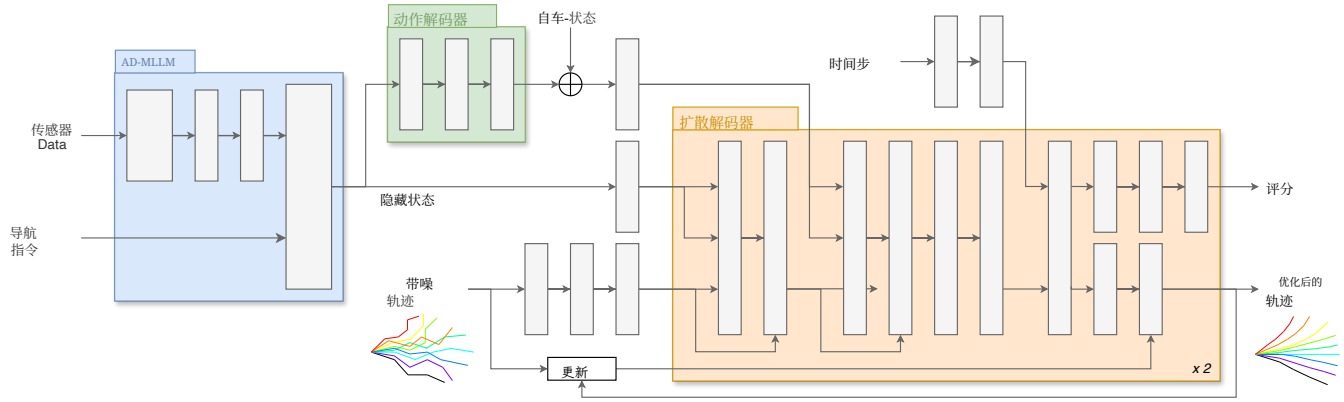


图2. LAD-Drive 的详细架构。该框架利用 AD-MLLM 骨干网络处理多视角图像、激光雷达点云和导航指令，并将其转换为上下文化隐藏状态。专用动作解码器（绿色）从这些状态中推断元动作分布，并将其与车辆的自车状态融合，形成联合状态-意图表征。扩散解码器（橙色）采用共享的基于Transformer的模块，在两个去噪步骤中迭代优化带噪轨迹锚点，每一步的输出都会更新下一步的输入。该细化过程利用连续的多头交叉注意力（MHCA）层，使轨迹生成以全局视觉-语言上下文和预测的概率意图为依据，最终输出优化后的轨迹及其对应的置信度分数。

隐式条件 [18] 或僵化的独热编码 [15]–[17]，LAD-Drive 采用专用动作解码器预测显式的元动作分布，从结构上将高层意图与低层规划解耦。这种概率条件使系统能够有效应对高不确定性场景，并引导连续扩散解码器生成严格遵循语义导航指令的稳健轨迹。

III. 方法

我们将端到端自动驾驶任务表述为学习条件分布 $P(\tau \mid \mathcal{O}, \mathcal{I})$，该分布将多模态传感器观测 $\mathcal{O}$ 和自然语言导航指令 $\mathcal{I}$ 映射为未来轨迹 $\tau = \{w_1, \ldots, w_T\}$，其中 $w_i \in \mathbb{R}^2$ 表示车辆局部坐标系中的航路点。基于多层感知机或门控循环单元的回归方法通过条件期望 $\mathbb{E}[\tau \mid \mathcal{O}, \mathcal{I}]$ 进行近似，但在高度多模态的场景中，由于多条路径都可能成立，这种表述可能会将不同路径平均化，从而产生次优轨迹。为此，我们提出显式保留驾驶场景的多模态性。具体而言，我们引入中间概率信念状态 $\hat{\mathbf{p}} = P(\mathcal{A} \mid \mathcal{O}, \mathcal{I})$，将高层意图与低层执行在结构上解耦，以表示离散动作空间 $\mathcal{A}$ 上的元动作分布。随后，将轨迹生成表述为一个同时以观测和该显式信念状态为条件的扩散过程：

$$P(\tau \mid \mathcal{O}, \mathcal{I}) \approx P(\tau \mid \hat{\mathbf{p}}, \mathcal{O}, \mathcal{I}) \tag{1}$$

为实现这一表述，我们的模型采用专为自动驾驶设计的多模态大语言模型（AD-MLLM）作为核心语义骨干网络，沿用 [6]的工作。该架构利用大语言模型处理多模态上下文，融合传感器输入与自然语言输入。具体而言，视觉编码器首先将多视角摄像头和LiDAR特征转换为统一的鸟瞰视图（BEV）表示。随后，模型利用Q-Former [32], 将这些高维BEV特征压缩为一组固定数量的可学习场景令牌，并将其投影到语言空间中。

通过将场景令牌和导航指令视为因果序列，大语言模型预测出具有上下文感知能力的隐藏状态潜在表示。为构建显式的信念状态，专用动作解码器直接根据这些状态推断元动作分布。这些输出与车辆的自车状态相结合，作为概率条件信号输入下游扩散解码器；该解码器通过迭代将带噪轨迹优化为运动学一致且以感知场景为语义依据的精化轨迹。该架构如图2所示。

