NavAI：面向虚拟现实环境导航任务的通用大语言模型框架

Xue Qin维拉诺瓦大学计算科学系美国维拉诺瓦
xue.qin@villanova.edu

Matthew DiGiovanni维拉诺瓦大学计算科学系美国维拉诺瓦 mdigio02@villanova.edu

arXiv:2601.03251v1 [cs.SE] 6 Jan 2026

摘要——**导航是虚拟现实（VR）自动化探索的一项基础任务。现有技术主要侧重于 360 度图像数据集和三维模拟器中的路径优化，无法直接应用于沉浸式虚拟现实环境。为弥补这一不足，我们提出 NavAI，这是一种通用的大语言模型导航框架，可在各种 VR应用程序中支持基本动作和复杂的目标导向任务。我们通过目标导向任务和探索性任务，在三个不同的虚拟现实环境中评估 NavAI。结果表明，该框架准确率较高，在目标导向任务中的成功率达到 89%。我们的分析还揭示了完全依赖大语言模型的局限性，尤其是在需要动态目标评估的场景中。最后，我们讨论了实验中发现的局限性，并为未来研究方向提供了见解。**

索引词——**虚拟现实、大语言模型、导航**

I. 引言

据行业预测，消费级虚拟现实和增强现实设备的出货量预计将从2023年的1,781万台增长至 2026 [1]年的3,088万台。凭借模拟物理环境的能力，虚拟现实如今已广泛应用于房地产导览、科学教育和制造业培训等领域。为确保大规模应用的质量，研究人员和开发者提出并应用了针对虚拟现实应用程序的自动化测试 [2] ，重点关注探索测试 [3],[4], 、单元测试 [5],和隐私增强 [6]等领域。

在所有这些自动化测试任务中，成功导航是探索的基本起点之一，有助于用户有效地与环境互动。在虚拟现实空间中导航——例如在陌生城市或茂密森林中——若没有引导可能颇具挑战；仅凭地图不足以解决空间导航、用户方位感和参与度方面的问题。目前最先进的导航解决方案称为视觉语言导航（VLN），要求智能体遵循人类语言指令，在此前未见过的环境中进行导航。视觉语言导航的类型包括目标导向型 [7] 和路线导向型 [8], [9], ，用户可以通过单轮或多轮交互使用视觉语言导航 [10]。

目前讨论最多的视觉语言导航技术，都是在通过360 度虚拟导览呈现的真实环境中，或在传统三维模拟器中开发和测试的。然而，这些环境与完全沉浸式的虚拟现实场景有显著差异。现有的虚拟现实导航支持通常仅限于预先编程环境中嵌入的预定义导览 [11] ，或为特定虚拟现实应用程序量身定制的内置人工智能功能 [12]。随着虚拟现实行业持续发展，许多新的复杂环境将不断涌现，也将设计出新的交互体验。我们迫切需要稳健且可扩展的导航解决方案，使其能够适用于当前及未来的虚拟现实应用程序。

本文提出了一种基于大语言模型（LLM）的导航解决方案框架，名为 NavAI，可应用于多种虚拟现实环境。该框架利用大语言模型的广泛知识，通过从 VR应用程序中捕获的屏幕截图来解读虚拟世界。具体而言，NavAI 首先使用综合解释器组件分析视觉场景，并生成视觉解读和文本描述。随后，多智能体决策投票器解析用户的导航请求，并将其归类为新定义的导航类型之一。该投票器还用于判断预期目标是否已经实现。如果导航任务仍在进行，决策到控制映射会将意图匹配到相应的控制函数，并执行导航动作。NavAI 目前同时支持基本动作指令和复杂的目标导向导航任务。

我们在三种不同的虚拟现实环境中评估了 NavAI 的有效性，并为目标驱动型和探索性场景分配了不同的导航任务。该框架在“向前移动”和“向左转”等基本动作任务上表现稳定，成功执行了全部 21 个基本运动指令，且开销极低（平均值为 0.74 秒）。NavAI 在直接目标导向任务中也取得了 89% 的成功率（18 个任务中完成 16 个）。然而，其效率因环境而异，解释组件和决策投票器会引入明显的延迟，尤其是在需要灵活停止条件的任务中，例如探索性扫描。最后，我们深入讨论了 NavAI 的可用性，并针对观察到的局限性提出了潜在的未来解决方案。

