arXiv:2601.14679v1 [cs.MM] 21 Jan 2026

HCVR场景生成：面向扩展重定向行走的高兼容性虚拟现实环境生成

Yiran Zhang、Xingpeng Sun 和 Aniket Bera

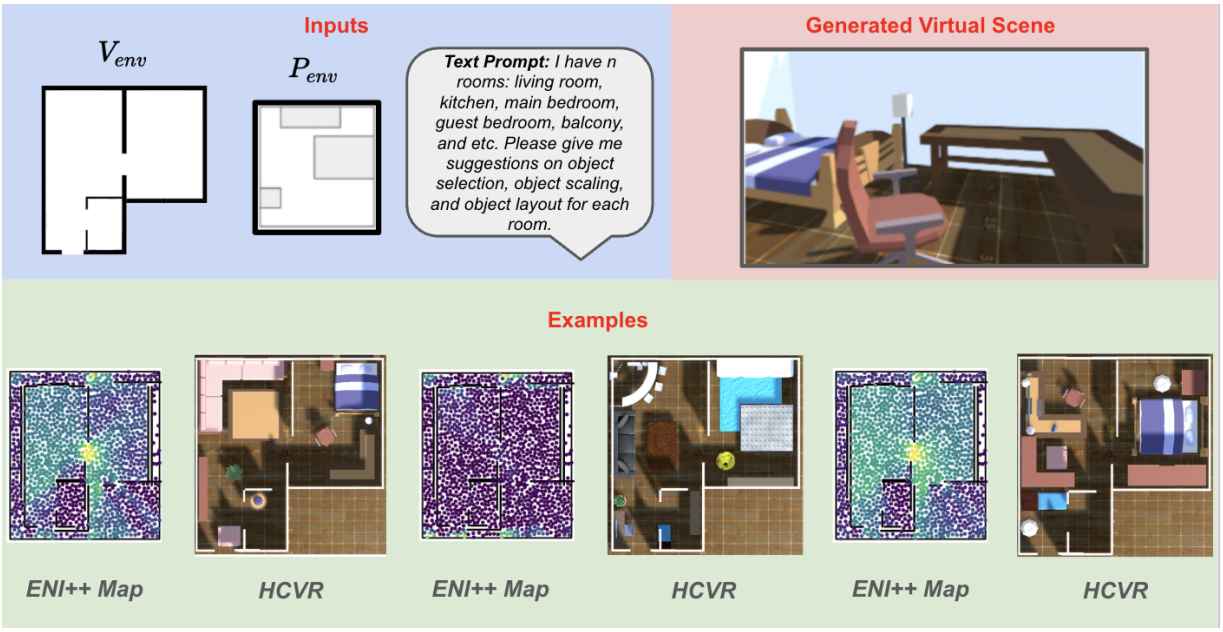


图 1：这是我们面向虚拟现实导航优化的 HCVR虚拟场景生成框架概览，该框架兼容基于对齐的重定向行走控制器，可为用户提供独特的自然行走界面。给定一对虚拟V_{env} 和物理P_{env} 场景以及一条文本提示，我们的模型会处理 ENI++ 分数图，以评估各区域的场景兼容性。ENI++ 场景图展示了采样点，其颜色从黄色（ENI++ 分数高，表示不利于重定向行走）到紫色（ENI++ 分数低，表示更有利于重定向行走）渐变。蓝色方块表示输入 HCVR 架构的内容，红色方块展示了生成的虚拟场景示例。绿色方块展示了基于不同 ENI++ 分数图生成的其他一些 HCVR 定向场景。

摘要

自然行走能够增强用户在虚拟环境（VEs）中的沉浸感，但物理空间限制和障碍物会妨碍探索，尤其是在大型虚拟场景中。重定向行走（RDW）技术通过轻微操控虚拟摄像机，引导用户避开预定义虚拟环境中的物理碰撞，从而缓解这一问题。然而，当物理环境和虚拟环境之间存在显著的几何差异时，RDW 的效果会大幅降低，导致无法避免的碰撞。现有的场景生成方法主要关注物体关系或布局美学，往往忽视有效 RDW 所必需的物理兼容性。为此，我们提出 HCVR（高兼容性虚拟现实环境生成），这是一种专为基于对齐的重定向行走控制器优化、能够生成虚拟场景的新型框架。HCVR 首先采用 ENI++，这是一种新颖的边界敏感度量，通过比较旋转敏感可见性多边形来评估物理空间与虚拟空间之间的不兼容性。在 ENI++兼容性图和用户提示的引导下，HCVR 利用大语言模型（LLM）进行上下文感知的 3D资产检索和初始布局生成。随后，该框架会策略性地调整物体选择、缩放和通过放置来最大化对虚拟不兼容区域的覆盖，从而有效引导用户沿重定向行走可行路径行进。评估物理碰撞和布局质量的用户研究表明，HCVR在使用HCVR生成的场景时效果显著：与结合重定向行走的基于大语言模型的生成相比，物理碰撞减少了22.78倍，ENI++ 分数也低了35.89%；此外，用户对布局设计的反馈评分提高了12.5%。

关键词：虚拟现实、3D场景生成、大语言模型空间表征、可导航性指标、运动、重定向行走

1 引言

在虚拟环境中自然行走，为VR用户带来了独特的探索体验 [2, 47]。这使用户能够自然地与虚拟环境互动，从而提升任务表现[34]。先前的研究提出了重定向行走（RDW）技术，并指出该技术能够让用户在受限物理空间中自由探索虚拟环境 [18, 46,50, 51]。重定向行走算法会调整虚拟场景的旋转和平移，将用户引导至无碰撞路径，同时不被用户察觉。这些重定向行走方法，如向中心引导 [33], 、环绕或以物体为目标 [32], 、基于势场的重定向 [46], 、基于强化学习的重定向 [24, 36], 和基于对齐的重定向 [50], 已被证明

在各种场景中都有效。然而，为了避免用户在虚拟体验过程中察觉到重定向，平移和旋转的重定向幅度受到人类感知阈值的限制 [13, 19, 21, 35, 53] [17]。因此，当用户的物理空间与设计的虚拟空间极不兼容时，碰撞便无法避免。

先前研究已证实，环境的几何结构和形状会显著影响用户导航行为 [8,11,37]。此外，研究还探讨了基于不同虚拟环境布局实现无碰撞导航的可行性 [49]。因此，我们建议不要直接在现有虚拟场景中应用重定向行走控制器，而是生成与用户物理环境相适配的虚拟场景，充分发挥重定向行走控制器的优势，为用户带来安全的虚拟行走体验。

现有的3D场景生成方法主要关注视觉效果和空间合理性。有些生成场景是在预定义设计的基础上修改而来 [48,56]，另一些则通过训练来学习物体之间的空间关系 [55,58]。然而，这些方法受到多方面限制：一方面，需要投入大量精力，手动追踪各种物体类别之间的空间关系；另一方面，对数据集所包含布局的多样性要求较高。借助大语言模型（LLM）的强大上下文理解能力，近期基于大语言模型的场景生成方法 [57]显著减轻了明确指定物体空间关系的负担，同时无需进行布局标注或训练，即可生成更多样化的场景配置。尽管取得了这些进展，现有场景生成方法并非专为支持虚拟现实中的自然行走界面而设计。为弥补这一不足，我们提出了HCVR架构，用于生成兼具空间连贯性且针对重定向行走（RDW）应用进行了优化的虚拟场景。我们还展示了HCVR生成的场景在布局质量方面的优势，以及其促进无碰撞重定向行走路径的能力。

图 2 展示了 HCVR 架构的概览。给定用户的物理环境，HCVR 使用一种对边界敏感且可解释的环境导航不兼容性指标（ENI++）来评估虚拟平面图的重定向行走可行性，该指标是对原始 ENI 指标的扩展 [52]。生成 ENI++ 分数图以可视化虚拟空间中不同区域的兼容性后，HCVR 使用 BUDAS [12] 对虚拟平面图中的房间进行分割。随后，HCVR 利用大语言模型选择合适的 3D 资产，定义所选物体之间的关系约束，并以视觉协调且空间一致的方式缩放这些资产。此外，HCVR 还集成了越界检查器（OOB），以确保所有放置的物体均位于相应的虚拟房间边界内。最后的调整阶段会进一步优化物体定位和缩放，以最大限度地增加适合重定向行走的虚拟区域，从而提升实现有效重定向行走的潜力。

总而言之，我们的主要贡献如下：

- 一种新颖的交互式虚拟现实环境生成框架（HCVR），利用 ENI++ 进行大语言模型引导的场景生成，从而生成设计合理且适用于重定向行走的虚拟场景。
- ENI++指标在原始 ENI [52] 的基础上，纳入了虚拟边界和障碍物边界的约束，以及旋转敏感的可见性多边形比较，从而提供了一种更易描述且更有效的物理空间与虚拟空间兼容性度量。
- 针对 ENI = 16 生成的 VR场景的满意度以及输出场景中的无碰撞导航开展了广泛的用户研究实验（N++ ）。在整体布局方面，HCVR 导向的场景比基线大语言模型生成的场景表现高出 7.75% 的反馈。在物体选择方面，它也获得了同等程度的反馈，在物体缩放方面则高出 3.38%。尽管定位与旋转的满意度降低了 3.72%，但在五项设计各异的实验中，与在大语言模型生成的场景中引导用户相比，在 ENI++ 生成的场景中引导用户可避免多 22.78 倍的物理碰撞。