A. 动作解码器

动作解码器根据大语言模型的上下文化隐藏状态 $\mathbf{h}_{\text{LLM}} \in \mathbb{R}^{4096}$ 推断高层驾驶动作。通过对该模块进行显式监督，我们确保指令与动作对齐，从而弥合语义意图与几何执行之间的差距，进而减少含糊不清或因果关系不一致的机动动作。此外，输出离散且易于人类理解的动作，显著提升了模型行为的整体可解释性。遵循 [18], 中定义的横向动作空间，我们将离散横向机动动作集合定义为 $\mathcal{A}_{\text{lat}}$，包含六类：车道跟随、直行、左转、右转、向左变道和向右变道。如图2所示，解码器由一个多层前馈神经网络（FFN）构成，通过末端 Softmax 层输出概率分布 $\hat{\mathbf{p}}_{\text{lat}} \in \mathbb{R}^{|\mathcal{A}_{\text{lat}}|}$。

虽然我们的架构旨在支持解耦的横向和纵向头，但实证分析表明，仅以横向意图作为扩散过程的条件即可取得更优性能。我们推测，这是因为纵向动力学主要受车辆自车状态和即时空间约束支配，而高层

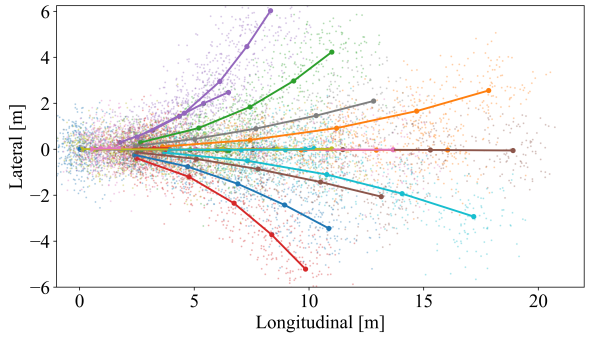


图 3. 通过对训练轨迹进行 k均值聚类生成的锚点轨迹，为截断扩散过程的初始化提供运动先验。

纵向头会引入冗余的语义噪声。因此，在我们的主要配置中，概率信念状态 $\hat{\mathbf{p}}$ 完全由横向动作分布 $\hat{\mathbf{p}}_{\text{lat}}$构成。这提供了一种聚焦的条件信号，能够消除导航歧义，同时不损害低层规划的稳定性。

B. 扩散解码器

对于低层规划，我们提出了一种基于 [21] 中引入的截断扩散策略的生成式扩散解码器。不同于利用显式空间表征（如 BEV地图）的现有方法，我们的解码器直接作用于上下文化隐藏状态 $\mathbf{h}_{\text{LLM}}$。由于这些嵌入包含了对驾驶场景、历史上下文和导航指令丰富且紧凑的表征，生成过程可以灵活地以这些语义知识为条件，从而自然保留驾驶环境的多模态性。

a) 锚点编码：扩散解码器需要一组锚点轨迹 $\mathbf{A} = \{\mathbf{a}_i \in \mathbb{R}^{T\times 2}\}_{i=1}^{N_a}$，其中$T = 5$ 航路点和 $N_a = 20$锚点。以往工作 [21] 使用从 Navsim 等数据集中提取的静态锚点 [33], ，而我们则提出对锚点进行域对齐，以充分捕捉运行域的内在属性。因此，如图 3 所示，我们直接对目标训练分布中的专家轨迹应用 k均值聚类，生成这组经过域对齐的锚点。这种数据驱动的初始化确保运动先验能够准确反映评估环境特有的运动学约束和速度剖面，使模型能够以更低的规划复杂度优化出可行路径。为嵌入这些运动先验，一个两层多层感知机编码器将它们映射到共享潜在空间，得到$\mathbf{z}_a \in \mathbb{R}^{N_a\times d}$，其中 $d = 512$。

b) 特征瓶颈：尽管ORION [18] 等近期方法直接将原始的高维大语言模型隐藏状态用作“规划令牌”来条件化轨迹预测，但它们报告了将这些推理特征与基于扩散的动作空间对齐时所面临的重大挑战。我们的实证分析证实了这一点。我们观察到，将完整的4096维大语言模型表征直接耦合到扩散解码器，会导致严重退化的驾驶性能（详见第IV节）。我们将此归因于严重的表征失配与维度过载。由于原始的大语言模型嵌入主要针对自回归语言预测进行了高度优化，它们封装了宽泛而抽象的世界知识，而这些知识对于低层空间规划而言是多余的。将这种高维语义噪声直接注入去噪过程，会压倒轨迹细化。

