# SpaceBlender: Creating Context-Rich Collaborative Spaces Through Generative 3D Scene Blending


Nels Numan*
Microsoft Research
United States
University College London
United Kingdom
nels.numan@ucl.ac.uk

Shwetha Rajaram*
Microsoft Research
United States
University of Michigan
United States
shwethar@umich.edu

Balasaravanan Thoravi Kumaravel
Microsoft Research
United States
bala.kumaravel@microsoft.com

Nicolai Marquardt
Microsoft Research
United States
nicmarquardt@microsoft.com

Andrew D. Wilson
Microsoft Research
United States
awilson@microsoft.com


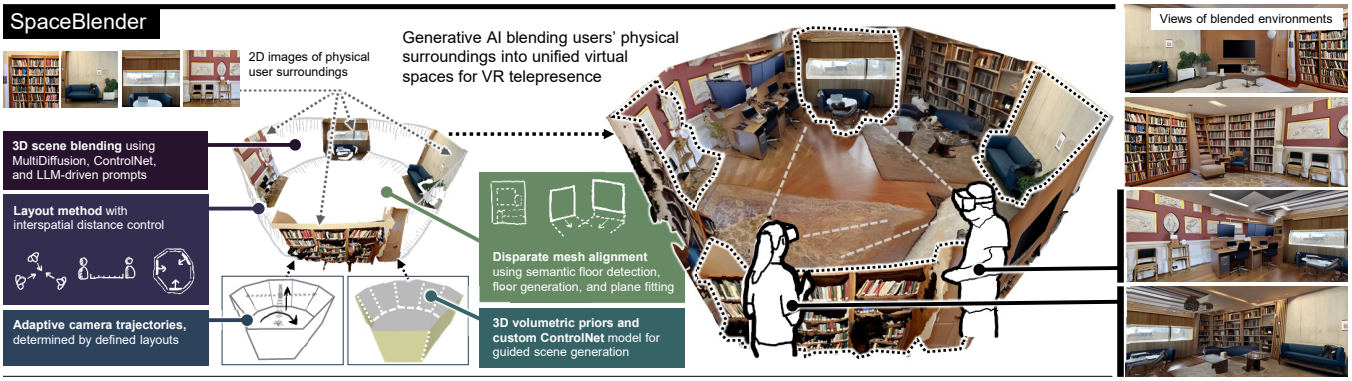


**Figure 1: Overview of SpaceBlender, a pipeline that extends state-of-the-art generative AI models to blend users' physical surroundings into unified virtual environments for VR telepresence.**


## ABSTRACT

There is increased interest in using generative AI to create 3D spaces for Virtual Reality (VR) applications. However, today's models produce artificial environments, falling short of supporting collaborative tasks that benefit from incorporating the user's physical context. To generate environments that support VR telepresence, we introduce SpaceBlender, a novel pipeline that utilizes generative AI techniques to blend users' physical surroundings into unified virtual spaces. This pipeline transforms user-provided 2D images into context-rich 3D environments through an iterative process consisting of depth estimation, mesh alignment, and diffusion-based space completion guided by geometric priors and adaptive text prompts. In a preliminary within-subjects study, where 20 participants performed a collaborative VR affinity diagramming task in pairs, we compared SpaceBlender with a generic virtual environment and a state-of-the-art scene generation framework, evaluating its ability to create virtual spaces suitable for collaboration. Participants appreciated the enhanced familiarity and context provided by SpaceBlender but also noted complexities in the generative environments that could detract from task focus. Drawing on participant feedback, we propose directions for improving the pipeline and discuss the value and design of blended spaces for different scenarios.



## CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; **Collaborative and social computing systems and tools**; • **Computing methodologies** → **Artificial intelligence**.

## KEYWORDS

generative AI, VR telepresence

**ACM Reference Format:**
Nels Numan, Shwetha Rajaram, Balasaravanan Thoravi Kumaravel, Nicolai Marquardt, and Andrew D. Wilson. 2024. SpaceBlender: Creating Context-Rich Collaborative Spaces Through Generative 3D Scene Blending. In *The*


*This work was done while the first two authors were interns at Microsoft Research. Both authors contributed equally to the paper.



*UIST '24, October 13–16, 2024, Pittsburgh, PA, USA*



ACM ISBN 979-8-4007-0628-8/24/10
https://doi.org/10.1145/3654777.3676361

*37th Annual ACM Symposium on User Interface Software and Technology (UIST '24), October 13–16, 2024, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 25 pages. https://doi.org/10.1145/3654777.3676361

## 1 INTRODUCTION

There is increased interest in integrating generative models into Virtual Reality (VR) development workflows to accelerate content creation in commercial tools[1] and enable end-user customization [12, 14, 54]. The recent proliferation of generative AI tools introduces low-effort techniques for end-users to create 3D objects [42, 53], panoramic images [6, 63–65], and 3D scenes [3, 19, 30, 58, 70, 79]. Many operate with minimal input such as text and images, offering an easier alternative to the conventional, labor-intensive process of modeling 3D scenes and paving the way for new forms of interactive systems.

In this work, we leverage these developments to explore creating custom virtual environments for VR telepresence systems. Prior research demonstrated various benefits of incorporating users' familiar real-world context into virtual environments in remote collaboration scenarios, such as supporting deixis [25, 48, 51, 61]), mutual awareness [29, 67, 69], and information recall [13, 18, 35, 36]. Motivated by these findings, we explore a generative approach to creating spaces by blending together multiple users' environmental contexts. This extends the body of work on aligning dissimilar remote spaces for mixed reality collaboration, e.g., via common object anchors [9, 25, 29, 34] or mesh overlays [29, 57].

We identify two key challenges with using today's generative AI models to augment the creation of 3D environments for VR telepresence. First, most models are aimed at producing fully synthetic output that is not grounded in real-world spaces [19, 21]. Models that attempt to do such grounding require input beyond text and image and can only ground themselves in a single space [30, 58, 65]. Second, the 3D meshes generated by these models are not explicitly optimized for use as VR environments. Through our development process, we found that these generated environments can pose core usability issues for VR telepresence, such as non-navigable pathways, distracting visual and geometric artifacts, and uncanny spaces that detract from user comfort.

To address these challenges, **we developed SpaceBlender, a novel pipeline that leverages and extends state-of-the-art generative AI techniques to blend users' physical surroundings into unified virtual spaces suitable for VR telepresence**. This pipeline transforms user-provided 2D images of distinct spaces into context-rich 3D environments through a multi-stage process. First, we transform the 2D images into 3D meshes based on depth estimation, depth alignment, and backprojection. We then employ a RANSAC-based alignment technique to align the disparate 3D meshes, ensuring a uniform floor level. Finally, we use a diffusion-based method for space completion, guided by geometric priors and text prompts defined by a Large Language Model (LLM) acting as an interior architect.

As a preliminary assessment of the suitability of environments generated by SpaceBlender in supporting collaborative VR, we conducted a comparative study with 20 participants. In pairs, they performed a VR-based affinity diagramming task in three different virtual environments: *(1)* Generic3D: a generic, low-poly room; *(2)* an environment generated with Text2Room [30]; and *(3)* a SpaceBlender environment incorporating input images of familiar physical locations provided by participants. Overall, participants experienced increased physical comfort and navigability in the Generic3D and SpaceBlender compared to Text2Room due to greater consistency in the room geometry. Furthermore, some leveraged recognizable environmental features in the SpaceBlender space to complete the clustering task. While participants envisioned future use cases where incorporating familiar or personal contextual details could provide value, they recommended improvements to the visual quality and realism of SpaceBlender's environments to better enable these use cases.

In summary, our work contributes *(1)* the SpaceBlender generative AI pipeline for creating VR telepresence spaces by blending users' physical surroundings into unified 3D environments; *(2)* a preliminary user study with 20 users that elicited potential benefits, limitations, and use cases of SpaceBlender, laying the groundwork for future generative AI tools for creating blended environments.

## 2 RELATED WORK

SpaceBlender builds on computational techniques for generating 3D artifacts and prior work that motivates the representation of physical spaces in VR telepresence.

### 2.1 Computational Generation of 3D Spaces

The domain of computational generation of 3D scenes has been significantly revitalized by recent advancements in generative methods. Unlike traditional procedural generation techniques that depend on predefined rules and asset libraries to assemble 3D environments [7, 10, 62, 68], recent approaches can generate entirely new spaces via novel generative techniques. These approaches use a variety of scene representations, including well-known explicit formats like 2D panoramic views [6, 63, 64] and meshes [19, 30, 58, 59], as well as recent implicit representations such as Neural Radiance Fields (NeRFs) [3, 75], 3D Gaussian Splats [8], and Signed Distance Functions (SDFs) [32].

The majority of recent scene generation methods are grounded in the 2D image domain due to the wide availability of image-caption datasets and state-of-the-art generation models trained on these [55]. By leveraging depth estimation models [1, 4], 2D images can be transformed into 3D representations by predicting a depth value for each pixel and backprojecting these values into 3D space. Utilizing models that can handle panoramic images, combined with image upscaling techniques to move beyond image generation models' limited output resolution (e.g., [2, 33]), systems such as Skybox AI[2] and LDM3D [63, 64] can generate high-quality 360° skyboxes based on user-provided text prompts. However, we observe that such techniques generate 3D spaces that do not stay spatially consistent during navigation due to their single-view generation approach, making them less suited for VR telepresence.

Other image-based scene generation methods address this limitation by generating multi-view image sequences [21, 30, 65]. One such method is Text2Room, which, given a camera trajectory with

[1] *Unity Muse*: https://muse.unity.com, *Bezi AI*: https://www.bezi.com/ai

[2] https://skybox.blockadelabs.com

matching text prompts and an optional starting image, iteratively expands a 3D mesh through text-conditioned inpainting of 2D rendered views of the mesh [30]. In each step, this method estimates a depth image for a rendered view, aligns it with known depth values, and backprojects the depth values to integrate them with the mesh. This is repeated for each viewpoint in the camera trajectory. Finally, an adaptive trajectory is defined at runtime based on a set of viewpoints with the highest number of missing pixels to complete the remaining holes in the mesh. While this approach results in high-quality scenes with improved multi-view consistency for scene navigation, the generated scenes still exhibit structural and visual irregularities, as well as contextual repetitions (e.g., multiple bathtubs in a single bathroom), which limit the navigability and realism required for telepresence.

To address geometric consistency and controllability, MVDiffusion [65] as well as recent parallel work such as Ctrl-Room [19] and ControlRoom3D [58] take additional inputs to act as spatial constraints, such as untextured 3D meshes [3, 65] or semantic scene maps [19, 58]. However, in typical camera-based telepresence systems, such priors are commonly unavailable. Furthermore, like Text2Room, these systems are unable to process multiple disparate images. SpaceBlender advances these concepts by not only accepting multi-image inputs from distinct spaces of collaborating users but also by leveraging the contextual insights from a Visual Language Model (VLM) [40], an LLM [47], and a semantic segmentation model [31] to autonomously generate a suitable layout, geometric prior, and text prompts for the generation of an environment directly from user-provided image frames.

## 2.2 Physical Spaces in VR Telepresence

A large body of prior work has studied telepresence systems for supporting users in collaborative tasks. A key objective in these systems is supporting mutual awareness, which refers to the shared understanding of *where* other users are and *what* they are doing [17, 26]. Much of this research focused on establishing awareness in unidirectional settings, where a singular physical space is captured and represented to one or more remote users [38, 45, 46]. This approach supports tasks centered on a single environment, such as remote assistance, by providing remote users with a view into a specific physical space without mutual visibility.

Recent work is increasingly focused on achieving bidirectional awareness to enable new interaction concepts in collaborative settings that not only resemble but also further extend face-to-face collaboration metaphors. These systems often integrate physical and virtual elements belonging to local user's or remote user's space into a common interaction space, which is referred to as *Extended Collaborative Space* (*xspace*) by Kumaravel and Hartmann [67]. Based on a literature review, Herskovitz et al. [29] identified three categories of techniques for creating such shared spaces: (1) object-centric methods, using specific objects to align spaces [9, 25, 34]; (2) perspective-driven methods, such as portals [37, 69] and world-in-miniature views [43, 52, 66]; and (3) mesh-based methods such as mesh overlays [43, 48, 57]. For example, Loki [66] used a world-in-miniature volumetric representation to provide real-time awareness cues of remote users' workspace contexts; RealityBlender [25] used multiple anchor objects (e.g., whiteboards, tables) to establish an object-centric interaction space; and Slice of Light [69] used a combination of interactive portals and mesh overlays to enable users to peek into and enter multiple distinct virtual environments.