本文的贡献如下：

- 我们设计了一种具有泛化能力的框架，能够执行

在任何VR环境中的导航任务，将用户请求与大语言模型（LLM）所作出的理解和决策连接起来。

- 我们在多种虚拟现实环境中，针对不同任务类型开展了三组实验。结果表明，NavAI在各种场景中均取得了较高的成功率和合理的效率。
- 我们分析了基于大语言模型的导航的可用性，着重指出当前的局限，并提出未来改进性能和实时适用性的方向。

II. 框架

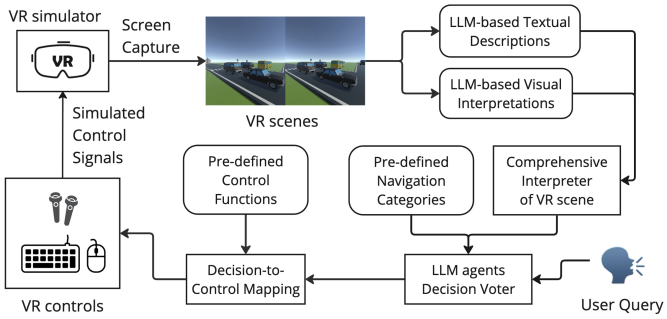

图1. NavAI框架概览

本节将介绍NavAI框架。该框架会自动捕获并处理来自VR应用程序的立体图像，分析用户查询，然后发送相应的模拟控制信号。如图1所示，NavAI包含以下用于导航自动化的核心组件：1）综合解释器——利用大语言模型生成VR场景的文本描述和视觉解读。2）预定义导航类别——涵盖常见导航类型及其子类型的数据集，用于指导用户查询分类。3）决策投票器——汇总多个大语言模型智能体的决策，以对预期的导航动作进行分类。4）决策到控制映射——将分类后的查询与相应的控制函数相匹配，并在VR模拟器上执行这些函数。其他辅助组件，例如屏幕捕获模块和用户查询接收器，因并非导航自动化流程的核心内容，不在本节讨论范围内。

A. 综合解释器

由于所有 VR 场景都是此前从未训练过或见过的，综合解释器的主要目标是让大语言模型生成对 VR环境的基本理解，以便用户能够结合上下文轻松提出查询。认知理论——尤其是双重编码理论 [13]——指出，人脑在处理物体时会同时编码视觉信息和语言信息。基于这一观点，我们在分析 VR 截图（代表用户的视场，FOV）时为大语言模型分配了两类任务：视觉解释和文本描述。

具体而言，对于文本描述，我们采用现有 VR 研究 [14] 中的思维链设计，收集观察到的物体名称及其特征描述。对于视觉解释，我们在截图上自动叠加网格，并要求大语言模型提供坐标信息。考虑到时间成本，我们没有将屏幕上的每个物体逐一分割并裁剪出来。我们还认为，坐标更有用，因为它们可以间接表示物体位置与用户站立点或观察点之间的空间关系。最后，我们将这两类信息汇总为上下文，供决策投票器决定应执行哪项导航动作。

B. 预定义导航类别

Wu等人通过一项综述，根据人类指令构建了现有的导航任务分类体系 [15]。该综述考察了嵌入具身人工智能的三维模拟器中的各种导航任务，其中内置的人工智能智能体经过大量训练，能够在环境中执行动作并与环境互动。本文重点关注支持自动化探索的导航任务，尤其是前往访问视场内可见或存在于VR场景中的现场物体。图2展示了我们完善后的分类体系，其中包含三种主要类型。

- 第一类是**语义解释器**，用于处理“我在哪里？”等非导航查询。这类任务无需发送控制信号或触发移动即可完成。
- 第二类是**动作导航器**，用户会明确请求执行一项基本动作，例如“向前移动”或“左转”。该类别分为两种子类型：移动执行器和稳定性执行器，因为它们调用不同组的控制函数。
- 第三类是**目标导航器**，用户会指定一个与视场中可见目标对象相关联的导航请求，例如“我想去黄色巴士那里”。这种类型更为复杂，因为用户没有指定具体动作。此任务通常需要一系列动作，并会在多个转弯次数内完成。我们进一步将其分为两种子类型：直接目标导航器，指当前视场中可见目标的导航目标；探索性目标导航器，则需要先扫描或搜索目标，再执行导航。