为了解决这一对齐问题并稳定生成过程，我们引入一种结构化特征瓶颈，通过两层多层感知机将大语言模型的隐藏状态$\mathbf{h}_{\text{LLM}}$ 映射到紧凑的 d 维潜在空间中，得到$\mathbf{h}'_{\text{LLM}} \in \mathbb{R}^d$。如图2所示，该投影基于信息瓶颈原理充当语义过滤器，确保扩散解码器仅处理与导航意图和空间上下文最相关的特征。

c) 状态-意图条件化：为了让轨迹解码有据可依，我们引入联合状态-意图表征。虽然我们通过以连续的概率信念状态 $\hat{\mathbf{p}}$为条件，明确保留导航不确定性，但仅有高层语义意图不足以支持低层规划。为确保生成路径在运动学上可行，轨迹表述必须严格锚定于车辆当前的物理动力学。因此，如图2所示，我们将信念状态 $\hat{\mathbf{p}}$ 与车辆的自车状态$\mathbf{e}$ （当前速度和偏航角）拼接起来，形成统一的条件向量$\mathbf{c} = \mathbf{e} \oplus \hat{\mathbf{p}}$。这种融合明确地将车辆打算做什么与它当前如何运动联系起来，提供全面的运动学语义先验。最后，通过两层多层感知机将这个联合向量映射到共享潜在空间，得到$\mathbf{z}_{\text{si}} \in \mathbb{R}^d$。

d) 迭代细化：在建立了必要的几何、语义和运动学表征后，扩散解码器开始迭代生成。遵循截断扩散范式 [21],，解码器对以高斯噪声初始化的 K-means 锚点执行$N = 2$ 次细化步骤。由于从这些噪声先验而非纯噪声开始，渐进式去噪过程能够高效地将候选轨迹调整为适应场景且满足运动学约束的路径。

为确保生成的轨迹能够感知上下文，锚点嵌入$\mathbf{z}_a$ 首先通过多头交叉注意力（MHCA）层关注投影后的大语言模型隐藏状态$\mathbf{h}'_{\text{LLM}}$ ，其形式为$\mathbf{h}_t = \text{MHCA}(\mathbf{z}_a, \mathbf{h}'_{\text{LLM}}, \mathbf{h}'_{\text{LLM}})$。该模块作为上下文注意力机制，将轨迹置于全局视觉-语言上下文中，使规划器能够聚焦于感知到的场景布局和给定的导航指令。随后，第二个 MHCA 层整合状态-意图表征$\mathbf{z}_{\text{si}}$：$\mathbf{h}_t = \text{MHCA}(\mathbf{h}_t, \mathbf{z}_{\text{si}}, \mathbf{z}_{\text{si}})$。此步骤作为引导注意力机制，明确依据预测意图和物理状态对细化过程进行条件约束。这确保生成路径遵循动作解码器作出的高层决策，同时严格遵守车辆的动力学限制。通过采用这些顺序条件约束步骤来构建扩散过程，模型有效地弥合了高层

语义规划与低层执行之间的脱节。

融合特征$\mathbf{h}_t$ 随后由前馈神经网络处理，并由时间步嵌入层调制，该层根据当前扩散步通过缩放和平移参数动态调整潜在特征。最后，解码器分支为两个专用多层感知机头：航点头预测偏移量，通过残差连接迭代形变初始带噪锚点，分类头则生成相应的轨迹置信度分数。

C. 训练策略

为训练扩散解码器，我们采用来自 [21]的截断去噪调度。扩散解码器 f_{dec}处理带噪锚点$\mathbf{Y}^{\text{noisy}}$ 以及我们显式定义的条件表示，以预测 N_a 精化轨迹$\hat{}\mathbf{Y} = \{\hat{}\mathbf{y}_i\}_{i=1}^{N_a}$ 及其置信度分数$\hat{}\mathbf{s} = \{s_i\}_{i=1}^{N_a}$：

$$\hat{\mathbf{Y}}, \hat{\mathbf{s}} = f_{\text{dec}}(\mathbf{Y}_{\text{noisy}}, \mathbf{h}'_{\text{LLM}}, \mathbf{z}_{\text{si}}). \tag{2}$$

为在不引发模式坍缩的情况下监督预测的多模态性，我们采用最近锚点匹配策略。与真实轨迹$\mathbf{y}_{\text{GT}}$ 距离最小的锚点被赋予正标签 $s_i = 1$，其余锚点则标记为 $s_i = 0$。规划目标将作用于所有锚点的二元交叉熵（BCE）分类损失与仅应用于正锚点$\hat{}\mathbf{y}^{\text{pos}}$的L1回归损失（$\mathcal{L}_{\text{wp}}$）相结合：

$$\mathcal{L}_{\text{plan}} = \lambda_{\text{wp}}\mathcal{L}_{\text{wp}}(\hat{\mathbf{y}}_{\text{pos}}, \mathbf{y}_{\text{GT}}) + \lambda_{\text{cls}} \sum_{i=1}^{N_a} \text{BCE}(\hat{s}_i, s_i), \tag{3}$$

其中， $\lambda_{\text{wp}} = 8$ 和 $\lambda_{\text{cls}} = 10$ 是用于平衡这两个目标的超参数。

最终的训练目标包含标准交叉熵（CE）损失，以显式监督动作解码器。使用独热编码的真实动作 $\mathbf{a}_{\text{GT}}$，总损失表示为：

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{plan}} + \lambda_{\text{action}}\text{CE}(\hat{\mathbf{p}}, \mathbf{a}_{\text{GT}}). \tag{4}$$