2 背景与相关工作 2.1 虚拟现实场景生成

2.1.1 场景布局生成

随着深度学习技术的融入，场景布局生成取得了显著进展，尤其关注场景图和标签集等结构化输入。早期方法（如 LayoutVAE）采用变分自编码器，根据标签集随机建模物体布局，从而实现概率式生成 [22]。随后，LayoutTransformer [16]等方法利用自注意力机制，捕捉不同领域中布局元素之间的上下文关系，支持生成和补全任务。近期研究着重于概念和空间多样性：LT-Net [55] 引入了基于Transformer的架构，从场景图中推断隐式物体关系；而新视角预测模型 [31] 则通过物体上下文转换模块，探索了跨相机视点的布局合成。对于三维场景，SceneCraft 展示了通过多视图代理图和扩散模型实现布局引导生成的方法，使生成复杂室内环境不再受限于单个房间 [56]。这些进展凸显了一种趋势，即结合几何推理与语义理解，以应对二维与三维布局合成的挑战。然而，这些方法仍受限于必须预先定义空间关系，或训练数据中空间多样性有限。

2.1.2 LLMLayout 布局生成方法

近年来，大语言模型（LLM）已展现出在各个领域进行上下文推理的有效性 [14,30,43,44]。近期研究也利用大语言模型创建房间布局 [4, 10, 25, 57]。然而，大语言模型对数值 [15, 59] 和空间关系 [54] 的理解存在局限，可能导致位置问题，例如物体重叠和物体越界问题（OOB） [10, 25]。

类似地，Holodeck [57] 和 I-Design [4]都在布局提示中纳入空间约束，并在生成布局后采用优化技术来解决越界问题（OOB）。I-Design 使用八种空间关系——左、右、前、后、上方、上、下方和角落——来描述物体放置；Holodeck 则使用涵盖五个类别的 10 种空间关系：全局、距离、位置、对齐和旋转。这两种系统都实现了算法，在后处理阶段调整物体位置，以纠正基于 GPT 的布局方法所产生的空间误差，证明了其生成合理布局和解决空间问题的有效性。然而，这些方法并非专为虚拟布局生成而设计，也未考虑虚拟现实设置中的自然行走适应性，或虚拟环境与物理环境之间的空间差异。

2.1.3 SceneModification forVRNavigation

Tailored Reality [9] 提出了一种创新算法，可调整虚拟环境，使其与用户的物理环境相匹配。传统的虚拟现实（VR）场景通常是预先设计好的，因此用户在实际环境中移动时往往会遇到不一致的情况。这种不匹配会破坏沉浸式体验，并妨碍自然移动。为应对这一挑战，作者提出了一种感知感知重定向技术，通过缩放并稍微重新调整虚拟场景中物体的位置，让用户能够不受干扰或视觉失真的影响，自由探索。该算法会综合考虑多种因素，包括动态可见性和

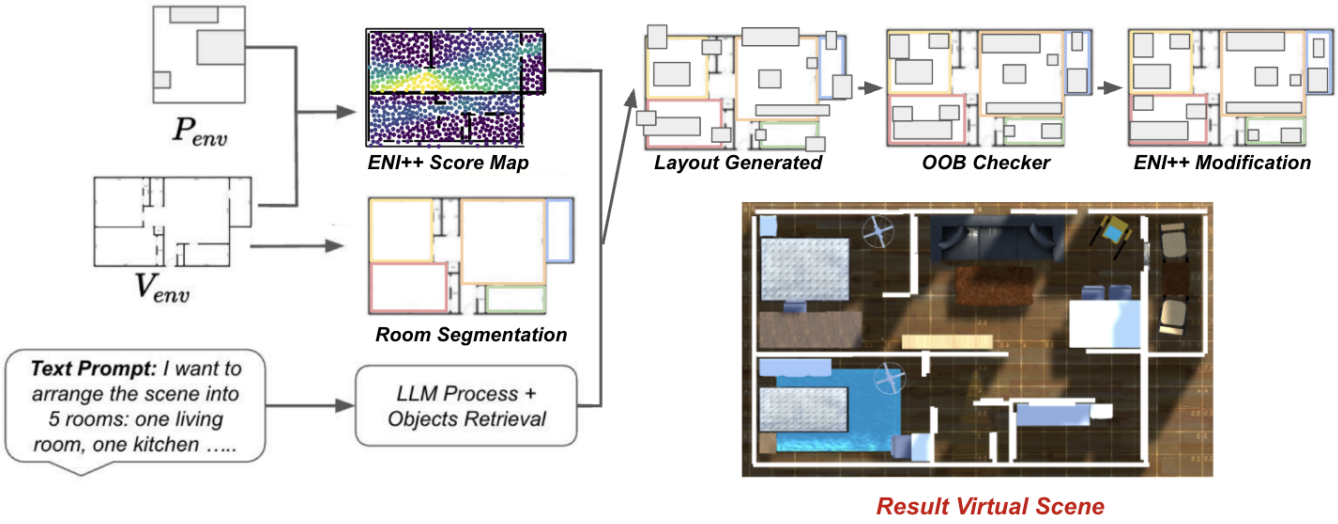


图 2：**HCVR 架构**：一个带有障碍物的物理环境 P_{env} 和一个具有平面图的虚拟空间V_{env} 能够使用 ENI++ 指标处理分数图。我们使用 BUDAS [12] 识别虚拟平面图中的房间。随后将生成每个房间内的物体、物体之间的空间关系以及房间本身。在初始布局生成过程中，每个房间的兼容性信息会与大语言模型提示一同处理。采样点分布在场景图中，颜色从黄色渐变至紫色，以表示 ENI++ 分数——黄色表示高分，紫色表示低分。因此，越界检查器（OOB）会确保初始布局中的所有物体都位于各自房间内，且互不重叠。最后，我们检索分数图，在物体的空间约束范围内重新调整其位置，使它们尽可能覆盖不兼容区域。最终，HCVR 留下了可供用户探索的兼容且适合重定向行走的区域。

从第一人称视角观察到的物体关系，从而确保虚拟空间保持连贯且便于探索。

该方法仍需要预先设计的虚拟场景来进行修改，而对现有不兼容虚拟场景的修改仍然限制了重定向行走控制器的能力。为克服这些局限性，我们的 HVCR 架构将 ENI++ 指标与基于大语言模型的布局建议相结合。该集成方法最小化了为用户导航指定的未占用虚拟区域中的平均 ENI++ 得分，从而促进沿无碰撞路径进行更有效的重定向。人工评估研究还表明，用户对生成的基于大语言模型的布局具有较高的满意度，凸显了我们方法的有效性。

2.2 虚拟场景中的自然行走

自然行走为探索虚拟空间提供了一种直观方式，促使研究人员开发相关技术，将可用的虚拟空间扩展到超出用户有限的物理环境范围。其中一个研究方向聚焦于基于视觉的重定向方法。Sun 等人 [42]通过扫视重定向，推动了 VR 环境中连续自然运动的发展。他们的工作聚焦于动态扫视重定向，这是一种新颖的方法，利用用户快速的眼动（扫视）以难以察觉的方式改变其行走路径，使用户能够在有限的物理空间中获得无限行走体验。在此基础上，Joshi 等人 [20]优化了扫视重定向策略，降低了延迟，使重定向过程更加顺畅。此外，研究人员还探索了使用视觉干扰物来增强 VR 中的重定向行走（RDW）体验 [5, 7, 29]。这些干扰物还有助于提升虚拟行走体验的自然性和沉浸感，从而支持更有效的 RDW 实现。

感知阈值增益是指用户在旋转、速度和曲率方面的物理行走与虚拟行走之间允许存在的不匹配程度 [2, 13, 23, 26, 28, 39, 40]。这些阈值促成了各种重定向行走（RDW）算法的开发 [24, 32, 36, 46, 50],，这些算法会逐帧更新头戴式显示器（HMD）在虚拟空间中的位置和旋转。

重定向行走中的引导方法，例如向中心引导、环绕和目标引导，会将用户重定向到中心物理空间周围、沿着物理空间中的圆形路径，或朝向某个目标物理位置 [32]。基于势场的重定向 [46] 通过在物理空间中构建人工势场，引导用户远离物理边界和障碍物。强化学习也被用于优化重定向路径，通过碰撞检测训练模拟来改进重定向策略 [24, 36]。另一种重定向行走算法 [50] 着重使用户与障碍物之间的物理距离，与其面朝方向上物体的虚拟距离相匹配，从而进一步增强虚拟导航体验的真实感。