C. 决策投票器

决策投票器汇总综合解释器和预定义导航类别的输出，并根据用户查询为导航系统确定合适的响应。具体而言，它会先将查询归类为某一导航类别，然后判断用户的需求是否已得到满足，或任务目标是否已达成。如果目标尚未达成，投票器便进入动作决策阶段，以引导下一步。

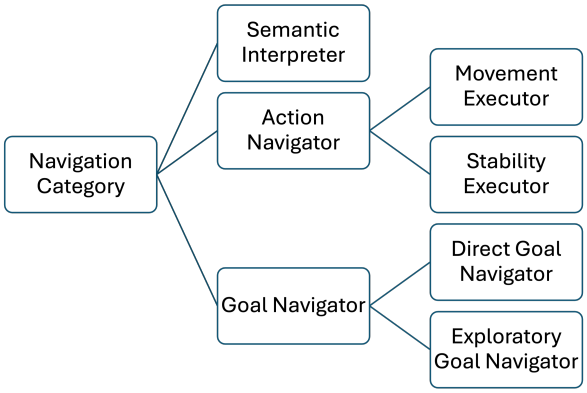

图2. 探索的导航类别

算法1导航决策投票器

过程DECISIONVOTER(*query*)
  $ctx \leftarrow$ GETSCENECONTEXT()
  $cat \leftarrow$ 分类类别($query, ctx$)
  如果 $cat =$ 语义解释器 **则**
    返回语义解释器_
  否则如果 $cat =$ 动作导航器 **那么**
    返回动作导航器_
  else
    $v \leftarrow$ 智能体投票($query,\ ctx$)
    如果 $v =$已到达 **那么**
      停止导航()
      返回已达成目标
    else
      返回目标进展_
    结束 if
  结束 if
结束过程

详细算法如算法1所示。DECISIONVOTER 程序接收以自然语言表达的用户查询，并确定合适的导航动作。首先，在第2行通过方法*GetSceneContext*()获取虚拟现实场景上下文。随后，在第3行将查询归为三种类别之一。如第4行至第7行所示，如果类别类型符合前两种情况，程序将返回语义解释器或动作导航器。对于基于目标的导航，多个大语言模型智能体会根据场景上下文，对用户查询中所指的预期目标是否已达成进行投票。如果确认目标已达成，导航将停止，系统返回已达成目标；否则，系统返回目标进展，后续的决策到控制映射组件将决定并执行下一项控制动作。

D. 决策到控制映射

确定目标状态后，最终的决策到控制映射组件会从一组预定义基本动作中识别并执行适当的控制函数。

在实践中，这种映射将用户的导航意图与 VR环境中的低级控制命令连接起来。我们发现，OpenAI 的函数调用功能非常适合这一需求，因为它允许我们将动作函数公开为 JSON 对象，并让模型生成最合适的调用，包括参数和值。这种机制清晰地分离了高级自然语言解释与具体执行。预定义基本动作的完整集合分为移动和稳定性两类，如表 I 所示。移动动作控制位置变化，例如向前移动；稳定性动作则让用户保持原位，同时调整方向或视角，包括进行360度场景扫描以全面探索。

表 I 预定义的基本导航动作

Type	动作函数
移动	move_forward(duration) move_left(duration) move_right(duration)
稳定性	原地向左旋转_____（持续时间） 原地向右旋转_____（持续时间） 向上看_（持续时间） 向下看_（持续时间） scan_360()

III. 评估

在本节中，我们评估 NavAI框架的有效性。具体而言，我们测量不同任务类型在多个环境中的成功率，并分析执行时间，以评估整体可用性。我们不将语义解释器任务纳入评估，因为这些任务主要关注人工智能的图像识别准确率，而非导航性能。

A. 数据集与设置

我们选择了三种不同的 Unity VR 资产作为测试环境。第一种是 高速公路 场景，其中包含多辆汽车和一个交叉路口。第二种是 船场景，其特点是在狭窄空间内设有开放式甲板。第三种是 乡村住宅，这是一个包含多个房间和物体的室内环境。为运行这些资产，我们使用 Unity 模拟器来模拟虚拟现实环境，无需实体头戴式设备。该模拟器非常适合功能测试，这与我们评估导航决策算法有效性的重点相一致。对于综合解释器 和决策到信号映射组件，我们根据 Qi 等人 [14]的研究，采用 GPT-4o [16], 。在决策投票器 组件中，我们整合了三种不同的大语言模型，用于主要的投票策略：GPT-4o [16],、Grok-2 [17],和 Gemini-2.5-Flash [18]。

