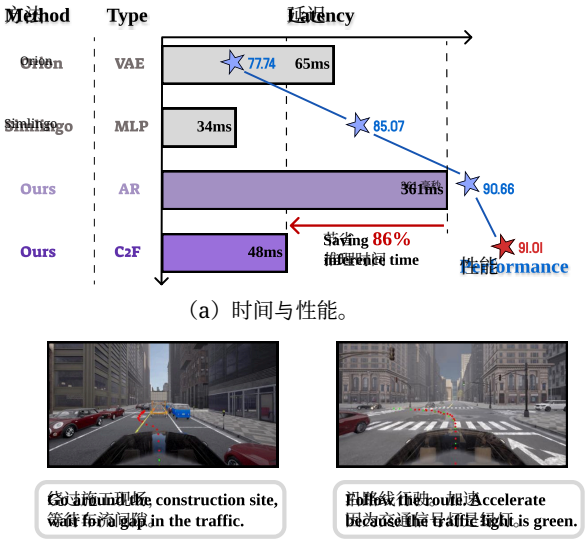
arXiv:2603.01441v1 [cs.CV] 2 Mar 2026

统一自动驾驶中的语言-动作理解与生成

Xinyang Wang[1, 2, *] Qian Liu[2, *] Wenjie Ding[2, *] Zhao Yang[2, †]
Wei Li[2] Chang Liu[2] Bailin Li[2] Kun Zhan[2] Xianpeng Lang[2] Wei Chen[1]
[1]浙江大学计算机辅助设计与图形学国家重点实验室，浙江大学 [2]理想汽车

摘要

视觉-语言-动作（*VLA*）模型正成为端到端自动驾驶的一种有前景的范式，因其有望利用世界知识并对复杂驾驶场景进行推理。然而，现有方法存在两项关键局限：语言指令与动作输出之间持续存在错位，以及典型自回归动作生成本身效率低下。本文提出***LinkVLA***，这是一种旨在直接解决上述挑战、同时提升对齐程度和效率的新型架构。首先，我们将语言和动作词元统一到一个共享离散码本中，并在单个多模态模型内进行处理，从而建立结构联系。这从根本上强制实现跨模态一致性。其次，为建立深层语义联系，我们引入辅助动作理解目标，训练模型根据轨迹生成描述性说明，促进双向语言-动作映射。最后，我们采用两步由粗到细的生成方法（$C2F$）取代缓慢的逐步生成，高效解码动作序列，将86%推理时间。闭环驾驶基准测试的实验结果表明，指令遵循准确率和驾驶性能均持续提升，同时推理延迟有所降低。



（a）时间与性能。

（b）指令遵循能力的可视化。

图 1. LinkVLA 在闭环评估中实现了更高的性能和更低的延迟，并具备出色的指令遵循能力。

1. 引言

端到端学习 [3, 5, 13, 14, 22, 30, 31, 42, 51, 55, 60, 61] 已成为开发自动驾驶系统的主导范式，它从原始传感器输入中学习直接的感知运动控制策略，以用于车辆规划。传统端到端模型虽然擅长在熟悉场景中进行反应式控制，但在复杂推理、长尾事件和人机交互方面往往表现不佳 [41, 46]。视觉语言模型（VLM）[11, 16] 近年来，相关方法因能够利用丰富的世界知识和强大的推理能力而受到广泛关注，并被引入驾驶场景以增强其泛化能力 [2, 6, 23–25, 27, 32–35, 45, 49, 50, 52]。视觉-语言-动作（VLA）方法将视觉语言模型扩展到动作生成领域，并已在机器人学 [1, 36] 和自动驾驶 [8, 21, 28, 29, 37, 38, 44, 56, 58, 63, 64]中得到探索，旨在根据视觉和语言输入生成物理上可行的动作。视觉-语言-动作模型有望带来范式转变：从学习隐式、反应式策略，转向开发具备显式推理和长时域规划能力的智能体。这些模型提升了交互性和指令遵循的灵活性，使语言成为传达明确规则和组合逻辑的强大媒介，这是迈向更通用、更值得信赖的自主智能体的关键一步。

在理想汽车实习期间完成的工作。
* 共同贡献。 † 项目负责人。

视觉-语言-动作范式的核心在于遵循自然语言指令的能力。指令遵循是现实世界部署的一项关键功能，可实现动态任务调整，并通过透明交互增强用户信任。然而，当前视觉-语言-动作模型面临一个根本性挑战：语言理解与物理动作之间持续存在错位 [10, 37, 49]。模型可能正确生成“向左变道”这一决策，却输出车道保持轨迹。这种指令遵循失败削弱了视觉-语言-动作模型的核心承诺，并对其安全性和可靠性构成重大障碍。

为解决错位问题，研究者提出了多种方案。一些方法着重改进数据收集技术 [10, 37, 49], ，以此绕开根本性的建模挑战。另一些方法则依赖通过强化学习进行事后策略优化 [64], ，将对齐视为事后补救措施。还有一类方法试图通过潜在空间中的隐式分布匹配来对齐模态 [8], ，但缺乏直接且可验证的监督。与这些方法不同，我们认为，这种语义鸿沟要求在主要的监督学习阶段中融入明确的双向关联。为此，我们提出了***LinkVLA***，这是一种专为强化这种关联而设计的新型视觉-语言-动作模型。

首先，我们在架构层面建立结构性关联。通过将语言指令和轨迹统一到共享离散码本中，我们从设计上消除了模态鸿沟，迫使这两个概念进入共同的表征空间。其次，我们通过一种新颖的双向学习目标建立深层语义关联。我们引入了明确的动作理解目标，促使模型将规划动作反向转换为描述性文本。这种协同作用确保了双向一致性，使语言与动作之间的关联具有基础性且可验证。尽管这种紧密耦合的表征有利于对齐，但其自回归特性造成了推理瓶颈。为了使我们的框架实用，我们以由粗到精的两步生成过程取代传统的逐步解码。该机制首先生成轨迹的高层结构概要，然后将其细化为完整的精细路径。如图1所示，*LinkVLA*不仅大幅降低了推理延迟，还持续提升了指令遵循准确率和驾驶性能。主要贡献如下：

- 一种统一的令牌化框架 通过学习语言与动作共享的码本，在结构上弥合模态差距，以增强对齐。
- 一个明确的动作理解目标，用于确保双向语义一致性。
- 一种由粗到细的动作生成方案，可大幅降低推理延迟。
- 最先进的性能在具有挑战性的闭环驾驶基准测试中表现出色，证明了指令遵循准确率和驾驶能力均取得显著提升。

2. 相关工作

2.1. 端到端自动驾驶

近年来，端到端自动驾驶取得了巨大成功。UniAD [14]采用以规划为导向的设计理念，将感知、预测和规划整合到单一网络中。VAD [22] 将驾驶场景建模为完全向量化表示，从而消除计算密集型的栅格化表示，并大幅提升运行速度。ParaDrive [51] 提出了一种完全并行的端到端架构，并全面探索了模块化技术栈的设计空间。VADv2 [3]将规划动作空间离散化为大型规划词汇表，并利用大量驾驶示范学习规划动作的概率分布。SparseDrive [42] 通过稀疏实例表示统一多项任务，并提出了对称稀疏感知模块和并行运动规划器。GenAD [61]将自动驾驶视为生成式建模问题。DriveTransformer [20]采用任务并行和稀疏表示，以流式方式运行，从而提高效率。DiffusionDrive [31] 使用锚点截断扩散过程，从而获得更好的性能和更高的推理效率。GoalFlow [55] 通过引入目标点来约束生成轨迹，并将其作为流匹配的条件，以生成多模态轨迹。尽管这些方法擅长在已知运行域内进行反应式控制，但在复杂逻辑推理、长尾事件处理和人机交互任务执行方面仍存在关键局限。

2.2. 自动驾驶中的视觉语言模型与视觉-语言-动作模型

早期的视觉语言模型主要遵循语言中心范式，通过视觉问答来描述动作 [25, 34, 35]。DriveGPT4 [56] 采用大语言模型（LLM）生成场景解释和控制信号。DriveVLM [46] 使用视觉语言模型来增强场景理解和规划能力。Drivemlm [50] 将决策状态标准化，以弥合语言决策与车辆控制信号之间的鸿沟，并利用大语言模型作出驾驶决策和生成解释。EMMA [15]利用 Gemini 的多模态能力，将所有非传感器数据——包括导航指令、车辆状态、轨迹和三维位置——表示为文本序列，从而将预训练大语言模型的世界知识迁移到自动驾驶任务中。

