arXiv:2602.10154v1 [cs.CR] 9 Feb 2026

PRISM-XR：借助多模态大语言模型赋能注重隐私的扩展现实协作

Jiangong Chen* Mingyu Zhu† Bin Li‡

电气工程系，宾夕法尼亚州立大学，大学公园，宾夕法尼亚州 16802，美国

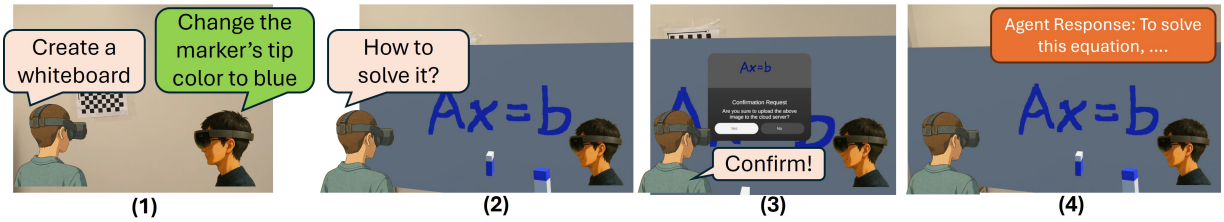


图 1：PRISM-XR 中多用户协作示例。在此示例中，两位用户在同一物理空间中协作创建虚拟白板、编辑对象属性，并利用多模态大语言模型（MLLM）智能体协助解决数学问题。用户只需说出请求，PRISM-XR 便会自动执行指令并同步虚拟物体。为保护隐私，系统会自动裁剪采集到的帧，并在将裁剪后的帧上传至云端多模态大语言模型之前请求用户确认。

摘要

多模态大语言模型（MLLM）将自然语言与视觉输入相结合，能够灵活创建物体和动画，从而增强扩展现实（XR）环境中的协作。然而，扩展现实头显采集的视觉数据包含现实世界背景，其中可能含有无关或敏感的用户信息，例如留在桌上的信用卡或其他用户的面部身份信息。将这些帧上传至云端多模态大语言模型会带来严重的隐私风险，尤其是在未经用户明确同意的情况下处理此类数据时。此外，商业扩展现实应用程序接口中现有的共址和同步机制依赖耗时且侵犯隐私的环境扫描，并且难以适应集成多模态大语言模型的扩展现实环境高度动态的特性。本文提出 PRISM-XR，这是一种新颖的框架，通过提供注重隐私的多模态大语言模型集成，促进扩展现实中的多用户协作。PRISM-XR 在边缘服务器上进行智能帧预处理，以过滤敏感数据并移除无关上下文，然后再与云端生成式人工智能模型通信。此外，我们还引入了轻量级注册流程和完全可定制的内容共享机制，以实现用户间高效、准确且保护隐私的内容同步。数值评估结果表明，该平台在满足用户请求方面的准确率接近 90%，注册时间不到 0.27 秒，同时空间不一致性小于 3.5 厘米。此外，我们开展了一项经机构审查委员会（IRB）批准、包含 28 名参与者的用户研究，结果显示，我们的系统能够在超过 90% 的场景中自动过滤高度敏感的物体，同时保持较强的整体可用性。

索引术语：多模态大语言模型、人工智能智能体、扩展现实、用户隐私。

*共同第一作者。电子邮箱：jiangong@psu.edu
†共同第一作者。电子邮箱：mintrrey@psu.edu
‡电子邮箱：binli@psu.edu

1 引言

将多模态大语言模型集成到扩展现实应用中（见 [51,56,57,63]），彻底改变了沉浸式环境中的用户协作。近年来，在元宇宙日益普及的推动下，VRChat和Meta Horizon Worlds等社交扩展现实应用受到了广泛关注。多模态大语言模型的集成增强了这些体验，使用户能够实时协作构建和修改虚拟世界（见 [12, 16, 38]），从而丰富互动并促进更紧密的社交联系。除虚拟协作之外，多模态大语言模型还增强了对物理环境的情境理解（见 [9, 52]），使用户在共享同一物理空间时能够无缝互动。例如，在办公工作区或家庭环境中，多名佩戴扩展现实头显的用户可以通过语音指令动态定制周围环境。一个人可能会说：“在这里添加一块虚拟白板和一些彩色笔。”另一位用户可以开始在白板上绘图，然后提示：“总结我们写下的内容，并根据绘图生成数字物体。”系统会立即作出响应，在共享空间中放置交互对象，使所有参与者都能实时与数字叠加层互动并对其进行修改。

尽管潜力巨大，将多模态大语言模型集成到协作式扩展现实应用中仍面临若干挑战：i）XR 设备捕获的视觉输入可能包含敏感的用户信息（见[1, 58]），例如遗留在桌上的信用卡，或其他用户的面部身份信息；在用户面对面互动的物理环境或私人空间中，这一问题尤为突出。未经明确的用户同意，将此类数据上传至由商业实体管理的云服务器，会引发严重的隐私问题（见 [11, 23]）。ii）现实世界环境通常包含大量无关物体，导致多模态大语言模型难以准确提取满足用户请求所需的上下文（见 [26, 50]）。iii）在物理空间中与共享数字物体进行无缝交互，需要用户之间实现精确的坐标对齐（见 [37, 61]）。然而，现有的共址解决方案要么需要耗时且侵犯隐私的环境扫描（见 [19, 39, 40]），要么对保持基准标记可见有着难以接受的限制（见 [17, 43]）。iv）由多模态大语言模型驱动的扩展现实应用具有高度动态性，这给传统网络同步机制带来了兼容性挑战（见 [28]）；而传统机制主要

是为预定义场景和用户操作而设计的。

由此引出了以下研究问题：i）如何精简视觉输入，在保护用户隐私、减少上下文噪声的同时，仍提供足够的信息以实现无缝交互？ii）是否存在一种低成本、高效、准确且注重隐私的共址策略，能够确保现实用户获得沉浸式协作体验？iii）考虑到由多模态大语言模型驱动的扩展现实应用具有高度动态的特性，如何在用户之间高效同步扩展现实环境中的物体和动画？先前研究（参见 [6, 13, 18]）探索了将多模态大语言模型集成到扩展现实应用中的各种方法，主要采用脚本生成或结构化数据。然而，这些研究尚未解决与基于云的多模态大语言模型服务器通信时的隐私泄露风险。尽管一些研究（参见 [11, 25, 45]）提出了在扩展现实环境中模糊或替换敏感物体以保护用户隐私的方法，但这些方法可能导致环境上下文缺失或被篡改，从而妨碍多模态大语言模型的有效集成。

本文提出 PRISM-XR，这是一个兼顾隐私保护与共享的协作平台，将多模态大语言模型集成到扩展现实环境中。用户只需通过简单的语音命令创建虚拟对象和动画，即可借助 PRISM-XR 与现实环境无缝交互并展开协作。PRISM-XR 利用本地上下文收集、边缘服务器上的隐私保护处理，以及功能强大的闭源商业人工智能（AI）模型，通过多阶段管线处理用户的语音输入，并融合多模态数据，生成可转换为扩展现实任务的结构化 JSON 输出。为保护隐私并消除上下文噪声，PRISM-XR 会将敏感视觉输入转换为文本描述，或在将裁剪帧上传至云端前明确征求用户确认。对于处于同一物理空间的用户，一次性且不到一秒的注册流程可确保后续整个游戏会话中的所有用户操作无缝同步。PRISM-XR 是一个简化的扩展现实交互与协作平台，专为受过教育但没有编程或扩展现实技术经验的普通用户设计。

PRISM-XR 的关键设计要点总结如下。首先，PRISM-XR 在边缘服务器上采用最先进的目标检测模型来分析用户的视场（FoV），并生成文本摘要，用以替代发送给多模态大语言模型的提示词中的原始图像帧。若生成的描述不足以满足用户请求，PRISM-XR 将根据该描述指定裁剪区域，并在上传裁剪部分之前请求用户确认。其次，PRISM-XR 通过轻量级注册流程，将商用扩展现实设备的本地扩展现实跟踪系统与基于马克笔的跟踪方法相融合。用户可以自由探索物理环境，而无需依赖基准标记；基准标记仅在初始化期间使用，以促进虚拟内容对齐。最后，PRISM-XR 具备一种轻量且完全可定制的同步机制，专门针对多模态大语言模型响应和扩展现实用户交互。扩展现实相关物体及其属性存储在边缘服务器上，并在本地同步，以降低延迟和带宽消耗。

本文的主要贡献如下：

- 我们提出了一种隐私感知帧处理方法，可自动排除敏感信息，避免将其上传至云服务器，从而保护用户隐私。
- 我们提出了轻量且可定制的用户空间对齐与虚拟内容同步技术，使多模态大语言模型驱动的扩展现实应用能够实现有效的多用户协作。
- 我们开发并开源[1] 一个可全面部署的端到端系统，将所有组件串联起来。该系统具有一个用户友好的界面，支持自然语言输入，并且设计上易于扩展，可直接用于实际部署。
- 我们进行了系统层面的数值评估和初步用户研究，以评估 PRISM-XR 的可用性。结果展现了其强大的功能，同时指出了进一步优化的机会。