所有重定向行走技术都旨在减少预定义的虚拟与物理环境配置中的碰撞。现有对这些算法的分析往往忽略了物理环境的兼容性对虚拟空间中碰撞的影响。在本研究中，我们比较评估了低兼容性虚拟环境设计与采用 ENI++框架生成的高兼容性设计，两者使用相同的重定向行走算法。此外，我们的结果表明，在某些实验中，虚拟环境设计的兼容性对碰撞结果的影响可能比重定向行走算法本身的选择更为显著。

2.3 虚拟环境兼容性

环境复杂度指标用于衡量用户在不同主题的虚拟环境中执行任务的能力。ENI [52] 提供了一种量化用户在虚拟空间中自由行走能力的方法。物理空间与虚拟空间中两个可见性多边形的形状匹配，可以描述用户在当前位置和旋转角度下进行重定向时自由行走的能力 [51]。ENI会计算每对物理环境和虚拟环境中所有采样点的这些形状匹配结果。因此，ENI得分图可以表示虚拟空间各个区域内自由行走的可能性。ENI++可以视为ENI的更新版本。ENI ++ 不再考虑整体分数图，而只计算可达位移范围内的可见性多边形。ENI++ 还将旋转计算限制为边界敏感。

环境复杂度指标用于评估用户在各种情境下于虚拟环境中执行任务的能力。

ENI指标 [52]提供了一种量化用户在虚拟空间中自由导航能力的方法。通过比较物理空间与虚拟空间中可见性多边形的形状匹配情况，ENI可根据用户当前的位置和旋转，评估在重定向下自由行走的可行性 [51]。针对成对的物理环境和虚拟环境中的采样点计算这些形状匹配情况，从而生成ENI得分图，展示虚拟空间不同区域内自由行走的可能性。

ENI++ 指标与 ENI 指标类似，两者都会为物理环境和虚拟环境生成分数图，以表示各虚拟区域与对应物理空间的兼容性。ENI++ 并不评估整个环境，而是聚焦于有限距离内的可达空旷空间，并将旋转计算限制为边界敏感，从而更局部、更精细地衡量自由行走潜力。

3 方法

3.1 概述

给定物理环境配置 P_{env}、虚拟平面图V_{env}以及文本场景描述，我们的方法将生成一个针对步行体验优化的虚拟布局（图 2）。本文将详细介绍：（1）第 3.2 节中的 ENI++ 指标；（2）第 3.3 节中的基于大语言模型的对象与关系生成；（3）第 3.4 节中的越界检查器（OOB）；（4）第 3.5 节中使用 ENI++ 分数图进行的最终对象调整。

3.2 ENI++：ENI 指标的扩展

我们的环境导航指标 ENI++ 是在 ENI[52] 指标的基础上扩展而来，后者用于量化虚拟区域的物理兼容性。与 ENI 类似，对于给定的虚拟环境和物理环境对 (V_{env},P_{env})，我们分别从两个环境中采样点，使得 $V_p \in V_{env}$ 且 $P_p \in P_{env}$。对于每个采样点对 (v_p, p_p)，其中 $v_p \in V_p$ 且 $p_p \in P_p$，我们将 v_{poly} 和 p_{poly} 定义为由从 v_p 和 p_p出发的平移距离所限定的多边形区域。每对多边形 (v_{poly},p_{poly})通过可达距离和旋转进行比较，以使重叠区域最大化。每个采样点 v_p 对应的 ENI++得分，计算为 P_p中观测到的所有最大相似度区域值的最小值。

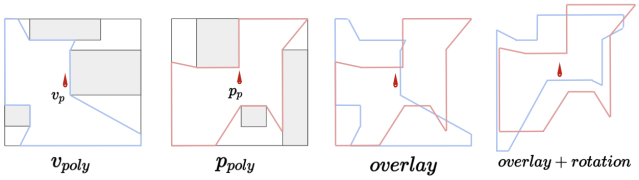


图3：蓝色边界区域表示从点v_p 计算出的可见性多边形，红色边界区域表示从p_p计算出的可见性多边形。由于P_{env}中的前方障碍物，两个可见性多边形的重叠区域受到限制，用户无法朝当前方向行走。不过，如果对用户进行重定向，使其在行走时旋转一定角度，就能解决这个问题。重叠 + 旋转 显著扩大了重叠区域，并为用户提供了更大的可访问区域。

3.2.1 点的选择

给定的虚拟空间边界及其内部的障碍物，限定了V_{env} 和P_{env} 环境的可用区域。在我们的测试环境中，初始输入V_{env} 为平面图，因此可用区域受给定平面图的轮廓及其内部检测到的墙体障碍物限制。P_{env} 中的边界和障碍物应与每位虚拟现实用户的物理环境相匹配。我们没有采用随机采样，而是使用德劳内三角剖分 [6]，确保在两种环境中非障碍区域内的点均匀采样。在德劳内三角剖分过程中，我们根据每种环境的大小限制每个三角形的尺寸，使采样结果适用于不同的（V_{env}，P_{env}）组合。因此，德劳内三角剖分生成的点随后被选作V_p 和P_p，分别对应V_{env} 和P_{env}。

3.2.2 可见性多边形和相似度比较

在 ENI++ 中，我们仅考虑可达区域比较，也就是说，仅比较在可用虚拟平移和旋转下的虚拟可见性多边形与相应人体阈值下可用的物理可视多边形。当给定较大或复杂的V_{env} 时，局部可见性多边形的比较可以得到更具描述性的结果。

对于环境中的任意给定点 v_p 或 p_p，可见性多边形 v_{poly} 或 p_{poly} 是平面上的一个区域，从 v_p 或 p_p 始终可见，如图 3 所示。

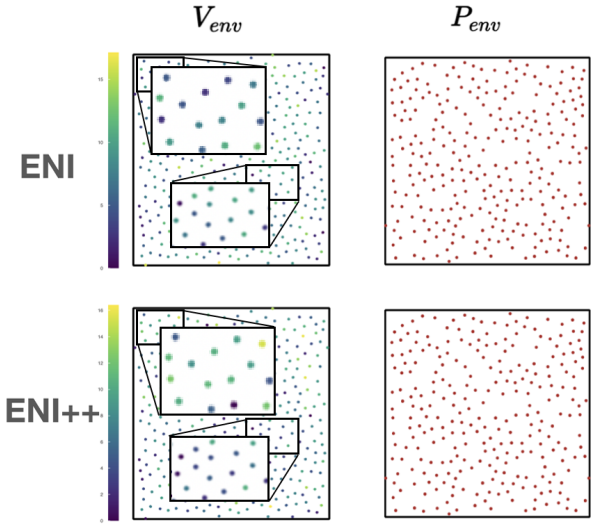


图4：该图展示了在虚拟与物理场景对相同的情况下，ENI 与 ENI++ 在所得绘图中的差异。虚拟环境V_{env} 被定义为一个 20×20 正方形，而P_{env}是一个 20x25 的矩形区域。放大的局部图显示了 ENI 与 ENI++ 之间的差异，这是由于实现了边界敏感和旋转敏感可见性多边形比较。

我们在每个 v_p 处对所有可用的旋转进行插值，并使用旋转增益，将物理重定向的强度控制在感知阈值以内。对于每个虚拟旋转方向 v_{rot}，都有对应的可重定向物理旋转 p_{rot}，其范围介于 0.67 处的最小旋转增益 RG_{min} 和 1.24 [53] 处的最大旋转 RG_{max} 之间。出于计算效率的考虑，我们还在 4 中的实验里采用 4 米 * 4 米作为默认局部平移距离。这样一来，在每个 v_p 处构建的可见性多边形，便可与旋转增益为 (RG_{min},RG_{max})、缩放虚拟旋转方向后的每个 p_p 进行比较。对于每个 v_p，我们从 360 度中插值得到 36 个方向作为 v_{rot}s。一对 (v_p, p_p) 的可见性多边形，仅在 v_{poly} 的全部 36 个方向中，分别与每个方向在旋转增益 (RG_{min},RG_{max}) 范围内的 10 种旋转以及 p_{poly} 的平移进行形状相似度比较。v_{poly} 与 p_{poly} 的形状相似度是两个区域的重叠，因此可按以下方式计算：[51]

$$S(v_{poly}, p_{poly}) = v_{poly} \setminus (v_{poly} \setminus p_{poly}) \tag{1}$$

ENI++ 分数v_p 与一个p_p 可以据此按照算法1计算。应用算法1 处理所有v_p以及所有p_p后，我们便可得到所有++p的总体 *ENI* 分数$_p$:

$$\text{ENI++}(v_p) = \arg\min\{\forall p_p \in P_p,\ \text{ENI++}(v_p, p_p)\}. \tag{2}$$