B. 动作导航器任务的有效性

表 II 报告了当用户的查询属于动作导航器类别时，NavAI 在三种虚拟现实环境中的执行时长。每个动作的设计时长为 2

秒，且均成功执行。所有动作和环境的平均测得开销约为 0.74 秒。该开销反映了 GPT-4o 将请求映射为信号函数、在 NavAI 中执行已定义函数，以及 Unity 环境带来的模拟和渲染成本的综合耗时。总体而言，结果表明，NavAI 能够在多种虚拟现实场景中稳定、高效地处理并执行基本的动作查询。

表 II 三种虚拟现实环境中动作导航器查询的执行时长（秒）及平均开销。

动作	高速公路	乡村住宅	Ship	开销
向前移动	2.68	2.74	2.72	0.71
向左移动	2.72	2.62	2.61	0.65
向右移动	2.60	3.49	2.75	1.28
向左旋转	2.59	2.82	2.63	0.68
向右旋转	2.45	2.80	2.65	0.63
向上看	2.76	2.95	2.70	0.80
向下看	2.60	2.91	2.67	0.73
总体平均值	2.63	2.90	2.67	**0.74**

C. 直接目标导航任务的有效性

我们进一步评估了 NavAI 在直接目标导向导航任务中的有效性，重点考察准确性（成功率）和效率（执行时间）。我们在三个虚拟现实环境中测试了具有代表性的任务。在高速公路场景中，任务是“到达黄色巴士的后方，并绕开其他车辆，避免撞上它们。”在乡村住宅中，任务是“穿过左侧的门口，进入卧室。”在船舶环境中，任务是“走到你右侧的大炮旁。”表 III、IV、V 展示了各环境中多次尝试的执行结果，以及按每次导航转弯统计的详细效率测量结果。

在所有环境中，18 次尝试最终有 16 次成功，表明 NavAI 能够可靠地到达指定目标。然而，效率分析显示，导航会产生显著的时间成本。在三个测试环境中，每次转弯的平均执行时间约为 49.4 秒，每次尝试平均需要 8 次转弯。这意味着，完成一次典型的目标导向导航任务总计最多可能需要 395 秒。在每次转弯中，综合解释器占据主要耗时，约为 72%（35.6 秒），其中包括视觉解释（19.8 秒，占 40%）和文本描述（15.8 秒，占 32%）。决策投票器约占 17%（8.3 秒），而动作执行约占 9%（4.6 秒）。总之，基于目标的导航效率主要受解释和决策开销的限制，而动作执行所需的时间相对较短。

D. 探索性目标导航任务的有效性

我们在三个环境中评估了360度探索扫描任务。表 VI展示了每个环境中的三次尝试。第 3 列表示扫描被视为完成前的旋转次数。平均而言，完成一次扫描耗时 41.1 秒，不到一分钟。此外，每次旋转仅耗时 4.6 秒，这表明框架 NavAI 能够及时完成任务。然而，不同环境中的旋转次数差异很大，少则 2 次，多则 12 次。这种差异表明，大语言模型在探索性任务中难以始终如一地判断合适的停止条件。

IV. 讨论

本节将根据 NavAI 的评估结果，讨论采用大语言模型解决方案执行导航任务的可用性，并指出未来工作中需要解决的不足。

A. 可用性讨论

动作导航器。该组件始终展现出较高的准确性，能够正确分析 21 条用户查询的意图，并成功将其映射到预定义动作。此外，其开销较低（约 0.72 秒），表明它在处理实时导航请求方面具有很大潜力。总之，这些结果凸显了动作导航器可靠的可用性，以及对整体性能的影响微乎其微。