However, few systems provide *xspaces* that coherently and flexibly include the physical contexts of all users simultaneously. SpaceBlender aims to enhance such collaborative settings where the inclusion of the physical context of all users is beneficial. Unlike prior systems with distinct boundaries, SpaceBlender leverages state-of-the-art generative AI models to create cohesively and smoothly blended contextual transitions between disparate spaces. While not explicitly studied in the current work, by incorporating familiar spaces in visually faithful ways, we seek to lay the groundwork for exploring the potential positive effect of these spaces in human information and memory recall [11, 13, 36, 39, 44].

Several works have employed spatial manipulation techniques to alter or combine spaces, such as Remixed Reality [43], which allowed users to make various changes in a live 3D reconstruction of their space, and PointShopAR [71], a tablet-based AR tool for modifying 3D point clouds of physical spaces. In contrast, SpaceBlender automates the customization of captured spaces using images of familiar places to create blended virtual environments, eliminating the need for manual intervention. This approach is particularly relevant for VR telepresence applications, where users often lack the time or ability to manually design or adjust virtual environments for each meeting.

# 3 SPACEBLENDER SYSTEM

This section details the SpaceBlender pipeline, designed to integrate images of the physical context of multiple users into cohesive virtual environments. SpaceBlender builds upon the Text2Room pipeline due to its extensibility through the usage of off-the-shelf 2D image models, transparent iterative generation process, and ability to initialize generation based on single 2D input images, which are commonly available in telepresence scenarios.

This section starts by outlining the system requirements in Sec. 3.1, followed by an overview of the proposed SpaceBlender pipeline in Sec. 3.2, and then provides a detailed description of its two phases in Secs. 3.3 and 3.4.

## 3.1 Requirements

The assumptions, implementation details, and limitations of the Text2Room framework present significant barriers to generating unified spaces from disparate input images for VR usage. Below, we outline the requirements of SpaceBlender, alongside the associated challenges and our approach to addressing them.

**Requirement 1: Enabling multiple disparate spatial inputs from diverse perspectives and locations.**

*Challenge*: Text2Room accepts at most one input image and does not register the resulting mesh in a global coordinate space, while SpaceBlender must be able to process and align multiple images with various viewpoints to support scene blending.

*Approach*: We introduce a floor plane alignment process that identifies the floor of each mesh by backprojecting semantic values into 3D space, which is then used for global alignment (Sec. 3.3.2). For

**Figure 2: A birds-eye view of two meshes that failed to blend due to the lack of geometric guidance and context throughout the iterative mesh completion process.**

the case where no floor is visible, we propose a technique to synthesize floor sections before alignment. The aligned meshes are then arranged with a parameter-based layout technique (Sec. 3.3.3).

**Requirement 2: Enabling coherent scene blending for realistic and context-rich spaces.**

*Challenge*: Text2Room uses low-resolution square images in its iterative view inpainting process. This limits the reference frame of the inpainting model, leading to mismatched mesh segments between disparate meshes with harsh geometric and visual boundaries and artifacts, as shown in Fig. 2.

*Approach:* We incorporate MultiDiffusion-based [2] image inpainting to complete wider images, extending the model's contextual reference window and enabling smooth blends.

*Challenge*: Text2Room's lack of control over room shape causes issues when blending disparate spaces, as shown in Fig. 2.

*Approach:* We propose the usage of a geometric prior defined as the convex hull of the disparate meshes (Sec. 3.3.4) and a custom ControlNet model for guided scene generation (Sec. 3.4.1).

**Requirement 3: Enabling users to create blended environments without the need for extensive manual configuration.**

*Challenge*: The manual configuration required by Text2Room for trajectory and prompt adjustments is infeasible for SpaceBlender's intended application context of VR telepresence for end-users, as the process of trajectory and prompt definition is time-consuming and requires expertise.

*Approach*: We introduce contextually adaptive prompt inference based on a VLM and an LLM (Sec. 3.3.5) as well as adaptive trajectories (Sec. 3.4.3), enabling cohesive and automated space blending.

**Requirement 4: Supporting core VR usability requirements for end-users including comfortable navigation and viewing.**

*Challenge*: The depth estimator used by Text2Room commonly produces slanted and discontinuous floors and walls, which is problematic for VR navigation and spatial orientation.

*Approach*: To achieve a consistent room structure, SpaceBlender performs semantic segmentation on inpainted images and copies the depth values for wall, floor, and ceiling pixels from the rendered depth image of the geometric prior. These values are also used to inform depth completion for remaining pixels.

## 3.2 System Overview

The SpaceBlender pipeline processes $n$ input images to produce a 3D mesh that integrates the context of each image into a cohesive blended environment. The pipeline has two main stages: the first runs once per generation, while the second is iterative, similar to Text2Room. Below, we give a brief overview of these stages, with detailed descriptions available in the following subsections.

In *Stage 1*, each input image is first preprocessed, after which depth values of each pixel are estimated and backprojected to create a 3D mesh (Sec. 3.3.1). We refer to the resulting $n$ meshes as *submeshes* throughout the remainder of this paper. Next, the submeshes are aligned to a common floor plane with a RANSAC-based method applied to floor vertices identified by a semantic segmentation model (Sec. 3.3.2), optionally including a floor generation step if no floor is visible in the image. The aligned submeshes are then positioned based on a parameter-based layout technique (Sec. 3.3.3) based on which a geometric prior mesh is created to define the shape of the blended space (Sec. 3.3.4). Lastly, text prompts describing the blended regions (i.e., the empty space between submeshes) of the environment are generated with an LLM based on captions inferred by a VLM (Sec. 3.3.5).

In *Stage 2*, the submeshes are blended through a process that involves repeatedly inpainting and integrating 2D rendered views of the mesh. For each iteration, based on the submesh layout defined in *Stage 1*, geometric image priors are rendered to function as a guide for the shape of the space (Sec. 3.4.1). These are combined with the generated text prompts from *Stage 1* to guide the content and appearance of the space (Sec. 3.4.2). Once the blending process completes, an adaptive mesh completion trajectory is followed to fill remaining gaps in the environment (Sec. 3.4.3).

*Implementation.* Like the original Text2Room implementation, we use Stable Diffusion 1.5 [55] for image generation and inpainting and IronDepth [1] for depth estimation and inpainting. Furthermore, SpaceBlender uses the BLIP-2 [40] VLM, GPT-4 [47] LLM, and OneFormer [31] semantic segmentation model. We decoupled the image inpainting process through the usage of a local API A1111 WebUI server API endpoint[3] to provide enough GPU memory for the usage of the various models in our pipeline. The server and pipeline run on separate machines, each equipped with an NVIDIA RTX 4090 GPU in our local setup. It takes about 55-60 minutes to generate a SpaceBlender environment with this configuration.

## 3.3 *Stage 1*: From 2D Images to 3D Layout

The pipeline's first stage establishes the spatial structure of the blended environment. It begins with image preprocessing and depth estimation, converting 2D images into 3D submeshes. These submeshes are aligned to a common floor plane and arranged in a circle, based on which a geometric prior is defined. Finally, prompts for the blended regions are generated by an LLM.

*3.3.1 From 2D Images to 3D Submeshes (**Fig. 3A**).* In this step, the $n$ input images are projected into 3D space. First, a semantic segmentation model is used to detect people in each input image. If a person is detected, that region is removed and inpainted using a prompt inferred by the VLM. The resulting image is then cropped

[3]https://github.com/AUTOMATIC1111/stable-diffusion-webui

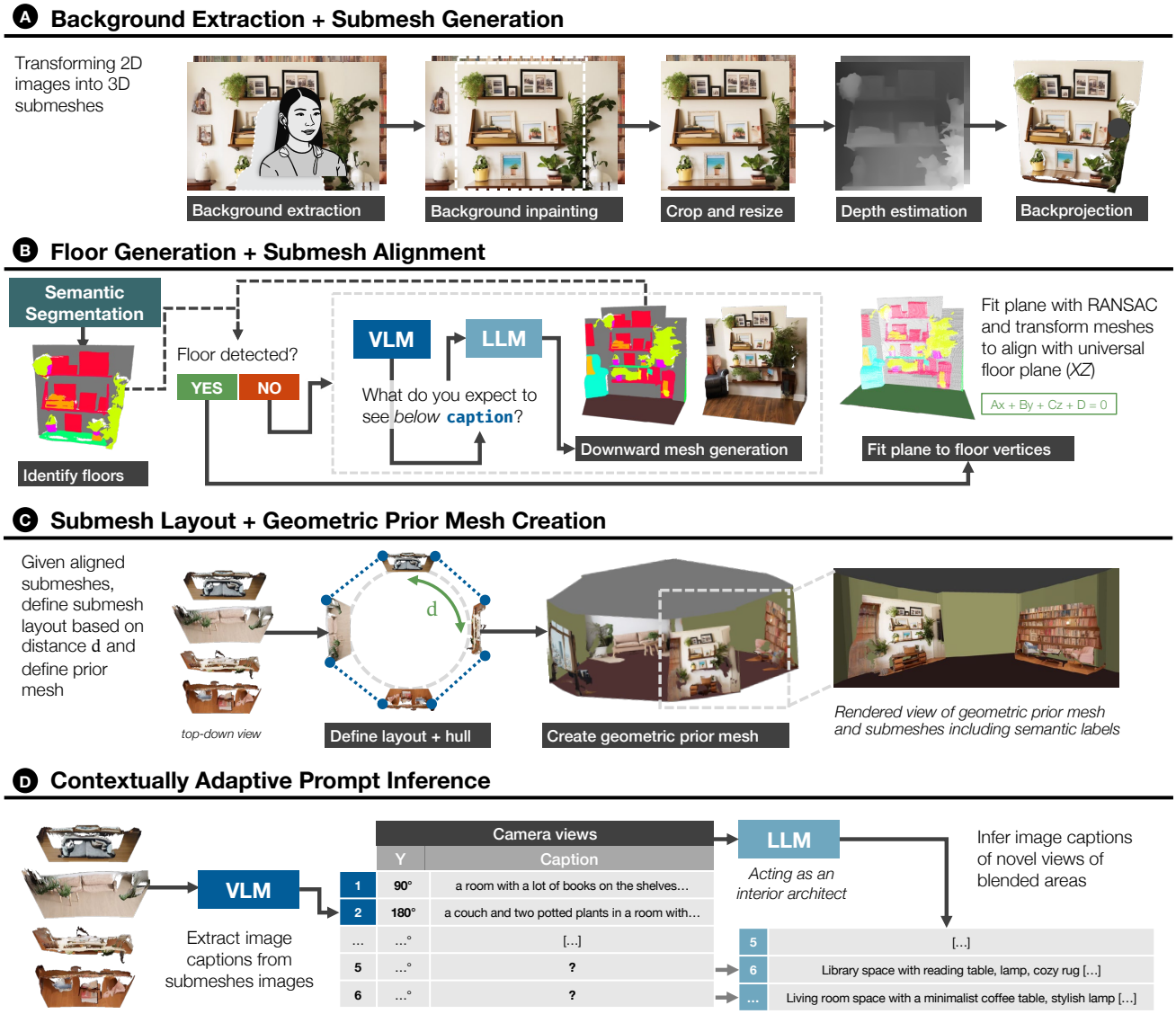


**Figure 3: Overview of *Stage 1* components as described in Sec. 3.3.**

to 512 × 512 pixels to ensure compatibility with subsequent models in the pipeline. Next, using Text2Room's existing functionality, a 3D mesh is created. This involves using the depth estimation model to estimate depth values, aligning them with known depth values, and backprojecting them into 3D space along with the colors of the input image.