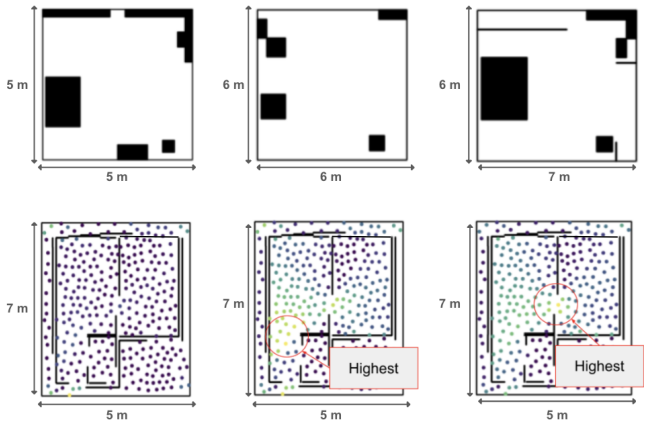


图5：此图展示了在给定不同P_{env}时，将不同的 ENI++ 分数图绘制在同一V_{env} 上的效果。ENI++ 分数以从紫色到黄色（由低到高）的颜色范围绘制。可以明显看出其变化。第一个场景显示出较高的兼容性。第二组的 ENI++ 分数最低处是左侧标记区域，而第三个 ENI++绘图显示最低分数位于中间区域。

ENI++ 分数图vp 在虚拟空间的边界和障碍物处均包含边界与角点。对于vp 附近的虚拟边界线和障碍物边界线，朝向这些边界线的虚拟旋转将被忽略。随后即可得到 ENI++ 分数图，如图5 所示。该图还展示了相同的V_{env} 在不同P_{env} 以及内部安装障碍物时的兼容性。

算法1 ENI++ 分数：vp，pp

要求： 计算v^{poly} 的vp 和p^{poly} 的pp。

对于所有头部方向v_{rot}，从 0 ° 到 360 ° **do**
  初始化形状比较 ← []
  对于所有 旋转 p_{rot} 从$RG_{\min}$ 到 $RG_{\max}$ **执行**
    形状比较.添加($S(v^{\text{poly}}, p_{\text{rot}}(p^{\text{poly}}))$)
  结束循环
  ENI++(vp,v_{rot}) ← 取最大值{形状比较}
结束循环
ENI++(vp，pp) ← 平均值{$\forall v_{\text{rot}}$，ENI++(vp，v_{rot})}
返回

由于 ENI++ 对边界很敏感，因此边缘附近的评分差异更为明显。相比之下，中心区域的变化预计很小（图 4）。这是因为 ENI 通过评估每个虚拟点的所有可能旋转来计算相似度，所得结果通常与 ENI++的结果一致，因为两者最终都考虑了不同旋转下的形状相似度。此外，ENI++还对旋转增益施加了额外约束，使其能够生成更精确且信息更丰富的绘图。例如，图 4 展示了 ENI++ 补丁：虚拟空间的中部兼容性更高（颜色较深），而虚拟空间的角落兼容性较低（颜色较浅）。ENI++ 生成了更准确的地图，因为在某些 $vrot$下，角落更难实现重定向，而空旷空间的中部更适合重定向。

3.3 LLMLayout 生成

输入的虚拟场景由墙壁和边界构成平面图。BUDAS 房间检测算法 [12] 可输出房间分割结果，其中包含房间数量、房间面积和位置。使用 ENI++ 指标，我们可以计算初始每个房间 R_{score} 的容量评分 R 如下：

$$R_{score} = Avg(\sum_{v_p \in R} \text{ENI++}(v_p)). \tag{3}$$

每个房间的容量评分 R_{score} 基于房间内采样的所有 vps 的平均 ENI++ 评分。随后，我们将该容量评分归一化到 1 到 10 的范围内，其中数值越大表示需要占用的面积越大。房间的大小 m计算如下

面积(R) 面积(虚拟)。由于大多数大语言模型对上下文语言比对数字更敏感，因此房间的规模可以分为五个等级：超大（$m > 0.25$）、大（$0.166 < m < 0.25$）、中（$0.125 < m < 0.166$）、小（$0.1 < m < 0.125$）和极小（$m < 0.1$）。房间尺寸描述及其容量评分作为可变输入提供给大语言模型提示，以辅助后续的物体选择和缩放流程。

大语言模型文本提示可帮助为虚拟场景选择 3D资产、合理安排所有资产的位置，并以视觉上合理的方式缩放资产。给定一条大语言模型文本提示 T，其中包含容量评分、尺寸描述以及所有房间的功能（图6），大语言模型能够从资产中选择尺寸和数量合适的物体，从而覆盖所有高 ENI++ 评分区域。在大语言模型物体选择流程中，房间的功能有助于生成物体名称列表。容量评分决定了房间至少应包含多少件物体。在实验中，我们使用

$$R_{score}/(\frac{Area(Virtual)}{Area(Physical)}) \tag{4}$$

作为每个房间的最少物体数量。随后，我们要求大语言模型在3D模型资产中找出最适合的物体名称，作为每个房间选定的物体。我们在大语言模型中使用上下文训练，以确保其选择与房间尺寸相匹配的物体（图 7）。尺寸描述还会在资产选择过程中筛除一些大型物体。例如，如果房间很小，所选物体的长×宽总面积不能超过场景面积的十分之一。

系统指令：
你是一名熟练的室内设计师。你的任务是解读每个房间的给定指令，包括房间的功能、尺寸描述以及放入房间的最少物体数量。你的工作是根据指令，列出每个房间中应出现的物体。你的回答必须严格遵循所提供的格式。不要写任何其他内容。

示例：
输入：一套公寓包含：一个至少放置5件物体的大型客厅。
输出：客厅（沙发、餐桌、椅子、电视、书架、装饰画、咖啡桌）。

图6：用于物体建议的示例大语言模型提示，其中将容量分数用作房间物品数量的下限。每个房间的功能和尺寸描述都会影响大语言模型输出的物品及其数量（完整提示详情见补充材料）。

然后，对于选定的物品，我们还会查询大语言模型，以返回它们之间合理的空间关系（图8）。定义的空间关系包括场景之间的关系（靠近/远离墙X、房间中央、墙X 与墙Y的拐角处）以及物体之间的关系（物体X的左侧、上方、右侧、下方、前方、后方、左上方、右上方、左下方、右下方、上方中央和下方中央）。给定提示的输出包括一份缩放后的列表

上下文训练
示例 1：
输入：客厅（沙发、餐桌、椅子、电视、书架、装饰画、咖啡桌）。房间（3*3），沙发（1*4），餐桌（2*2），椅子（0.4*0.4），.....输出：False，因为沙发的一个轴向尺寸明显大于房间。修改建议：沙发。删除建议：无。

示例 2：
输入：前门位于（落地灯、鞋架、椅子、衣帽架）处。房间（1*0.6），落地灯（0.3*0.3），鞋架（0.8*0.4），椅子（0,5*0.5），衣帽架（0.3*1）。输出：错误，因为总面积小于所有物体的面积之和。修改建议：衣帽架、鞋架。删除建议：衣帽架。

图7：物体选择的上下文训练演示。此图展示了两个案例：一个物体有一条轴长于房间，以及所有物体的总占地面积大于房间面积（补充材料中包含更多上下文训练的详细信息）。

每个房间中的物体及其彼此之间以及与房间之间的空间关系。本文展示的所有实验中，我们均使用默认参数的 GPT-4o。

系统指令：
你是一位经验丰富的室内设计师。根据每个房间的物体列表，你的任务是返回这些物体彼此之间以及它们与环境之间的空间关系建议。回答必须严格遵循提供的格式。不要写任何额外内容。
示例：
输入：客厅（沙发、餐桌、椅子、电视、书架、装饰画、咖啡桌）。输出：客厅（沙发靠近上方的墙，咖啡桌位于沙发前方，椅子A位于餐桌左侧，椅子B位于餐桌右侧，装饰画靠近上方的墙，书架靠近右侧的墙，电视靠近下方的墙）。

图8：根据每个房间给定的物体列表生成布局建议的大语言模型提示示例（更多提示详情见补充材料）。

3.4 越界修改

在生成空间关系并对生成结果进行初步绘图后，一些物体可能会出现在场景边界之外。对于每个房间，该房间中对应的物体组首先检查是否有资产位于边界之外。如果某个物体与边界重叠或完全位于边界之外，我们会沿重叠的边缘或靠近该物体的边缘移动该物体，使其能够出现在房间内部，如图 9中的步骤 A 所示。随后，我们检查每个房间内发生重叠的物体，并一次修改一个轴，直到它们能够在没有任何重叠的情况下适应房间，如图 9 中的步骤 B 所示。如果某个物体不可避免地发生重叠，如图 9中的步骤 C 所示，我们会将它们放置在重叠区域最小的位置，并缩小发生重叠的轴，直到它们不再重叠。然而，在我们的实验中，缩放流程被限制在其原始大小的 (0.8, 1.2) 范围内。如果仍然无法避免重叠，模型会忽略该物体，并将物体名称返回给大语言模型，以进行另一轮物体检索过程。