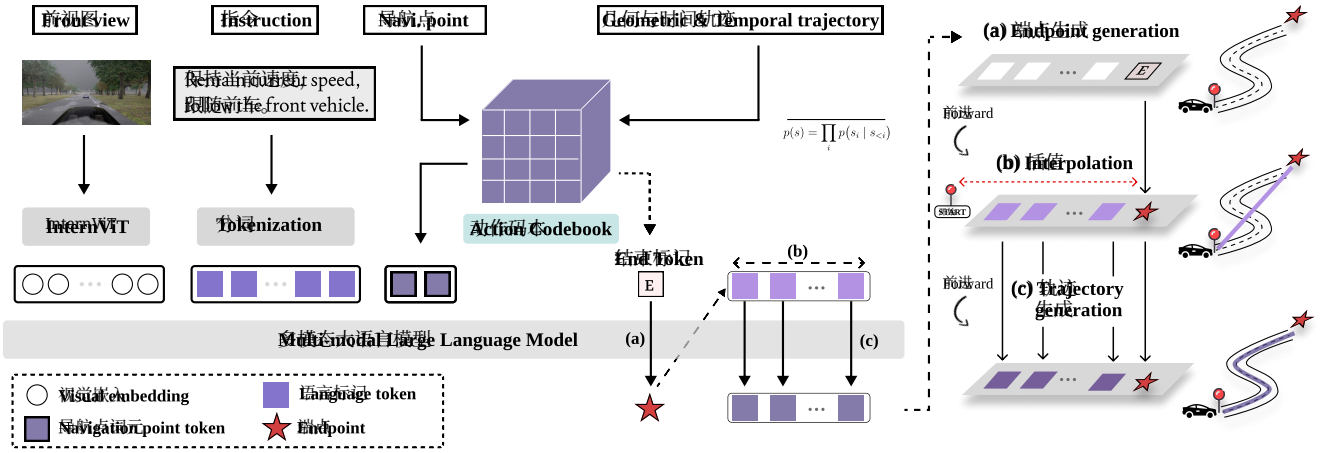


图2。**LinkVLA**架构概览。**该模型由预训练的 InternViT** [4] 视觉骨干网络和 Qwen2- 0.5B [43] 大语言模型组成。LinkVLA 的核心是将语言标记和动作标记（用于导航点和轨迹）统一到一个共享码本中。训练采用统一目标，同时实现语言-动作理解与生成，从而确保深层语义对齐。推理采用高效的粗到细过程：首先，(a) 模型预测轨迹终点；然后，(b) 将其插值为粗略航点；最后，(c) 将其细化为平滑的最终轨迹。

视觉-语言-动作模型直接从原始输入输出动作。Open-drivevla [63]将视觉词元投射到统一的语义空间中，以弥合模态鸿沟。DriveMoE [57] 引入视觉混合专家和动作混合专家，以处理多样化场景。Recogdrive [29]将视觉语言模型学习到的驾驶先验注入扩散规划器，并利用强化学习弥合语言与动作之间的鸿沟。Dif-fVLA [21] 提出了一种由视觉语言模型（VLM）赋能的混合稀疏-稠密扩散策略，从而实现高效的多模态驾驶行为生成。然而，模态鸿沟仍然存在，导致语言与动作之间出现不对齐。

2.3. 语言与动作对齐

AutoVLA [64] 将推理与动作生成相结合，并采用 GRPO [39]来提升规划性能。AutoVLA 逐步进行自回归动作生成，因此在实际部署中效率不高。ORION [8] 通过将生成式规划器与 VLM 架构相结合，弥合推理与动作空间之间的鸿沟，并联合优化视觉问答和规划。SimLingo [37] 专注于对齐语言理解与驾驶动作。然而，由于模态差异，ORION 和 SimLingo 中语言指令与动作之间仍存在持续的错位。CAST [10] 利用视觉语言模型生成反事实标签，以扩充现有的机器人数据集。OmniDrive [49]提出了一个整体性的视觉-语言数据集，利用反事实推理使智能体模型与三维驾驶任务对齐。在统一的多模态理解与生成模型中，研究发现理解与生成能够相互促进 [47, 54, 62]。受此启发，我们提出了一种新颖的架构，通过提升动作生成能力，并引入动作理解目标，而无需额外的数据整理。

3. 方法

我们提出的模型 ***LinkVLA*** 是一种视觉-语言-动作模型，旨在提升自动驾驶中的语言-动作对齐和推理效率。我们的方法包含三项关键创新，如图 2所示。首先，我们建立了一个统一的自回归框架，在单一离散空间中对语言和动作词元进行建模（第 3.1 节）。其次，为增强语义对齐，我们提出了一种新颖的动作理解目标，以促进语言与轨迹之间的双向映射（第 3.2 节）。第三，我们用高效的由粗到细生成机制取代了缓慢的顺序解码，大幅降低推理延迟（第 3.3 节）。

3.1. 统一令牌化框架

我们认为，自动驾驶中的语言-动作失配，是模态之间架构分裂的直接结果。为消除这一问题，我们的方法建立在统一原则之上：在单一统一框架内，对从理解指令到生成轨迹的整个过程进行建模。我们的方法将语言指令 L 和动作轨迹 $\mathcal{T}$ 都映射为由离散令牌组成的统一序列，随后由视觉语言模型骨干网络进行处理。对于语言，我们利用视觉语言模型现有的分词器。对于本质上连续的动作，我们设计了一种空间感知动作令牌化方案。模型不再回归连续值，而是预测一系列动作令牌

来自分词码本。

统一令牌空间。 我们的方法为语言和动作建立了统一令牌空间。为此，我们首先对连续轨迹进行量化：将局部鸟瞰视角（BEV）空间划分为由K_{action} 个网格单元组成的网格，每个网格单元定义一个唯一的动作令牌。因此，通过将每个航点映射到其对应的网格单元，轨迹 $\mathcal{T} = \{w_1, \ldots, w_T\}$ 会被转换为由动作令牌 $A = \{a_1, \ldots, a_T\}$ 组成的序列。随后，该动作码本（$\mathcal{C}_{action}$）与模型的文本词汇表（大小为 K_{text}）合并，形成一个统一的、大小为$K = K_{text} + K_{action}$的码本 $\mathcal{C}$。动作令牌嵌入通过端到端方式学习，促使模型将语言概念和空间概念映射到共享表示中，从而使单个视觉语言模型能够同时处理二者。在推理过程中，每个预测出的动作令牌都会被简单地解码回其对应网格单元的中心。

动作令牌化。 朴素令牌化将航点映射到均匀鸟瞰图网格上，并使用独热标签，但存在两个关键问题。首先，均匀网格将分辨率平均分配，无法提供近场控制所必需的精细精度。其次，独热标签的硬分配丢弃了网格固有的空间拓扑，使空间先验的学习更加困难。为缓解这些问题，我们提出两项关键改进：对数坐标变换和空间软标签化。

1) 对数坐标变换。 我们设计了一种非均匀量化方案，以优先保证自车附近的精度。具体而言，首先沿各轴分别对航点坐标 $(x,\ y)$ 应用非线性变换。具体来说，每个坐标$z \in \{x, y\}$ 都通过带符号的对数函数进行变换：

$$z' = \text{sign}(z) \cdot \log(1 + k \cdot |z|). \tag{1}$$

其中，k 是一个正缩放因子，用于控制原点周围线性区域的大小。随后，对变换后的空间 $(x',\ y')$ 进行均匀量化，以生成动作标记。

2) 空间软标签。 为了将关于动作空间连续性的物理先验嵌入我们的学习目标，我们采用空间软标签策略。标准的独热目标提供离散的监督信号，但并未显式考虑动作网格的空间拓扑。我们的方法通过定义一个承认空间邻接关系的平滑目标分布来对此加以改进。

具体而言，对于真实动作词元 a_{gt}，我们将所有动作标记 $a \in \mathcal{C}_{action}$ 上的目标分布 $q(a)$ 构造为以该词元为中心的归一化二维高斯分布，

半径为 R 的 a_{gt} 坐标：

$$q(a) = \frac{1}{Z} \exp\left(-\frac{\|\text{pos}(a) - \text{pos}(a_{gt})\|_2^2}{2\sigma^2}\right), \tag{2}$$