本文其余部分安排如下：第2节总结相关领域的现有研究。第3节介绍所提系统的概述与工作流程。第4节详细阐述系统的关键设计方面。第5节介绍系统的软件与硬件实现。第6节和第7节介绍针对人类受试者开展的系统性能与可用性评估。第8节总结全文。

2 相关工作

本节总结四个关键领域的相关文献：大语言模型智能体、扩展现实中的大语言模型、多用户XR应用，以及扩展现实中的用户隐私保护。

大语言模型智能体。随着多模态大语言模型的近期进展，人工智能智能体已成为热门研究主题。研究人员探索了多种方法，以利用多模态大语言模型自动、智能地处理复杂任务，而无需人工干预。一些先前研究（见 [44, 54]）尝试在电子游戏中应用大语言模型，其中非玩家角色（NPC）展现出可信的类人行为。其他研究（见 [5, 55, 62]）将人工智能智能体融入端到端机器人控制，从而实现涌现式语义推理或交互式机器人行为。与此同时，较新的研究（见 {v5）开发人工智能智能体以加速科学发现，例如开展自主文献综述、提出研究想法以及规划实验。

扩展现实中的大语言模型。近期研究还显示出将大语言模型智能体应用于扩展现实环境的新兴趋势。这些环境通过支持用户与突破物理限制的虚拟物体交互，让用户沉浸其中。一些先驱性研究（见 [6, 9, 13, 18, 63]）将人工智能智能体引入扩展现实环境；然而，这些研究主要依赖文本上下文，并生成代码脚本或JSON数据来控制交互对象或动画。其他研究（见 [51,56]）借助多模态大语言模型探索现实世界图像输入，但并未解决用户隐私问题，因为用户采集的图像会在未通知用户的情况下直接共享。此外，这些研究大多针对单用户交互，忽略了多用户XR应用面临的挑战与机遇，例如共址问题和高效同步。较新的一项研究（见 [28]）探讨了利用大语言模型协作设计虚拟现实世界，但未考虑增强现实/混合现实场景，尤其是用户位于同一现实世界空间中的情形。关于扩展现实与人工智能融合的更广泛综述，请参阅 [22, 53]。

多用户扩展现实应用。 除了用户、人工智能智能体与扩展现实环境之间的交互外，基于扩展现实的协作还在广泛的应用场景中引入了额外的沉浸感。研究人员已经证明，多用户扩展现实系统在协同构建、 [59], 消防员训练 [15], 和远程学习等领域具有良好的效果（见 [7, 8]）。然而，这些研究并未探讨将多模态大语言模型智能体集成到协同扩展现实应用中的潜在益处。此外，真实世界环境中的多用户协作相比单用户场景面临更大的定位挑战。每位用户可能拥有不同的视角和运动轨迹，这使得合并个体地图并在共享环境中保持准确定位变得十分困难。一些工作（见 [14, 32, 35]）通过关注高效的全局地图构建，以及降低多用户与服务器之间的通信延迟来解决这些问题。另一些工作（见 [34, 48]）则研究交换视觉特征或关键帧的方法，以对齐用户的坐标系。

[1] https://github.com/SNeC-Lab-PSU/PRISM-XR

然而，这些研究主要关注相机定位，并未深入探讨多模态大语言模型驱动的扩展现实协作的动态，尤其是虚拟物体或动画的同步问题。

扩展现实中的用户隐私保护。随着人工智能驱动的分析和人际互动不断融合，扩展现实中的隐私问题已引起研究者的关注 [47]。以往研究强调了诸多风险，例如旁观者数据泄露，以及在用户不知情的情况下提取个人信息的隐蔽机器学习操作 [30, 42], ，而对意外暴露和现实感知偏差的担忧也进一步阻碍了用户采用扩展现实系统 [29]。近期的隐私增强解决方案，包括通过眼动追踪和空间感知实现旁观者选择性保护[11] ，以及通过模式化追踪替换物体，以在扩展现实直播中隐藏敏感物品 [25], ，通过设备端处理、选择性遮蔽和安全推理流程来降低这些风险。然而，在协作式扩展现实中集成多模态大语言模型也带来了新的隐私挑战 [60], ，因为这些模型通常依赖云端处理，可能会在缺乏用户明确控制的情况下上传敏感视觉数据。从本地处理转向远程服务器，意味着需要制定新的策略，以保护多模态大语言模型驱动的扩展现实环境中的隐私。如需全面了解扩展现实环境中的隐私问题，我们建议读者参阅 [2, 20, 47]。

3 系统概述

本节概述 PRISM-XR。这一框架支持多名用户在同一物理空间内协作，通过语音输入创建虚拟对象和动画。PRISM-XR 扩展了 LLMER [9] 框架，以支持多用户交互，并引入了将现实世界图像输入纳入其中的新方法，从而增强场景理解。
PRISM-XR 的系统架构包括客户端数据采集，以及在边缘端和云服务器上的多层处理，如图 2所示。

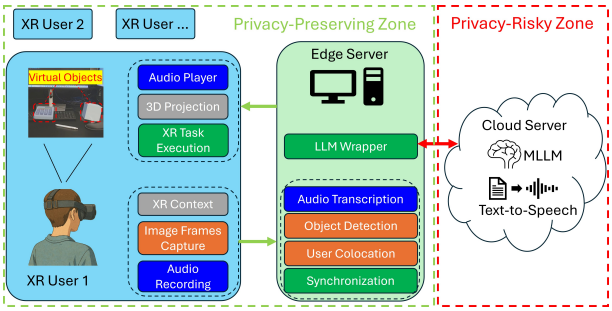


图 2：PRISM-XR 的系统架构。

不同于依赖繁琐手动输入 [13], 基于手势的音频触发 [9], 或基于控制器的音频激活 [18], 的现有系统，PRISM-XR 采用了更灵活的关键词激活方式。用户以预先设定的关键词开头发出请求，检测到该关键词时便会自动开始音频录制，并在检测到静音时停止。录制的音频片段使用 OpenAI Whisper 模型进行转录 [46]。为确保用户隐私，所有处理均在本地完成：关键词检测和音频录制直接在客户端设备上执行，而音频转录则在边缘服务器上执行。这种设计无需将敏感音频数据发送至外部服务。此外，XR 设备会采集一帧反映用户视场（FoV）的画面，并记录采集时的相机位姿。边缘服务器不会将原始帧上传至云端，而是采用注重隐私的处理方式：提取该帧的文本描述并将其纳入提示词，或裁剪该帧，并在将裁剪后的帧纳入提示词之前征求用户同意。收集到用户的语音命令和上下文信息后，边缘服务器会与云端 AI 服务器通信，以处理请求并解析云端返回的 JSON 输出，从而确保在客户端设备上可靠渲染。

为促进多用户协作，PRISM-XR 要求用户完成轻量级注册流程。该流程采用半标记式方法，高效、准确地对齐用户的坐标系，同时让用户能够灵活地在真实世界环境中移动，不受视场约束。多模态大语言模型的所有响应都会广播给已注册用户，使他们能够同时观察到在同一物理空间中渲染的物体和动画。PRISM-XR 还采用基于所有权感知和变更驱动的数据通信方式，确保可抓取立方体等可交互对象无缝同步。该设计支持沉浸式扩展现实环境中的协作操作，例如交接虚拟物体。考虑到用户处于同一物理空间时可能出现的关键词冲突，我们为每位用户分配不同的关键词。下一节将详细阐述本系统的设计原则。

4 系统设计

本节将讨论 PRISM-XR 的三项关键设计原则，这些原则借助多模态大语言模型和真实世界视觉输入，实现多用户协作。首先，我们介绍 PRISM-XR 的隐私感知帧处理，这对于向多模态大语言模型提供简洁且安全的真实世界上下文至关重要。接着，我们概述基于半马克笔的用户注册流程，旨在确保高效、准确且灵活的用户共址。最后，我们介绍 PRISM-XR 的内容同步机制，该机制专为满足多模态大语言模型支持的动态环境需求而设计。

4.1 隐私感知帧处理

为了在扩展现实世界中实现更精细的交互，多模态大语言模型需要视觉输入，因为视觉输入能够提供超越粗略文本描述的上下文信息。然而，真实世界环境往往包含敏感信息，尤其是在多用户协作场景中。例如，当一名用户在与虚拟棋盘互动时面对另一名用户，采集的屏幕截图可能包含此人的面部；若将其上传至云服务器，就会引发隐私顾虑 [11, 23]。为解决隐私顾虑，
PRISM-XR 提出了一种隐私感知帧处理机制，通过自动帧裁剪和明确的用户同意请求来保护隐私，其工作流程如图 3所示。