*3.3.2 Submesh Alignment (**Fig. 3B**).* We introduce a floor plane alignment technique to reconcile differing perspectives in input images, ensuring the spatial consistency that is needed for scene blending (Fig. 3B). First, labels for floor-like objects (e.g., floor, carpet) within each submesh are obtained using a semantic segmentation map derived from the input image. These semantic labels are then backprojected into 3D space, replacing the submesh's RGB colors with semantic label values. To handle any discrepancies between the depth estimation and semantic segmentation model output, floor vertices more than 0.3 meters above or below the median Y-coordinate of the floor-like vertices are excluded.

Next, RANSAC is used to identify a plane corresponding to these floor-like vertices. To ensure a hypothetical plane is a floor, we use three additional heuristics: (1) the plane's orientation must be within 45° of the target plane normal; (2) the normal vector must have a positive Y-component; and (3) the extent of the inlier points in the X and Z axes should be at least 0.5 meters. After selecting the best floor plane candidate, we rotate the mesh to align the plane's normal with the Y-axis. Next, we translate the floor to $Y = 0$ and set the minimum Z-coordinate to 0. Figure 4 shows an example of submeshes aligned to a common floor plane.


**Figure 4: Comparison between unaligned submeshes and submeshes aligned with our semantic floor alignment technique. The unaligned spaces have floors at different levels and inclines that can be jarring to navigate.**

*Floor Generation (**Fig. 3B**).* If input images lack a floor (e.g., only show a wall), preventing floor plane fitting, we employ a generative technique to extend the submesh downward to create a floor for alignment. This method follows a five-step trajectory that interpolates between the following: gradually looking downward (from -5 to -30 degrees), moving backward (from 1 to 1.5 meters), and moving upward (from 0.3 to 1 meter) relative to the initial camera view. For each step, we use a custom floor description generated by the LLM based on the caption of the image obtained with a VLM. If floor generation fails after ten attempts, the submesh remains unaligned for the rest of the generation process.

*3.3.3 Submesh Layout (**Fig. 3C**).* With submeshes aligned to a universal floor plane, each submesh is transformed to form a layout resembling an open space. This is achieved by positioning the front side (i.e., the side facing the camera used to capture the input image) of each submesh on the perimeter of a circle as seen from a top-down view. The diameter of this circle is determined by a configurable parameter $d$, which controls the distance between the submeshes. Each submesh faces the center of the unified space, ensuring a clear line of sight between all submeshes. This design choice was made in consideration of the importance of mutual awareness in collaborative scenarios [17, 26].

*3.3.4 Geometric Prior Mesh (**Fig. 3C**).* Given the aligned submeshes, a geometric prior mesh is generated to define the shape of the blended space. This involves the definition of a mesh based on the convex hull of the submesh layout, with faces assigned to represent the floor, walls, and ceiling. The height of this mesh is set to the height of the tallest submesh, or 2.5 meters if none is taller. The floor, ceiling, and wall faces are colored based on their respective semantic label colors from the ADE20K dataset [78]. This mesh is used in rendering prior images for the iterative image inpainting process, as described in Sec. 3.4.

As the convex hull effectively forms straight walls between submeshes, the number of submeshes and their shapes directly impact the overall shape of the geometric prior, and conversely, the blended space. For example, four submeshes with straight walls create an octagon-like shape, while four submeshes with straight corners result in a square-like shape. A visual explanation and example of spaces with different numbers of input images and submesh shapes are available in Figs. A.3 and A.4, respectively.

*3.3.5 Contextually Adaptive Prompt Inference (**Fig. 3D**).* In preparation of the iterative mesh generation process in *Stage 2*, text prompts are generated to describe the intended contents of the blended regions. This begins by obtaining image descriptions for each submesh using the VLM, along with a rotation value indicating their relative orientation from a top-down view. This data is then passed to an LLM instructed to act like a highly creative interior architect and photographer skilled at designing spaces with diverse contexts and appearances while avoiding repetitive objects. Based on the rotation values and known image descriptions, the LLM returns a set of new image descriptions paired with rotation values corresponding to the regions to be blended (i.e., the blank regions between submeshes). The full system prompt of this LLM is provided in Sec. A.1.

## 3.4 *Stage 2*: Iterative Blending Guided by Geometric Priors and Contextual Prompts

Utilizing the submesh layout, geometric prior mesh, and text prompts from Stage 1, this stage iteratively blends the disparate submeshes into a unified environment.

*3.4.1 Room Shape Guidance with Geometric Prior Images (**Fig. 5E**).* To enable geometrically coherent space blending, SpaceBlender uses a collection of prior images to guide the iterative text-conditioned image inpainting step of the mesh blending and completion process via ControlNet [76]. Each time the process renders a view of the mesh for inpainting, it also renders a set of prior images from the same camera viewpoint based on the geometric prior mesh aligned with the submesh layout. SpaceBlender is capable of generating three types of prior images which each have a distinct effect on the output of the image inpainting process:

- **Depth Prior:** This prior type can act as a hard constraint for generating spaces with shapes and contents similar to the geometric prior. It is defined by rendering the relative depth for a specified view of the geometric prior mesh.
- **Layout Prior:** This prior type enables constraining the shape of the space without limiting its content (i.e., furniture). It is defined by calculating depth gradients using the Sobel operator to form surface normals based on the depth prior. Subsequently, the magnitude of these normals is calculated and processed with Canny edge detection to produce an image that effectively outlines the geometric prior mesh with white lines outlining the walls, floor, and ceiling on a black background.

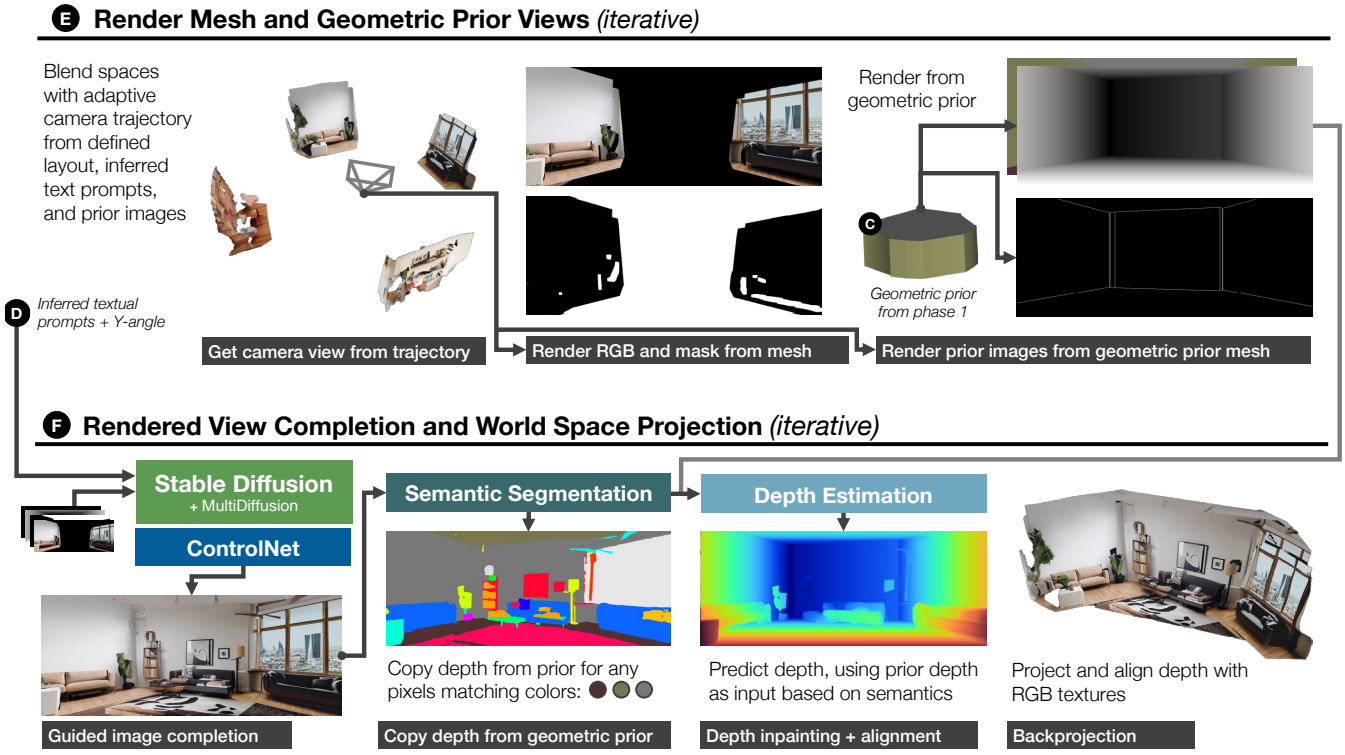


**Figure 5: Overview of *Stage 2* components as described in Sec. 3.4.**

- **Semantic Prior:** This prior type can act as an additional hard constraint to guide the semantic contents of the inpainted views. The current geometric prior mesh definition of SpaceBlender only defines semantic labels for the walls, floor, and ceiling, making it suitable to serve as room layout composition guidance when an empty open space is desired.

These prior images can be stacked and combined using MultiControlNet[4], which allows for adjusting each prior's influence on the image output. For example, using only the layout prior guides the model to generate a space with a specific room structure while allowing the room content (e.g., furniture) to be generated freely. A depth prior can be added to guide the image inpainting model to generate furniture commonly positioned near the walls (e.g., sofas and bookshelves). An example of the depth and layout priors' influence on the output is shown in Fig. 6, demonstrating the varying effects on room contents, with more examples shown in Fig. A.1.

While the depth and semantic prior images are used with pre-trained ControlNet models[5], the layout prior is used with a custom ControlNet model, `ControlNet-Layout`. We describe the training process of this model below.

*Training ControlNet-Layout.* We trained `ControlNet-Layout` on a dataset of 13,182 images. Instead of using an existing dataset, we created our own by using the pre-trained ControlNet segmentation model[5] using semantic maps inferred from SUN-RGBD [60] and LSUN [74, 77], resized to 512×512 pixels. This was repeated several times with unique seeds to enhance image quality and diversify the dataset by generating multiple images per segmentation map. The training process was initialized with the weights of the pre-trained M-LSD model of ControlNet[5] and used a learning rate of $1 \times 10^{-5}$ and a batch size of 4. Training was halted after one epoch due to satisfactory performance on the validation set. Fig. A.2 shows examples of `ControlNet-Layout` output, including a comparison to a similar model from recent parallel work [5].

*3.4.2 Iterative Space Blending (**Fig. 5F**).* This step unifies disparate submeshes iteratively, according to the geometric prior images and prompts defined in Sec. 3.3.5. To enable SpaceBlender 's blending capabilities, we broaden the context window of the image inpainting model by increasing the resolution from 512 × 512 (as used by Text2Room) to 512×1280 while maintaining the rendering camera's field-of-view of 55°. This is enabled by an A1111 WebUI plugin implementation of MultiDiffusion[6] [2].