其中，$\text{pos}(a)$ 将动作词元映射到其在空间网格中的二维坐标，$\|\cdot\|_2^2$ 表示欧几里得距离的平方，σ 是控制分布扩散范围的超参数，而 Z 是归一化常数，确保 $\sum_{a \in \mathcal{C}_{action}} q(a) = 1$。对于动作生成，随后使用交叉熵损失优化模型的预测分布 $p(a)$，使其与该软目标相匹配：

$$\mathcal{L}_{\text{generation}} = -\sum_{a \in \mathcal{C}_{action}} q(a) \log p(a). \tag{3}$$

该目标鼓励模型不仅将概率质量分配给正确的词元，还依据高斯形状将其分配给空间邻域。这有助于形成局部平滑的动作流形，使模型对轻微的真值误差更加稳健。在引入思维链时，会在该目标中额外加入一项用于语言生成的标准交叉熵损失。

3.2. 统一的语言-动作理解与生成

为加强语言与行为之间的联系，我们引入了一种双向训练目标，其灵感来自图像描述与文本到图像生成的对偶性。在生成式视觉语言建模中，生成文本 L 来自图像 V（$p(L|V)$）以及由文本生成图像（$p(V|L)$）这两项任务是互逆的 [47, 54, 62]。在这两个目标上训练模型已被证明能够产生更加稳健且对齐的联合嵌入空间。

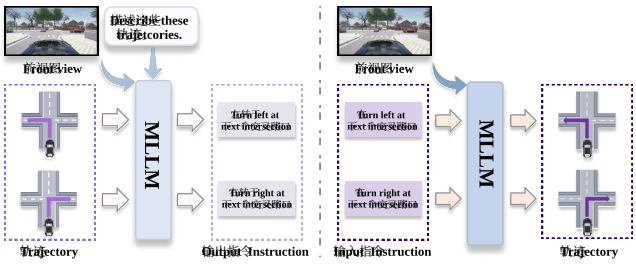


图3. 动作理解（左）与动作生成（右）的示意图。

我们认为，视觉-语言-动作模型中的语言-动作映射也存在类似的二元性。传统任务是动作生成，即由语言指令引导动作序列的预测 A ($p(A|L)$)。这类似于文本到图像合成。我们提出与之互惠的任务：动作理解，即利用已执行的动作序列 A推断原始语言指令（$p(L|A)$）。这与图像描述类似，因为

模型必须生成语言描述，解释观察到的行为。至关重要的是，这两种映射都以共享的视觉上下文为基础 V，它提供了必要的环境感知。形式上，除了$\mathcal{L}_{generation}$(方程 3)之外，我们还引入一个互惠目标，用于重构语言指令 L，条件是给定视觉和动作输入：

$$\mathcal{L}_{\text{understanding}} = -\sum_{j} \log p(l_j | V, A, l_{<j}), \tag{4}$$

其中，$l_{<j}$ 表示前面的真实标记。

最终的训练目标是这两项损失的加权和：$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{generation}} + \lambda \mathcal{L}_{\text{understanding}}$，其中 λ是一个平衡超参数。通过迫使模型解决这一逆问题，我们在共享嵌入空间中确保双向一致性。这一过程丰富了动作标记的语义落地，使其与描述性语言概念内在关联，从而更好地遵循指令。在实践中，只需交换 L和 A作为预测目标，即可由同一个解码器处理这两项任务。

3.3. 粗到细动作生成

对包含 T 个航点的长轨迹进行自回归生成，需要 T 次顺序前向传播，计算开销大且会显著增加推理延迟。为解决这一问题，我们将 T步的顺序依赖压缩为一个两阶段流程：（1）端点预测与粗轨迹初始化；（2）并行轨迹细化。按照 [37], ，我们的动作输出包括时间速度航点和几何路径航点。由于二者都被视为点序列，并在整个框架中以对称方式处理，为清晰起见，下文将它们统称为轨迹，其组成部分称为航点。

使用粗轨迹先验进行训练。我们精心设计的训练目标使由粗到细的推理成为可能。为直接预测端点，我们在解码器输入序列的开头加入特殊词元。训练期间，真实目标序列会重新排序，$\{w_T, w_1, w_2, ..., w_{T-1}\}$，从而教会模型将特殊词元与端点预测相关联。对于细化阶段，我们使用真实端点通过线性插值模拟粗轨迹，将其量化为粗航点标记作为模型输入，并训练模型将这些粗标记映射到相应的细粒度轨迹。

粗轨迹初始化。 在推理过程中，我们首先为轨迹建立强有力的结构先验，以引导后续的生成步骤。借助我们修改后的训练序列，模型只需进行一次前向传播，即可专门预测最终航点 $\hat{w}_T$。尽管这一初始步骤受到目标点预测方法 [55], 的启发，但我们的方法有着本质区别：我们在单一、统一的 Transformer架构中整合了端点预测与轨迹优化。

给定起点 w_0 （自车在 (0, 0) 处的原点）和预测的终点 w_T，我们通过线性插值构造粗轨迹， $\mathcal{W}_{coarse}$：

$$w_i^{coarse} = w_0 + \frac{i}{T}(w_T - w_0) \quad \text{for } i \in \{1, \ldots, T\} \tag{5}$$

随后，这些航点会被词元化为轨迹词元，作为后续细化阶段的初始框架。

并行轨迹细化。第二步推理会将粗略的直线路径细化为动态可行的轨迹。我们将其建模为一种保留结构的细化过程，其中每个粗航点 w_i^{coarse}都会映射到相应的细化航点 w_i^{fine} 。以词元化的粗航点作为初始输入，LinkVLA 并行预测 T 个细化点。通过交叉注意力以视觉语言上下文为条件，细化后的路径能够遵守车道边界、避开障碍物，并遵循语言指令。

4. 实验

我们首先介绍基准测试、评估指标（第4.1节）和实现细节（第4.2节）。随后，我们展示并分析闭环结果及指令遵循能力（第4.3节），最后详细介绍消融实验和对比结果（第4.4节）。

4.1. 设置

Bench2Drive。我们使用 Bench2Drive 基准测试 [19], 训练和评估 LinkVLA。该基准测试 [19], 在广泛使用的 CARLA 模拟器 [7]中提供了一系列交互式场景。我们遵循 SimLingo [37], 并使用开源专家模型 PDM-lite [41] 在 CARLA 模拟器中收集驾驶数据集。我们的评估遵循 CARLA v2 端到端自动驾驶闭环协议，包含 44 个交互式场景，每个场景有 5 条路线，总计 220 条官方路线，涵盖多种天气条件。我们使用基准测试的官方指标报告性能：驾驶得分（DS）、成功率（SR）、效率、舒适度和多能力。

指令跟随评估。 我们使用来自 SimLingo [37] 的 *ActionDreaming* 数据集评估模型的指令跟随能力。该数据集旨在评估模型不仅能否理解语言中的特定场景知识，还能否将这种理解转化为相应的动作空间。

表1. **Bench2Drive中的主要结果与多能力**。*表示专家特征蒸馏。

方法	闭环指标 ↑		多能力 (%) ↑							
	DS	成功率 (%)	效率	舒适度.	并线	超车	制动	给予-Way	交通-Sign	Mean
TCP* [53]	40.70	15.00	54.26	47.80	16.18	20.00	20.00	10.00	6.99	14.63
TCP-ctrl*	30.47	7.27	55.97	**51.51**	10.29	4.44	10.00	10.00	6.45	8.23
TCP-traj*	59.90	30.00	76.54	18.08	8.89	24.29	51.67	40.00	46.28	34.22
ThinkTwice* [18]	62.44	31.23	69.33	16.22	27.38	18.42	35.82	50.00	54.23	37.17
DriveAdapter* [17]	64.22	33.08	70.22	16.01	28.82	26.38	48.76	50.00	56.43	42.08
AD-MLP [59]	18.05	0.00	48.45	22.63	0.00	0.00	0.00	0.00	4.35	0.87
UniAD-Tiny [14]	40.73	13.18	123.92	47.04	8.89	9.33	20.00	20.00	15.43	14.73
UniAD-Base [14]	45.81	16.36	129.21	43.58	14.10	17.78	21.67	10.00	14.21	15.55
VAD [22]	42.35	15.00	157.94	46.01	8.11	24.44	18.64	20.00	19.15	18.07
DriveTransformer [20]	63.46	35.01	100.64	20.78	17.57	35.00	48.36	40.00	52.10	38.60
Orion [8]	77.74	54.62	151.48	17.38	25.00	71.11	78.33	30.00	69.15	54.72
AutoVLA [64]	78.84	57.73	146.93	39.33	-	-	-	-	-	-
SimLingo [37]	85.07	67.27	**259.23**	33.67	53.75	68.89	81.67	50.00	82.11	67.28
LinkVLA（我们的方法）	**91.01**	**74.55**	255.84	34.62	**60.00**	**80.00**	**93.33**	**50.00**	**83.68**	**73.40**