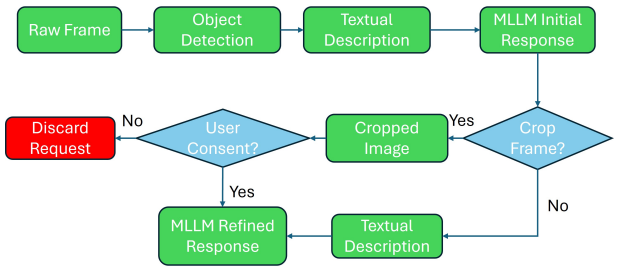
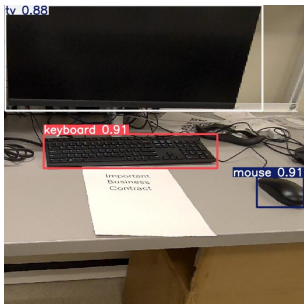
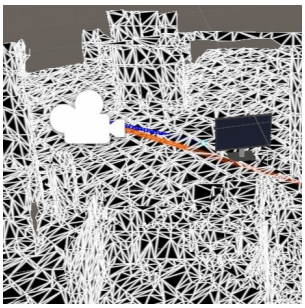


图 3：隐私感知帧处理工作流程。

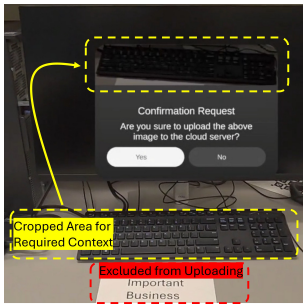
PRISM-XR 不会将 XR 设备采集的原始帧上传到云服务器，而是由边缘服务器将图像转换为文本描述，以保护用户隐私。系统采用先进的 YOLO v11 模型 [27], ，该模型因准确率高、泛化能力强且在各种场景中得到广泛应用而被选用。模型的输出包括检测到的物体名称、中心像素坐标、边界框和置信度。检测结果示例如图 4a所示，用户视场的文本描述格式如下：



(a) 对包含敏感物体的原始图像进行 YOLO 检测的结果。



(b) 真实世界环境的碰撞体与射线投射。



(c) 确认对话框，其中已排除敏感物体。

图 4：隐私感知帧处理关键步骤示例。

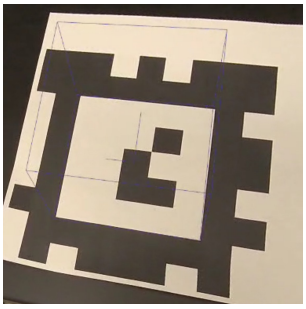


图 5：使用线框立方体成功注册用户。

键盘，中心点 (327.80, 352.12)，边界框 (66.47, 66.03, 589.13,

这些数据会纳入多模态大语言模型的提示词中，以提供相关的现实世界上下文。例如，当检测到用户的手时，多模态大语言模型便能感知用户的身体手势，从而理解“把物体移到这里”这类模糊请求。由于目标检测仅输出图像帧内的像素坐标，因此需要进行额外处理来计算准确的三维世界位置。与 [51], 类似, PRISM-XR 会记录帧捕获时的相机位姿，并从相机位置沿着由像素坐标确定的方向发射射线。射线会与真实世界环境中的碰撞体相交；该环境由微软 HoloLens 和 Meta Quest 3 等 XR 设备动态构建。随后，将命中点设为目标对象的初始位置。为了让渲染出的物体看起来更加自然，PRISM-XR 会进行位姿校正。此过程根据物体的大小调整其位置，以确保其正确附着于表面，并使其朝向与表面法线对齐。图 4b展示了在 HoloLens 设备上根据像素坐标创建物体的示例。请注意, LLMER [9], 提出的在世界坐标中创建物体或动画的方式，也受 PRISM-XR 支持，且无需经过此帧处理流程。

在仅靠文本描述不足以满足需求的场景中，PRISM-XR 会利用多模态大语言模型自动裁剪帧。具体而言，我们将文本描述纳入提示词，并要求模型返回一个指定待裁剪区域的“CropArea”属性。若有必要，多模态大语言模型会返回一个表示该区域的射线数组；如果请求不需要帧信息，则只会返回“None”。图 4c展示了一个示例：用户请求获取键盘的颜色信息，而仅凭文本描述无法捕捉这一信息。在这种情况下，YOLO 模型会检测键盘的位置，多模态大语言模型会识别相关裁剪区域，而 PRISM-XR 则会生成一帧排除敏感内容的裁剪帧，供用户审核。只有在用户确认裁剪图像不包含敏感信息后，它才会被纳入发送至云服务器的提示词中；未经裁剪的帧仅由边缘服务器处理，不会上传至云端。请注意，还可以采用其他方法处理帧，例如对敏感物体进行模糊处理或灰度化。不过，这些方法需要准确检测敏感物体，而这通常需要通过微调来实现更个性化的检测。相比之下，我们的方法借助 YOLO 模型检测常见物体，无需微调，并利用多模态大语言模型的能力智能确定裁剪区域。除了保护用户隐私外，裁剪后的帧还能去除无关上下文，使多模态大语言模型专注于关键细节。例如，裁剪图像会排除桌子和椅子等无关元素，从而减少对多模态大语言模型的干扰。此外，商业 API 通常根据像素数计算图像输入的 token 消耗量，这意味着裁剪图像可以显著降低 token 成本。

4.2 基于半标记的用户注册

多用户扩展现实应用的一项关键要求，尤其是涉及现实世界背景且多名用户身处同一物理位置的应用，就是对齐用户的坐标系。这样可以确保无论各个用户的视角如何，虚拟物体都被感知为附着在相同的物理空间中。然而，Meta共享空间锚点(SSA) [39], 微软 Spatial Anchor Sharing [40], 和 Google Cloud Anchors [19] 等商业API要求用户扫描环境并将数据上传到云服务器。这个过程不仅耗时，还会引发严重的隐私问题。

相比之下，PRISM-XR采用一种轻量且注重隐私的方法来对齐用户的坐标系。该方法集成了基于标签的定位系统 AprilTag [43], , 以便高效完成每位用户的注册流程。尽管 AprilTag是一项成熟的技术，我们的主要贡献在于将其无缝集成到商业设备的内部定位算法中。具体而言，我们仅使用 AprilTag来对齐用户的世界坐标系，而不覆盖头戴式显示器原有的定位能力。用户只需识别附近固定位置的标签并说出“注册”，PRISM-XR便会采集屏幕截图，并应用AprilTag检测器来识别标签，估算其在用户世界坐标系中的位姿。如果成功识别标签，边缘服务器会将标签位姿保存为用户的注册信息，将位姿发送回用户，并在用户的XR 设备上于标签位置渲染一个线框立方体，如图5所示。如果未检测到标签，边缘服务器会通知用户注册失败，并建议再次尝试。完成注册后，可通过简单的基变换和计算，将一个物体在某位用户坐标系中的位姿转换到另一位用户的坐标系中；同时，用户的移动不会受到 AprilTag的限制，例如必须让标签始终处于视野内。请注意，每次启动应用时都需要注册；为保护用户隐私，所有处理均在本地完成，不使用云服务器。此外，如果用户发现其他用户同步的物体出现错位，可以随时重新执行注册流程以纠正偏差。

4.3 内容同步

除坐标变换之外，多用户协作系统还必须确保场景中虚拟内容的及时同步，例如物体的创建或修改。我们不依赖 Photon Fusion 等商业API——这些API会将用户数据上传并存储到云服务器——而是利用边缘服务器，并设计一种定制的同步协议，以适配

由多模态大语言模型驱动的动态环境。该方法在保持隐私并尽量降低延迟的同时，确保高效的内容共享。我们将内容同步分为两大类：多模态大语言模型响应和用户交互，每一类都需要不同的同步策略。

4.3.1 同步多模态大语言模型响应

回顾一下，来自云端 AI 服务器的所有响应，例如创建物体或动画的响应，均以 JSON数据 的形式构成。这些响应可以广播给所有用户，以确保虚拟世界中的一致性。由于 JSON 字符串相对较短，因此占用的带宽极少，延迟也很低。然而，某些多模态大语言模型响应（例如在用户视场中的特定位置创建物体）依赖于特定用户的上下文，其他用户无法获取这些上下文。为此，我们定义了多模态大语言模型生成的虚拟内容姿态所使用的三个坐标空间：i）世界空间：指发出请求的用户在世界坐标系中的绝对三维姿态。ii）局部空间：用于作为子级附着于其他物体的物体，或在局部坐标系内运行的动画。iii）像素空间：与射线投射过程配合使用，详见 第 4.1 节，并如 图 4b所示。创建虚拟内容的用户被指定为所有者用户。每个多模态大语言模型响应都会发送给所有者用户，由其将内容的姿态从局部空间或像素空间转换到世界空间，然后应用必要的坐标变换并广播给其他用户。

4.3.2 同步用户交互

用户驱动的修改（例如用户创建的可交互对象的位姿更新）需要单独的同步机制。PRISM-XR 采用变化检测策略，主动监测并同步虚拟对象属性的变化。只有当变化超过预设阈值，且距上次同步的时间超过指定间隔时，才会发送更新。这些参数经过设定，以平衡带宽消耗与同步流畅度。在多用户场景中，多个用户与同一虚拟对象交互时可能会产生冲突。为解决这一问题，PRISM-XR 区分自有对象和同步对象：i) 自有对象：用户直接交互的物体。其属性由客户端上传至边缘服务器。ii) 同步对象：用户未直接拥有的物体。其属性通过接收来自边缘服务器的数据进行更新。当用户与某个对象交互时，该对象会自动加入其自有对象列表，并从同步对象列表中移除。同时，边缘服务器会通知原所有者相应地更新其列表。