By increasing the width of the images generated throughout the blending process, we broaden the inpainting model's environmental reference frame and enable it to blend the space between neighboring submeshes in a single inpainting step (see Fig. 6), yielding higher fidelity results compared to step-wise blending, which results in harsh boundaries and artifacts such as shown in Fig. 2. However, due to its circular layout method, when SpaceBlender is given three or fewer input images, the large distances between submeshes still prevent blending in a single step. In these cases, SpaceBlender uses LLM-based prompts to create intermediate submeshes with

[4]https://github.com/Mikubill/sd-webui-controlnet#multi-controlnet
[5]https://huggingface.co/lllyasviel/ControlNet
[6]https://github.com/pkuliyi2015/multidiffusion-upscaler-for-automatic1111

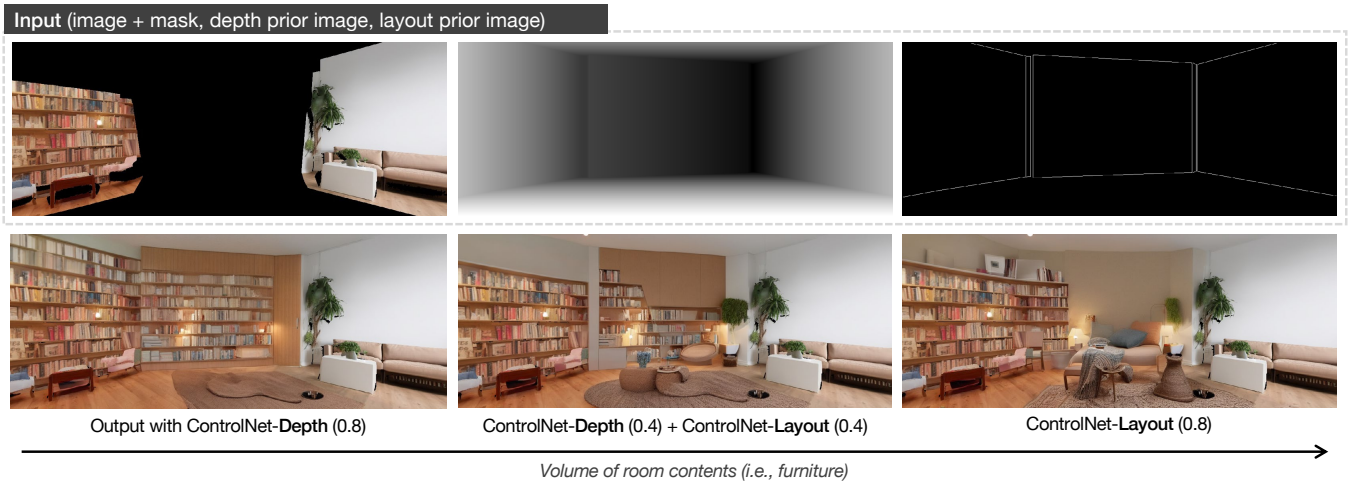


**Figure 6: Comparison of output generated with varying weights of ControlNet depth and layout models, impacting the prior's impact on the output (generated with fixed seed). Top: input images including the input image, depth prior image, and layout prior image rendered from geometric prior. Bottom: results with varying weights are indicated in parentheses.**

an image generation model and integrates these with the layout to bridge the gaps between submeshes and enable blending. The rightmost example in Fig. A.3 demonstrates this approach.

*3.4.3 Mesh Completion Trajectory.* The initial iterative blending process creates a mesh that horizontally integrates the submeshes, defining the blended space from a central perspective. However, at this stage, the floor and ceiling are largely incomplete, and the mesh contains significant gaps that need filling. To address this, an additional set of camera trajectories is employed.

First, interpolation-based trajectories are generated to cover the missing sections of the floor and ceiling. Then, trajectories for each submesh are defined. These paths interpolate the position and rotation of the camera viewpoint, starting centrally within the unified space and initially directed at a specific submesh. The interpolation trajectory ends at the submesh center, facing either the left or right adjacent submesh. Throughout this process, the text prompt passed to the image inpainting model is defined as the text prompt of the blended area that is most closely aligned with the camera's view.

Lastly, an additional trajectory simulates a user looking around the unified space from the center point of their submesh, ensuring the mesh accounts for gaps visible from typical user perspectives. To mimic natural gaze variations, a degree of randomness is introduced into these viewpoints. Once all trajectories are completed, the blended space is ready for use in a VR telepresence system.

## 4 PRELIMINARY USER STUDY

Our preliminary user study explored the effects of *space blending* in the context of a collaborative affinity diagramming task within a VR telepresence environment. The study used a within-subjects design with three conditions that emphasized different visual and geometric qualities of virtual environments. With the selection of conditions below, we sought to explore variations across the dimensions of environmental visual and geometric complexity, fidelity, and familiarity to study their impacts on user behavior and strategies.

- **Generic3D** served as a baseline representing low-poly environments commonly used in current social VR platforms such as Meta Horizon and Recroom. This space was designed with 3D models from the public domain[7].
- **Text2Room** served as a baseline representing an environment generated with a state-of-the-art 3D scene generation method, containing salient landmarks that contrast with the simple landmarks of the Generic3D condition. This environment was produced by the Text2Room framework [30], which we extended to develop SpaceBlender.
- **SpaceBlender** represents our pipeline with participant-provided input images of spaces familiar to them.

The order of conditions was counterbalanced. The Generic3D and Text2Room environments were consistent for all pairs and are shown in Fig. 7. For the Text2Room condition, we used the pipeline's public source code and trajectory files to generate twelve environments and then selected the best one based on subjective visual analysis of their geometric and visual quality. For the SpaceBlender condition, a new environment was generated for each pair to embed a familiar physical context for each participant, which are shown in Fig. 9. This involved collecting images via an online form sent to participants before the study, where they could upload a photo of a familiar space (e.g., a library, cafe, living room, or desk). Participants who did not upload a photo could choose from seven photographs of various spaces within our institution. Those who found none of the spaces familiar were excluded from the study. Uploaded images were cropped to a 1:1 aspect ratio, excluding any personally identifiable content (e.g., portraits). The study was approved by the University College London Research Ethics Committee (Study ID UCL/CSREC/R/16).

[7] https://kenney.nl/assets/furniture-kit


**Figure 7: Overview of the environments used for the Generic3D and Text2Room conditions.**

## 4.1 Task

The task involved clustering virtual sticky notes with predefined text. Initially, each participant independently clustered twelve sticky notes from one of two datasets (*fruits* or *vegetables*) by color (e.g., placing "banana" near "lemon"). After two minutes of individual work elapsed, participants were given three more minutes to collaboratively reorganize their clusters into new groups.

## 4.2 Recruitment

We recruited 20 participants (10 pairs) through internal mailing lists. Participants (7 Female, 13 Male) had an average age of 26 years (SD = 6.9) and were mainly students and professionals; seven had backgrounds in computer graphics, and four in UX/HCI. All participants had used a VR headset at least once, with eight using them monthly, and three using VR telepresence platforms monthly. Five pairs knew each other before the study. Participants received £15 gift cards as compensation.

## 4.3 Implementation and Setup

The VR telepresence system for our study was built with Unity3D and the Ubiq framework [22], providing voice chat, networked objects, and low-poly floating-body avatars. Each participant used a Meta Quest 3 VR headset tethered to a desktop computer with an NVIDIA 4090 RTX GPU in separate physical spaces. Navigation was achieved using the controller's joystick.

Two stacks of sticky notes were placed in the virtual environments. Participants could grab the top note by pressing and holding the grip button when their virtual hand was near the stack. Given the high complexity of the meshes generated by generative models, sticky notes could be placed anywhere without physics constraints. The stacks were manually placed (e.g., on a table) within arm's reach of the participant spawn points before the study.

## 4.4 Procedure

Upon arrival, the experimenter introduced the study's structure and goals to the participants, including an explanation of the usage of the VR equipment. Participants then read and signed an informed consent form and completed a pre-study questionnaire covering demographics and VR experience. Participants were guided to randomly assigned spaces and equipped with VR headsets. The experimenter then joined the environment from a desktop computer in a separate space, explained the task, and guided participants through a test environment to acclimatize them to the navigation controls and manipulation techniques of the virtual sticky notes.

Participants were then teleported to opposite sides of an environment matching the current condition. In the SpaceBlender condition, teleportation points were defined to be at the center of the submesh generated based on the image provided by the respective participant. Participants started by individually clustering the sticky notes. After two minutes, the experimenter instructed them to continue the task collaboratively, combining their individual work for three more minutes (Fig. 8). After each condition, participants completed a post-task questionnaire.

After the final condition, participants completed a post-study questionnaire. The study concluded with the experimenter guiding participants to a common physical space for a semi-structured interview, including a brief scenario walkthrough. This walkthrough featured two scenarios: a collaborative study session and a cooking class with friends, as shown in Fig. A.5. The interview questions are shown in A.6.

## 4.5 Data Collection & Measures

The post-task questionnaire included questions to measure participants' perceived spatial presence from an existing questionnaire [28], specifically focusing on two dimensions: Self-Location, which assesses the sensation of being physically present within the virtual environment, and Possible Actions, which measures the perceived capability for interaction within that space. We also incorporated questions to measure perceived Copresence to evaluate participants' perceptions of sharing the virtual environment with others [27]. For each question, "Do not agree at all" was treated as 1, and "Fully Agree" as 5.

Additionally, we included four custom questions to assess how various factors influenced task execution: (1) *layout*, (2) *visual quality*, (3) *level of familiarity*, and (4) *navigation controls*. Each question asked, "To what extent did [$X$] help or hinder you in conducting the task?" with $X$ representing one of the factors. Participants responded using a 5-point Likert scale: *significantly hindered* (1), *slightly hindered*, *neither helped nor hindered*, *slightly helped*, and *significantly helped* (5). We averaged the scores of each measure to arrive at a single score for each.

# 5 RESULTS

In this section, we present *(1)* our quantitative analysis of the participants' self-reported measures; *(2)* qualitative themes around the benefits and limitations of the three environments for the clustering task; and *(3)* participants' suggestions for future requirements and potential use cases of SpaceBlender's environments.


**Figure 8: Captures of participants manipulating sticky notes while represented by Ubiq avatars in the individual (left) and collaborative (right) task phases.**

## 5.1 Self-Reported Questionnaire Results

To analyze the self-reported questionnaire data, we first applied the Friedman test to identify overall differences across conditions. Following this, we conducted Wilcoxon signed-rank tests for post-hoc comparisons. Key findings are presented below, with additional results provided in Sec. A.7. Fig. 10 shows distributions of participants' scores for Possible Actions, Self-Location, Copresence, and task impact factors. Furthermore, Fig. 11 shows participants' ranked preferences for using the Generic3D, Text2Room, and SpaceBlender environments. Most participants ranked the Generic3D environment as their first choice for completing the clustering task, followed closely by SpaceBlender, which received only first or second choice ratings.

*Possible Actions.* A statistically significant difference in Possible Actions scores was found between Generic3D (M = 3.89, SD = 0.71) and Text2Room (M = 3.13, SD = 0.95) ($W = 5.0$, $p = 0.0021$), indicating a diminished perception of possible actions within the environment under the Text2Room condition.

*Self-Location.* A statistically significant difference in Self-Location scores was found between Text2Room (M = 3.46, SD = 0.78) and SpaceBlender (M = 3.91, SD = 0.68) ($W = 0.0$, $p = 0.0039$), indicating a decreased sense of being physically present within the environment under the Text2Room condition.

*Impact of Environmental Factors.* Among the impact of environmental factors, we found statistically significant differences for the impact of *Layout*, *Visual Quality*, and *Familiarity*. For *Layout* we found statistically significant differences between Generic3D (M = 3.92, SD = 0.68) and Text2Room (M = 3.20, SD = 0.75) ($W = 10.0$, $p = 0.0043$); and between Text2Room and SpaceBlender (M = 3.85, SD = 0.70) ($W = 4.0$, $p = 0.0036$). We also found statistically significant differences for *Visual Quality* between conditions Generic3D (M = 4.00, SD = 0.00) and Text2Room (M = 3.17, SD = 0.75) ($W = 0.0$, $p = 0.0010$); and between Text2Room and SpaceBlender (M = 3.90, SD = 0.30) ($W = 3.5$, $p = 0.0049$). Lastly, we found statistically significant differences for *Familiarity* between Text2Room (M = 3.10, SD = 0.83) and SpaceBlender (M = 3.95, SD = 0.22) ($W = 0.0$, $p = 0.0039$).

## 5.2 Benefits and Limitations of Environments for the Clustering Task

Next, we discuss four themes from participants' post-task reflections on the clustering task and to what extent the three styles of virtual environments supported their work. We refer to paired participants as P#A and P#B, where # represents their pair ID.

*5.2.1* ***Environments typically played a passive role in supporting spatial organization, but some participants adapted their clustering strategies to SpaceBlender's environments' distinct or familiar features.*** When asked what strategies they adopted to complete the clustering task across all three conditions, a majority of participants described using the center of the environment as a staging area for finalized clusters. Preferences varied for storing and comparing ungrouped notes either in the middle or in individually assigned regions.