综合解释器。综合解释器能够准确感知导航上下文，帮助人工智能模型做出可靠的决策。因此，NavAI 成功完成了 18 项直接面向目标的导航任务中的 16 项。然而，该组件也带来了显著的延迟，约占每轮导航执行时间的 72%，其中视觉解释占 40%，文本描述占 32%。这表明这两个子流程都是主要瓶颈，需要进行优化以提升系统整体性能。总而言之，尽管综合解释器适用于功能测试等对时间不敏感的任务，但其较高的延迟限制了它在实时支持等实时应用中的可用性。

决策投票器。决策投票器整合了多个大语言模型，以实现稳健的决策，在 18 次尝试中成功 16 次，成功率高达 89%。平均额外开销为 8.3 秒，主要用于向三个独立的人工智能模型发送请求并等待其响应。因此，每个模型的平均时间成本为 2.76 秒，更易于接受。然而，在探索性扫描任务中，旋转次数从 2 次到 12 次不等，表明大语言模型难以持续确定扫描终止条件。这种不一致性暴露了模型在理解一系列图像输入方面的局限性，也表明可能需要更复杂的上下文管理。总之，尽管决策投票器在处理离线测试方面展现出有效性，但其在实时导航支持中的可用性有所下降。

总体可用性评估。在不同环境中，NavAI框架展现出较高的成功率（约

表 III 高速公路环境中直接目标导向导航的评估结果。

尝试	成功	# 转弯次数	决策投票器（秒）	综合解释器		动作（秒）	总计/回合（秒）
				视觉（秒）	文本（秒）		
1	Yes	9	8.35	13.02	15.00	3.60	40.51
2	Yes	11	10.33	14.60	16.13	4.52	46.23
3	Yes	10	11.61	13.62	16.18	5.31	47.32
4	Yes	7	13.48	19.78	17.90	3.99	55.88
5	Yes	7	8.26	15.39	16.80	3.82	44.95
6	Yes	13	9.09	17.97	17.92	4.39	50.09
平均值	-	**9.5**	**10.19**	**15.73**	**16.66**	**4.27**	**47.50**

表 IV 乡村住宅环境中的直接目标导向导航评估结果

T.

尝试	成功	# 转弯次数	决策投票器（秒）	综合解释器		动作（秒）	总计/回合（秒）
				视觉（秒）	文本（秒）		
1	Yes	5	6.38	17.37	17.32	4.75	46.65
2	Yes	5	5.39	11.23	13.58	3.49	34.17
3	Yes	5	5.53	10.94	15.38	3.08	35.37
4	Yes	5	19.55	17.38	17.38	5.32	63.43
5	Yes	4	7.57	7.12	20.07	7.49	42.59
6	No	19	6.49	14.63	15.92	6.98	44.77
平均值	-	**7.2**	**8.82**	**13.11**	**16.61**	**5.18**	**44.50**

表 V 船舶环境中直接目标导向导航的评估结果。

尝试	成功	# 转弯次数	决策投票器（秒）	综合解释器		动作（秒）	总计/回合（秒）
				视觉（秒）	文本（秒）		
1	Yes	17	7.35	29.53	18.42	5.33	62.00
2	No	4	5.66	20.63	18.31	3.62	49.30
3	Yes	6	6.34	31.51	20.27	3.37	63.14
4	Yes	4	5.18	14.13	15.58	4.47	40.02
5	Yes	7	6.45	27.14	19.30	4.32	58.72
6	Yes	6	6.03	33.92	18.75	4.33	64.40
平均值	-	**7.3**	**6.17**	**26.48**	**18.77**	**4.24**	**56.26**

探索性扫描任务的评估结果

Env.	注意力旋转		时间/旋转（秒）	决策（秒）	总计/注意（秒）
高速公路	1	4	3.65	4.74	19.35
	2	5	3.75	4.92	23.64
	3	5	3.42	3.33	20.44
国家	1	10	5.21	12.67	64.76
	2	8	5.31	5.92	48.42
	3	2	2.87	5.74	14.02
Ship	1	9	4.90	6.66	50.77
	2	10	5.12	8.39	59.57
	3	12	5.32	11.15	75.04

89%），并为大语言模型驱动的导航提供了有价值的见解。然而，基于大语言模型的解读和决策成本限制了其整体效率，而不一致的终止行为进一步降低了扫描任务的效率。总而言之，NavAI解决方案适用于离线测试和评估，但若不进行进一步优化，其可用性在实时应用或实时支持应用中有限。