我们采用两阶段训练策略，将空间运动的学习与语义推理解耦。

a) 空间落地： 在初始阶段，动作解码器的损失贡献会被屏蔽（$\lambda_{\text{action}} = 0$）。主要目标是训练 Q-Former [32] 和扩散解码器，使其生成物理上有效且与视觉场景严格一致的路径。通过将扩散过程显式地落地于提取出的场景令牌，模型能够在进行任何复杂的语义对齐之前，为运动学可行的轨迹生成和空间感知奠定稳健的基础。

b) 语义对齐： 在第二阶段，动作解码器解除掩码（$\lambda_{\text{action}} = 1$），并参与总损失的计算。此阶段旨在促使模型在优化后的轨迹与其对应的高层语义动作之间建立潜在映射。具体而言，扩散解码器学会根据概率信念状态 $\hat{}\mathbf{p}$调整其空间优化过程，从而使车辆生成的机动动作与高层意图相一致。

IV. 实验

A. 基准与指标

为了在具有挑战性的指令跟随场景中评估我们的方法，我们在 CARLA 模拟器（ [34]）中使用 LangAuto 基准（ [6]）。鉴于我们的方法专为处理自然语言指令而设计，而非离散目标点，因此我们采用 LangAuto 作为目前唯一支持此类引导的闭环评估框架。该基准要求智能体根据路线长度，在八个不同的城镇中完成三种赛道：极短（$< 150m$）、短 （$150 - 500m$）和长 （$> 500m$）。我们报告 CARLA 的标准指标：路线完成度（RC），表示已行驶距离的百分比；违规评分（IS），作为针对碰撞和交通违规的乘法安全惩罚；以及主要综合指标驾驶评分（DS），其定义为 RC 与 IS 的乘积。按照 LangAuto协议，所有报告结果均为三次独立评估运行的平均值。

B. 实现细节

1) 智能体与控制器优化：除了标准基线评估外，我们还针对 LMDrive 中两个关键的工程瓶颈进行了改进，显著提升了其基线性能，并打造出极具竞争力的基准。首先，为防止长时域内过时的时间噪声不断累积并产生分布漂移，我们移除了基线中用于基于指令的历史重置的预测头，并在推理期间强制采用严格的 40 帧滑动窗口，使其与训练分布的感受野相匹配。其次，我们发现，官方 LMDrive 检查点在低层横向执行方面存在明显的校准不足。根据 [7], 的见解，我们通过优化 PID 控制器系数来稳定横向动态：将比例增益降至 1.2，同时分别将积分增益和微分增益提高至 0.8 和 0.35。

2）训练配置：我们使用官方 LMDrive 检查点 [6], 初始化 AD-MLLM，并冻结 LLaVA-v1.5 [35] 骨干网络和视觉编码器，以保留预训练语义知识；仅对 Q-Former 进行微调，并从头开始训练所提出的动作解码器和扩散解码器。我们以高效的截断去噪策略取代标准回归头，扩散解码器从 $N_a = 20$ 聚类锚点初始化，在训练和推理过程中均仅执行两步去噪。模型在 GraphPilot 数据集 [36]的一个子集上训练。我们使用原始传感器数据和导航指令，但有意省略场景图标注，以确保训练方案与原始 LMDrive 基线保持可比。在推理过程中，我们选择得分最高的轨迹，以引导纵向和横向 PID 控制器生成离散控制命令。我们使用八块 NVIDIA L40s GPU，全局批量大小为 16，权重衰减为 0.1，并采用线性预热余弦学习率调度：学习率从 5×10^{-7}开始，升至 2×10^{-5}后再衰减至 1×10^{-6}。训练共进行六轮，平均分配到所提出的两个阶段，整个训练约需七小时。

表 I LangAuto 性能。最佳结果以粗体突出显示。

方法	发表 venue	Tiny			短			Long			Mean		
		DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑
LMDrive [6]（报告值）	[illegible]	66.5	77.9	0.85	50.6	60.0	0.84	36.2	46.6	0.81	51.1	61.5	0.83
LMDrive [6]（检查点）	[illegible]	60.7	71.0	0.82	41.3	57.0	0.79	26.8	36.5	0.77	42.9	54.8	0.79
AD-H [15]	2024年arXiv	77.5	85.1	0.91	56.1	68.0	0.78	44.0	53.2	0.83	59.2	68.8	0.84
BEVDriver [7]	IROS'25	70.2	81.3	0.87	66.7	77.8	0.87	48.9	**59.7**	0.82	61.9	72.9	0.85
SToRM [27]	arXiv'26	78.8	86.9	0.92	64.5	74.7	0.88	44.2	56.8	0.82	62.5	72.8	0.87
VLDrive [26]	ICCV'25	81.9	85.5	0.94	67.4	78.1	0.85	43.8	54.5	0.84	64.4	72.7	0.88
AdaDrive [25]	ICCV'25	80.9	**87.6**	0.90	70.6	**85.3**	0.81	42.9	53.4	0.82	64.8	**75.4**	0.84
LAD-Drive（我们的方法）		**83.5**	87.0	**0.95**	**71.3**	78.1	**0.89**	**49.8**	58.3	**0.86**	**68.2**	74.5	**0.90**