Although the task did not require explicit use of the environment, some participants in SpaceBlender utilized its unique features. First, P5A and P6A expressed that SpaceBlender's more detailed environments with distinct segments helped to establish mental models of where to organize notes: "I just put the green objects in the green area of the environment" (P6A). This strategy contrasts with P6A's experience in Generic3D, where a lack of environmental cues led to a more deliberate strategy of labeling areas of the environment to place specific clusters: "We had to actually allocate areas because the wall was just gray." Additionally, several participants found value in the familiar details of SpaceBlender's environments (i.e., spatial landmarks preserved from their input images) to inform their clustering strategies (P4A, P7A, P10A). P7A noted: "Familiarity helped because this is pretty much like where I work a lot"; "it just feels a bit more like comfortable, like, thinking in that area." This sentiment was echoed by P10A, who envisioned aligning sticky notes to their own table in the virtual workspace.

*5.2.2* ***Participants had mixed preferences for minimalistic and realistic environments for supporting their focus on the clustering task.*** A majority of participants favored the simplistic design of Generic3D as they believed it enabled them to be more engaged with the task. P8A felt that as the "the cleanest environment," the Generic3D environment supported task efficiency. Although P3A perceived the Generic3D environment to be "cartoonish" and

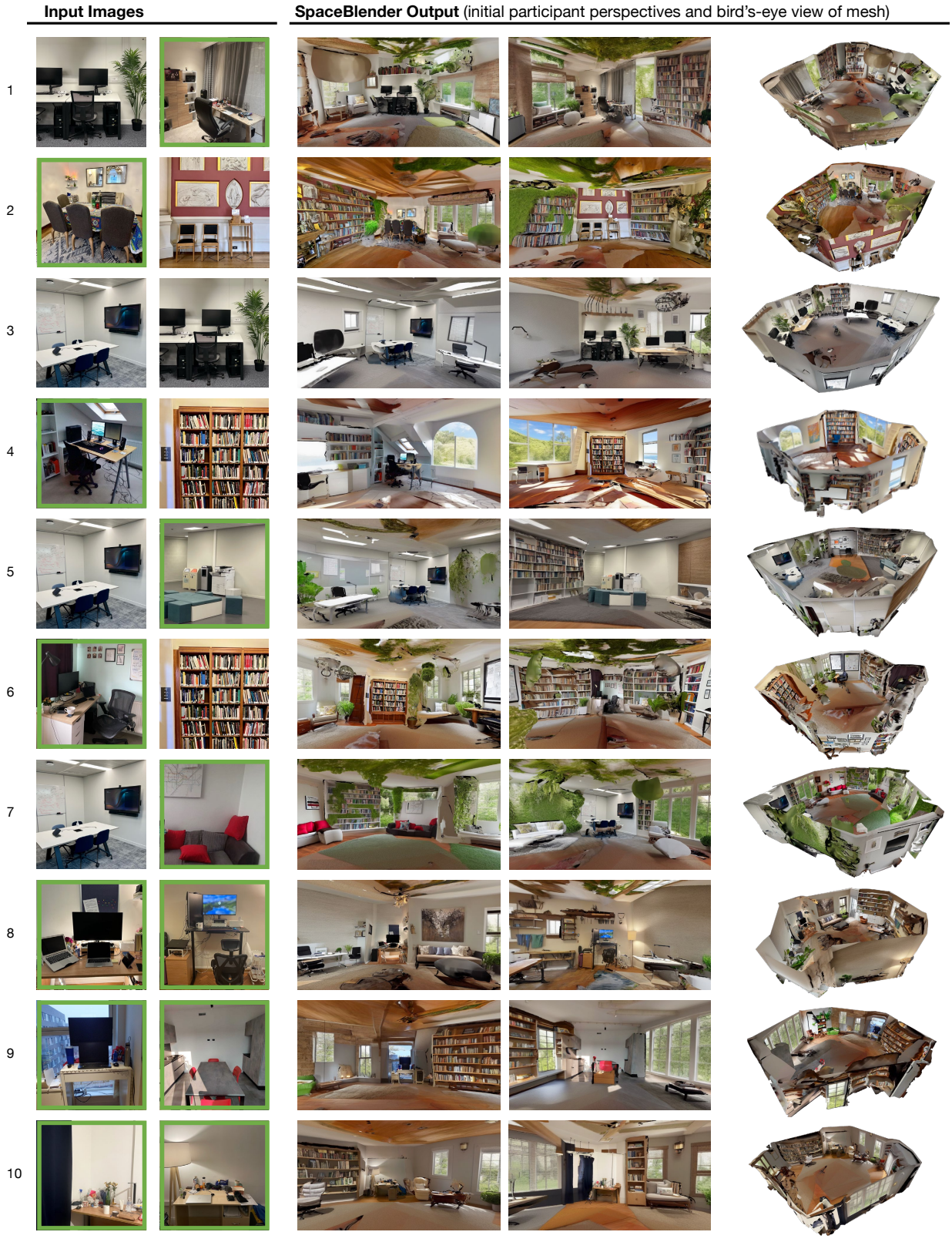


**Figure 9: Overview of SpaceBlender meshes generated based on input images uploaded (with green outline) or selected by participants (no outline).**

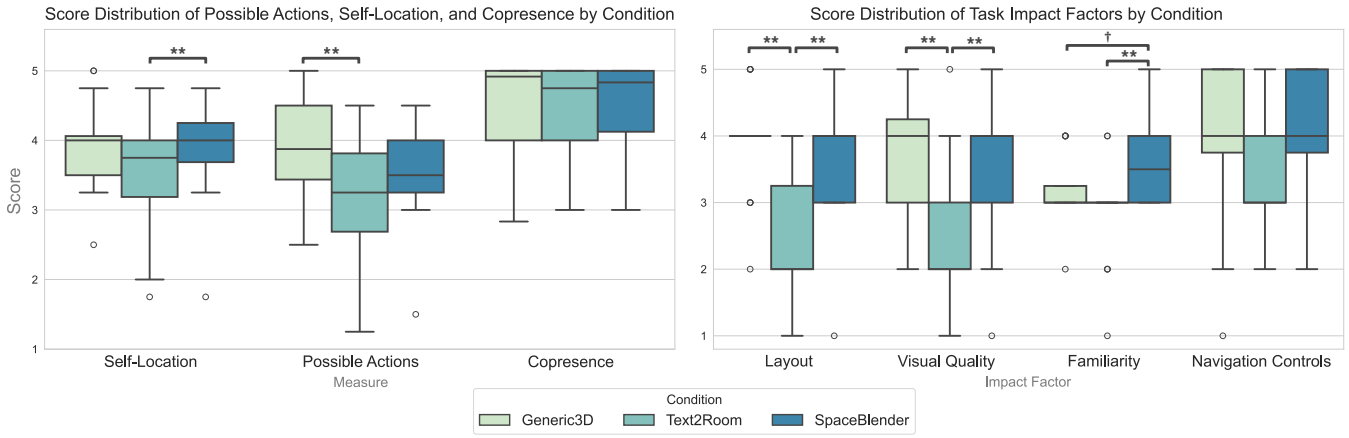


**Figure 10: Plot of score distributions of Possible Actions, Self-Location, Copresence, and task impact factors. Levels of statistical significance: $^{\dagger}$ for $p < 0.05$ (before Bonferroni correction), $^{*}$ for $p < 0.05$, $^{**}$ for $p < 0.01$, and $^{***}$ for $p < 0.001$ (after Bonferroni correction).**

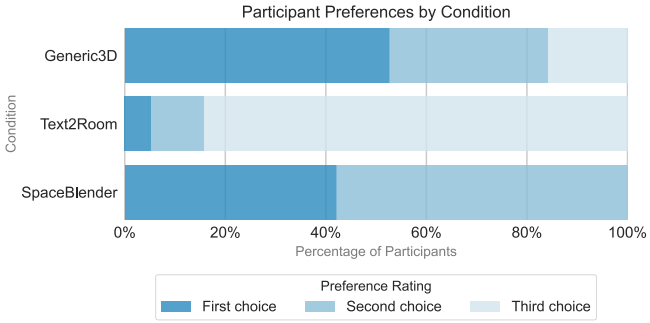


**Figure 11: Plot of participant preferences per condition.**

"not real," they considered it the "highest quality" environment. P9A identified benefits to having both low-fidelity environments and virtual avatars:"I know my figure in the space was artificial, so I'm more aware I'm in the game."

However, not all participants preferred the minimalistic environments: "[Generic3D] is my least favorite, because it didn't feel realistic" (P4B). Some participants found value in working in the more detailed Text2Room and SpaceBlender environments: "because it feels more realistic, it feels more immersive," despite the potential for these details to detract from their task engagement.

*5.2.3* ***Participants perceived increased physical comfort and navigability in the SpaceBlender and Generic3D conditions, as opposed to the Text2Room condition.*** All participants noted Text2Room's inconsistent floor geometry as a major source of navigational difficulty and discomfort, with P8A noting that "stuff sticking out of the floor" significantly hindered their ability to navigate in the space. Similarly, P3B pointed to "noise and also the distortion in the floor" in Text2Room, but noted that this issue was not present in SpaceBlender.

However, a majority of participants highlighted that the both generative environments exhibited low texture resolution, noisy artifacts, and incoherent geometry, which could detract from the realism and usability of the spaces. Both the Text2Room and SpaceBlender environments were criticized for their geometric inaccuracies that led to physical objects like "the table and a couple of the chair legs" being mapped onto the floor (P5B).

At times, the physical discomfort reported by participants extended beyond visual annoyance. For instance, P10A stated that they "did not want to move much in [Text2Room]," attributing their simulator sickness to the environment's poor construction and excessive clutter. The impact of these challenges was so pronounced that some participants stated having to close their eyes while navigating in Text2Room spaces (P5B, P10A, P11A, P11B).

In contrast, the Generic3D environments' visual and geometric consistency were universally praised for being "navigable" and "clean". Participants found the simplistic and stylized nature of the Generic3D environment not only visually appealing but also conducive to task performance. P6A appreciated the environment being "more spacious and a lot more easy to navigate." Similarly, P8A described their experience in the Generic3D environment as "clean and easy to perform the task in."

## 5.3 Future Requirements and Potential Use Cases for Blended Spaces

Finally, we discuss participants' comments regarding future scenarios where SpaceBlender environments could benefit VR telepresence and suggestions for improvement of the blended environments.

*5.3.1* ***Participants envisioned a variety of future scenarios where blending familiar context into virtual environments could explicitly or implicitly provide value.*** After walking through additional SpaceBlender environments, participants brainstormed how SpaceBlender may be applied within VR telepresence environments in the future. For professional use cases, participants were interested in meeting spaces incorporating inspiring physical locations (e.g., "areas where you can have a coffee... and discussions") to be "more conducive for ideation" (P6B) and learning

environments that use different parts of the blended space to structure educational activities (P3A, P5A, P6B). They also saw value in supporting social interactions (e.g., gaming with friends, family gatherings) and personal well-being (e.g., providing familiar environments for therapy and recording personal memories) (P3A, P3B, P4A P8A). Some participants expressed concerns regarding the exposure of personal spaces (P3A, P3B, P8A, P7A). For instance, P3A expressed that depending on the scenario, they may feel uncomfortable sharing "a part of [their] life".

Reflecting on these scenarios, participants distinguished between SpaceBlender environments providing *explicit* and *implicit* value for collaboration in different contexts. For example, virtual training applications may need realistic depictions of users' surroundings, while social use cases could benefit indirectly from familiar details that "invoke a sense of being in a cozier space" (P8A). This distinction aligns with participants' feedback on the clustering task: some felt the SpaceBlender environment did not explicitly support their clustering strategies, while others described implicitly using environmental features (e.g., colors, familiar furniture) to establish organization patterns.

*5.3.2* ***Enabling future use cases for blended environments in VR telepresence requires quality improvements and real-world alignment.*** While participants could see the benefits of using SpaceBlender environments in VR telepresence systems, participants universally emphasized the need to improve visual quality and realism to fully realize these benefits. First, we observed instances where environments deviating from participants' memories of familiar locations caused confusion or disorientation. P9A expressed concerns about the accuracy of input images reconstructed in the blended space: "it made me feel a bit weird that I could not recognize some things on my desk." P1A mentioned needing to adjust expectations due to partial reconstructions of familiar environments: "it takes a lot of getting used to if there's... somewhere you're familiar with, but then you turn around and it's not what you're familiar with." Some participants also noted that the shape of the blended spaces did not match their expectations (P1B, P1A): "[…] closed spaces usually don't look like that" (P1B), while P4A suggested a specific use case for the circular shape of the blended space, imagining "[...] students in a circle, and the teacher standing in the middle."