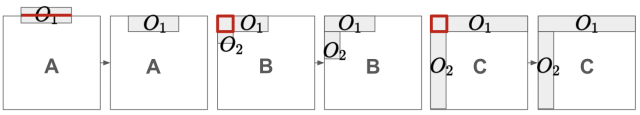


图 9：步骤 A：表示物体 O_1 位于线上，并通过修改 y 轴在边界内移动。步骤 B：展示检测到的重叠区域（红色正方形），并将O_2作为两个发生重叠且向下移动的物体之间较小的一个。步骤 C：表示 O_1 与 O_2, O_2 之间不可避免的重叠情况，其中两个物体之间较大的物体仅沿 y 轴进行缩放以缩小。

3.5 ENI++定向房间修改

当所有物体都出现在预期的房间中，且不存在任何重叠或越界错误后，ENI++ 分数图便可进行适当的最终梯度调整，以确保场景适合步行体验。场景中的每个物体都会覆盖一定数量的采样点$v_{covered} \in V_P$，并将其余点（$V_p \setminus v_{covered}$）留作$v_{uncovered} \in V_P$。这样可以最大限度地降低$V_{env}$中未被物体占据的所有点的 ENI++ 分数的平均值，即：

$$Min\{Avg\{\ \text{ENI++}\ (V_p \setminus v_{covered})\}\}. \tag{5}$$

与此同时，我们希望在梯度下降过程中合理分配资产，因此根据大语言模型输出的各个房间的空间关系来约束修改；在每次修改迭代中，我们都会检查资产之间的空间关系。经过足够多次迭代后，放置在虚拟环境中的物体便能与物理环境保持高度兼容。

4 实验

参与者 我们招募了 16 名参与者（年龄 22–30 岁，8 名女性、8 名男性）参与研究。所有参与者的视力正常或经矫正后正常，无听力障碍病史，也无现有或既往残疾。其中，7 人此前没有虚拟现实体验，5 人有一些虚拟现实体验，4 人熟悉虚拟现实技术并经常使用。所有人均不了解研究假设。本研究已获所在机构的机构审查委员会（IRB）批准，所有参与者均在参与前提供了知情同意。

实验设置 在研究期间，受试者佩戴 Meta Quest 2 头戴式显示器进行视觉显示。受试者在虚拟现实中显示的虚拟环境中进行探索，并期望在一片较大的空旷区域内自然行走。在测试期间，系统开展了用户偏好调查，并记录了受试者与所设计物理场景之间发生的物理碰撞。同时，在研究开始前，要求参与者在默认场景中行走 1 分钟，以尽量减少其初始感知对测试区域及后续行走行为的影响。

在研究期间，为保持无缝且自然的行走体验，用户在与系统中设计的物理环境发生碰撞时不会收到通知。物理碰撞次数分别按用户（=16）、实验（=5）和消融研究（=3）单独记录。

实验条件 我们在五组物理环境和虚拟环境组合中开展了实验。这五个场景涵盖了不同大小的房间，包括：(1) 一个由（$P_{env} > V_{env}$）组成的场景；(2) 一个由（$P_{env} = V_{env}$）组成的场景；(3) 一个由（$P_{env} < V_{env}$）组成的场景，其中V_{env} 的长度是P_{env}的 1.5 倍；(4) 一个由（$P_{env} < V_{env}$）组成的场景，其中VE 的宽度和长度均是P_{env}的 1.5 倍；以及 (5) 一个由P_{env} 组成的场景，其中V_{env} 的宽度和长度均是P_{env} 的 2 倍（参见附录中的实验组合绘图）。

对于每一对（P_{env}, V_{env}），我们在三个生成案例之间进行了消融研究，以验证我们的方法的有效性

布局设计与重定向行走相结合，可以将用户引导至一条“无碰撞”路径。我们收集了以下情况下的用户满意度和碰撞数据：(1) LLM&ARC：LLM提示生成的场景 + 在场景探索期间使用 OOB 检查器和 ARC [50]；(2) ENI++：ENI++指标引导的场景（包括将 ENI++ 分数图输入大语言模型，以及由 ENI++ 引导的修改流程），在用户测试期间不应用重定向行走算法；(3) ENI++&ARC：ENI++ 指标引导的场景，在场景探索期间使用 ARC [50]。因此，我们的用户研究中，每位受试者都进行了 (# 场景对 = 5)×(# 条件 = 3) = 15 次试验。

这五组物理场景与虚拟场景配对旨在测试各种条件。实验1描述的情况是（$PE > VE$）。实验2展示了（$PE = VE$）时发生的物理碰撞。实验3、4、5分别展示了以下情况：在一个轴上，$PE < VE$中PE的尺寸是VE的1.5倍；在两个轴上，PE的尺寸均为VE的1.5倍；以及在两个轴上，PE的尺寸均为VE的1.85倍。

5 结果分析

5.1 数据收集

对于每组设计的（P_{env},V_{env}）场景配对，我们生成一组三个场景，分别为 LLM&ARC、ENI++ 和 ENI++&ARC，用于用户测试。对于每个场景，我们从两个方面进行检测：物理碰撞，以及用户对房间布局的满意度。由于 ENI++ 和 ENI++&ARC 共享相同的场景，因此在每组测试中，无论先测试哪个，都会收集 ENI++ 的用户满意度。用户还会从不同位置开始探索相同的 ENI++ 和 ENI++&ARC 场景，以确保受试者充分探索两个场景，从而收集碰撞数据。

5.2 结果分析：对生成场景的人工评估

参与者从四个方面对场景中的每个房间进行评估：图10右侧所示的整体布局评分，以及图10左侧所示的物体选择、物体缩放和物体定位（放置与旋转）。他们对每个方面按 5 分制打分。我们记录了他们在 LLM 和 ENI++ 或 ENI++&ARC 场景下对所有方面的回答。每轮测试结束后，受试者还会从中选出自己更偏好的布局。

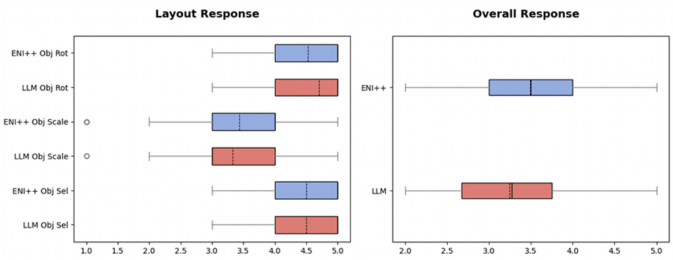


图 10：左图展示了用户对五种不同场景中 LLM 生成场景和在 ENI++ 引导下创建的场景的物体选择、缩放和定位的反馈。右图展示了用户对 ENI++ 或 LLM 生成的房间设计的总体偏好。

在四个评估方面中，图10 显示，参与者总体上更偏好 ENI++ 场景。图 10 表明，在物体选择能力方面，参与者对大语言模型和 ENI++ 并无明显偏好。这是因为二者使用相同的大语言模型文本提示，并生成相似的物体资产。向大语言模型提示的 ENI++ 信息不会影响大语言模型为房间选择合适三维模型的能力。在物体缩放方面，ENI++ 获得了略高的偏好度。另一方面，大语言模型在物体定位方面表现出更强的能力。ENI++ 引导的修改牺牲了部分定位能力，以在布局设计与虚拟环境同

表1：进入物理边界之外的虚拟房间

方法	LLM&ARC	ENI++	HCVR（ENI++&ARC）
实验1	0.00%	0.00%	0.00%
实验2	0.00%	10.40%	0.00%
实验3	3.13%	17.20%	1.56%
实验4	6.25%	34.38%	1.56%
实验5	10.00%	35.00%	6.26%

物理世界之间的兼容性达到平衡。总之，结果表明，对于 ENI++ 生成的场景，在所有四个维度上，用户满意度相当。

除了评估的四个方面之外，用户还会在每组的两个不同场景中选择其偏好的布局（共三个场景，但 ENI++ 和 ENI++&ARC 相同）。在五组场景中，大多数标注者更偏好 ENI++ （50%）而非 LLM&ARC （ 37.5% ），但有时也未表现出明确倾向（ 12.5% ）。