表 II 违规与效率分析详情，包括模型总延迟平均值 [MS] 和解码器参数量 [M]。

方法	DS ↑	RC ↑	IS ↑	CP ↓	CV ↓	CL ↓	RL ↓	SS ↓	关闭 ↓	RD ↓	TO ↓	AB ↓	DP ↓	延迟 ↓
LMDrive	42.9	54.8	0.80	0.08	2.83	4.28	2.31	**0.00**	3.79	11.95	2.93	**1.63**	33.61	**40.15**
LAD-Drive（我们的方法）	**68.2**	**74.5**	**0.90**	**0.02**	**0.67**	**1.20**	**1.60**	0.11	**1.10**	**2.31**	**2.68**	2.48	**32.13**	47.04

CP：行人碰撞，CV：车辆碰撞，CL：布局碰撞，RL：红灯，SS：停车标志，Off：驶出道路，RD：路线偏离，TO：超时，AB：智能体阻塞，DP：解码器参数量。延迟在单块 NVIDIA L40s GPU 上测量。

C. 结果

表 I 展示了我们的方法与基线 LMDrive [6] 以及当前最先进方法的比较结果。LAD-Drive 取得了新的最先进性能，平均驾驶分数（DS）达到 68.2，大幅超越 LMDrive 基线（42.9），提升了 +59%，并优于 AdaDrive（64.8）和 VLDrive（64.4）等近期方法。

我们将相较于基线在舒适性（RC）方面取得的显著提升归因于意图与执行之间的结构解耦。基线的 RC 值显著较低，在短和 长 基准中尤为明显，这表明其指令遵循能力存在根本局限，常导致路线偏离并进而造成回合提前终止。相比之下，我们的方法缓解了这些问题，能够在较长时间范围内持续稳健地遵循导航目标。此外，即便与 BEVDriver、VLDrive 和 AdaDrive 等 RC 水平相当的最先进方法相比，LAD-Drive 的交互安全性（IS）仍始终更高。这一差异验证了解耦架构的有效性。通过显式地以预测得到的元动作分布为扩散过程提供条件，我们的概率动作条件化能够有效规范驾驶行为，确保驾驶进程不会以牺牲安全为代价。

与物理控制相关的违规行为，例如碰撞布局（CL）和驶出道路（Off）事件，分别减少了约 72% 和 71%。如图 4 中间列所示，这证实了我们的扩散解码器生成的轨迹不仅在语义上保持一致，而且在物理上也符合道路边界。路线偏离（RD）指标显著改善，从 11.95 降至 2.31。如图 4 左侧列所示，这一降幅主要归功于有效的概率元动作条件化，使扩散解码器始终严格与预期的导航目标保持一致。相比之下，标准的基于回归的基线通常计算负担过重，因为它们试图在单个头中同时处理高层语义意图和低层空间控制。这种双重负担经常导致错误的横向决策，例如在交叉路口错误转向，进而使智能体过早终止回合。通过将这些任务解耦，LAD-Drive 有效地规范了轨迹生成过程，从而避免导航失败。这一架构改进在实现显著性能跃升的同时，与基线相比还将解码器规模减少了 1.48M 个参数。尽管两步去噪过程使推理开销略有增加（+6.89ms），该模型仍保持了极具竞争力的 47.04ms 总实时延迟。

D. 消融实验

为进一步探究这些改进的来源，表 II 详细列出了违规类型。与官方 LMDrive检查点相比，我们的方法大幅降低了与动态交通参与者的碰撞率。与车辆碰撞（CV）下降了 76%（从 2.83 降至 0.67），与行人碰撞（CP）下降了 80%（从 0.08 降至 0.02）。图 4 右侧各列直观展示了安全性的提升。同样重要的是空间落地能力的改善。违规行为

为验证各个架构组件的作用，我们在表 III 中进行了系统的消融研究。初步结果表明，对基础模型进行微调并优化智能体（第 2 行），相较于原始检查点实现了显著的性能提升（驾驶分数从 42.9 提升至 60.6）。这为后续实验建立了一个强有力的回归基线，确保后续收益源于架构创新，而非训练方案或参数改进。仅将回归