为提高通信效率，PRISM-XR 使用专门的数据结构来同步可交互对象，而非依赖 JSON 字符串。该结构将属性编码为原始字节，使已注册用户之间每个对象的同步消息大小仅为 48 字节（对象 ID 占 4 字节，3D 位置占 12 字节，四元数形式的 3D 旋转占 16 字节，3D 缩放占 12 字节，特殊事件处理占 4 字节）。此外，我们使用原生套接字和局部坐标变换，进一步降低通信开销。

5 系统实现

本节详细介绍 PRISM-XR 在客户端、边缘服务器上的软硬件实现，以及评估所用的物理环境。

客户端。 PRISM-XR 的客户端应用以 Unity 项目的形式使用 C# 开发，构建于开源框架 LLMER [9] 之上。请注意，原始 LLMER 框架仅支持 Meta Quest 系列，而我们的平台同时支持 Meta Quest设备和 Microsoft HoloLens 2。在 HoloLens 版本中，我们集成了 Microsoft 混合现实工具包 3（MRTK3），以支持扩展现实交互，例如抓取虚拟物体。屏幕截图通过 Unity.Windows 包中的 PhotoCapture API 采集。该 API 提供所采集帧的相机位姿和投影矩阵。关键词激活机制通过 MRTK3 的 KeywordRecognitionSubsystem 实现，而音频录制和静音检测则由 Unity 的 AudioModule 管理。初始音频录制时长设为 2 秒，之后每次延长 1 秒，直至音频幅度持续低于预定义阈值。所有这些处理均直接在客户端设备上离线执行。在 Quest 版本中，我们使用 Meta XR 一体化 SDK 构建基本的扩展现实交互。该 SDK 中的 Passthrough Camera API 用于采集用户的视场，并支持在现实世界环境中进行射线投射。我们使用 Porcupine Unity 包实现关键词激活，同时采用与 HoloLens 相同的录音和静音检测机制。在两种设备上，我们使用 System.Net.WebSockets 包与边缘服务器建立套接字通信，通过该方式在 第 4.3 节 中提到的 JSON 字符串和序列化字节通过双向 TCP 连接传输。为确保用户之间的交互流畅并避免带宽过载，用户交互的同步频率设为 60 Hz。

边缘服务器。 PRISM-XR 的边缘服务器使用 Python 开发，同时支持 Linux 和 Windows 操作系统。它使用 OpenAI 官方 Python API 与云服务器交互，以生成多模态大语言模型响应，并使用文本转语音（TTS）模型生成合成语音。GPT-4o 模型的结构化输出模式可确保响应符合指定的 JSON 模式，并强制执行相关约束，例如要求物体位置为由三个浮点数组成的数组。然而，模型确定精确位置的准确性取决于所提供的上下文及其自身能力。为实现注重隐私的音频转录和帧处理，服务器部署了离线版 OpenAI Whisper 模型 [46] 和 YOLO v11 [27] 模型。套接字服务器基于 Python 中的 WebSocket API 构建。

硬件与物理环境。我们已在运行 Windows 10 Holographic OS 的 Microsoft HoloLens 2 和运行 Meta Horizon OS v74 的 Meta Quest 3 上部署客户端应用程序。在初步测试过程中，我们发现，Meta Quest设备上的用户体验全面优于 HoloLens。除非另有说明，实验结果均来自部署在 Meta Quest 3 设备上的客户端，其每只眼睛的分辨率为 2064×2208 像素。边缘服务器运行于 La mbda 工作站，配置如下：Intel Core i9-10980XE CPU @3.00GHz × 36、NVIDIA GeForce RTX 3070 显卡 × 4、128 GB 内存、4 TB 磁盘，以及 Ubuntu 22.04 LTS。边缘服务器通过以太网连接到 TP-Link Archer AX50 路由器，客户端设备则通过 WiFi 5 (802.11ac) 连接到路由器。服务器使用额外的网络端口访问互联网，以便与云端生成式人工智能服务器通信。所有实验和用户研究均在一间研究实验室（约 550 平方英尺）内进行，XR 设备、边缘服务器和路由器均置于其中。虽然大多数操作可以站立完成，但用户也可以在房间内自由走动，以获得更沉浸式的体验。

6 数值研究

我们首先开展一系列数值研究，以评估 PRISM-XR 的系统级性能，包括云端或边缘端各种多模态大语言模型生成响应的准确率、共置准确率与效率，以及同步延迟。

6.1 多模态大语言模型响应的准确率

PRISM-XR 涉及与多模态大语言模型通信的两个阶段：初始阶段 旨在解析并分类请求，获取

裁剪区域及相应类别；精化阶段 则根据初始阶段的响应整合适当的提示词，从而获取结构化响应。我们评估了两个阶段中多模态大语言模型生成响应的准确率，并比较了云端模型与运行在边缘服务器上的开源模型的性能。具体而言，我们测试了作为云端模型的 OpenAI o1、o3-mini 和 GPT-4o，以及通过 Ollama 框架部署的边缘端模型 LLaVA 13B 和 Llama 3.2 Vision 11B。对于通常不专门用于大规模 AI 模型的普通研究实验室而言，更大的模型超出了边缘服务器的能力范围；而较小的模型则难以生成结构化输出，常常导致处理陷入无限循环。

为了评估初始阶段，我们针对不同类别提出了 15 个典型的用户请求，并为其配备相应的真实世界上下文（视场描述，详见 第 4.1 节）。其中，七个请求需要来自裁剪图像的视觉上下文（例如，翻译真实世界书籍上的文字），其余八个仅凭文本描述即可完成（例如，在文本上下文中将物体移向检测到的物体）。在需要裁剪图像的请求中，四个明确指定了目标区域，另外三个则未明确指定。为确保生成的请求符合实际用户请求，我们手动筛除不恰当的请求，并优化预期响应。通过大语言模型生成请求并进行适当修正，相较于直接提出用户请求，可以消除偏差并增强通用性。我们采用以下四项指标评估性能：分类准确率、裁剪召回率、裁剪漏报率和生成时间。分类准确率表示正确分类的请求所占比例。裁剪召回率衡量返回的裁剪区域与预期区域之间的交并比（IoU）超过 0.5 的情况所占比例；这一数值足够高，表明已恰当提供真实世界视觉上下文。裁剪漏报率统计预期结果为“无”但错误生成了裁剪区域的情况，这可能导致不必要的弹出窗口。理想情况下，模型应在保持较低裁剪漏报率和生成时间的同时，实现较高的准确率和裁剪召回率。

指标	云端			Edge	
	o1	o3-mini	GPT-4o	LLaVA	Llama V
解析请求、获取裁剪区域和类别的初始阶段					
CA（%）	88.67	72.67	**89.33**	30.67	23.33
CR（%）	71.43	72.86	**78.57**	1.43	7.14
CF（%）	**0**	5.00	1.25	95.00	15.00
GT（秒）	11.407	8.305	**2.415**	7.032	3.164
获取结构化响应的精化阶段					
准确率$_{obj}$ (%)	96	85	**97**	1	29
时间$_{obj}$ (s)	20.285	9.832	**2.509**	7.192	3.835
准确率$_{ani}$ (%)	**100**	100	99	0	1
时间$_{ani}$ (s)	9.708	9.929	**3.611**	10.345	4.469

表 1：各模型的性能比较。CA、CR、CF 和 GT 分别是分类准确率、裁剪召回率、裁剪漏报率和生成时间的简称。准确率$_{obj}$ 和 准确率$_{ani}$ 分别表示对象创建或动画创建响应正确的成功率，而 时间$_{obj}$ 和时间$_{ani}$ 则分别表示它们所消耗的时间。

我们对 OpenAI 的模型和通过 Ollama 本地部署的模型均采用结构化输出，以确保符合指定的 JSON 模式。所有受测模型使用完全相同的提示词，并将每个请求重复 10 次，以计算平均性能指标。如表 1所示，云端模型在分类准确率、裁剪召回率和裁剪漏报率方面始终优于边缘端模型，这凸显了采用云端处理的必要性，但也带来了亟待解决的重大隐私问题。有趣的是，与 GPT-4o 相比，o1 和 o3-mini 的裁剪召回率较低，这可能是因为它们更倾向于对裁剪区域返回“None”。请注意，裁剪召回率的效果受用户请求详细程度的影响：用户提供清晰、明确的指示时，正确进行图像裁剪的可能性会显著提高。具体而言，仅考虑明确描述目标区域的请求时，GPT-4o 的裁剪召回率可达 100%。此外，与较小的边缘端模型相比，GPT-4o 的生成时间更短。这种效率得益于云服务器更强的处理能力，也因为较小的模型难以进行结构化输出生成，往往需要多次迭代，从而增加延迟。