给定一条自然语言指令，模型应生成与该指令相对应的动作序列。评估在 CARLA Town 13 dreamer 数据集的验证集上进行。指令分为六类之一：减速、加速、达到目标速度、变道、以物体为中心。模型表现以成功率衡量。

DriveLM-hard（视觉问答）和评论。我们在 DriveLM-hard 基准测试上评估视觉问答和评论生成 [37]。这个具有挑战性的验证集源自 DriveLM [41]，聚焦于 CARLA Town 13 环境。为确保测试集均衡并包含罕见案例，该基准测试通过对每种答案类型均匀抽取 10 个样例构建，而非简单随机抽样。最终数据集包含 330 种视觉问答答案类型和 190 种评论类型。我们使用以下指标报告得分：*SPICE*、*BLEU*和*ROUGE-L* 。

表 2. **性能与延迟对比**。所有指标均在 CARLA 基准上评估。延迟是每步的平均推理时间，在 H20 GPU 上测量。

ID	方法	Type	延迟↓	驾驶得分↑
1	Orion [8]	VAE	65ms	77.74
2	SimLingo [37]	MLP	34ms	85.07
3	Ours	AR	361毫秒	90.66
4	Ours	C2F	48ms	91.01

4.2. 实现细节

动作离散化。该框架在鸟瞰图（BEV）空间中运行，其坐标范围为 $x \in [0, 50]$米和 $y \in [-30, 30]$米。为构建离散动作空间，首先使用第3.1 节详述的对数函数对这些坐标进行变换（超参数为$k = 5$），随后以 0.1 的步长将其离散化为均匀网格。由此得到一个 56×101 网格，构成包含 $K_{action} = 5{,}656$ 个离散动作令牌的词汇表。空间软标签化过程中，邻居加权半径设为 $R = 10$ 个单元，高斯标准差设为 $\sigma = 1.2$ 。此外，为了支持粗到细（C2F）框架的分层动作生成，引入了两个特殊标记：路径目标令牌和航点目标令牌。

训练细节。 我们采用 Mini-InternVL 系列中的 InternVL2-1B [9] 作为主要架构。InternVL2-1B [9]模型由视觉编码器 InternViT-300M-448px（视觉变换器） [4] 和作为大语言模型（LLM）的 Qwen2-0.5B-Instruct [43] 组成。我们使用 AdamW 优化器 [26]训练模型，并采用余弦学习率调度。超参数设置如下：基础学习率为 1e-4，权重衰减为 0.1， $\beta_1 = 0.9$、$\beta_2 = 0.999$，丢弃率为 0.1。模型在 32 块 H20 GPU 上训练 30 个训练轮次，批量大小为 48。在模型适配方面，我们遵循 SimLingo [37]，采用低秩适配 [12]，秩设为 32，并使用 $\alpha = 64$。

在推理过程中，我们采用思维链（CoT）方法。首先，模型会为动作生成文本推理。然后，以这段评论为条件，模型预测最终的

动作序列。每帧的输出由 20 个几何路径词元和 10 个时间航点词元组成。

统一理解与生成。我们随机将一个（V、 L、 A）元组拼接至：1） $[V, A, L]$ 并通过L 监督动作理解，或 2） $[V, L, A]$ 并通过 A监督动作生成。两者与 $\mathcal{L}_{total}$ 一同训练。

4.3. 结果

Bench2Drive 结果。表 1报告了 Bench2Drive 上的闭环指标和多能力评估结果。LinkVLA 在保持效率和舒适度相当的同时，取得了最高的驾驶得分和成功率。具体而言，LinkVLA 的驾驶得分为 91.01，成功率为 74.55，分别以 5.94 和 7.28 分超越此前的最先进方法 SimLingo（驾驶得分 85.07，成功率 67.27），相对提升分别为 6.98% 和 10.82%。在效率方面，LinkVLA 达到 255.84，显著超过 Orion（151.48）和 AutoVLA（146.93）等早期方法。在舒适度方面，LinkVLA（34.62）略胜 SimLingo（33.67），并大幅优于 Orion（17.38）。

多能力结果表明，LinkVLA 在多样化交互场景中展现出稳健的驾驶行为。LinkVLA 在并线（60.00）、超车（80.00）、制动（93.33）和交通标志（83.68）方面取得了最高分，并在让行（50.00）方面达到并列最佳表现。在交互密集型和危险响应型技能上，其提升尤为明显：相比 SimLingo，并线、超车和制动分别提高了 6.25、11.11 和 11.66 分。总体多能力性能达到 73.40，超过 SimLingo（67.28）6.12 分（相对提升 9.09%），并比 Orion（54.72）高出 18.68 分，表明 LinkVLA 更适合闭环驾驶性能。

推理延迟。 表2 展示了 Bench2Drive 基准测试中延迟与性能的对比分析。延迟以每帧生成一条紧凑轨迹所需的平均推理时间进行量化。我们省略了思维链（CoT）过程的计算成本，因为其具有随查询变化的特性，将会对延迟分析引入混杂因素。

在各基线模型中，SimLingo 的速度最快，每步延迟仅为 34 ms，但其驾驶得分为 85.07。相比之下，Orion（变分自编码器）的速度较慢，延迟为 65 ms。我们的自回归（自回归）变体取得了 90.66 的高驾驶得分，但延迟高达 361 ms，难以接受。

相比之下，我们提出的粗到细（C2F）方法实现了更出色的平衡。它将延迟从 361 毫秒降至 48 毫秒，同时将驾驶得分提升至 91.01，在所有对比方法中最高；与 Orion 相比，驾驶得分高出 13.27 分，延迟则降低 26%。与速度最快的基线 SimLingo 相比，我们的方法仅增加 14 毫秒延迟，性能便提升了 5.94 分。

4.4. 对比分析

指令跟随评估。 表3中的详细结果表明，逐步加入动作令牌化、粗到细（C2F）轨迹生成以及用于对齐的动作理解目标（表中简称*align.*），能够显著提升性能。与基线（ID 1，平均成功率为 70.11%）相比，仅引入分词（ID 2）便将总体成功率提升至 81.63%。这一提升主要得益于停车任务的近乎完美表现（99.88%），以及变道（88.49%）和以物体为中心（84.34%）任务的显著改善。随后加入 C2F 生成（ID 3），进一步提升了加速（93.16%）、目标速度（69.24%）和变道（95.45%）等任务的表现。最后，引入对齐模块（ID 4）获得了最优且均衡的性能。该最终配置的平均成功率最高，达到 87.16%，并在加速（96.48%）、目标速度（74.73%）和变道（97.42%）任务上创下新高，展现了我们方法令人惊讶的指令跟随能力，也证明了所提出的动作理解目标的有效性。

语言能力评估。 表4 表明，在两个基准测试中，启用分词（ID 2）相比基线（ID 1）都能在 *SPICE*、*BLEU*和 *ROUGE-L*方面带来稳定提升。加入 C2F（ID 3）后，语义保真度进一步增强：在 DriveLM-VQA 上，*SPICE* 提升至 71.3；在评论任务上，则提升至 53.6。在分词和 C2F 的基础上进一步引入对齐训练（ID 4）后，模型在 VQA 和评论基准测试中均取得了最佳的语言理解性能，所有指标均有所提升。得益于统一令牌空间的设计，我们的模型实现了更出色的语言能力。

消融实验。 表5 表明，分词带来了显著提升，使驾驶得分从 85.07% 提高至 89.57%，成功率从 67.27% 提高至 73.18%。在未进行对齐的情况下加入 C2F，驾驶得分略有提升（达到 89.85%），而成功率则略有下降（降至 72.27%）。在分词和 C2F 的基础上加入对齐，取得了最佳性能（驾驶得分为 91.01%；成功率为 74.55%），优于所有先前配置。

软标签化的效果。 表S2 表明，加入软标签监督可持续提升闭环性能。驾驶得分从 90.85 提高至 91.01（提升 0.16 个百分点），成功率从 72.73% 提高至 74.55%（提升 1.82 个百分点）。这些结果表明，软标签监督能够有效利用空间先验，从而提升驾驶性能和任务完成的稳健性。

导航模态。表7 表明，这两种导航模态实现了相当的闭环性能。这表明LinkVLA能够遵循基本



图 4. 在包含多种语言指令的复杂环境中的可视化结果。生成的轨迹准确遵循语言指令，同时在复杂环境中保持安全且可行。

表 3. **指令遵循评估** 在 *Action Dreaming* 数据集 [37]中。Align. 指的是与统一训练的对齐 g.