To improve realism, some participants wanted the blended virtual environment to "connect with [their] real environment" by matching the geometry of their physical surroundings (P9B). With this approach, participants envisioned enabling mixed reality collaboration for physical tasks (P1B, P3A, P10A, P10B, P11B), such as in our cooking class scenario (Fig. A.5).

# 6 DISCUSSION AND FUTURE WORK

We structure our discussion around *(1)* avenues for further studies to explore the impact of blended virtual environments on VR collaborative tasks; *(2)* themes for potential improvement and extension of SpaceBlender; *(3)* the limitations of our work.


**Figure 12: Examples of SpaceBlender pipeline components errors: (A) protrusion of outdoor structure, (B) distorted table, (C) depth estimation error, (D) depth alignment error, (C) depth estimation error, (F) floor expansion error.**

## 6.1 Opportunities for Supporting VR Telepresence with Blended Environments

*6.1.1* ***Leveraging familiar context in VR telepresence scenarios.*** Several participants noted that familiar elements in the SpaceBlender environment supported their clustering strategies by providing contextual cues similar to their real-world experiences. Similarly, questionnaire responses indicated that familiarity had a greater impact on the clustering task in the SpaceBlender condition compared to the other two conditions. Although our study was preliminary and requires further empirical research, these early insights suggest that familiarity is a promising direction for future research on collaborative spaces.

Participant feedback indicated that the use and desired functionality of blended spaces may vary between professional and social contexts. In professional settings, blended spaces might be used to establish workspace context and create environments that stimulate creativity. With continued usage, we envision that familiar spaces could be tools for preparation and augmenting memory for collaborative tasks in blended spaces. Furthermore, after collaboration, the uniquely blended regions of the SpaceBlender spaces could provide a lasting artifact of the collaborative work [54]. In social contexts, familiarity may play a more implicit role, recreating comfortable and meaningful environments. Future research could explore how blending familiar personal spaces affects social presence and personal identity.

Despite the potential of blending, some participants expressed discomfort with merging familiar and novel spaces, highlighting the need for careful consideration of privacy and personal boundaries. Future research could investigate mechanisms that allow users to control the degree of blending and ensure only intended elements of their familiar spaces are shared.

*6.1.2* ***Extending SpaceBlender environments to support explicit interactions for collaborative scenarios.*** The SpaceBlender environments primarily implicitly supported the clustering task in our user study by grounding users in familiar spaces. We envision several ways to extend SpaceBlender to explicitly support collaboration. One such extension could enable users to manipulate virtual objects within the scene by semantically segmenting scene components and applying matching functionalities, such as *drawing* on a *whiteboard* [14, 23]. In line with this, while generating 3D scenes is currently time-consuming, we foresee future models allowing for real-time user-driven changes to customize spaces, similar to those available in 2D generative systems, with future studies possibly exploring customization preferences [54].

Future work may also explore extending the layout, geometric prior, and trajectory definition techniques to support more ecologically valid submesh arrangements, including multi-room or multi-story structures, enabling features like breakout rooms and meeting context transitions [24]. To further incorporate spatial familiarity, these layouts could be modeled after existing building floor plans, with submeshes aligned to these layouts instead of the current parameter-based layout approach (Sec. 3.3.3). Future work may study how these spatially familiar layouts impact user navigation, collaboration efficiency, and spatial awareness in virtual environments.

## 6.2 Opportunities to Improve and Extend SpaceBlender

In this section, we discuss potential ways to improve and extend SpaceBlender based on feedback shared by our study participants.

*6.2.1* ***Improving the quality and physical comfort of SpaceBlender environments.*** Participants highlighted the need for enhanced visuals and geometry to support their envisioned future uses of SpaceBlender. A prevalent issue associated with this feedback was inaccurate depth estimation and alignment, leading to objects merging with walls, floors, or ceilings, or displaying implausible depth values, diminishing the realism and coherence of the scene (Fig. 12A–E). Due to the iterative nature of the pipeline and the dual use of the mesh representation for rendering and completion, these inaccuracies tended to amplify during generation. One way to address this may be to replace IronDepth with a more recent monocular depth estimation model [4, 73]. However, the replacement model should also support depth image *inpainting*, a task that is less commonly supported. Additionally, semantic priors could be extended to objects to improve geometric consistency, as used by the recent framework ControlRoom3D [58]. While this framework requires the manual definition of these priors, it could be combined with our geometric prior definition technique and existing furniture layout synthesis methods for an automated approach [20, 49]. Alternatively, SDF-based methods may be used to define the mesh directly, which could be extended to make the generation conditional on the geometry of input submeshes to support blending.

Furthermore, several participants noted the low resolution of SpaceBlender environments. While the current MultiDiffusion-based inpainting process would support higher resolution images if paired with depth estimation and inpainting model capable of processing these, adopting more recent image generation models that produce higher-resolution images could further improve visual quality [56]. Some recent models [41, 50] also offer improved controllability, which could help prevent failure modes such as the one in Fig. 12F, where our floor generation method produced a kitchen island instead of a floor, even though the prompt specified otherwise. Lastly, recent video generation models [16, 72] might replace iterative image inpainting to enhance visual quality through multi-view consistency without relying on a mesh as an intermediate representation.

*6.2.2* ***Aligning SpaceBlender environments with the real world.*** Our participants commonly expressed a desire to align the blended space with their physical environment (Sec. 5.3.2). This may be achieved by extending SpaceBlender to accept meshes or point clouds as input pre-captured by users or, alternatively, captured by RGB-D cameras mounted in the user's local space in real-time [43]. These representations could be registered to the user's local physical space, who could then use a mixed-reality headset to observe the blended space.

## 6.3 Limitations

Our user study included both VR novices and periodic VR users, whose perceptions may not generalize to experienced VR users. However, we note that our novice VR users helped surface the most critical challenges with SpaceBlender's environments, particularly those involving core VR interaction requirements (e.g., physical comfort and navigability) that pertain to experienced users as well.

Furthermore, some elements of our study design limited our ability to isolate the impacts of familiar context and fidelity on participants' task performance and collaboration patterns. First, we did not control the familiarity of participants' input images to generate SpaceBlender environments, as some users used preselected images instead of uploading their own. Second, our choice of baseline conditions may limit the generalizability of some of our findings. A number of participants preferred the higher realism of the generative environments over the Generic3D condition. However, since the baseline and generative conditions had vastly different types of texture quality, this participant preference for the generative conditions might be different if compared to a higher-fidelity version of the Generic3D baseline. Future work could compare SpaceBlender with other environment types, including such higher-fidelity virtual environments[8], 3D scans, or manually designed environments. Third, the clustering task did not require participants to interact with the virtual environment explicitly, allowing them to choose whether to use environmental features for the spatial organization of sticky notes. We chose affinity diagramming as a fair task across all three conditions because the Generic3D and Text2Room environments would be insufficient for tasks requiring access to users' physical surroundings. Future work may explore studying blended spaces combined with VR tasks that require explicit environmental interaction. Lastly, considering the novelty of generative 3D environments incorporating familiar environments, participant responses may have been subject to response bias [15].

[8]E.g., *Spatial*: https://spatial.io, *Microsoft Mesh*: https://microsoft.com/mesh

Additionally, we acknowledge that our sample size was insufficient to reliably calculate order effects. Participants who began with the Generic3D might have been primed to organize notes in mid-air rather than aligning them with the environment in subsequent conditions due to the environment's lack of distinctive visual features.

## 7 CONCLUSION

To enable the creation of context-rich virtual spaces for VR telepresence, our work contributes SpaceBlender, a pipeline that leverages generative AI to incorporate and extend users' physical surroundings into blended virtual environments. SpaceBlender makes key improvements to current state-of-the-art generative models by projecting multiple user-provided images into 3D segments, aligning mesh segments to a uniform floor level, and blending those segments via diffusion-based space completion methods guided by geometric priors and dynamic text prompts. Through a preliminary within-subjects study with 20 participants, we explored how varying the virtual environment (using Generic3D, Text2Room, and SpaceBlender environments) affects their behavior and strategies when completing a collaborative clustering task. Overall, participants experienced increased physical comfort and navigability in the Generic3D and SpaceBlender compared to Text2Room due to greater consistency in the room geometry. Furthermore, some leveraged recognizable environmental features in the SpaceBlender space to complete the task. Additionally, participants envisioned a rich set of professional, social, and personal use cases where embedding familiar contextual details into virtual environments could provide value for collaboration. However, to fully realize the potential benefits, they desired further aligning SpaceBlender environments to real-world spaces and enhancing their visual and geometric quality.

Given the current gap in the HCI community's understanding of deploying generative AI environments in interactive systems, our studies around SpaceBlender lay the groundwork for future generative AI-based systems for VR environment creation. We note promising avenues for future work to extend and deploy our pipeline in VR telepresence systems to further study the impact of blended environments on collaborative processes.

## ACKNOWLEDGMENTS

We thank Anthony Steed for providing valuable feedback and suggestions on our study design. We also extend our gratitude to the reviewers for their insightful comments and our study participants for their time. The preliminary user study of this research was partially supported by the European Union's Horizon 2020 Research and Innovation program under grant agreement No. 739578.

## REFERENCES

[1] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. 2022. Irondepth: Iterative refinement of single-view depth using surface normal and its uncertainty. In *British Machine Vision Conference (BMVC)*.

[2] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. 2023. MultiDiffusion: fusing diffusion paths for controlled image generation. In *Proceedings of the 40th International Conference on Machine Learning (ICML'23, Vol. 202)*. JMLR.org, Honolulu, Hawaii, USA, 1737–1752.

[3] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, Afshin Dehghan, and Josh Susskind. 2022. GAUDI: a neural architect for immersive 3D scene generation. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NIPS '22)*. Curran Associates Inc., Red Hook, NY, USA, 25102–25116.

[4] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. 2023. ZoeDepth: Zero-shot Transfer by Combining Relative and Metric Depth. https://doi.org/10.48550/arXiv.2302.12288 arXiv:2302.12288 [cs].

[5] Shariq Farooq Bhat, Niloy Mitra, and Peter Wonka. 2024. LOOSECONTROL: Lifting ControlNet for Generalized Depth Conditioning. In *ACM SIGGRAPH 2024 Conference Papers (SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3641519.3657525

[6] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. 2022. Text2Light: Zero-Shot Text-Driven HDR Panorama Generation. *ACM Trans. Graph.* 41, 6 (Nov. 2022), 195:1–195:16. https://doi.org/10.1145/3550454.3555447

[7] Lung-Pan Cheng, Eyal Ofek, Christian Holz, and Andrew D. Wilson. 2019. VRoamer: Generating On-The-Fly VR Experiences While Walking inside Large, Unknown Real-World Building Environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, Osaka, Japan, 359–366. https://doi.org/10.1109/VR.2019.8798074

[8] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. 2023. LucidDreamer: Domain-free Generation of 3D Gaussian Splatting Scenes. https://doi.org/10.48550/arXiv.2311.13384 arXiv:2311.13384 [cs].

[9] Ben J. Congdon, Tuanfeng Wang, and Anthony Steed. 2018. Merging environments for shared spaces in mixed reality. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology (VRST '18)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10/gj6qrn

[10] Bob Coyne and Richard Sproat. 2001. WordsEye: an automatic text-to-scene conversion system. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01)*. Association for Computing Machinery, New York, NY, USA, 487–496. https://doi.org/10.1145/383259.383316

[11] Tim Dalgleish, Lauren Navrady, Elinor Bird, Emma Hill, Barnaby D Dunn, and Ann-Marie Golden. 2013. Method-of-loci as a mnemonic device to facilitate access to self-affirming personal memories for individuals with depression. *Clinical Psychological Science* 1, 2 (2013), 156–162.