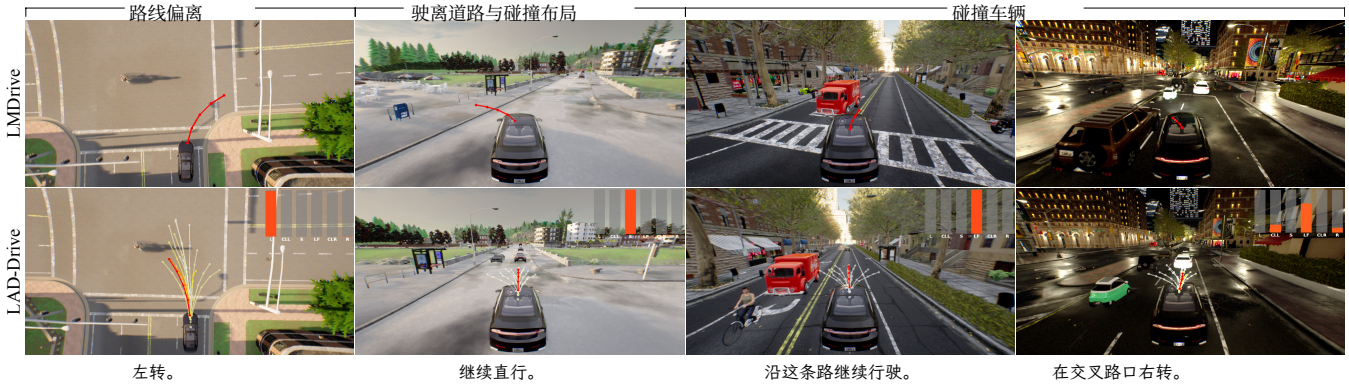


图 4. LMDrive（上排）与 LAD-Drive（下排）的定性比较。对于 LAD-Drive，生成的多模态轨迹以热力图配色方案进行可视化，路径颜色从白色（低置信度）到红色（高置信度）不等。每个 LAD-Drive 面板的右上角显示动作解码器预测的元动作分布（L：左转，CLL：向左变道，S：直行，LF：车道跟随，CLR：向右变道，R：右转）。第 1 列（路线偏离）：尽管指令要求左转，LMDrive 却向右转。LAD-Drive 则执行了正确的意图，同时安全避让骑行者。第 2 列（驶离道路与碰撞布局）：LMDrive 未能理解道路布局和指令，驶离道路并撞上围栏；而 LAD-Drive 成功继续直行。第 3 和第 4 列（碰撞车辆）：在第三种情形中，LMDrive 与一辆卡车迎面相撞，而 LAD-Drive 安全地沿车道行驶。在第四种情形中，LMDrive 无视右转导航指令，开始向左变道并驶入交通流，造成危险。相比之下，LAD-Drive 能有效应对动态交通参与者，并始终安全地行驶在道路上；即使动作解码器存在轻微的不确定性，也展现出良好的鲁棒性。

表 III 模型组件消融研究。

轨迹头	微调与优化	横向动作	纵向动作	Ego	DS ↑	RC ↑	IS ↑
基线					42.9	54.8	0.79
基线	✓				60.6	66.1	**0.91**
扩散	✓				59.4	66.0	0.89
扩散	✓	✓			60.0	68.1	0.88
扩散	✓			✓	64.8	72.2	0.88
扩散	✓	✓		✓	**68.2**	**74.5**	0.90
扩散	✓	✓	✓	✓	65.9	72.4	0.90

FT & Opt.：微调与优化，Lat. Act.：横向动作，Long. Act.：纵向动作，Ego：自车状态。

表 IV 扩散解码器潜在维度的消融研究。

潜在维度	DS ↑	RC ↑	IS ↑
128	63.6	72.3	0.87
256	66.9	73.9	0.90
512	**68.2**	**74.5**	0.90
1024	63.2	68.1	0.91
4096	48.5	52.4	**0.92**

潜在维度：潜在维度。

头替换为扩散解码器（第 3 行）会导致性能略有下降（驾驶分数从 60.6 降至 59.4）。这证实了，尽管生成模型具备多模态能力，但在缺乏充分状态引导的情况下，其本质上并不稳定，往往会生成多样但在物理上不一致的轨迹。当我们单独考察条件机制时，可以观察到一个关键的权衡。仅集成横向动作解码器（第 4 行）能够为“做什么”提供语义引导，使整体性能略有提升（驾驶分数为 60.0）。然而，由于缺乏物理状态感知，模型无法在时间维度上持续一致地执行这些意图。相比之下，仅依赖自车状态 grounding（第 5 行）则通过确保时间一致性，显著提升了物理稳定性（驾驶分数为 64.8）；但由于缺少显式的动作预测，模型难以解决复杂场景中的导航歧义。组合配置（第 6 行）取得了最高性能（驾驶分数为 68.2），验证了二者协同作用的必要性。高层横向动作引导能够消除语义歧义，而低层自车 grounding 则确保物理可行性。最后，引入纵向动作头（第 7 行）反而导致性能下降（驾驶分数为 65.9）。这一实证结果印证了我们的假设：由于纵向动力学主要由车辆的自车状态和即时空间约束决定，强行让扩散过程也以高层纵向指令为条件会引入冗余的语义噪声，从而干扰细化过程。

表 IV 详细说明了扩散解码器潜在维度大小的影响。我们观察到，在$d = 512$处性能达到明显峰值，该设置在特征压缩与表征能力之间实现了最佳平衡，从而取得最高性能（驾驶分数 68.2）。过度限制瓶颈（$d = 128$）会削弱模型捕捉复杂场景动力学的能力，而将其扩展到 512 以上则会降低导航精度。值得注意的是，绕过瓶颈、直接连接原始的 4096 维大语言模型嵌入，会导致性能严重崩溃（驾驶分数 48.5）。这一实证下降明确证实了我们的结构性假设：在缺乏语义过滤器的情况下，模型会遭受严重的表征失配和维度过载。

五、结论

我们提出了 LAD-Drive，以弥合高层语言推理与连续轨迹规划之间的模态鸿沟。LAD-Drive 采用动作解码器来推断概率元动作分布，从结构上将语义意图与空间执行解耦。这形成了一种显式信念状态，保留了关键的导航不确定性。将该分布与车辆的运动学自车状态融合后，