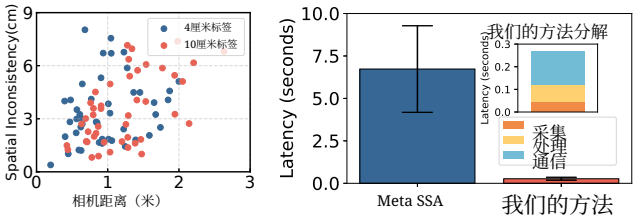
评估精化阶段时，某些请求需要进行主观人工评估（例如，判断响应是否符合用户意图）。在第7 节进行全面的主观评估之前，我们先在此关注确定性任务，例如在指定位置生成物体，以及创建目标位于特定现实世界位置的动画。如果对象创建响应正确指定了预制体名称、坐标空间（像素空间）和像素位置（位于检测到的物体边界框 1.5 倍范围内），则视为成功。同样，动画响应必须具有正确的动作类型、物体名称、坐标空间和像素位置。请注意，尽管我们的框架支持三种不同的坐标空间，但我们仅评估需要使用像素空间的请求，因为这最符合现实世界交互。此外，我们采用 1.5 倍系数来容许较粗略的位置描述，以涵盖物体“在附近”等情况，确保左侧和右侧的位置都被视为有效。与初始阶段类似，我们为每个类别（对象/动画）提出 10 个典型的用户请求，并重复实验 10 次以计算平均值。如表 1所示，结果表明云端模型仍优于边缘端模型，且精化阶段的准确率更高，这是因为请求中的歧义有所减少。鉴于 GPT-4o 表现出色，我们在用户研究中选用它。

6.2 共址精度与效率

PRISM-XR 采用半标记式方法来对齐不同用户间虚拟物体的位姿，其对齐准确率可使用 [48] 提出的空间不一致性指标进行评估。该指标量化了多个用户渲染相同虚拟内容时的偏差。我们从不同位置和角度注册客户端设备，并计算 XR 设备内部定位系统估算的变换矩阵与根据 AprilTag 得出的变换矩阵之间的位置差异。

为分析相机距离（即用户注册设备所在位置之间的距离）与空间不一致性之间的关系，我们使用两种不同尺寸的标签，收集了约 70 对用户的数据。结果如 图6a 所示，其中每个数据点代表一对已注册的用户。我们的分析表明，当用户在 3 米范围内注册时，空间不一致性较低，平均为 3.46 厘米。总体而言，用户之间距离越近，不一致性越低。此外，在距离较远时，使用较大的标签可以提升性能，因为采集帧的分辨率有限，导致在较远距离难以检测较小的标签。值得注意的是，注册完成后，空间不一致性不会随时间增加，因为后续定位仅依赖 XR 设备的内部定位系统。例如，据全面测量报告，Meta Quest 3 的追踪性能稳定，相对位姿误差小于 1 厘米，绝对位姿误差小于 8 厘米 [24]。

除了空间不一致性外，我们还评估了用户注册过程中的延迟，测量范围从检测到用户的注册命令到渲染出线框立方体。我们以 Meta SSA 作为基线，这是一种由 Meta Quest设备支持的流行共址解决方案。我们运行一个示例



(a) 准确率。 (b) 延迟。

图 6：注册评估。

共址应用 10 次，并记录两位用户的初始化时间。我们的结果如 图 6b 所示，我们的注册总延迟平均仅约为 0.27 秒，比 Meta SSA 测得的 6.73 秒快近 25 倍。

6.3 延迟分析

本节重点评估从用户语音请求到系统执行以及多用户之间虚拟内容同步的端到端延迟。理想情况下，我们期望系统在数秒内呈现结果，同时确保同一物理空间中的所有用户同时感知虚拟对象的创建与修改。然而，由于网络和处理延迟，发起操作的用户与观察到相应变化的其他用户之间存在差异。为全面分析系统延迟，我们测量以下关键指标：

- **语音到操作延迟**：从用户语音录制结束到优化后的响应被执行的时间，例如渲染虚拟物体或动画。
- **响应同步延迟**：从所有者执行多模态大语言模型响应到其他用户渲染同步响应之间所经过的时间。
- **交互同步延迟**：一名用户与虚拟对象交互的时刻与其他用户观察到该对象所应用更改的时刻之间的时间差。

由于不同设备间的时钟同步可能导致超过 100 毫秒的偏差，[41], 我们引入了一名伪用户，以准确测量同步延迟。这名伪用户与真实用户使用同一设备，接收服务器转发的消息，并与真实用户共享完全相同的时间线。随后，一名真实用户在现实世界环境中操作 PRISM-XR 约 20 分钟，自由发出与周围环境有关的语音请求，同时记录系统延迟。在实验期间，系统的平均帧率达到 72.34 帧/秒（FPS）。此外，我们为语音到操作延迟和响应同步延迟各收集了约 40 个样本，并为交互同步延迟收集了约 40,000 个样本。样本量差异的原因在于，前两种延迟类型取决于用户生成的请求，通常每分钟只有几次；而交互同步延迟则以每个对象 60 Hz 的频率进行采集。我们在 表 2中展示了各类延迟的均值和标准差。需要注意的是，语音到操作延迟不包括在边缘服务器上通过目标检测模型进行屏幕截图捕获和图像处理所耗费的时间。由于这些过程会在用户开始说话时启动，通常会在语音录制结束前完成。此外，本地处理时间包括上下文数据收集、射线投射，以及边缘服务器和客户端上的任何文件 I/O 或序列化操作。系统总处理时间的计算不包括用户确认，因为不同用户所需的确认时间各不相同。

延迟类型		数值（秒）
语音到操作单用户请求	转录	0.453 ± 0.188
	初始阶段	1.905 ± 1.125
	用户确认	1.566 ± 2.760
	文本转语音	1.549 ± 1.099
	精化阶段	3.162 ± 1.277
	本地处理	0.520 ± 0.623
	通信	0.177 ± 0.194
	不含确认的总计	7.744 ± 2.086
响应同步		1.169 ± 1.023
交互同步		0.016 ± 0.006

表 2：同步的延迟分解。

平均而言，PRISM-XR 处理每个用户请求约需 7.744 秒，虽然稍显漫长，但正如我们在第7节的用户研究所示，仍在可接受范围内。此外，响应同步延迟平均约为 1.169 秒，用户对此并不会特别明显地察觉。可交互对象创建后，交互同步延迟低至 15.53 毫秒，这种响应性是基于云端的同步机制无法实现的。

7 初步用户研究

除了对系统性能进行数值研究之外，我们还希望通过人类参与者评估主观用户体验。我们还希望收集用户反馈，以指明进一步改进的方向。在本节中，我们开展一项初步用户研究，并分析结果，以便更全面地评估我们系统的可用性与实用性。

7.1 流程

我们招募了 28 名参与者（13 名女性、15 名男性；平均年龄：27.57 岁），组成 14 对小组参与用户研究[2]。参与者人数的选择遵循先前研究针对“调试”测试所建议的指南（见 [3]）。参与者包括学生和在职人员，学科专业背景多样，涵盖工程学（50%）、科学（29%）和文科（21%）。这一参与者群体聚焦于在教育和游戏领域进行创新、背景各异的个人，与我们框架的目标用户相符。在 28 名参与者中，大多数是扩展现实新手（64%），较少的参与者自认为处于中级水平（25%）或专家水平（11%）。在软件编程经验方面，参与者的分布较为均衡：29% 的参与者经验不足 1 年，25% 超过 5 年，其余为 1 至 4 年。在游戏引擎经验方面，54% 的参与者经验不足 1 年，另有 32% 为 1 至 2 年，只有 14% 超过 3 年。这一分布体现了参与者背景的多样性。

每次用户研究约持续一小时，流程均相同。用户研究开始时，我们会向参与者简要介绍研究动机以及通过使用我们的系统可以完成的操作。此外，我们还会要求他们根据第7.2.1节中的详细说明，将环境中的一系列物体划分为不同的隐私级别。随后是一个为期10分钟的培训环节，让他们熟悉基本操作，例如通过表达请求来创建立方体、为其着色和移动它。之后，参与者需要提供人口统计信息，并按顺序完成一系列任务。每完成一项任务，他们都需要填写NASA任务负荷指数（TLX）[21] 表格，以衡量其感知工作负荷。所有任务完成后，参与者还可以用10分钟自由探索系统，提出他们感兴趣的任何请求。实验结束后，我们会要求参与者

[2]所有研究流程均遵循宾夕法尼亚州立大学机构审查委员会（IRB）的伦理指南，并根据协议编号 STUDY00026260 获得批准。

填写一份包含若干李克特量表问题的问卷，并通过半结构式访谈回答几个开放式问题。用户研究设计的详情如下：

• 预定义的用户任务。我们定义了以下任务，涉及对象创建、动画创建、场景理解和协作绘图。每项任务都有明确的目标，同时允许灵活执行。用户需在开始每项任务前完成注册流程，每组会交换角色并各执行每项任务两次。