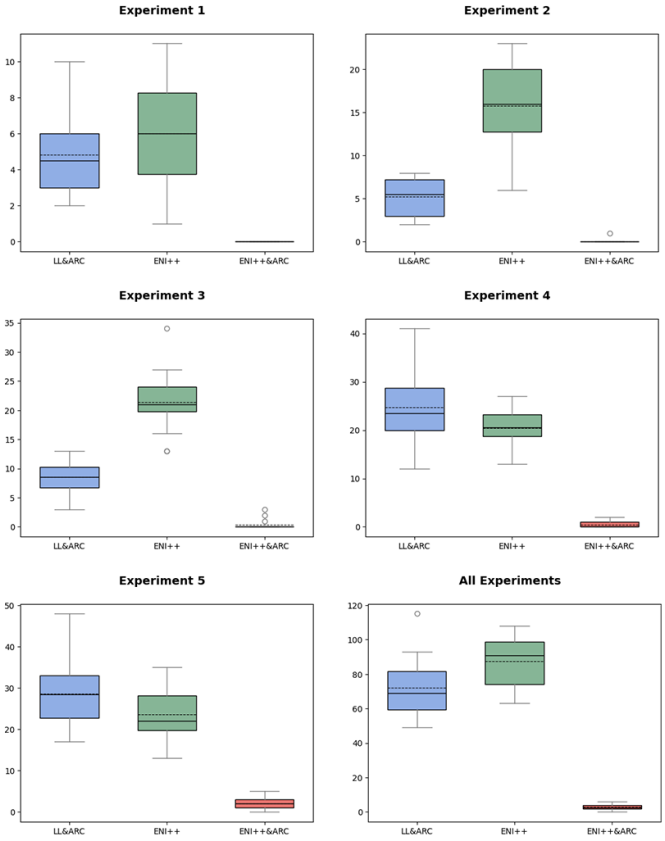


图11：该图展示了在全部五次实验中，使用重定向行走算法行走时 ENI++ 生成的场景、ENI++ 场景，以及使用重定向行走算法行走时大语言模型生成的场景中检测到的碰撞次数。该图可视化了不同虚拟与物理场景对设置下的碰撞结果。绘图表明，其碰撞次数总体上显著低于 LLM&ARC 和 ENI++。这凸显了 ENI++ 引导的场景在该情境下的优势，以及其与基于对齐的RDW算法的兼容性。

5.3 结果分析：物理碰撞检测

5.3.1 物理碰撞实验

我们绘制了五个实验场景中用户与物理边界和障碍物发生碰撞的次数（图11）。结果表明，在所有情况下，*HCVR（ENI++&ARC）*都显著减少了碰-撞次数。在实验5中，虚拟空间是物理

空间的四倍大，*HCVR*（*ENI++&ARC*）仍能保持有效的碰撞管理。这些结果验证了，我们的 ENI++ 引导的场景生成与重定向行走算法相结合，能够在典型场景中实现近乎无碰撞导航；而在空间不匹配的情况下，LLM&ARC 无法维持较低的碰撞率。

图 11 中的前三个实验还展示了 ENI++ 与 LLM&ARC 之间的比较。在实验1、2和3中，LLM&ARC 展现了 ARC 在避免碰撞方面的优势。以 ENI++ 作为参考，可以清楚地看到 ENI++&ARC 能够充分释放 ARC 在 HCVR 中的能力。同时，ENI++&ARC 在以更大、更复杂的虚拟空间作为输入时取得了更好的结果。因此，我们得出结论：当输入为更复杂的虚拟场景时，场景兼容性的影响会显著增加。

实验1图 11 展示了 P_{env} 在长度和宽度上均大于 V_{env} 时的碰撞结果。我们的方法 ENI++&ARC 在实验1中能够让所有用户实现0次碰撞。

稳健方差分析（AnalysisofVariance）统计检验显示，LLM&ARC、ENI++、ENI++&ARC 的碰撞次数存在显著差异。$f = 30.74148$。$p < .00001$。在 $p < .05$ 水平上，结果具有显著性。如图 11 所示，ENI++&ARC 方法的物理碰撞次数中位数和均值均为0，而 LLM&ARC 的中位数为4.5，均值为4.8125。未使用 ARC算法时，ENI++ 的碰撞次数中位数和均值均为6。

实验2图11 展示了当 P_{env} 与 V_{env}具有相同的长和宽时的碰撞结果。我们的方法 ENI++&ARC 在实验2中成功实现了用户间零碰撞，但有一名用户在测试期间发生了一次物理碰撞。

我们的结果显示 $f = 30.74148$，$p < .00001$。结果具有统计显著性，$p < .05$。图 11 表明，ENI++&ARC 方法的中位数为 0，平均数为 0.0625。LLM&ARC 的物理碰撞中位数为 5.5，平均数为 5.25。不使用 ARC算法时，ENI++ 的中位数为 15.75，平均数为 16。

实验3图11 展示了当 V_{env} 的尺寸因此是 P_{env}的两倍时的碰撞结果。我们的方法的碰撞次数显著少于 LLM&ARC，这证明了 ENI++指导的设计场景的有效性。同时，与不使用重定向行走的 ENI++ 相比，该方法实现了更少的物理碰撞，这证明了 ENI++ 指导的场景与重定向行走控制器（我们在实验中使用了 ARC）结合的优势。

LLM&ARC 与 ENI 的碰撞次数存在显著差异++，ENI++&ARC 导致了 $f = 147.26238$。该$p < .00001$。结果在以下水平上显著：$p < .05$。图 11 绘制了 ENI++&ARC 方法的优势：其碰撞次数中位数为 0，平均值为 0.375；而 LLM&ARC 的物理碰撞次数中位数和平均值均为 8.5。在测试过程中不使用重定向行走时，ENI++ 的中位数为 21.4375，平均值为 21。

实验4 图 11 展示了碰撞结果。当V_{env} 的长和宽均比P_{env}长一半时。因此，V_{env} 是P_{env}的 2.25 倍。我们的方法碰撞次数显著少于 LLM&ARC，这证明了 ENI++ 引导的设计场景的有效性。同时，其物理碰撞次数少于不使用重定向行走的 ENI++，证明了 ENI++ 引导的场景结合重定向行走控制器的优势（实验中使用了 ARC）。

图11 展示了 ENI++&ARC 方法所取得的显著成果，收集的数据为$f = 95.12786$，$p < .00001$。该结果在$p < .05$水平上具有显著性。ENI++&ARC 方法的中位数为 0，均值为 0.375；而 LLM&ARC 的物理碰撞中位数为 23.5，均值为 24.75。尽管 ENI++未应用重定向控制器，用户的碰撞次数中位数为 20.5，均值为 20.4375。

表2：LLM引导场景与ENI++引导场景中的ENI++ 得分

实验	平面图	大语言模型引导	ENI++引导
实验1	0	0	0
实验2	20.81	13.54	1.45
实验3	528.74	376.49	168.07
实验4	1187.63	852.72	352.54
实验5	2624.06	1763.35	765.89

实验5 图 11 展示了当 V_{env} 的长和宽均为 P_{env}的 1 倍时的碰撞结果。这使得V_{env} 4 为 P_{env}的倍数。绘图显示，其碰撞次数显著低于 LLM&ARC 和 ENI++。这凸显了 ENI++ 导向场景的优势，以及该场景下其与重定向行走算法的兼容性。

尽管虚拟场景远大于物理空间，ENI++&ARC仍保持$f = 98.75979$，p $< .00001$。该结果在 $p < .05$ 水平上具有显著性。如图 11 所示，ENI++&ARC方法的碰撞次数中位数为2，均值为2.125，而LLM&ARC的中位数为28.5，均值为28.6875次物理碰撞。在没有ARC算法的情况下，ENI++ 的碰撞次数中位数为22，均值为23.6875。

5.3.2 生成场景兼容性分析

在上一节介绍的五项实验中，我们将 ENI++作为生成的虚拟场景的定量指标，用于评估其与各个物理空间的兼容性。表2 展示了由大语言模型引导和由 ENI++ 引导的场景之间的平均 ENI++得分对比（数值越小表示兼容性越高）。结果表明，我们的架构能够生成兼容的虚拟场景，因此在自由行走时，更有可能引导用户沿着无碰撞的路径行走。

5.3.3 无法到达的房间分析

在一些实验中，虚拟场景比物理场景更大，因此受试者无法进入其中的所有房间。这意味着用户必须走出物理场景预设的边界，才能探索部分虚拟房间。为了在布局评分过程中让用户获得更沉浸的行走体验，当用户走出物理边界时，我们不会重置虚拟场景。走出物理边界被计作普通碰撞，而无法到达的房间可作为上述碰撞数据的补充。也就是说，当用户在探索某个房间时走出物理边界，该房间就会被标记为无法到达。ARC 技术会持续运行，直到用户重新进入预设的物理空间。我们统计了每次实验中无法到达的房间数量，并绘制了用户遇到无法到达房间情况的平均概率，如表1所示。

ENI++&ARC 在所有实验中都表现出显著成果。如果在实验过程中不应用 ARC，用户在进行场景探索时很容易走出物理区域。ENI++ 引导的场景生成还能够防止用户走出物理房间边界，同时 LLM&ARC 和 ENI++&ARC 共享同一个重定向行走控制器。

6 结论、局限性与未来工作

我们推出了 HCVR，这是一种新型界面，利用大语言模型生成适配各种物理环境的虚拟现实场景，使虚拟现实用户能够通过重定向行走自由探索。为量化场景与环境对齐程度，我们提出了 ENI++评分，用于评估每一对虚拟区域与物理区域的兼容性。通过比较大语言模型驱动的布局与 ENI++ 引导的布局，HCVR 在连贯设计与虚拟空间和物理空间的无缝融合之间实现了最佳平衡。本文还定量展示了 HCVR流程在