[12] Hai Dang, Frederik Brudy, George Fitzmaurice, and Fraser Anderson. 2023. WorldSmith: Iterative and Expressive Prompting for World Building with a Generative AI. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, 1–17. https://doi.org/10.1145/3586183.3606772

[13] Sauvik Das, David Lu, Taehoon Lee, Joanne Lo, and Jason I Hong. 2019. The memory palace: Exploring visual-spatial paths for strong, memorable, infrequent authentication. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 1109–1121.

[14] Fernanda De La Torre, Cathy Mengying Fang, Han Huang, Andrzej Banburski-Fahey, Judith Amores Fernandez, and Jaron Lanier. 2024. LLMR: Real-time Prompting of Interactive Worlds using Large Language Models. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*. Association for Computing Machinery, New York, NY, USA, 1–22. https://doi.org/10.1145/3613904.3642579

[15] Nicola Dell, Vidya Vaidyanathan, Indrani Medhi, Edward Cutrell, and William Thies. 2012. "Yours is better!": participant response bias in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 1321–1330. https://doi.org/10.1145/2207676.2208589

[16] Boyang Deng, Richard Tucker, Zhengqi Li, Leonidas Guibas, Noah Snavely, and Gordon Wetzstein. 2024. Streetscapes: Large-scale Consistent Street View Generation Using Autoregressive Video Diffusion. In *ACM SIGGRAPH 2024 Conference Papers (SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3641519.3657513

[17] Paul Dourish and Victoria Bellotti. 1992. Awareness and coordination in shared workspaces. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work (CSCW '92)*. Association for Computing Machinery, New York, NY, USA, 107–114. https://doi.org/10.1145/143457.143468

[18] Joey Ka-Yee Essoe, Nicco Reggente, Ai Aileen Ohno, Younji Hera Baek, John Dell'Italia, and Jesse Rissman. 2022. Enhancing learning and retention with distinctive virtual reality environments and mental context reinstatement. *npj Science of Learning* 7, 1 (Dec. 2022), 1–14. https://doi.org/10.1038/s41539-022-00147-6 Publisher: Nature Publishing Group.

[19] Chuan Fang, Xiaotao Hu, Kunming Luo, and Ping Tan. 2023. Ctrl-Room: Controllable Text-to-3D Room Meshes Generation with Layout Constraints. https://doi.org/10.48550/arXiv.2310.03602 arXiv:2310.03602 [cs].

[20] Weixi Feng, Wanrong Zhu, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Xuehai He, S. Basu, Xin Eric Wang, and William Yang Wang. 2023. LayoutGPT: Compositional Visual Planning and Generation with Large Language Models. *Advances in Neural Information Processing Systems* 36 (Dec. 2023), 18225–18250. https://proceedings.neurips.cc/paper_files/paper/2023/hash/3a7f9e485845dac27423375c934cb4db-Abstract-Conference.html

[21] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. 2023. SceneScape: Text-Driven Consistent Scene Generation. *Advances in Neural Information*

*Processing Systems* 36 (Dec. 2023), 39897–39914. https://proceedings.neurips.cc/paper_files/paper/2023/hash/7d62a85ebfed2f680eb5544beae93191-Abstract-Conference.html

[22] Sebastian J Friston, Ben J Congdon, David Swapp, Lisa Izzouzi, Klara Brandstätter, Daniel Archer, Otto Olkkonen, Felix Johannes Thiel, and Anthony Steed. 2021. Ubiq: A System to Build Flexible Social Virtual Reality Experiences. In *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology (VRST '21)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3489849.3489871

[23] Daniele Giunchi, Nels Numan, Elia Gatti, and Anthony Steed. 2024. DreamCodeVR: Towards Democratizing Behavior Design in Virtual Reality with Speech-Driven Programming. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. 579–589. https://doi.org/10.1109/VR58804.2024.00078 ISSN: 2642-5254.

[24] Carlos Gonzalez Diaz, John Tang, Advait Sarkar, and Sean Rintel. 2022. Making Space for Social Time: Supporting Conversational Transitions Before, During, and After Video Meetings. In *2022 Symposium on Human-Computer Interaction for Work*. ACM, Durham NH USA, 1–11. https://doi.org/10.1145/3533406.3533417

[25] Jens Emil Sloth Grønbæk, Ken Pfeuffer, Eduardo Velloso, Morten Astrup, Melanie Isabel Sønderkær Pedersen, Martin Kjær, Germán Leiva, and Hans Gellersen. 2023. Partially Blended Realities: Aligning Dissimilar Spaces for Distributed Mixed Reality Meetings. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–16. https://doi.org/10.1145/3544548.3581515

[26] Carl Gutwin and Saul Greenberg. 1996. Workspace awareness for groupware. In *Conference Companion on Human Factors in Computing Systems (CHI '96)*. Association for Computing Machinery, New York, NY, USA, 208–209. https://doi.org/10.1145/257089.257284

[27] Chad Harms and Frank Biocca. 2004. Internal consistency and reliability of the networked minds measure of social presence. In *Seventh Annual International Workshop: Presence*, Vol. 2004. Universidad Politecnica de Valencia Valencia, Spain.

[28] Tilo Hartmann, Werner Wirth, Holger Schramm, Christoph Klimmt, Peter Vorderer, André Gysbers, Saskia Böcking, Niklas Ravaja, Jari Laarni, Timo Saari, Feliz Gouveia, and Ana Maria Sacau. 2016. The Spatial Presence Experience Scale (SPES): A short self-report measure for diverse media settings. *Journal of Media Psychology: Theories, Methods, and Applications* 28, 1 (2016), 1–15. https://doi.org/10.1027/1864-1105/a000137 Place: Germany Publisher: Hogrefe Publishing.

[29] Jaylin Herskovitz, Yi Fei Cheng, Anhong Guo, Alanson P. Sample, and Michael Nebeling. 2022. XSpace: An Augmented Reality Toolkit for Enabling Spatially-Aware Distributed Collaboration. *Proceedings of the ACM on Human-Computer Interaction* 6, ISS (Nov. 2022), 277–302. https://doi.org/10.1145/3567721

[30] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. 2023. Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models. 7909–7920. https://openaccess.thecvf.com/content/ICCV2023/html/Hollein_Text2Room_Extracting_Textured_3D_Meshes_from_2D_Text-to-Image_Models_ICCV_2023_paper.html

[31] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. 2023. OneFormer: One Transformer to Rule Universal Image Segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2989–2998. https://doi.org/10.1109/CVPR52729.2023.00292 ISSN: 2575-7075.

[32] Xiaoliang Ju, Zhaoyang Huang, Yijin Li, Guofeng Zhang, Yu Qiao, and Hongsheng Li. 2024. DiffInDScene: Diffusion-based High-Quality 3D Indoor Scene Generation. 4526–4535. https://openaccess.thecvf.com/content/CVPR2024/html/Ju_DiffInDScene_Diffusion-based_High-Quality_3D_Indoor_Scene_Generation_CVPR_2024_paper.html

[33] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. 2023. Scaling up GANs for Text-to-Image Synthesis. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 10124–10134.

[34] Seonji Kim, Dooyoung Kim, Jae-Eun Shin, and Woontack Woo. 2024. Object Cluster Registration of Dissimilar Rooms Using Geometric Spatial Affordance Graph to Generate Shared Virtual Spaces. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. 796–805. https://doi.org/10.1109/VR58804.2024.00099 ISSN: 2642-5254.

[35] Griffin E. Koch and Marc N. Coutanche. 2024. Context reinstatement requires a schema relevant virtual environment to benefit object recall. *Psychonomic Bulletin & Review* (March 2024). https://doi.org/10.3758/s13423-024-02472-w

[36] Eric Krokos, Catherine Plaisant, and Amitabh Varshney. 2019. Virtual memory palaces: immersion aids recall. *Virtual reality* 23, 1 (2019), 1–15.

[37] André Kunert, Alexander Kulik, Stephan Beck, and Bernd Froehlich. 2014. Photoportals: shared references in space and time. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing (CSCW '14)*. Association for Computing Machinery, New York, NY, USA, 1388–1399. https://doi.org/10.1145/2531602.2531727

[38] Gun A. Lee, Seungwon Kim, Youngho Lee, Arindam Dey, Thammathip Piumsomboon, Mitchell Norman, and Mark Billinghurst. 2017. Improving Collaboration in Augmented Video Conference using Mutually Shared Gaze. *ICAT-EGVE 2017 - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments* (2017), 8 pages. https://doi.org/10/gm8b9d Artwork Size: 8 pages ISBN: 9783038680383 Publisher: The Eurographics Association.

[39] Eric LG Legge, Christopher R Madan, Enoch T Ng, and Jeremy B Caplan. 2012. Building a memory palace in minutes: Equivalent memory performance using virtual versus conventional environments with the Method of Loci. *Acta psychologica* 141, 3 (2012), 380–390.

[40] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning (ICML'23, Vol. 202)*. JMLR.org, Honolulu, Hawaii, USA, 19730–19742.

[41] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. 2023. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22511–22521.

[42] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. 2023. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 300–309.

[43] David Lindlbauer and Andy D. Wilson. 2018. Remixed Reality: Manipulating Space and Time in Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, Montreal QC Canada, 1–13. https://doi.org/10.1145/3173574.3173703

[44] Eleanor A Maguire, Elizabeth R Valentine, John M Wilding, and Narinder Kapur. 2003. Routes to remembering: the brains behind superior memory. *Nature neuroscience* 6, 1 (2003), 90–95.

[45] Nels Numan and Anthony Steed. 2022. Exploring User Behaviour in Asymmetric Collaborative Mixed Reality. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*. ACM, Tsukuba, Japan, 11. https://doi.org/10.1145/3562939.3565630

[46] Ohan Oda, Carmine Elvezio, Mengu Sukan, Steven Feiner, and Barbara Tversky. 2015. Virtual Replicas for Remote Assistance in Virtual and Augmented Reality. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 405–415. https://doi.org/10/ghp2qv

[47] OpenAI. 2023. GPT-4 Technical Report. https://arxiv.org/abs/2303.08774v3

[48] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L. Davidson, Sameh Khamis, Mingsong Dou, Vladimir Tankovich, Charles Loop, Qin Cai, Philip A. Chou, Sarah Mennicken, Julien Valentin, Vivek Pradeep, Shenlong Wang, Sing Bing Kang, Pushmeet Kohli, Yuliya Lutchyn, Cem Keskin, and Shahram Izadi. 2016. Holoportation: Virtual 3D Teleportation in Real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 741–754. https://doi.org/10.1145/2984511.2984517

[49] Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. 2021. ATISS: Autoregressive transformers for indoor scene synthesis. In *Advances in neural information processing systems (NeurIPS)*.

[50] Quynh Phung, Songwei Ge, and Jia-Bin Huang. 2024. Grounded Text-to-Image Synthesis with Attention Refocusing. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[51] Thammathip Piumsomboon, Gun A Lee, Jonathon D Hart, Barrett Ens, Robert W Lindeman, Bruce H Thomas, and Mark Billinghurst. 2018. Mini-me: An adaptive avatar for mixed reality remote collaboration. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–13.

[52] Thammathip Piumsomboon, Gun A. Lee, Jonathon D. Hart, Barrett Ens, Robert W. Lindeman, Bruce H. Thomas, and Mark Billinghurst. 2018. Mini-Me: An Adaptive Avatar for Mixed Reality Remote Collaboration. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173620

[53] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988* (2022).

[54] Shwetha Rajaram, Nels Numan, Balasaravanan Thoravi Kumaravel, Nicolai Marquardt, and Andrew D Wilson. 2024. BlendScape: Enabling End-User Customization of Video-Conferencing Environments through Generative AI. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3654777.3676326

[55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. 2022. High-Resolution Image Synthesis with Latent Diffusion Models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, New Orleans, LA, USA, 10674–10685. https://doi.org/10.1109/CVPR52688.2022.01042

[56] Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. 2024. Fast High-Resolution Image Synthesis with Latent Adversarial Diffusion Distillation. https://doi.org/10.48550/arXiv.2403.12015 arXiv:2403.12015 [cs].