1. 使用现实世界输入创建对象。第一位用户创建一个附着在现实世界物体上的虚拟对象（从候选列表中选择，包括立方体、球体、杯子、键盘等简单物体）。另一位用户将该对象的颜色改为与现实世界环境中的某个现有物体相似。2. 使用现实世界输入创建动画。第一位用户创建一个虚拟对象。另一位用户将该对象移至特定的现实世界位置，例如椅子旁边。3. 细粒度场景理解。第一位用户让智能体描述其视野中的物体。另一位用户让智能体提供有关现实世界物体的信息，例如简要介绍一本书。4. 协作绘图。第一位用户创建一块白板，并配备马克笔和橡皮擦。两位用户共同绘制和擦除内容。绘制完成后，第二位用户请智能体识别他们画的内容。

• 问卷。我们基于现有调查问卷设计了一份问卷 [9, 31, 36], , 从多个方面评估系统，包括沉浸感、可用性、交互性和隐私。问卷采用七点李克特量表，范围从 1（强烈不同意）到 7（强烈同意）。问卷包含以下陈述：

(Q1) 使用 PRISM-XR 时，我感受到了沉浸感。(Q2) 系统生成的内容符合我的预期。(Q3) 我喜欢系统提供的交互方式。(Q4) 系统响应迅速，能在合理时间内响应我的指令。(Q5) 系统操作简单。(Q6) 与另一位用户协作很简单。(Q7) 使用 PRISM-XR 时，我觉得自己的隐私得到了保护。(Q8) 我希望未来继续使用 PRISM-XR。(Q9) 我对整体体验感到满意。

• 半结构式访谈。我们提出了若干开放式问题，鼓励参与者在完成所有任务后分享他们的体验。这些问题涵盖了使用我们系统的各个方面，包括可用性、隐私保护、协作体验和期望。此外，参与者也可以自由分享问题中未涉及的想法。这一环节旨在收集用户真实的反馈，包括满意之处和批评意见。问卷与半结构式访谈之间虽有意设置了一些内容重叠，但半结构式访谈旨在引导参与者分享更深入的定性洞察，这些洞察无法仅通过量表作答表达。

7.2 定量结果与讨论

7.2.1 隐私指标

为评估我们的系统在保护用户隐私方面的有效性，我们将 12 类物体引入实验环境：杯子、书桌、椅子、书籍、包、笔记本电脑/手机、显示器、实验室设备和仪器、人脸、药品/医疗记录、身份证/驾驶证，以及邮件/个人信件/对账单。我们要求参与者将这些物体划分为三个隐私级别：不敏感、可能敏感和高度敏感。通过这一分类，我们可以评估系统根据用户感知的隐私风险过滤内容的有效性。对于根据用户请求采集的每一帧，我们都记录了应用隐私感知帧处理方法前后这些物体的出现情况。

隐私级别	原始帧	裁剪帧
不敏感的	96.43%	50.00%
可能敏感	82.14%	53.57%
高度敏感	53.57%	7.14%

表3：每个隐私级别中至少包含一个对象的帧。

如表 2所示，我们的方法显著减少了高度敏感物体的出现，将其占比从 53.57% 降至 7.14%，从而有效防止此类内容被上传至云端。边界情况通常涉及敏感物体与用户预期目标之间的重叠，例如，用户想要交互的书籍上放着一张信用卡。在这些情况下，我们的系统会通过提示用户手动确认是否应上传裁剪后的帧，提供最后一道防护。虽然包含不敏感物体和可能敏感的物体的帧的占比仍然相对较高，但这通常是必要的，因为任务要求与这些物体进行交互。尽管如此，观察到的减少仍然表明，我们的方法成功过滤了无关内容，帮助多模态大语言模型专注于目标对象，并减少潜在的隐私泄露。

7.2.2 可用性指标

每项任务和指标的 NASA-TLX 得分以箱线图的形式呈现在图 7中，得分范围为 1 至 20，得分越低表示感知工作负荷越低。在所有任务和指标中，得分均保持相对较低（所有平均得分均低于 6/20），这表明用户能够有效利用我们的系统，与现实世界环境实现无缝协作和交互。

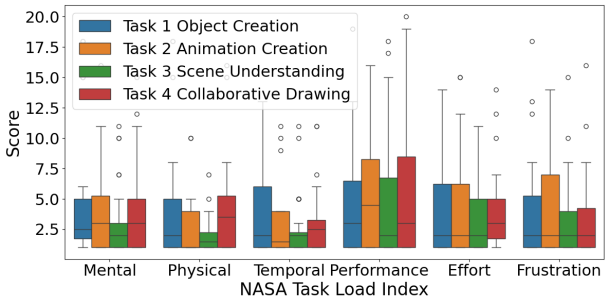


图7：所有任务的 NASA-TLX 得分。

我们还统计了任务完成时间和任务完成度，如表4 和图8所示。具体而言，任务完成时间是从用户开始执行任务，到用户表示任务完成所经过的时间。请注意，我们没有限制用户针对一项任务可以发出的请求数量。用户可以将所有指令整合在一起，也可以将其拆分成较小的请求。根据 [9]中的定义，每项任务的完成情况分为四个等级：成功、轻微问题、严重问题和失败。结果显示，用户完成任务1至3所用的时间（~2 分钟）相近，而完成任务4所用的时间更长。这是因为用户通常会花更多时间构思随机绘图或在白板上书写，并在上传到服务器之前修改内容。此外，超过91%

完成度被报告为成功或存在轻微问题，表明无论用户背景如何，系统都易于使用。

Item	均值（秒）	标准差（秒）
任务 1	129.31	67.58
任务 2	127.16	90.41
任务 3	137.41	73.18
任务 4	203.96	117.68

表4：TCT统计数据。

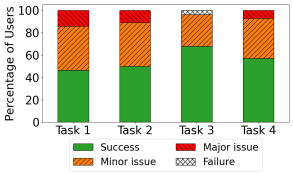


图8：完成度。

与NASA-TLX得分类似，问卷回答结果以箱线图的形式呈现于图9中。总体而言，用户对系统的反馈较为积极，所有问题的得分中位数均介于5至7分之间。尤其是，用户隐私保护和协作获得了较高评价，平均得分分别为5.6/7和5.8/7。这些结果表明，用户对系统的隐私感知帧处理和同步机制充满信心。另一方面，响应速度的平均得分最低，为5.0/7，说明这是影响整体用户体验的主要瓶颈。

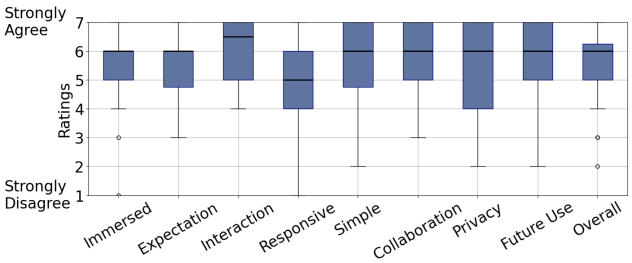


图9：问卷结果。

7.3 用户反馈与讨论

我们通过访谈收集了28名用户的248条评论，并进行了主题分析 [10]，以更好地了解用户使用我们系统的体验。

7.3.1 正面体验

沉浸式协作。许多参与者对沉浸式XR体验以及系统对交互式多用户协作的支持表示非常满意。用户称赞与他人一起在共享虚拟空间中协作十分引人入胜。这一主题被频繁提及，反映出系统成功营造了自然的共址式XR交互体验。

感知到的隐私保护。另一个突出的主题是用户对系统隐私保护设计的赞赏。几位用户肯定了保护个人数据的机制，还有一些用户甚至提出了额外的策略。一位参与者建议：“如果敏感物体无法排除，可以创建虚拟物体将其遮挡起来。”这表明用户积极认同通过虚拟遮挡来扩展隐私保护的理念。

无缝同步。参与者还称赞了虚拟内容在不同用户之间的流畅同步。这一主题反映了用户对多用户内容共享技术可靠性的认可，也与我们在表 2中报告的较低交互同步延迟相符。一位用户评论道：“看到虚拟立方体如此流畅地响应另一位用户的手势，真是令人惊叹。”

7.3.2 局限性与改进方向

进一步提升隐私保护。虽然目前系统要求用户在将裁剪图像上传到云端前手动确认，但反馈表明，用户有时可能会习惯性地确认，而不检查内容。为此，参与者建议添加文本警告，列出检测到的物体，或在裁剪图像中用方框标出这些物体。此外，对于敏感物体与非敏感物体相互重叠的情况，还可以采用动态遮挡技术作为补充方案，例如叠加虚拟遮罩或遮住敏感物体，然后上传部分遮挡帧。这些改进有助于促使用户更加审慎地确认，并进一步加强隐私保护。

语音识别的鲁棒性。唤醒词激活系统是一个常被提及的问题。参与者指出，关键词识别容易受到口音或语调变化的影响，有时还会误触发。此外，将用户语音转录为文本请求并不总是准确，而翻译不准确会导致系统执行方向有误。这一主题表明，需要更可靠、更便于用户使用的语音交互机制。通过集成更先进的声学处理技术，系统可以进一步提高识别准确率和整体用户体验。