ID	词元	C2F	对齐。	成功率（%）↑						
				加速	减速	目标速度	Lane 变化	物体	Stop	Mean
1	✗	✗	✗	81.42	61.83	66.27	75.53	74.69	60.93	70.11
2	✓	✗	✗	88.44	65.24	63.37	88.49	84.34	**99.88**	81.63
3	✓	✓	✗	93.16	55.86	69.24	95.45	85.38	92.14	81.87
4	✓	✓	✓	**96.48**	**65.57**	**74.73**	**97.42**	**91.41**	97.34	**87.16**

表4. **语言能力**在 DriveLM-VQA 和评述评估中的表现。S / B / R 分别指 *SPICE* / *BLEU* / *ROUGE-L*

ID	Token	C2F	对齐	DriveLM-VQA ↑			评论↑		
				S	B	R	S	B	R
1	✗	✗	✗	66.7	68.9	71.5	49.2	60.3	64.3
2	✓	✗	✗	69.7	70.5	73.1	53.3	63.7	68.0
3	✓	✓	✗	71.3	69.9	73.4	53.6	61.6	67.3
4	✓	✓	✓	**73.0**					

表5. **闭环性能**对我们提出的不同组件进行的消融研究。

ID	Token	C2F	对齐	闭环指标 ↑	
				驾驶得分	成功率(%)
1	✗	✗	✗	85.07	67.27
2	✓	✗	✗	89.57	73.18
3	✓	✓	✗	89.85	72.27
4	✓	✓	✓	**91.01**	**74.55**

表6. 软标签对闭环评估的影响。

方法	驾驶得分↑	成功率(%) ↑
我们的方法 不使用软标签	90.85	72.73
我们的方法 使用软标签	91.01	74.55

表 7. 不同导航形式的效果。

方法	驾驶得分↑	成功率(%)↑
GPS目标点	91.01	74.55
导航指令	91.25	73.18

导航指令和导航GPS目标点，并使用同一模型。

更多消融研究讨论。 由于篇幅所限，我们在补充材料中进一步对动作码本大小 K_{action}、对数变换中的缩放因子 k 以及空间软扩散中的扩散尺度参数 σ 进行了更全面的消融研究-

标注见补充材料。

5. 结论

我们提出了***LinkVLA***，这是一种新颖的视觉-语言-动作模型，可提升语言-动作对齐与效率。通过统一语言和

动作词元至共享码本中，并训练一种新颖的双向描述目标，我们实现了深层次的跨模态一致性。这一核心策略结合由粗到细解码器，将延迟降低了 86%，在闭环基准测试上的指令遵循和驾驶性能方面带来了显著提升。因此，我们的工作为开发可靠且响应迅速、可部署于现实世界的语言引导智能体提供了一条切实可行的路径。

参考文献

[1]Kevin Black、Noah Brown、Danny Driess、Adnan Esmail、Michael Equi、Chelsea Finn、Niccolo Fusai、Lachy Groom、Karol Hausman、Brian Ichter等。π0：面向通用机器人控制的视觉-语言-动作流模型。corr, abs/2410.24164, 2024. doi: 10.48550. arXiv预印本 ARXIV.2410.24164. 1[2] Kai Chen、Yanze Li、Wenhua Zhang、Yanxin Liu、Pengxiang Li、Ruiyuan Gao、Lanqing Hong、Meng Tian、Xinhai Zhao、Zhenguo Li等。自动驾驶极端案例上大型视觉语言模型的自动评估。见2025年IEEE/CVF计算机视觉应用冬季会议（WACV），第7817–7826页。IEEE，2025。1[3] Shaoyu Chen、Bo Jiang、Hao Gao、Bencheng Liao、Qing Xu、Qian Zhang、Chang Huang、Wenyu Liu和Xinggang Wang。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 1, 2 . . [4] Zhe Chen、Jiannan Wu、Wenhai Wang、Weijie Su、Guo Chen、Sen Xing、Muyan Zhong、Qinglong Zhang、Xizhou Zhu、Lewei Lu等。InternVL：扩展视觉基础模型并面向通用视觉语言任务进行对齐。见IEEE/CVF计算机视觉与模式识别会议论文集，第24185–24198页，2024。3, 6[5] Kashyap Chitta、Aditya Prakash、Bernhard Jaeger、Zehao Yu、Katrin Renz和Andreas Geiger。Transfuser：面向自动驾驶的基于Transformer的传感器融合模仿学习。IEEE模式分析与机器智能汇刊，45(11):12878–12895，2022。1[6] Kairui Ding、Boyuan Chen、Yuchen Su、Huan-ang Gao、Bu Jin、Chonghao Sima、Wuqiang Zhang、Xiaohui Li、Paul Barsch、Hongyang Li等。Hint-AD：端到端自动驾驶中的整体对齐可解释性。arXiv预印本 arXiv:2409.06702，2024。1 [7] Alexey Dosovitskiy、German Ros、Felipe Codevilla、Antonio Lopez和Vladlen Koltun。CARLA：一个开放的城市驾驶仿真器。见机器人学习会议，第1–16页。PMLR，2017。5, 1[8] Haoyu Fu、Diankun Zhang、Zongchuang Zhao、Jianfeng Cui、Dingkang Liang、Chong Zhang、Dingyuan Zhang、Hongwei Xie、Bing Wang和Xiang 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。见IEEE/CVF国际计算机视觉会议论文集，2025。1, 2, 3, 6[9] Zhangwei Gao、Zhe Chen、Erfei Cui、Yiming Ren、Weiyun Wang、Jinguo Zhu、Hao Tian、Shenglong Ye、Junjun He、Xizhou Zhu等。Mini-InternVL：一种灵活迁移的口袋参数量仅为 5%，性能却达到 90% 的多模态模型。Visual Intelligence，2(1):32，2024。 6[10] Catherine Glossop、William Chen、Arjun Bhorkar、Dhruv Shah 和 Sergey Levine。CAST：反事实标签提升视觉语言动作模型的指令遵循能力。arXiv 预印本 arXiv:2508.13446，2025。 2, 3[11] Daya Guo、Dejian Yang、Haowei Zhang、Junxiao Song、Ruoyu Zhang、Runxin Xu、Qihao Zhu、Shirong Ma、Peiyi Wang、Xiao Bi 等。DeepSeek-R1：通过强化学习激发大语言模型的推理能力。arXiv 预印本 arXiv:2501.12948, 2025。 1[12] Edward J Hu、Yelong Shen、Phillip Wallis、Zeyuan Allen-Zhu、Yuanzhi Li、Shean Wang、Lu Wang、Weizhu Chen 等。LoRA：大语言模型的低秩适配。ICLR, 1(2):3，2022。 6[13] Shengchao Hu、Li Chen、Penghao Wu、Hongyang Li、Junchi Yan 和 Dacheng Tao。ST-P3：通过时空特征学习实现端到端视觉自动驾驶。载于《欧洲计算机视觉会议》，第 533–549 页。施普林格，2022。 1[14] Yihan Hu、Jiazhi Yang、Li Chen、Keyu Li、Chonghao Sima、Xizhou Zhu、Siqi Chai、Senyao Du、Tianwei Lin、Wenhai 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》，第 17853–17862 页，2023。 1, 2, 6[15] Jyh-Jing Hwang、Runsheng Xu、Hubert Lin、Wei-Chih Hung、Jingwei Ji、Kristy Choi、Di Huang、Tong He、Paul Covington、Benjamin Sapp 等。EMMA：用于自动驾驶的端到端多模态模型。arXiv 预印本 arXiv:2410 23262，2024 2 . . [16] Aaron Jaech、Adam Kalai、Adam Lerer、Adam Richardson、Ahmed El-Kishky、Aiden Low、Alec Helyar、Aleksander Madry、Alex Beutel、Alex Carney 等。OpenAI o1 系统卡。arXiv 预印本 arXiv:2412.16720，2024。 1[17] Xiaosong Jia、Yulu Gao、Li Chen、Junchi Yan、Patrick Langechuan Liu 和 Hongyang Li。DriveAdapter：打破端到端自动驾驶中感知与规划的耦合障碍。载于《IEEE/CVF国际计算机视觉会议论文集》，第 7953–7963 页，2023。 6[18] Xiaosong Jia、Penghao Wu、Li Chen、Jiangwei Xie、Conghui He、Junchi Yan 和 Hongyang Li。驾驶前请三思：迈向可扩展的端到端自动驾驶解码器。载于《IEEE/CVF计算机视觉与模式识别会议论文集》，第 21983–21994 页，2023。 6[19] Xiaosong Jia、Zhenjie Yang、Qifeng Li、Zhiyuan Zhang 和 Junchi Yan。Bench2Drive：迈向闭环端到端自动驾驶的多能力基准测试。载于 NeurIPS 2024 数据集与基准测试赛道，2024。 5, 1[20] Xiaosong Jia、Junqi You、Zhiyuan Zhang 和 Junchi Yan。DriveTransformer：用于可扩展端到端自动驾驶的统一Transformer。载于国际学习表征会议（ICLR），2025。 2, 6 [21] Anqing Jiang、Yu Gao、Zhigang Sun、Yiru Wang、Jijun Wang、Jinghao Chai、Qian Cao、Yuweng Heng、Hao Jiang、Yunda Dong 等。DiffVLA：视觉语言引导的