通过虚拟场景设计，释放 ARC 的重定向行走能力，并引导用户进入无碰撞路径。

6.1 碰撞次数与重置评估

大多数重定向行走算法都会实现一个重置组件 [27, 50, 60]来评估其方法：每当记录到与物理边界发生物理碰撞时，就会通过符号、干扰物等方式停止用户。本研究采用了一种更直观的方法来收集碰撞数据，以避免影响用户满意度反馈。用户对虚拟环境中自然行走的感知能够增强具身感（SoE），从而积极影响其与场景的互动和反应 [3, 38]。

实验过程中，一旦用户开始与物理障碍物发生碰撞，我们的方法就会停用 ARC算法，直到碰撞结束，并将碰撞数据计数加一。用户重新进入物理区域时，碰撞数据也会再次累加。由于我们的测试场景大于虚拟空间，因此用户在没有 ARC 的情况下自由行走的距离也可以接受。

6.2 物体缩放

在结果展示阶段，图 10 中的用户满意度结果显示，物体缩放的整体评分较低。在我们的流程中，大语言模型建议的输出和 OOB 流程都可以缩放物体。实验期间，用户也指出物体高度比例失调的问题。这可能是 OOB检查器和 ENI++ 引导的修改过程中重新缩放所致。物体高度的重新缩放值根据其宽度和长度重新缩放值的平均值计算得出。不真实的缩放结果也可能是由我们所用大语言模型（gpt-4o [1]）空间推理能力有限造成的。未来可以使用专为 3D任务开发的一些大语言模型 [41, 45]来减少幻觉。另一种可取的方法是使用或训练一个对 3D 敏感且具备物体理解能力的模型，根据物体新的宽度和长度来缩放其高度。这可以成为一个新的研究课题，从而提升 HCVR，为用户带来更好的体验。

6.3 动态障碍物

尽管 HCVR 的结果颇具前景，但我们也必须承认本研究存在局限性。我们的实验在用户测试期间未纳入任何动态障碍物。这是因为，存在移动障碍物的物理环境无法生成静态 ENI++ 地图，而静态地图对于房间生成及后续流程至关重要。同时，在物理环境中难以计算移动障碍物的兼容性。我们认为，我们的模型可以与扫视行为重定向 [42] 方法结合，以便在测试期间处理动态障碍物。我们希望扩展该方法，使其能够处理静态和动态物理环境。我们的方法将着重于结合实时扫视重定向，并收集眼动数据以及人对物体移动的反应。

参考文献

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat 等。GPT-4 技术报告。arXiv 预印本 arXiv: *2303 08774*, 2023 9 。.

[2] G. Bruder, V. Interrante, L. Phillips 和 F. Steinicke。沉浸式虚拟环境中的自然导航：重定向行走与驾驶。《IEEE可视化与计算机图形学汇刊》，18 4 538 545, 2012 1, 3 ( ): – 。G. Calogiuri, S. Litleskare, K. A.

[3] Fagerheim, T. L. Rydgren, E. Brambilla 和 M. Thurston。通过沉浸式虚拟环境体验自然：模拟自然漫步中的环境感知、身体参与和情感反应。《心理学前沿》，8 2321, 2018 9 : 。A. Çelen, G. Han, K. Schindler, L. Van Gool, I.

[4] Armeni, A. Obukhov 和 X. Wang。I-Design：个性化大语言模型室内设计师。arXiv 预印本 arXiv: *2404 02838*, 2024 2 。. H. Chen 和 H.

[5] Fuchs。支持在大型虚拟环境中自由行走：利用沉浸式干扰物实现难以察觉的重定向行走。载于《计算机图形学国际会议论文集》，第 1 6, 2017 3 – 页。L. P. Chew。约束德劳内三角剖分。载于

[6] 《第三届计算几何年度研讨会论文集》，第 215–222 页，1987 年 4 月 – 。[7] R. Cools 和 A. L. Simeone。探究干扰物交互性对虚拟现实中重定向行走的影响。载于空间用户交互研讨会，第 1 5, 2019 3 – 页。R.

[8] Dalton, C. Hoelscher, T. Peck 和 V. Pawar。虚拟现实中对建筑复杂性与可导航性的判断。载于《2010年空间认知国际会议论文集》，第 49–64 页，2010 年。2[9] Z.-C. Dong, W. Wu, Z. Xu, Q. Sun, G. Yuan, L. Liu 和 X.-M. Fu。Tailored Reality：面向自适应虚拟现实导航的感知感知场景重构。《ACM图形学汇刊》（TOG），40(5):1–15，2021。2[10] W. Feng, W. Zhu, T.-j. Fu, V. Jampani, A. Akula, X. He, S. Basu, X. E. Wang 和 W. Y. Wang。LayoutGPT：利用大语言模型进行组合式视觉规划与生成。《神经信息处理系统进展》，36, 2024 2 。G. Franz 和 J. M. Wiener。探索

[11] 基于视域的空间行为与体验相关因素。载于第五届国际空间句法研讨会，第 503 517 2005 2 – 页。Techne Press，。Y. S. Gan, S.-Y.

[12] Wang, C.-E. Huang, Y.-C. Hsieh, H.-Y. Wang, W.-H. Lin, S.-N. Chong 和 S.-T. Liong。你需要几间卧室？基于建筑平面图图像的房地产推荐系统。《科学编程》，2021 1 9914557, 2021 2, 3, 5 ( ): 。T. Grechkin, J.

[13] Thomas, M. Azmandian, M. Bolas 和 E. Suma。重新审视重定向行走的检测阈值：结合平移增益与曲率增益。载于《ACM应用感知研讨会论文集》，第 113 120, 2016 2, 3 – 页。P. Guhan, U.

[14] Bhattacharya, S. Sarkhel, V. Azizi, X. Chen, S. Mitra, A. Bera 和 D. Manocha。Tame-RD：用于逆向设计的图像多重调整文本辅助复现。载于计算语言学协会 ACL *2024*, 10710 10727, 2024 2 论文集，第 –

[15] 页。S. Guo, A. Didolkar, N. R. Ke, A. Goyal, F. Huszár 和 B. Schölkopf。超越模式匹配的学习？评估大语言模型中的数学理解能力。arXiv 预印本 arXiv: *2405 15485*, 2024 2 。. K. Gupta, J.

[16] Lazarow, A. Achille, L. S. Davis, V. Mahadevan 和 A. Shrivastava。LayoutTransformer：利用自注意力进行布局生成与补全。载于《IEEE/CVF国际计算机视觉大会论文集》，第 1004 1014, 2021 2 – 页。J. Hildebrandt,

[17] P. Schmitz, A. Calero Valdez, L. Kobbelt 和 M. Ziefle。早日康复！人为因素对虚拟现实中重定向行走暴露后晕动症的影响。载于《虚拟、增强与混合现实：交互、导航、可视化、具身化与仿真：第十届国际会议 VAMR 2018，作为 HCI International 2018 的一部分于 2018 年 7 月 15—20 日在美国内华达州拉斯维加斯举行，论文集，第一部分》 *10*, 82 101 2018 2 ，第 – 页。Springer，。

[18] E. Hodgson 和 E. Bachmann。比较四种广义重定向行走方法：仿真与实时用户数据。《IEEE 《可视化与计算机图形学汇刊》，19(4)：634–643，2013. 1[19] C. Hutton、S. Ziccardi、J. Medina 和 E. S. Rosenberg。面向重定向行走的旋转增益阈值个性化校准。载于 ICAT-EGVE，第 61 64, 2018 2 – 页。Y. Joshi 和 C. Poullis。通过实时扫视预

[20] 测实现扫视重定向。《计算机动画与虚拟世界》， 34 3 4 2167, 2023 3 (-):e。S. Jung、C. W. Borst、S. Hoermann 和 R. W. Lindeman。重定向跳跃：曲率增益的感知检测率。载于第 32 届 ACM用户界面软件与技术研讨会论文集，第 1085 1092, 2019 2 – 页。A. A. Jyothi、T.

[21] Durand、J. He、L. Sigal 和 G. Mori。LayoutVAE：根据标签集生成随机场景布局。载于 IEEE/CVF国际计算机视觉大会论文集，第 9895 9904, 2019 2 – 页。L. Kruse、E. Langbehn 和 F. Steinicke。我

[22] 走路时能看见自己的双脚：可见双脚条件下对平移增益的敏感度。载于 2018 IEEE虚拟现实与三维用户界面会议（VR），第 305–312 页。IEEE，2018。3[24] D.-Y. Lee、Y.-H. Cho 和 I.-K. Lee。使用深度Q学习进行重定向行走的实时最优规划。载于 2019 IEEE虚拟现实与三维用户界面会议（VR），第 63–71 页。IEEE，2019。[25] Y. Lin、H. Wu、R. Wang、H. Lu、1, 5

X. Lin、H. Xiong 和 L. Wang。迈向语言引导的交互式三维生成：将大语言模型用作布局解释器并提供生成式反馈。arXiv 预印本 arXiv:2305 15808, 2023 2。。[26] K. Matsumoto、E. Langbehn、T. Narumi 和 F. Steinicke。虚拟现实及基于无人机的远程临场系统中垂直增益的检测阈值。载于 2020 IEEE虚拟现实与三维用户界面会议（VR），第 101 107 2020 3 – 页。IEEE，。A. Nguyen 和 A.