该分布为动作感知扩散解码器提供稳健的概率条件。结合结构特征瓶颈以过滤来自大语言模型的高维语义噪声，我们的框架能够高效地将抽象推理转化为安全、可执行的轨迹。在 LangAuto 基准上的广泛评估验证了我们的方法。LAD-Drive 实现了最先进的性能，驾驶评分比基线高出 59%。与此同时，该进展还显著提升了安全性和指令遵循能力，车辆碰撞和布局碰撞分别减少了 76% 和 72%，路线偏离则减少了 81%。

参考文献

[1] J. Mao、Y. Qian、J. Ye、H. Zhao 和 Y. Wang，“Gpt-driver：使用 GPT 学习驾驶”，arXiv 预印本 arXiv:2310.01415，2023 年。[2] Z. Xu、Y. Zhang、E. Xie、Z. Zhao、Y. Guo、K.-Y. K. Wong、Z. Li 和 H. Zhao，“Drivegpt4：通过大语言模型实现可解释的端到端自动驾驶”，IEEE Robotics and Automation Letters，第 9 卷，第 10 期，第 8186–8193 页，2024 年。[3] X. Tian、J. Gu、B. Li、Y. Liu、Y. Wang、Z. Zhao、K. Zhan、P. Jia、X. Lang 和 H. Zhao，“Drivevlm：自动驾驶与大型视觉语言模型的融合”，载于第 8 届机器人学习年会，2024 年。[4] Y. Li、K. Xiong、X. Guo、F. Li、S. Yan、G. Xu、L. Zhou、L. Chen、H. Sun、B. Wang 等，“Recogdrive：用于端到端自动驾驶的强化认知框架”，arXiv 预印本 arXiv:2506 08052，2025 年。[5] C. Xie、B. Sun、T. Li、J. Wu、Z. Hao、X. Lang 和 H. Li，“Latentvla：通过潜在动作预测实现高效的自动驾驶视觉语言模型”，arXiv 预印本 arXiv:2601.05611，2026 年。[6] H. Shao、Y. Hu、L. Wang、G. Song、S. L. Waslander、Y. Liu 和 H. Li，“LMDrive：使用大语言模型进行闭环端到端驾驶”，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2024 年，第 15 120–15 130 页。[7] K. Winter、M. Azer 和 F. B. Flohr，“BEVDriver：利用大语言模型中的 BEV地图实现稳健的闭环驾驶”，载于 IEEE/RSJ 智能机器人与系统国际会议，2025 年。[8] K. Renz、L. Chen、E. Arani 和 O. Sinavski，“SimLingo：通过语言-动作对齐实现仅视觉闭环自动驾驶”，载于计算机视觉与模式识别会议论文集，2025 年，第 11 993–12 003 页。[9] L. Yin、R. Ju、G. Guo 和 E. Cheng，“DiffRefiner：通过结合语义交互的扩散细化实现从粗到细的轨迹规划，用于端到端自动驾驶”，arXiv 预印本 arXiv:2511 17150，2025 年。[10] K. Oksuz、A. Buburuzan、A. Knittel、Y. Yao 和 P. K. Dokania，“自动驾驶中用于轨迹规划的基础模型：进展与开放挑战综述”，arXiv 预印本 arXiv:2512 00021，2025 年。[11] S. Chen、B. Jiang、H. Gao、B. Liao 等，“VADv2：通过概率规划实现端到端向量化自动驾驶”，arXiv 预印本 arXiv:2402.13243，2024 年。[12] Z. Zhou、T. Cai、S. Z. Zhao、Y. Zhang、Z. Huang、B. Zhou 和 J. Ma，“Autovla：结合自适应推理与强化微调的端到端自动驾驶视觉-语言-动作模型”，NeurIPS，2025 年。[13] J. Wei、N. V¨odisch、A. Rehr、C. Feist 和 A. Valada，“Parkdiffusion：使用扩散模型进行自动泊车的异构多智能体多模态轨迹预测”，载于 IEEE/RSJ 智能机器人与系统国际会议，2025 年，第 8297–8304 页。[14] J. Wei、A. Rehr、C. Feist 和 A. Valada，“Parkdiffusion++：基于自车意图条件的联合多智能体轨迹预测，用于使用扩散模型的自动泊车”，arXiv 预印本 arXiv:2602 20923，2026 年。[15] Z. Zhang、S. Tang、Y. Zhang、T. Fu、Y. Wang、Y. Liu、D. Wang、J. Shao、L. Wang 和 H. Lu，“AD-H：通过分层智能体实现自动驾驶”，arXiv 预印本 arXiv:2406.03474，2024 年。[16] B. Jiang、S. Chen、B. Liao、X. Zhang、W. Yin 等，“Senna：连接大型视觉语言模型与端到端自动驾驶”，arXiv 预印本 arXiv:2410.22313，2024 年。