自动驾驶融合规划。arXiv 预印本 arXiv:2505.19381，2025。 1, 3[22] Bo Jiang、Shaoyu Chen、Qing Xu、Bencheng Liao、Jiajie Chen、Helong Zhou、Qian Zhang、Wenyu Liu、Chang Huang 和 Xinggang Wang。VAD：用于高效自动驾驶的向量化场景表示。载于《IEEE/CVF国际计算机视觉会议论文集》，第 8340–8350 页，2023。 1, 2, 6[23] Bo Jiang、Shaoyu Chen、Bencheng Liao、Xingyu Zhang、Wei Yin、Qian Zhang、Chang Huang、Wenyu Liu 和 Xinggang Wang。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。 1[24] Bo Jiang、Shaoyu Chen、Qian Zhang、Wenyu Liu 和 Xinggang Wang。Alphadrive：通过强化学习与推理释放视觉语言模型在自动驾驶中的潜力。arXiv 预印本 arXiv:2503.07608，2025。 [25] Bu Jin 和 Haotian Liu。Adapt：动作感知驾驶描述Transformer。载于 CAAI 国际人工智能会议，第 473–477 页。施普林格，2023。 1, 2[26] Diederik Kinga、Jimmy Ba Adam 等。一种随机优化方法。载于国际学习表征会议（ICLR）。加利福尼亚；2015。 6[27] Boyi Li、Yue Wang、Jiageng Mao、Boris Ivanovic、Sushant Veer、Karen Leung 和 Marco Pavone。借助大语言模型策略适配实现随处驾驶。载于《IEEE/CVF计算机视觉与模式识别会议论文集》，第 14948–14957 页，2024。 1[28] Yingyan Li、Shuyao Shang、Weisong Liu、Bing Zhan、Haochen Wang、Yuqi Wang、Yuntao Chen、Xiaoman Wang、Yasong An、Chufeng Tang 等。DriveVLA-W0：世界模型放大自动驾驶中的数据缩放定律。arXiv 预印本 arXiv:2510.12796，2025。 1[29] Yongkang Li、Kaixin Xiong、Xiangyu Guo、Fang Li、Sixu Yan、Gangwei Xu、Lijun Zhou、Long Chen、Haiyang Sun、Bing 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。 1, 3[30] Zhenxin Li、Kailin Li、Shihao Wang、Shiyi Lan、Zhiding Yu、Yishen Ji、Zhiqi Li、Ziyue Zhu、Jan Kautz、Zuxuan Wu 等。Hydra-MDP：通过多目标 Hydra 蒸馏实现端到端多模态规划。arXiv 预印本 arXiv:2406 06978，2024 1 . . [31] Bencheng Liao、Shaoyu Chen、Haoran Yin、Bo Jiang、Cheng Wang、Sixu Yan、Xinbang Zhang、Xiangyu Li、Ying Zhang、Qian 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》，第 12037–12047 页，2025。 1, 2[32] Yingzi Ma、Yulong Cao、Jiachen Sun、Marco Pavone 和 Chaowei Xiao。Dolphins：用于驾驶的多模态语言模型。载于《欧洲计算机视觉会议》，第 403–420 页。施普林格，2024。 1[33] Jiageng Mao、Yuxi Qian、Junjie Ye、Hang Zhao 和 Yue Wang。GPT-Driver：使用 GPT 学习驾驶。arXiv 预印本 arXiv:2310.01415，2023。[34] Ana-Maria Marcu、Long Chen、Jan H¨unermann、Alice Karn-sund、Benoit Hanotte、Prajwal Chidananda、Saurabh Nair、Vijay Badrinarayanan、Alex Kendall、Jamie Shotton 等。LingoQA：面向自动驾驶的视觉问答. 收录于欧洲计算机视觉会议，页码 252–269. 施普林格，2024. 2[35] SungYeon Park、MinJae Lee、JiHyuk Kang、Hahyeon Choi、Yoonah Park、Juhwan Cho、Adam Lee 和 DongKyu Kim. VLAAD：面向自动驾驶的视觉语言助手. 收录于 IEEE/CVF计算机视觉应用冬季会议论文集，页码 980–987，2024. 1, 2[36] Karl Pertsch、Kyle Stachowicz、Brian Ichter、Danny Driess、Suraj Nair、Quan Vuong、Oier Mees、Chelsea Finn 和 Sergey Levine. FAST：面向视觉-语言-动作模型的高效动作令牌化. arXiv预印本 arXiv:2501 09747，2025 1 . . [37] Katrin Renz、Long Chen、Elahe Arani 和 Oleg Sinavski. SimLingo：通过语言-动作对齐实现纯视觉闭环自动驾驶. 收录于计算机视觉与模式识别会议论文集，页码 11993–12003，2025. 1, 2, 3, 5, 6, 8[38] Hao Shao、Yuxuan Hu、Letian Wang、Guanglu Song、Steven L Waslander、Yu Liu 和 Hongsheng Li. LMDrive：利用大语言模型实现闭环端到端驾驶. 收录于 IEEE/CVF计算机视觉与模式识别会议论文集，页码 15120–15130，2024. 1[39] Zhihong Shao、Peiyi Wang、Qihao Zhu、Runxin Xu、J unxiao Song、Xiao Bi、Haowei Zhang、Mingchuan Zhang、YK Li、Yang Wu 等. DeepSeekMath：突破开放语言模型数学推理的极限. arXiv预印本 arXiv:2402.03300，2024. 3[40] Chonghao Sima、Katrin Renz、Kashyap Chitta、Li Chen、Hanxue Zhang、Chengen Xie、Ping Luo、Andreas Geiger 和 Hongyang Li. DriveLM：通过图视觉问答实现驾驶. arXiv预印本 arXiv:2312.14150，2023. 2[41] Chonghao Sima、Katrin Renz、Kashyap Chitta、Li Chen、Hanxue Zhang、Chengen Xie、Jens Beißwenger、Ping Luo、Andreas Geiger 和 Hongyang Li. DriveLM：通过图视觉问答实现驾驶. 收录于欧洲计算机视觉会议，页码 256–274. 施普林格，2024. 1, 5, 6, 2 [42] Wenchao Sun、Xuewu Lin、Yining Shi、Chuang Zhang、Haoran Wu 和 Sifa Zheng. SparseDrive：通过稀疏场景表示实现端到端自动驾驶. 收录于 2025 IEEE国际机器人与自动化会议 (ICRA)，页码 8795–8801. IEEE，2025. 1, 2 – [43] . Qwen 团队等 . Qwen 技术报告 . arXiv 预印本 arXiv:2407.10671，2(3)，2024. 3, 6[44] Waywe Research Team 等. Lingo-2：使用自然语言驾驶，2024. 1[45] Kexin Tian、Jingrui Mao、Yunlong Zhang、Jiwan Jiang、Yang Zhou 和 Zhengzhong Tu. nuScenes-SpatialQA：面向自动驾驶视觉语言模型的空间理解与推理基准测试. arXiv预印本 arXiv:2504.03164，2025. 1[46] Xiaoyu Tian、Junru Gu、Bailin Li、Yicheng Liu、Yang Wang、Zhiyong Zhao、Kun Zhan、Peng Jia、Xianpeng Lang 和 Hang Zhao. DriveVLM：自动驾驶与大型视觉语言模型的融合. arXiv预印本 arXiv:2402.12289，2024. 1, 2