注册流程的可靠性。一些用户反映，轻量级注册流程存在困难。虽然每次尝试耗时很短，但一些参与者发现，为了与协作者实现精准对齐而多次重复这一流程令人沮丧。这一发现凸显了采用更稳健的注册机制的重要性，尤其是在存在遮挡或光照变化的情况下。未来的工作可以探索集成先进的视觉特征提取和定位算法，以改善坐标空间对齐，并维持流畅的用户体验。

响应延迟较高。如第 6.3 节所述，PRISM-XR 中语音到操作的总延迟平均约为 7.7 秒。尽管在研究中，大多数用户认为这一延迟可以接受，但对于实时扩展现实应用而言，这一延迟仍然相对较高。主要瓶颈在于云端处理时间，这受到商业平台共享使用负载以及大型多模态大语言模型推理延迟的影响。未来可着重于在边缘服务器上部署更小型的开源多模态大语言模型，以降低延迟，同时保持足够的能力。这一转变还有助于增强隐私保护并降低运营成本。

用户研究的可扩展性和范围有限。当前用户研究仅涉及双人交互和简短的预定义任务。为了更好地评估实际应用效果和系统稳定性，未来研究应纳入更长期或大规模的评估，涉及更多用户和更复杂的环境。此外，某些用户界面元素还可进一步改进。例如，当前通过语音或手势输入实现的确认机制可能导致用户注意力不集中地“直接点击通过”。因此，需要开展专门研究，以找出更可靠的方法来获取用户确认和反应。

8 结论

本文提出了一种由多模态大语言模型驱动的协作式隐私保护 XR 平台。通过将边缘辅助帧处理与明确的用户同意机制相结合，我们的系统能够对真实环境进行细粒度的情境理解，同时保护用户隐私。为便于同一物理空间中的多名用户无缝交互，我们引入了一种利用标签检测的半标记式注册流程，以及一种完全可定制的内容同步机制。为了评估系统的可用性，我们开展了系统级性能评估和经机构审查委员会批准的初步用户研究。结果表明，我们的方法效率出色，同时也指出了有待进一步优化的关键领域。

致谢

本文使用生成式人工智能工具润色文字，并为预告图片生成虚拟形象。本研究部分由美国国家科学基金会通过资助项目 CNS-2152658 和 M3X-2420351，以及美国国防高级研究计划局资助项目 HR0011-2420366 提供支持。

参考文献

[1] M. Abraham、P. Saeghe、M. Mcgill 和 M. Khamis。《扩展现实对隐私、安全和行为的影响：专家见解》。载于北欧人机交互会议，第 1 12, 2022 1 – 页。M.

[2] Alkaeed、A. Qayyum 和 J. Qadir。《人工智能与扩展现实（AI-XR）元宇宙中的隐私保护：综述》。《网络与计算机应用期刊》，第 103989, 2024 3 页。N.

[3] Bevan、C. Barnum、G. Cockton、J. Nielsen、J. Spool 和 D. Wixon。《“神奇数字5”对网页测试来说够用吗？》。载于 CHI'03 计算系统中的人因扩展摘要，第 698– 699, 2003 7 页。D. A. Boiko、R. MacKnight 和 G. Gomes。《大语言模型涌现出的自主科学研究能力》。arXiv 预印本 arXiv: *2304 05332*, 2023 2 。. A. Brohan、N. Brown、J. Carbajal、Y. Chebotar、X. Chen、K. Choromanski、T. Ding、D. Driess、A. Dubey、C. Finn 等。《RT-2：视觉-语言-动作模型将网络知识迁移至机器人控制》。arXiv 预印本 arXiv: *2307 15818*, 2023 2 。. J. Chen、T. Lan 和 B. Li。

[4] [5] [6] 《GPT-VR Nexus：由 ChatGPT 驱动的沉浸式虚拟现实体验》。载于 2024 IEEE 虚拟现实与三维用户界面会议摘要与研讨会（VRW），第 01 02 2024 2 – 页。IEEE，。J. Chen、F. Qian 和 B. Li。《利

[7] 用商用移动设备提升协作式虚拟现实的体验质量》。载于 2022 IEEE 第42届分布式计算系统国际会议（ICDCS），第 1018 1028 2022 2 – 页。IEEE，。J. Chen、X.

[8] Qin、G. Zhu、B. Ji 和 B. Li。《基于运动预测的无线调度：面向多用户全景视频流传输》。载于 IEEE INFOCOM 2021——IEEE 通信会议，第 1 10 2021 2 – 页。IEEE，。J. Chen、X. Wu、T. Lan 和

[9] B. Li。《 LLMER：利用大语言模型生成的 JSON 数据构建交互式扩展现实世界》。《IEEE可视化与计算机图形学汇刊》，2025年。1、2、3、4、5、8[10] V. Clarke 和 V. B. 和。《主题分析》。《积极心理学杂志》，12(3):297–298，2017年。doi：10.1080/17439760.2016.1262613 9[11] M. Corbett、B. David-John、J. Shang、Y. C. Hu 和 B. Ji。《BystandAR：保护增强现实系统中的旁观者视觉数据》。载于第21届移动系统、应用与服务国际年会论文集，第 370 382, 2023 1, 2, 3 – 页。H.

[12] Davies 和 L. Hjorth。《封控期间的 Roblox：理解年轻人在疫情中的数字社交游戏》。《疫情时期的游戏与玩家》，第 15, 2024 1 页。F. De La Torre、C. M. Fang、

[13] H. Huang、A. Banburski-Fahey、J. Amores Fernandez 和 J. Lanier。《LLMR：使用大语言模型实时提示交互式世界》。载于 CHI 计算系统中的人为因素会议论文集，第 1–22 页，2024年 2、3 – 。[14] A. Dhakal、X. Ran、Y. Wang、J. Chen 和 K. Ramakrishnan。《Slam-share：用于实时多用户增强现实的视觉同步定位与建图》。载于第18届新兴网络实验与技术国际会议论文集，第 293 306, 2022 2 – 页。Z. Dong、J. Chen 和

[15] B. Li。《基于协作混合现实的消防员训练》。载于 IEEE INFOCOM 2023——IEEE 通信会议研讨会（INFOCOM WKSHPS），第 1–2 页。IEEE, 2023 2 。S. Earle、F. Kokkinos、Y. Nie、J. Togelius 和 R.

[16] Raileanu。《Dreamcraft：在 Minecraft 中以文本引导生成具备功能的三维环境》。载于第19届数字游戏基础国际会议论文集，第 1 15, 2024 1 – 页。S.

[17] Garrido-Jurado、R. Munoz-Salinas、F. J. Madrid-Cuevas 和 M. J. Mar´ ´ın-Jim´enez。《遮挡条件下高可靠性基准标记的自动生成与检测》。《模式识别》，47(6):2280– 2292, 2014 1 页。D. Giunchi、N. Numan、E. Gatti 和 A. Steed。

[18] 《DreamcodeVR：通过语音驱动编程推动虚拟现实中的行为设计普及》。载于 2024 IEEE 虚拟现实会议与三维用户界面（虚拟现实），第 579–589 页。IEEE，2024年。2，3

[19] 谷歌。云锚点允许不同用户共享增强现实体验 | | 1, 4 ARCore 谷歌开发者。H. Hadan、

[20] D. M. Wang、L. E. Nacke 和 L. Zhang-Kennedy。沉浸式扩展现实中的隐私：探索用户感知、担忧与应对策略。载于《2024年 CHI 计算系统中的人为因素会议论文集》，第 1 24, 2024 3 – 页。S. Hart。

[21] NASA-TLX（任务负荷指数）的开发：实证与理论研究结果。《人类心理负荷》/爱思唯尔，1988年。7[22] T. Hirzle、F. M¨uller、F. Draxler、M. Schmitz、P. Knierim 和 K. Horn-bæk。扩展现实与人工智能相遇——关于扩展现实与人工智能的范围综述。载于《2023年 CHI 计算系统中的人为因素会议论文集》，第 1 45, 2023 2 – 页。J. Hu、A.

[23] Iosifescu 和 R. LiKamWa。Lenscap：一种用于增强现实应用精细化视觉隐私控制的分离式处理框架。载于《第19届年度国际移动系统、应用与服务会议论文集》，第 14 27, 2021 1, 3 – 页。T.

[24] Hu、F. Yang、T. Scargill 和 M. Gorlatova。苹果对比 Meta：对当前最先进扩展现实头显空间追踪的比较研究。载于《第30届年度国际移动计算与网络会议论文集》，第 2120 2127, 2024 6 – 页。Y.