B. 未来研究见解

为了解决综合解释器在可用性方面的局限，主要目标是缩短文本生成和图像识别的处理时间。目前，NavAI 完全依赖远程大语言模型来完成这些任务，因此延迟较长。一个很有前景的方向是将处理过程分解为子任务，并开发轻量级的本地AI模型，以更高效地处理这些任务。关键的研究挑战在于如何定义这些子任务，以及如何构建快速、准确的本地模型。这与近期基于边缘计算、低延迟大语言模型设计的趋势相契合 [19], [20]。

对于决策投票器而言，主要的可用性问题源于与多个大语言模型智能体进行顺序通信，以及缺乏有效的上下文管理。第一个问题——顺序处理导致的延迟——可以通过采用并行推理和减少令牌用量来缓解，这与 Plurals [21] 和 Live- Mind [22]中提出的策略类似。第二个问题则需要设计高效机制，以管理和同步多个智能体之间的上下文。

V. 结论

本文介绍了 NavAI，这是一种由大语言模型驱动的框架，旨在自动执行虚拟现实环境中的面向任务的导航。该框架整合了虚拟场景理解、导航查询分类、多智能体决策投票和控制映射，用以模拟导航动作。NavAI 在支持动作导航器和直接目标导向任务方面展现出显著成效，在三个虚拟现实环境中实现了 89% 的成功率。然而，场景理解和决策过程带来的额外开销显著降低了导航的可用性。最后，我们讨论了这些局限，并提出了未来改进方向。

参考文献

[1] Fortune Business Insights，《虚拟现实市场规模》，https://www.fortunebusinessinsights.com/industry-reports/virtual-reality-market-101378, 2025, [在线；访问日期：2025-07-28]。[2] D. E. Rzig、N. Iqbal、I. Attisano、X. Qin 和 F. Hassan，“真实环境中的虚拟现实（vr）自动化测试：一项基于 Unity 的虚拟现实应用案例研究”，载于第*32*届*ACM SIGSOFT*国际软件测试与分析研讨会论文集，系列出版物：ISSTA 2023。美国纽约州纽约：美国计算机协会，2023年，第1269–1281页。 [在线]。链接：https://doi.org/ 10 1145 3597926 3598134 。/。X. Wang，“[3] Vrtest：用于自动测试虚拟现实场景的可扩展框架”，载于*ACM/IEEE*第*44*届国际软件工程会议论文集：伴随论文集，系列出版物：ICSE ’22。美国纽约州纽约：美国计算机协会，2022年，第232–236页。 [在线]。链接：https://doi.org/ 10 1145 3510454 3516870 。/。X. Qin 和 G. Weaver，“利用生[4] 成式人工智能进行虚拟现实探索测试：一项案例研究”，载于第*39*届*IEEE/ACM*国际自动化软件工程会议研讨会论文集，系列出版物：ASEW ’24。美国纽约州纽约：美国计算机协会，2024年，第228–232页。 [在线]。链接：https://doi.org/ 10 1145 3691621 3694955 。/。A. Gil、T. Figueira、E.[5] Ribeiro、A. Costa 和 P. Quiroga，“虚拟现实应用的自动化测试”，载于*HCI International 2020*——最新研究海报，C. Stephanidis、M. Antona 和 S. Ntoa 编。瑞士查姆：施普林格国际出版公司，2020年，第145–149页。[6] J. Y. Kim、C. Zuo、Y. Zhao 和 Z. Lin，“Autovr：用于检测虚拟现实应用中敏感数据流暴露的自动化用户界面探索”，2025 [ ] 2508 12187 。在线。链接：[7] https://arxiv.org/abs/。Y. Qi、Q. Wu、P. Anderson、X. Wang、W. Y. Wang、C. Shen 和 A. van den Hengel，“REVERIE：真实室内环境中的远程具身视觉指代表达”，载于*2020*年*IEEE/CVF*计算机视觉与模式识别会议（*CVPR*）。美国加利福尼亚州洛斯阿拉米托斯：IEEE计算机学会，2020年6月，第9979–9988页。 [在线]。链接：https://doi.ieeecomputersociety.org/ 10 1109 42600 2020 01000 。/CVPR。。[8] P. Anderson、Q. Wu、D. Teney、J. Bruce、M. Johnson、N. Sunderhauf、I. Reid、S. Gould 和 A. van den Hengel，“视觉与语言导航：解读真实环境中以视觉为基础的导航指令”，载于*2018*年*IEEE/CVF*计算机视觉与模式识别会议（*CVPR*）。美国加利福尼亚州洛斯阿拉米托斯：IEEE计算机学会，2018年6月，第3674–3683页。 [在线]。链接：https://doi.ieeecomputersociety.org/ 10 1109 2018 00387 。/CVPR。。A.[9] Ku、P. Anderson、R. Patel、E. Ie 和 J. Baldridge，“跨房间导航：通过密集时空定位实现多语言视觉与语言导航”，载于*2020*年自然语言处理实证方法会议（*EMNLP*）论文集，B. Webber、T. Cohn、Y. He 和 Y. Liu 编。在线：计算语言学协会，2020年11月，第4392–4412页。 [在线]。链接：https://aclanthology.org/2020.emnlp-main.356/ [10] K. Nguyen、D. Dey、C. Brockett 和 B. Dolan，“通过模仿学习和间接干预实现基于视觉的导航与基于语言的辅助”，载于*2019*年*IEEE/CVF*计算机视觉与模式识别会议（*CVPR*），2019年，第12 519–12 529页。