[47] Shengbang Tong, David Fan, Jiachen Li, Yunyang Xiong, Xinlei Chen, Koustuv Sinha, Michael Rabbat, Yann Le-Cun, Saining Xie 和 Zhuang Liu. Metamorph：通过指令微调实现多模态理解与生成. 收录于 IEEE/CVF国际计算机视觉会议论文集，页码 17001–17012，2025. 3, 4[48] Martin Treiber, Ansgar Hennecke 和 Dirk Helbing. 实证观测与微观模拟中的拥堵交通状态. Physical Review E, 62(2):1805，2000. 2[49] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li 和 Jose M Alvarez. OmniDrive：一个面向自动驾驶、支持反事实推理的整体式视觉-语言数据集. 收录于计算机视觉与模式识别会议论文集，页码 22442–22452，2025. 1, 2, 3[50] Wenhai Wang, Jiangwei Xie, ChuanYang Hu, Haoming Zou, Jianan Fan, Wenwen Tong, Yang Wen, Silei Wu, Hanming Deng, Zhiqi Li 等. Drivemlm：将多模态大语言模型与自动驾驶的行为规划状态对齐. arXiv预印本 arXiv:2312.09245，2023. 1, 2[51] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang 和 Marco Pavone. Para-drive：用于实时自动驾驶的并行化架构. 收录于 IEEE/CVF计算机视觉与模式识别会议论文集，页码 15449–15458，2024. 1, 2[52] Katharina Winter, Mark Azer 和 Fabian B Flohr. Bev-driver：在大语言模型中利用鸟瞰图地图实现稳健的闭环驾驶. arXiv预印本 arXiv:2503.03074，2025. 1[53] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li 和 Yu Qiao. 面向端到端自动驾驶的轨迹引导控制预测：简单而强大的基线. 神经信息处理系统进展，35:6119–6132，2022. 6[54] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang 和 Mike Zheng Shou. Show-o：用单一 Transformer 统一多模态理解与生成. arXiv预印本 arXiv:2408.12528，2024. 3, 4[55] Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long 和 Wei Yin. GoalFlow：面向端到端自动驾驶的多模态轨迹生成目标驱动流匹配. 收录于计算机视觉与模式识别会议论文集，页码 1602–1611，2025. 1, 2, 5[56] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li 和 Hengshuang Zhao. DriveGPT4：通过大语言模型实现可解释端到端自动驾驶. IEEE Robotics and Automation Letters，2024. 1, 2[57] Zhenjie Yang, Yilin Chai, Xiaosong Jia, Qifeng Li, Yuqian Shao, Xuekai Zhu, Haisheng Su 和 Junchi Yan. DriveMoE：面向端到端自动驾驶视觉-语言-动作模型的混合专家. arXiv预印本 arXiv:2505.16278，2025. 3[58] Shuang Zeng、Xinyuan Chang、Mengwei Xie、Xinran Liu、Yifan Bai、Zheng Pan、Mu Xu 和 Xing Wei。Futuresight-drive：通过时空思维链进行视觉思考以实现自动驾驶。arXiv预印本 arXiv:2505.17685，2025。1[59] Jiang-Tian Zhai、Ze Feng、Jinhao Du、Yongqiang Mao、Jiang-Jiang Liu、Zichang Tan、Yifu Zhang、Xiaoqing Ye 和 Jingdong Wang。重新思考 nuScenes 中端到端自动驾驶的开环评估。arXiv预印本 arXiv:2305.10430，2023。6[60] Wenzhao Zheng、Weiliang Chen、Yuanhui Huang、Borui Zhang、Yueqi Duan 和 Jiwen Lu。OccWorld：学习用于自动驾驶的三维占用世界模型。发表于《欧洲计算机视觉会议》，第 55–72 页。施普林格，2024。1[61] Wenzhao Zheng、Ruiqi Song、Xianda Guo、Chenming Zhang 和 Long Chen。GenAD：生成式端到端自动驾驶。发表于《欧洲计算机视觉会议》，第 87–104 页。施普林格，2024。1, 2[62]Chunting Zhou、Lili Yu、Arun Babu、Kushal Tirumala、Michihiro Yasunaga、Leonid Shamis、Jacob Kahn、Xuezhe Ma、Luke Zettlemoyer 和 Omer Levy。Transfusion：使用单一多模态模型预测下一个词元并扩散图像。arXiv预印本 arXiv:2408.11039，2024。3, 4[63] Xingcheng Zhou、Xuyuan Han、Feng Yang、Yunpu Ma 和 Alois C Knoll。OpenDriveVLA：利用大型视觉-语言-动作模型实现端到端自动驾驶。arXiv预印本 arXiv:2503.23463，2025。1, 3[64] Zewei Zhou、Tianhui Cai、Seth Z. Zhao、Yun Zhang、Zhiyu Huang、Bolei Zhou 和 Jiaqi Ma。AutoVLA：一种用于端到端自动驾驶、具备自适应推理和强化微调能力的视觉-语言-动作模型。发表于《第三十九届神经信息处理系统年会》，2025。1, 2, 3, 6

统一自动驾驶中的语言-动作理解与生成

补充材料

附录

A. 动作令牌化

A.1. 对数坐标变换

我们对对数变换坐标空间进行可视化，以便直观比较。我们设计了一种非均匀量化方案，通过首先对各轴上的航点坐标（x， y）分别应用非线性变换，优先保证自车附近的精度。

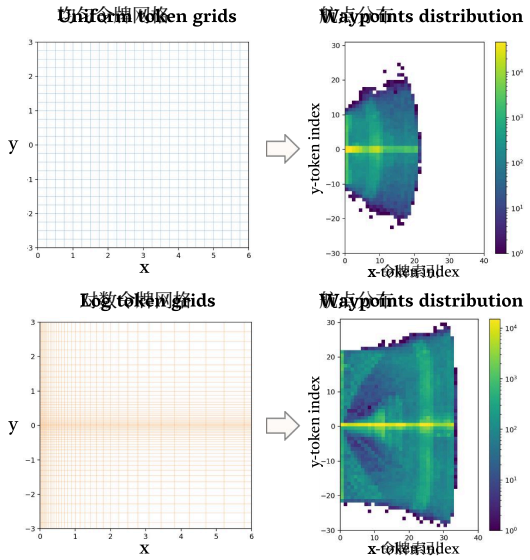


图 S1. 均匀与对数标记网格的比较，以及各网格对应的航点分布。

A.2. 动作标记数量

我们评估了动作标记数量对 Bench2Drive [19] 基准测试中驾驶性能的影响。为此，我们采用一种非均匀量化方案，通过先对航点坐标$(x,\ y)$ 沿每个轴分别进行非线性变换，优先提高自车附近的精度。具体而言，每个坐标 $z \in \{x, y\}$ 均使用带符号的对数函数进行变换：

$$z' = \mathrm{sign}(z) \cdot \log\big(1 + k \cdot |z|\big). \qquad \text{(S1)}$$

随后，我们将对称对数（symlog）缩放因子 k 从 5.0 调整至 10.0，使动作标记数量从 5,656 增至 7,245。参数 k 控制物理坐标（ x 和 y，单位均为米）到变换空间的映射，决定分箱前的压缩程度，进而影响原始坐标空间中的有效箱宽。

表 S1. 闭环评估中动作标记数量（由 k 控制）的影响 [7]。

方法	驾驶得分↑	成功率 (%)↑
k = 5.0（5,656 个词元）	91.01	74.55
k = 10.0（7,245 个词元）	89.85	70.45

A.3. 空间软标签

我们在 Bench2Drive [19] 基准测试上评估空间软标签化中的扩散尺度对驾驶性能的影响，将扩散参数 σ 从 1.2 调整到 3.0。较大的σ 会使高斯平滑范围更广，并将概率分布到更大的邻域，从而生成更软的目标。具体而言，对于真值词元 a_{gt}，我们将所有动作标记 $a \in \mathcal{C}_{\text{action}}$ 上的目标分布 $q(a)$构造为以 a_{gt}的坐标为中心的归一化二维高斯分布：

$$q(a) = \frac{1}{Z} \exp\left(-\frac{\|\mathrm{pos}(a) - \mathrm{pos}(a_{gt})\|_2^2}{2\sigma^2}\right), \qquad \text{(S2)}$$

其中，$\mathrm{pos}(a)$ 将动作令牌映射到其在空间网格中的二维坐标，$\|\cdot\|_2^2$ 表示平方欧氏距离，σ 是控制分布范围的超参数，Z 是确保$\sum_{a \in \mathcal{C}_{\text{action}}} q(a) = 1$的归一化常数。