[17] A. Jiang、Y. Gao、Z. Sun、Y. Wang、J. Wang、J. Chai、Q. Cao 等，“DiffVLA：用于自动驾驶的视觉语言引导扩散规划”，arXiv 预印本 arXiv:2505.19381，2025 年。[18] H. Fu、D. Zhang、Z. Zhao、J. Cui、D. Liang、C. Zhang、D. Zhang、H. Xie、B. Wang 和 X. Bai，“ORION：通过视觉语言指令动作生成实现的整体端到端自动驾驶框架”，载于 IEEE/CVF 国际计算机视觉会议论文集，2025 年，第 24 823–24 834 页。[19] Y. Hu、J. Yang、L. Chen、K. Li、C. Sima、X. Zhu、S. Chai、S. Du、T. Lin、W. Wang 等，“面向规划的自动驾驶”，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2023 年，第 17 853–17 862 页。[20] Y. Zheng、R. Liang、K. Zheng、J. Zheng、L. Mao、J. Li、W. Gu、R. Ai、S. E. Li、X. Zhan 等，“具有灵活引导的基于扩散的自动驾驶规划”，载于国际表示学习会议，2025 年。[21] B. Liao、S. Chen、H. Yin、B. Jiang、C. Wang、S. Yan、X. Zhang、X. Li、Y. Zhang、Q. Zhang 等，“DiffusionDrive：用于端到端自动驾驶的截断扩散模型”，载于计算机视觉与模式识别会议论文集，2025 年，第 12 037–12 047 页。[22] Z. Xing、X. Zhang、Y. Hu、B. Jiang、T. He、Q. Zhang、X. Long 和 W. Yin，“GoalFlow：用于端到端自动驾驶中多模态轨迹生成的目标驱动流匹配”，载于计算机视觉与模式识别会议论文集，2025 年。[23] S. Liu、W. Chen、W. Li、Z. Wang、L. Yang、J. Huang、Y. Zhang、Z. Huang、Z. Cheng 和 H. Yang，“BridgeDrive：用于自动驾驶闭环轨迹规划的扩散桥策略”，arXiv 预印本 arXiv:2509.23589，2025 年。[24] D. Fu、X. Li、L. Wen、M. Dou、P. Cai、B. Shi 和 Y. Qiao，“像人一样驾驶：使用大语言模型重新思考自动驾驶”，载于 IEEE/CVF 冬季计算机视觉应用会议研讨会，2024 年，第 910–919 页。[25] R. Zhang、J. Xie、W. Zhang、W. Chen、X. Tan、X. Wan 和 G. Li，“AdaDrive：用于语言基础自动驾驶的自适应慢速-快速系统”，载于 IEEE/CVF 国际计算机视觉会议论文集，2025 年，第 5112–5121 页。[26] R. Zhang、W. Zhang、X. Tan、S. Yang、X. Wan、X. Luo 和 G. Li，“VLDrive：用于高效语言基础自动驾驶的视觉增强轻量级多模态大语言模型”，载于 IEEE/CVF 国际计算机视觉会议论文集，2025 年，第 5923–5933 页。[27] S. H. Kim、J. B. Park、D. Y. Koo、H. G. Park 和 I. Y. Chun，“SToRM：面向高效端到端自动驾驶的多模态大语言模型监督式令牌缩减”，arXiv 预印本 arXiv:2602.11656，2026 年。[28] F. Schmidt、N. M. K. A. Nazar、M. Enzweiler 和 A. Valada，“通过模块化交通灯和交通标志识别增强基于大语言模型的自动驾驶”，arXiv 预印本 arXiv:2511.14391，2025 年。[29] Y. Ma、Y. Cao、W. Ding、S. Zhang、Y. Wang、B. Ivanovic、M. Jiang、M. Pavone 和 C. Xiao，“dVLM-AD：通过可控推理增强用于驾驶的扩散视觉语言模型”，arXiv 预印本 arXiv:2512.04459，2025 年。[30] J. Li、J. Wu、D. Hu、X. Huang、B. Sun、Z. Hao、X. Lang、X. Zhu 和 L. Zhang，“SGDrive：从场景到目标的分层世界认知，用于自动驾驶”，arXiv 预印本 arXiv:2601.05640，2026 年。[31] Z. Liu、R. Huang、R. Yang、S. Yan、Z. Wang、L. Hou、D. Lin、X. Bai 和 H. Zhao，“DrivePI：用于统一自动驾驶理解、感知、预测和规划的空间感知四维多模态大语言模型”，arXiv 预印本 arXiv:2512.12799，2025 年。[32] J. Li、D. Li、S. Savarese 和 S. Hoi，“BLIP-2：使用冻结图像编码器和大语言模型引导语言-图像预训练”，载于国际机器学习会议，2023 年。[33] D. Dauner、M. Hallgarten、T. Li、X. Weng、Z. Huang 等，“Navsim：数据驱动的非反应式自动驾驶车辆仿真与基准测试”，NeurIPS，第 37 卷，第 28 706–28 719 页，2024 年。[34] A. Dosovitskiy、G. Ros、F. Codevilla、A. Lopez 和 V. Koltun，“CARLA：一种开放的城市驾驶模拟器”，载于第 1 届机器人学习年会论文集，2017 年，第 1–16 页。[35] H. Liu、C. Li、Y. Li 和 Y. J. Lee，“通过视觉指令微调改进基线”，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2024 年，第 26 296–26 306 页。[36] F. Schmidt、M. Enzweiler 和 A. Valada，“Graphpilot：基于场景落地的场景图条件控制，用于基于语言的自动驾驶”，arXiv 预印本 arXiv:2511.11266

，2025年。