[11] J. Collins、K. M. Nicholson、Y. Khadir、A. Stevenson Won 和 S. Azenkot，“为盲人提供人工智能导览，以提升社交虚拟现实的无障碍体验”，载于第*26*届国际*ACM SIGACCESS*计算机与无障碍会议论文集，系列出版物：ASSETS ’24。美国纽约州纽约：美国计算机协会， 2024 [ ] 10 1145 3663548 3688498 。在线。链接：[12] https://doi.org/。/。M. Konenkov、A. Lykov、D. Trinitatova 和 D. Tsetserukou，“Vr-gpt：用于智能虚拟现实应用的视觉语言模型”，2024年。 [ ] 2405 11537 在线。链接：[13] https://arxiv.org/abs/。A. Paivio，《心理表征：双重编码方法》。牛津大学出版社，1990年9月。[在线]。链接：https://doi.org/ 10 1093 9780195066661 001 0001 。/acprof:oso/。。Z. Qi、[14] H. Li、H. Qin、K. Peng、S. He 和 X. Qin，“利用大语言模型进行虚拟现实探索测试：一项案例研究”， 2025 [ ] 2501 05625 。在线。链接：[15] https://arxiv.org/abs/。W. Wu、T. Chang 和 X. Li，“视觉语言导航：综述与分类体系”， 2022 [ ] 2108 11544 。在线。链接：[16] https://arxiv.org/abs/。OpenAI，“Gpt-o技术报告”，https://openai.com/index/gpt-o-system-card/，2024年，访问日期：2025年8月18日。[17] xAI，“Grok-2发布介绍”，https://x.ai/blog/grok-2，2024年，访问日期：2025年8月18日。[18]G. DeepMind，“Gemini 2.5技术概述”，https://deepmind.google/models/gemini/flash/，2025年，访问日期：2025年8月18日。[19] D. Pollini、B. V. Guterres、R. S. Guerra 和 R. B. Grando，“降低基于大语言模型的自然语言命令处理在机器人导航中的延迟”，2025 [ ] 2506 00075 。在线。链接：[20] https://arxiv.org/abs/。Y. Zheng、Y. Chen、B. Qian、X. Shi、Y. Shu 和 J. Chen，“边缘大语言模型综述：设计、执行与应用”，*ACM*计算调查，第57卷，第8期，2025年3月。 [在线]。链接：https://doi.org/ 10 1145 3719664 。/ J. Ashkinaze、E. Fry、[21] N. Edara、E. Gilbert 和 C. Budak，“Plurals：通过模拟社交群体引导大语言模型的系统”，载于*2025*年*CHI*计算系统中的人为因素会议论文集，系列出版物：CHI ’25。美国纽约州纽约：美国计算机协会，2025年。 [在线]。链接：https://doi.org/ 10 1145 3706598 3713675 。/。C. Chen、G. L. Zhang、X. Yin、C.[22] Zhuo、U. Schlichtmann 和 B. Li，“Livemind：具备同步推理能力的低延迟大语言模型”，2024年。 [在线]。链接：https://arxiv.org/abs/2406.14319