[25] Hu、M. Zhu、Q. Jin、F. Qian 和 B. Li。MagicCloth：保护增强现实流媒体中的用户隐私。载于《第1届 ACM 移动沉浸式计算、网络与系统研讨会论文集》，第222–228页，2023年。2, 3[26] Y. Huang、Y. Wang、Z. Xu、C. Gao、S. Wu、J. Ye、X. Chen、P.-Y. Chen 和 X. Zhang。打破专注：大语言模型中的上下文干扰诅咒。arXiv 预印本 arXiv: *2502 01609*, 2025 1 。。G. Jocher、

[27] J. Qiu 和 A. Chaurasia。Ultralytics YOLO，2023年1月。3, 5[28] A. Kobenova、C. DeVeaux、S. Parajuli、A. Banburski-Fahey、J. A. Fernandez 和 J. Lanier。社交幻化：借助人工智能协同构建虚拟三维世界的多用户运行时协作。arXiv 预印本 arXiv:2410.00274，2024年。1, 2[29] L. Lammerding、T. Hilken、D. Mahr 和 J. Heller。真实得令人不适：衡量消费者对增强现实信息隐私的担忧。载于《增强现实与虚拟现实：沉浸式技术新趋势》，第 95 108 2021 3 –页。施

[30] 普林格，。S. M. Lehman、A. S. Alrumayh、K. Kolhe、H. Ling 和 C. C. Tan。隐藏在眼前：探索移动增强现实应用的隐私风险。《ACM 隐私与安全汇刊》， 25 4 1 35, 2022 3 ( ): – 。J. R. Lewis。IBM 计算机可用性满意度问卷：

[31] 心理测量评估与使用说明。《国际人机交互期刊》，7 1 57 78, 1995 8 ( ): – 。F. Li、S.

[32] Yang、X. Yi 和 X. Yang。CORB-SLAM：一种面向多机器人的协作式视觉 SLAM 系统。载于《协同计算：网络、应用与工作共享国际会议》，第 480 490 2017 2 – 页。施普林格，。R. Li、T.

[33] Patel、Q. Wang 和 X. Du。MLR-Copilot：基于大语言模型智能体的自主机器学习研究。arXiv 预印本 arXiv: *2408 14033*, 2024 2 。。

[34] T. Li、N. S. Nguyen、X. Zhang、T. Wang 和 B. Sheng。Promar：实用的基于参考物体的多用户增强现实。载于《IEEE INFOCOM 2020—IEEE计算机通信会议》，第1359–1368页。IEEE，2020年。2[35] L. Liu 和 M. Gruteser。EdgeSharing：边缘辅助的城市街道实时定位与物体共享。载于《IEEE INFOCOM 2021—IEEE计算机通信会议》，第1–10页。IEEE，2021年。2[36] A. M. Lund。使用问卷12衡量可用性。《可用性界面》，8 2 3 6, 2001 8 ( ): – 。H. Mecheri、X.

[37] Robert-Lachaine、C. Larue 和 A. Plamondon。八种对齐两个坐标系方向的方法的评估。《生物力学工程学期刊》，138(8):084501，2016年。1[38] T. Merino、M. Charity 和 J. Togelius。用于生成 Minecraft 结构的交互式潜变量进化。载于《第18届数字游戏基础国际会议论文集》。

游戏》，第 1 8, 2023 1 – 页。Meta。共享空间锚点 [39] | 1, 4 Meta [40] Horizon OS 开发者。微软。空间锚点共享。 1, 4
[41] D. L. Mills。《计算机网络时间同步：网络时间协议》。CRC 出版社, 2006 7 。 [42] J. O'Hagan、P. Saeghe、J. Gugenheimer、D. Medeiros、K. Marky、M. Khamis 和 M. McGill。隐私增强技术与日常增强现实：理解旁观者对知情与同意的不同需求。《ACM 交互、移动、可穿戴与泛在技术会议论文集》，6 4 1 35, 2023 3 ( ): – 。
[43] E. Olson。AprilTag：一种稳健且灵活的视觉基准标记系统。载于《2011年 IEEE 国际机器人与自动化会议》，第3400–3407页。IEEE，2011年。1, 4 [44] J. S. Park、J. O'Brien、C. J. Cai、M. R. Morris、P. Liang 和 M. S. Bernstein。生成式智能体：人类行为的交互式拟像。载于《第36届年度 ACM 用户界面软件与技术研讨会论文集》，第 1 22, 2023 2 – 页。F. Qian 和 B. Li。借助魔法绳提升远程多用户增强现实隐私。载于《第20届年度国际移动系统、应用与服务会议论文集》，第583 584页，2022年 2 – 。[46] A. Radford、J. W. Kim、T. Xu、G. Brockman、C. McLeavey 和 I. Sutskever。通过大规模弱监督实现稳健的语音识别。载于《国际机器学习会议》，第28492–28518页。PMLR，2023年。3, 5[47] S. Rajaram、C. Chen、F. Roesner 和 M. Nebeling。为多用户增强现实引出兼顾安全与隐私的共享技术。载于《2023年 CHI 计算系统中的人为因素会议论文集》，第 1 17, 2023 3 –页。X. Ran、C. Slocum、Y.-Z. Tsai、K. Apicharttrisorn、M. Gorlatova 和 J. Chen。通信高效且空间一致的虚拟物体多用户增强现实。载于《第16届新兴网络实验与技术国际会议论文集》，第 386 398, 2020 2, 6 – 页。S. Schmidgall、Y. Su、Z. Wang、X. Sun、J. Wu、X. Yu、J. Liu、Z. Liu 和 E. Barsoum。Agent Laboratory：使用大语言模型智能体担任研究助理。arXiv 预印本 arXiv: *2501 04227*, 2025 2 。。
F. Shi、X. Chen、K. Misra、N. Scales、D. Dohan、E. H. Chi、N. Sch¨ arli 和 D. Zhou。大语言模型很容易受到无关上下文的干扰。载于《国际机器学习会议》，第31210–31227页。PMLR，2023年。1[51] S. Srinidhi、E. Lu 和 A. Rowe。XaiR：一个将大语言模型与物理世界相结合的扩展现实平台。载于《2024年 IEEE 混合现实与增强现实国际研讨会（ISMAR）》，第 759 767 2024 1, 2, 4 – 页。IEEE，。S. Srinidhi、E. Lu、A. Singh、S. Kartik、A. Lin、T. Laroia 和 A. Rowe。一个将大语言模型与物理世界相结合的扩展现实平台。载于《第23届 ACM 嵌入式网络传感器系统会议论文集》，第 700 701, 2025 1 – 页。
Y. Tang、J. Situ、A. Y. Cui、M. Wu 和 Y. Huang。大语言模型与扩展现实的集成：当前趋势、挑战与未来展望的全面综述。载于《2025年 CHI 计算系统中的人为因素会议论文集》，第1–24页，2025年。2[54] G. Wang、Y. Xie、Y. Jiang、A. Mandlekar、C. Xiao、Y. Zhu、L. Fan 和 A. Anandkumar。Voyager：一种基于大语言模型的开放式具身智能体。arXiv 预印本 arXiv: *2305 16291*, 2023 2 。。Y.-J. Wang、B. Zhang、J. Chen 和 K. Sreenath。借助大语言模型提示机器人行走。arXiv 预印本 arXiv:2309 09969，2023年 2 。。[56] Y. Xiu、T. Scargill 和 M. Gorlatova。LOBSTAR：基于语言模型的增强现实遮挡检测。载于《2024年 IEEE 混合现实与增强现实国际研讨会增刊（ISMAR-Adjunct）》，第 335 336 2024 1, 2 – 页。IEEE，。Y. Xiu 、 T. Scargill 和 M. Gorlatova。ViDDAR：基于视觉语言模型的增强现实任务有害内容检测。arXiv 预印本 arXiv: *2501 12553*, 2025 1 。。T. Yamakami。扩展现实应用中的隐私威胁模型。载于《互联网、数据与网络技术进展：第8届新兴互联网、数据与网络技术国际会议（EIDWT-*2020* 384 394 2020 1)，第 – 页。施普林格，。X. Yao、J. Chen、T. He、J. Yang 和 B. Li。用于远程协作式乐高设计的可扩展混合现实平台。载于 *IEEE INFO- COM 2022-IEEE*计算机通信会议通信研讨会（*INFOCOM WKSHPS*），第 1 2 2022 2 – 页。IEEE，。Y. Yao、J. Duan、K. Xu、Y. Cai、Z. Sun 和 Y. Zhang。大语言模型（LLM）安全与隐私综述：优点、缺点与不足。 *High-Confidence Computing*，第 100211, 2024 3 。K. You、Q. Chen、P. Xie 和 S. Song。用于协作式移动传感器网络定位的基于距离的坐标对齐。 *IEEE*网络控制系统汇刊, 7 3 1379 1390, 2020 1 ( ): – 。W. Yu 、 N. Gileadi 、 C. Fu 、 S. Kirmani 、 K.-H. Lee 、 M. G. Arenas 、 H.-T. L. Chiang、T. Erez、L. Hasenclever、J. Humplik 等。面向机器人技能合成的语言到奖励。 *arXiv*预印本 *arXiv:2306 08647*，2023年2。。[63] M. Zhu、J. Chen 和 B. Li。当生成式人工智能遇上扩展现实：实现可扩展且自然的交互。 *IEEE*互联网计算，第 1–10 页，2026年。doi: 10.1109/MIC.2025.3619462 1，2