表 S2. 闭环评估 [7]中空间软标签化的扩散尺度参数 σ 的影响。

方法	驾驶得分↑	成功率（%）↑
$\sigma = 1.2$	91.01	74.55
$\sigma = 3.0$	89.73	69.55

B. 数据集

B.1. Action Dreaming

我们在*Action Dreaming* [37]数据集及其离线非反应式模拟器上开展实验。该模拟器会生成替代的自车轨迹，并评估其在碰撞规避和交通规则遵守方面的可行性。自车轨迹预测通过运动学自行车模型实现，并由 PID 控制器（PDM-lite [41]）控制；控制方式可以是受扰动的真实动作，也可以是根据预定义路径航点和目标速度计算出的 PID 指令。数据集提供模拟器状态和动态物体的短时域预测，以便进行碰撞检测。模拟器支持多种模式（物体/碰撞、加速、减速、目标速度、变道、停车），以产生多样的行为和轨迹。我们使用成功率作为指标。各类别的成功定义各不相同，具体如下：

• 物体（碰撞）：描述驶向或撞击特定物体的任务。首先评估路径。如果专家轨迹与真实梦者轨迹的路径不同（平均位移误差 $ADE > 1.0$），那么当预测路径比专家路径更接近真实梦者路径时，即判定为成功（$ADE_{pred2expert} > ADE_{pred2dreamer}$）。如果梦者路径与专家路径几乎相同（$ADE < 1.0$），则该指令考察的是速度预测是否正确（例如，如果指令是“驶向一个动态物体”，那么准确预测速度而不只是路径就很重要）。此时，成功的定义为 $ADE_{pred2dreamer} < 1.0$，且平均预测速度与真实梦者速度的差距在 30% 以内。

• 加速（提升速度）：对于每个预测速度航点，推导未来各时间步的目标速度，并拟合线性回归以获得斜率 s。令 v 表示序列起始时的当前自车速度。成功定义为 $s > 0.05\,v$。

• 减速（减速）：计算方式与加速相同。成功定义为 $s < -0.05\,v$。

• 目标速度：由于在航点的预测时域内可能无法达到目标速度，因此我们将预测结果与真实动作进行比较，而不是直接与目标速度比较。成功由两条规则定义：第一，根据最后两个航点推断出的预测目标速度与指令目标速度的差距在 20% 范围内；第二，根据最后两个航点推断出的预测目标速度与真实速度对应的最后两个航点的速度的差距在 20% 范围内。由于加速度率受到限制，这一速度可能不同于指令目标速度。

车辆。

• 变道：将预测路径的最终航点与真实梦者路径和真实专家路径的最终航点进行比较。当预测最终位置与梦者的最终位置的距离小于其与专家最终位置的距离时，即视为成功。

• 停车：当整个序列中的最小预测速度低于 0.1m/s 时，即视为成功。

B.2. VQA-DriveLM

SimLingo [37] 的视觉问答数据来源于 DriveLM-Carla [40] 数据集，并使用其数据生成流程生成。问答对取自所采用的数据集，而非最初发布的 DriveLM；训练划分包含 Town 12 中 100 万帧上的 2800 万个问答对。评估遵循 DriveLM 的关键帧选择方式，以聚焦信息丰富的帧；验证划分按答案类型均衡分布。标签由启发式方法自动生成，数据集还包含基于 GPT-4 的释义增强（每个问答最多 20 个变体），以缓解短语级过拟合；这些变体在加载时采样。

B.3. 评论

SimLingo [37]中的解说标签是利用启发式、基于模板的规则，从已保存仿真器状态的一个子集自动生成的。每个标签包含：(1) 带有依据的路线动作——默认为“沿路线行驶”，仅在需要车道偏离的场景中才会替换（例如障碍物、侵入车辆），并针对偏离前、偏离中和偏离后使用分阶段模板；(2) 速度动作，类别为保持停止、立即停车、保持（或保持降低的）速度、提高速度或减速；当在偏离前处于静止状态时，还有一种特殊情形：“变道前等待空隙”；以及 (3) 由智能驾驶员模型 [48]特征得出的速度原因，这些特征用于识别前方目标（车辆/行人/静态交通控制）及其属性/状态，并据此撰写简洁理由（例如由于行人过街、位于红色 SUV后方、因为信号灯为红色/绿色）。接近路口时，还会附加一条提示，概括其他车辆的位置与运动情况（例如路口畅通、车辆正在驶离、对向来车）。

C. 定性结果

图 S2 展示了闭环评估中更多的定性结果，说明了我们的模型在多种驾驶场景中的表现。

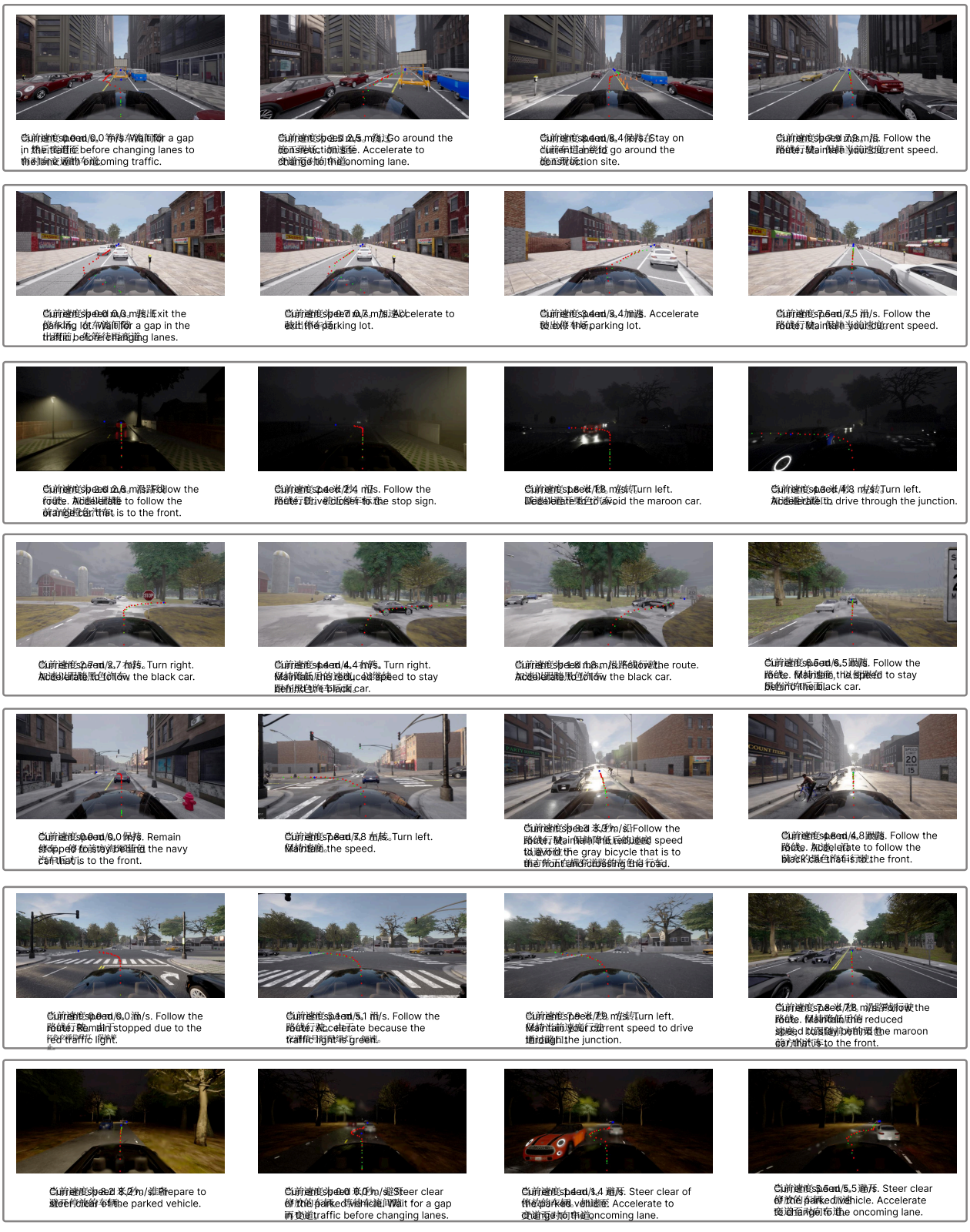


图 S2。我们的模型在 CARLA 模拟器中进行闭环评估时的定性结果。该图展示了具有代表性的驾驶场景，例如通过交叉路口和避开障碍物。