[23] Kunz。眨眼期间的离散场景旋转及其对重定向行走算法的影响。载于第 24 届 ACM虚拟现实软件与技术研讨会论文集，第 1 10 页，2018 9–。[28] A. Paludan、J. Elbaek、M. Mortensen、M. Zobbe、N. C. Nilsson、R. Nordahl、L. Reng 和 S. Serafin。通过伪装旋转增益实现虚拟现实中的重定向行走：视觉密度的影响。载于 2016 IEEE虚拟现实（VR），第 259 260 2016 3 – 页。IEEE，。T. C. Peck、H.

[24] Fuchs 和 M. C. Whitton。借助干扰物改善重定向：大规模真实行走移动界面及其对虚拟环境导航的影响。载于 2010 IEEE虚拟现实会议（VR），第 35 38 2010 3 – 页。IEEE，。P.

[25] Pham、A. N. Mathur、O. Sharma 和 A. Bera。MvGaussian：借助多视图引导和表面稠密化实现高保真文本到三维内容生成。arXiv 预印本 arXiv: *2409 06620*, 2024 2 。。X.

[26] Qiao、G. P. Hancke 和 R. W. Lau。学习物体上下文以生成新视角场景布局。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第 16990–16999, 2022 2 页。S. Razzaque。重定向行走。北卡罗来纳大学教堂山

[27] 分校， 2005 1, 3 。S. Razzaque、D. Swapp、M. Slater、M. C. Whitton 和 A. Steed。原地重定向行走。载于 EGVE，第 2, 123 130, 2002 1 卷，第 –

[28] 页。R. A. Ruddle 和 S. Lessels。使用行走界面导航虚拟环境的优势。《ACM人机交互汇刊》（TOCHI），16 1 1 18, 2009 1 (-):–。P. Schmitz、J.

[29] Hildebrandt、A. C. Valdez、L. Kobbelt 和 M. Ziefle。你让我晕头转向：重定向行走中旋转增益的有限沉浸阈值。《IEEE可视化与计算机图形学汇刊》，24 4 1623 1632, 2018 2 (-):–。W. Shibayama 和

[30] S. Shirakawa。基于强化学习的重定向控制器，用于虚拟迷宫环境中的高效重定向行走。载于《计算机图形学进展》：第 37 届计算机图形学国际会议 CGI 2020，日内瓦，瑞士，2020 年 10 月 *20 23, 2020,* *37*, 33 45 2020 1, 3 – 论文集，第 –

[31] 页。Springer，。M. Simpson 、 K.-F. Richter 、 J. O. Wallgrün 和 A. Klippel。量化空间，理解心智：一种视觉摘要方法。《空间信息科学杂志》，(14):95–136，2017。2

[38] P. Spangenberger、S.-C. Freytag 和 S. M. Geiger。在沉浸式虚拟现实中融入自然：多感官刺激对增强自然联结和亲环境行为是否至关重要？《计算机与教育》，212:104964，2024。9[39] F. Steinicke、G. Bruder、J. Jerald、H. Frenz 和 M. Lappe。对人类重定向行走敏感度的分析。载于 2008 年 ACM虚拟现实软件与技术研讨会论文集，第 149 156, 2008 3 – 页。F. Steinicke、G. Bruder、J. Jerald、H. Frenz 和 M. Lappe。重定向行走技术检测阈值的估算。《IEEE可视化与计算机图形学汇刊》，16(1):17–27，2009。3[41] F.-Y. Sun、W. Liu、S. Gu、D. Lim、G. Bhat、F. Tombari、M. Li、N. Haber 和 J. Wu。LayoutVLM：通过视觉语言模型对三维布局进行可微优化。arXiv 预印本 arXiv: *2412 02193*, 2024 9 。。Q. Sun、A. Patney、L.-Y. Wei、O. Shapira、J. Lu、P. Asente、S. Zhu、M. McGuire、D. Luebke 和 A. Kaufman。迈向虚拟现实中的无限行走：动态扫视重定向。《ACM图形学汇刊》（TOG），37 4 1 13, 2018 3, 9 (-):–。X. Sun、H. Meng、S. Chakraborty、A. Bedi 和 A. Bera。超越文本：利用声音线索改善大语言模型在机器人导航任务中的决策。《机器学习研究汇刊》，2024。2[44] X. Sun、Y. Zhang、X. Tang、A. S. Bedi 和 A. Bera。TrustNavGPT：通过对不确定性建模提升音频引导的大语言模型机器人导航的可信度。arXiv 预印本 arXiv: *2408 01867*, 2024 2 。。M. R. Team。SpatialLM：用于空间理解的大语言模型。https://github.com/manycore-research/SpatialLM，2025。9[46] J. Thomas 和 E. S. Rosenberg。利用人工势函数实现重定向行走的通用反应式算法。载于 2019 IEEE虚拟现实与三维用户界面会议（VR），第 56–62 页。IEEE，2019。1，3[47] M. Usoh、K. Arthur、M. C. Whitton、R. Bastos、A. Steed、M. Slater 和 F. P. Brooks Jr. 行走> 原地行走> 虚拟环境中的飞行。载于第 26 届计算机图形学与交互技术年会论文集，第 359 364, 1999 1 – 页。X. Wang、C. Yeshwanth 和 M. Nießner。SceneFormer：基于Transformer模型的室内场景生成。载于 2021 年国际三维视觉会议（ *3* 106 115 2021 2 DV），第 – 页。IEEE，。J. M. Wiener、G. Franz、N. Rossmanith、A. Reichelt、H. A. Mallot 和 H. H. Bülthoff。等视域分析能够捕捉与运动和体验相关的空间属性。《感知》，36 7 1066 1083, 2007 2 (-):–。N. L. Williams、A. Bera 和 D. Manocha。ARC：面向复杂环境中重定向行走的基于对齐的重定向控制器。《IEEE可视化与计算机图形学汇刊》，27 5 2535 2544, 2021 1, 3, 7, 9 (-):–。N. L. Williams、A. Bera 和 D. Manocha。利用可见性多边形在静态与动态场景中实现重定向行走。《IEEE可视化与计算机图形学汇刊》，27 11 4267 4277, 2021 1, 3, 4 (-):–。N. L. Williams、A. Bera 和 D. Manocha。ENI：量化虚拟现实中自然行走的环境兼容性。载于 2022 IEEE虚拟现实与三维用户界面会议（VR），第 419–427 页。IEEE，2022。2，3，4[53] N. L. Williams 和 T. C. Peck。考虑视场、性别和干扰物的旋转增益阈值估算。《IEEE可视化与计算机图形学汇刊》，25 11 3158 3168, 2019 2, 4 (-):–。Y. Yamada、Y. Bao、A. K. Lampinen、J. Kasai 和 I. Yildirim。评估大语言模型的空间理解能力。arXiv 预印本 arXiv:2310.14540，2023。2[55] C.-F. Yang、W.-C. Fan、F.-E. Yang 和 Y.-C. F. Wang。LayoutTransformer：兼具概念与空间多样性的场景布局生成。载于 IEEE/CVF计算机视觉与模式识别会议论文集，第 3732 3741, 2021 2 – 页。X. Yang、Y. Man、J. Chen 和 Y.-X. Wang。SceneCraft：布局引导的三维场景生成。《神经信息处理系统进展》，37 82060 82084, 2024 2 :–。Y. Yang、F.-Y. Sun、L. Weihs、E. VanderBilt、A. Herrasti、W. Han、J. Wu、N. Haber、R. Krishna、L. Liu 等。Holodeck：语言引导的具身人工智能环境生成。载于 *IEEE/CVF*计算机视觉与模式识别会议，第 16227 16237, 2024 2 – 页。Z. Yang、D. Liu、C. Wang、J. Yang 和 D. Tao。面向复杂场景生成的图像构图建模。载于 *IEEE/CVF*计算机视觉与模式识别会议论文集，第 7764 7773, 2022 2 – 页。R. Zhang、D. Jiang、Y. Zhang、H. Lin、Z. Guo、P. Qiu、A. Zhou、P. Lu、K.-W. Chang、P. Gao 等。Mathverse：你的多模态大语言模型真的能看懂视觉数学问题中的图表吗？*arXiv* 预印本 *arXiv:2403.14624*，2024年。2[60] S.-H. Zhang、C. Chen 和 S. Zollmann。用于虚拟现实重定向行走的一步式异位重置。*IEEE*可视化与计算机图形学汇刊，29(7):3327–3339，2022年。9