[57] Jonas Schjerlund, Kasper Hornbæk, and Joanna Bergström. 2022. OVRlap: Perceiving Multiple Locations Simultaneously to Improve Interaction in VR. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3491102.3501873

[58] Jonas Schult, Sam Tsai, Lukas Höllein, Bichen Wu, Jialiang Wang, Chih-Yao Ma, Kunpeng Li, Xiaofang Wang, Felix Wimbauer, Zijian He, Peizhao Zhang, Bastian Leibe, Peter Vajda, and Ji Hou. 2024. ControlRoom3D: Room Generation using Semantic Proxy Rooms. 6201–6210. https://openaccess.thecvf.com/content/CVPR2024/html/Schult_ControlRoom3D_Room_Generation_using_Semantic_Proxy_Rooms_CVPR_2024_paper.html

[59] Liangchen Song, Liangliang Cao, Hongyu Xu, Kai Kang, Feng Tang, Junsong Yuan, and Zhao Yang. 2023. RoomDreamer: Text-Driven 3D Indoor Scene Synthesis with Coherent Geometry and Texture. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23)*. Association for Computing Machinery, New York, NY, USA, 6898–6906. https://doi.org/10.1145/3581783.3611800

[60] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. 2015. SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite. 567–576. https://openaccess.thecvf.com/content_cvpr_2015/html/Song_SUN_RGB-D_A_2015_CVPR_paper.html

[61] Maximilian Speicher, Jingchen Cao, Ao Yu, Haihua Zhang, and Michael Nebeling. 2018. 360anywhere: Mobile ad-hoc collaboration in any environment using 360 video and augmented reality. *Proceedings of the ACM on Human-Computer Interaction* 2, EICS (2018), 1–20.

[62] Misha Sra, Sergio Garrido-Jurado, and Pattie Maes. 2018. Oasis: Procedurally Generated Social Virtual Spaces from 3D Scanned Real Spaces. *IEEE Transactions on Visualization and Computer Graphics* 24, 12 (Dec. 2018), 3174–3187. https://doi.org/10.1109/TVCG.2017.2762691

[63] Gabriela Ben Melech Stan, Diana Wofk, Estelle Aflalo, Shao-Yen Tseng, Zhipeng Cai, Michael Paulitsch, and Vasudev Lal. 2023. LDM3D-VR: Latent Diffusion Model for 3D VR. https://doi.org/10.48550/arXiv.2311.03226 arXiv:2311.03226 [cs].

[64] Gabriela Ben Melech Stan, Diana Wofk, Scottie Fox, Alex Redden, Will Saxton, Jean Yu, Estelle Aflalo, Shao-Yen Tseng, Fabio Nonato, Matthias Muller, and Vasudev Lal. 2023. LDM3D: Latent Diffusion Model for 3D. https://doi.org/10.48550/arXiv.2305.10853 arXiv:2305.10853 [cs].

[65] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. 2024. MVDiffusion: enabling holistic multi-view image generation with correspondence-aware diffusion. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, 51202–51233.

[66] Balasaravanan Thoravi Kumaravel, Fraser Anderson, George Fitzmaurice, Bjoern Hartmann, and Tovi Grossman. 2019. Loki: Facilitating Remote Instruction of Physical Tasks Using Bi-Directional Mixed-Reality Telepresence. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans LA USA, 161–174. https://doi.org/10.1145/3332165.3347872

[67] Balasaravanan Thoravi Kumaravel and Björn Hartmann. 2022. Interactive mixed-dimensional media for cross-dimensional collaboration in mixed reality environments. *Frontiers in Virtual Reality* 3 (2022), 766336.

[68] Roland Van Der Linden, Ricardo Lopes, and Rafael Bidarra. 2014. Procedural Generation of Dungeons. *IEEE Transactions on Computational Intelligence and AI in Games* 6, 1 (March 2014), 78–89. https://doi.org/10.1109/TCIAIG.2013.2290371

[69] Chiu-Hsuan Wang, Chia-En Tsai, Seraphina Yong, and Liwei Chan. 2020. Slice of Light: Transparent and Integrative Transition Among Realities in a Multi-HMD-User Environment. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, Virtual Event USA, 805–817. https://doi.org/10.1145/3379337.3415868

[70] Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. 2021. SceneFormer: Indoor Scene Generation with Transformers. In *2021 International Conference on 3D Vision (3DV)*. 106–115. https://doi.org/10.1109/3DV53792.2021.00021 ISSN: 2475-7888.

[71] Zeyu Wang, Cuong Nguyen, Paul Asente, and Julie Dorsey. 2023. PointShopAR: Supporting Environmental Design Prototyping Using Point Cloud in Augmented Reality. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–15. https://doi.org/10.1145/3544548.3580776

[72] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. 2024. MotionCtrl: A Unified and Flexible Motion Controller for Video Generation. In *ACM SIGGRAPH 2024 Conference Papers (SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3641519.3657518

[73] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. 2024. Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[74] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. 2016. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. https://doi.org/10.48550/arXiv.1506.03365 arXiv:1506.03365 [cs].

[75] Jingbo Zhang, Xiaoyu Li, Ziyu Wan, Can Wang, and Jing Liao. 2023. Text2NeRF: Text-Driven 3D Scene Generation with Neural Radiance Fields. http://arxiv.org/abs/2305.11588 arXiv:2305.11588 [cs].

[76] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023. Adding Conditional Control to Text-to-Image Diffusion Models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Paris, France, 3813–3824. https://doi.org/10.1109/ICCV51070.2023.00355

[77] Y. Zhang, F. Yu, S. Song, P. Xu, A. Seff, and J. Xiao. 2016. Large-scale scene understanding challenge: Room layout estimation. http://lsun.cs.princeton.edu/2016/

[78] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2017. Scene Parsing through ADE20K Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Honolulu, HI, 5122–5130. https://doi.org/10.1109/CVPR.2017.544

[79] Xiaoyu Zhou, Xingjian Ran, Yajiao Xiong, Jinlin He, Zhiwei Lin, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. 2024. GALA3D: Towards Text-to-3D Complex Scene Generation via Layout-guided Generative Gaussian Splatting. http://arxiv.org/abs/2402.07207 arXiv:2402.07207 [cs].

# A APPENDIX

## A.1 Contextually Adaptive Prompt Inference: Full System Prompt

*System Prompt.* You are a helpful assistant that acts like highly creative interior architect and photographer. You are given descriptions of images captured by a camera on a tripod placed exactly in the middle of an open indoor space at a height of 1.5 meters. This camera has a field-of-view of 55 degrees and only takes square images. You will be given a set of image descriptions with Y rotation values of the camera, as well as Y rotation values without a description. From your perspective as a creative interior architect, your task is to describe what you expect to see for the image that will be taken at the Y rotation value with unknown contents. You receive the rotation value and descriptions in JSON format. Do not use words such as 'blend, 'transition', 'fusion', 'mix', 'transformation' (or synonyms), but concretely describe the novel contents and salient objects you expect to see in the area without repeating objects. Use a similar format to the other given descriptions. It is not always obvious what the camera will capture as the contents of the space can be highly diverse in style and content, so please be creative and focus on coming up with new objects and artifacts that fit in. You can assume that the objects appearing in the known image description do not show up in the other image (all objects are fully contained within the image frame). Do not include mentions of the shape of the room (e.g., corner). Use comma-separated descriptions (e.g., instead of 'On the sticky note wall, a whiteboard marker tray holds colorful pens, a spark of color in an otherwise monochrome environment.', write 'Sticky note wall with whiteboard marker holding colorful pens, monochrome environment'. Always start descriptions with '... space with' where '...' is the type of the room (e.g., living room, kitchen, etc.). Only return the descriptions with the set_description function, without any explanation. Keep your descriptions short please, without adding too many different items/objects, with 20 words or less for each description.

*User Prompt.* The size of the room is space_size_str (WxHxL) meters and the camera is positioned in the middle. These are the Y rotation values and descriptions of the images that were already taken: y_rotations_and_descriptions. What do you expect for the following Y rotation values: y_rotations_without_descriptions? Consider the theme of "" when coming up with the descriptions

## A.2 ControlNet-Layout: Additional Output Samples


**Figure A.1: Additional results of the ControlNet-Layout model. Each of these output images was generated by combining Control-Layout and ControlNet-Depth with weights 0.6 and 0.3, respectively.**

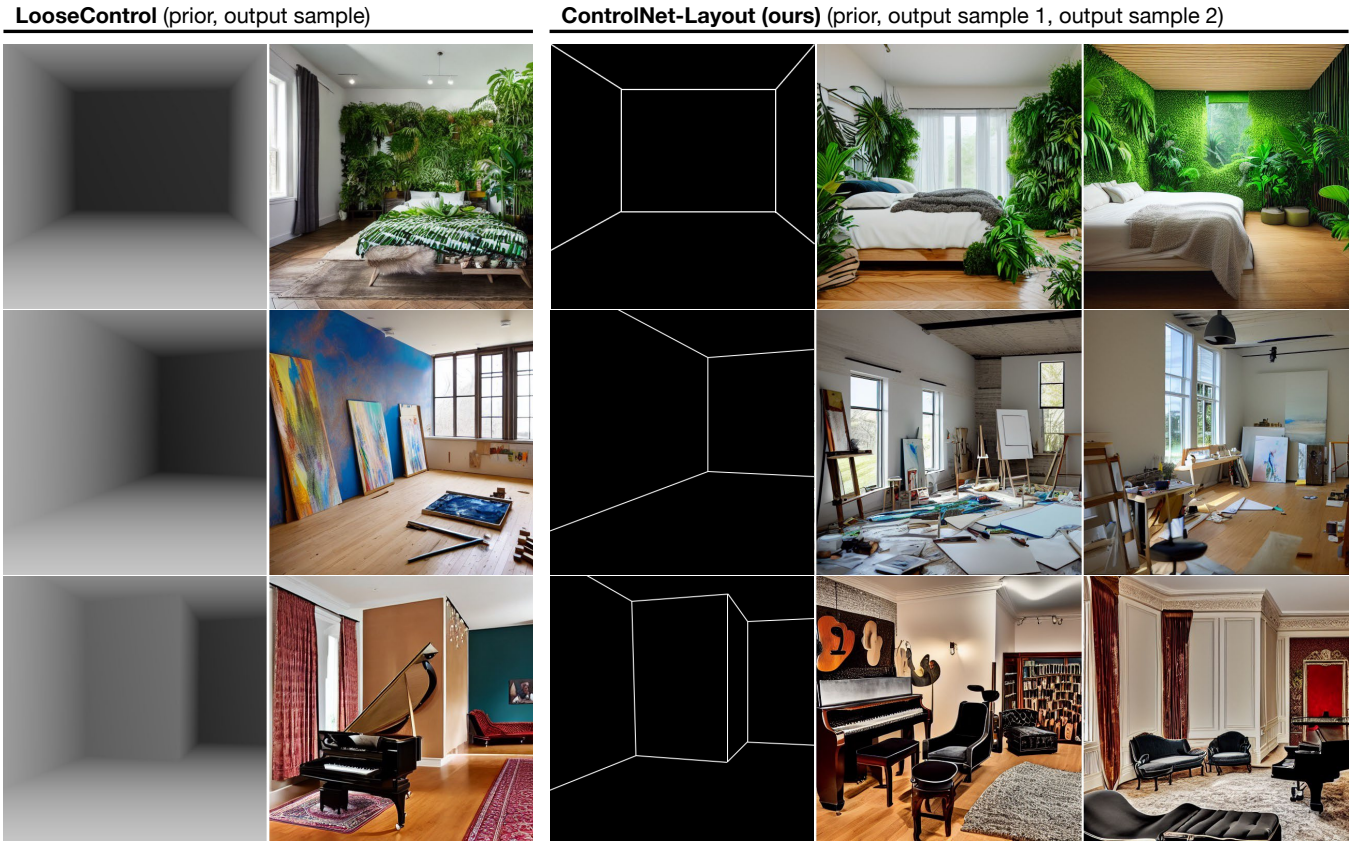


**Figure A.2: Comparative image generation output of the recent LooseControl model and our ControlNet-Layout model, including prior images and output images each. The LooseControl prior and output images in this figure were reproduced from the paper's web page (https://shariqfarooq123.github.io/loose-control) with permission from the authors. The ControlNet-Layout prior images were manually created to match the room structure depicted in the LooseControl prior images.**

## A.3 Influence of Submesh Shapes on Geometric Prior Shape

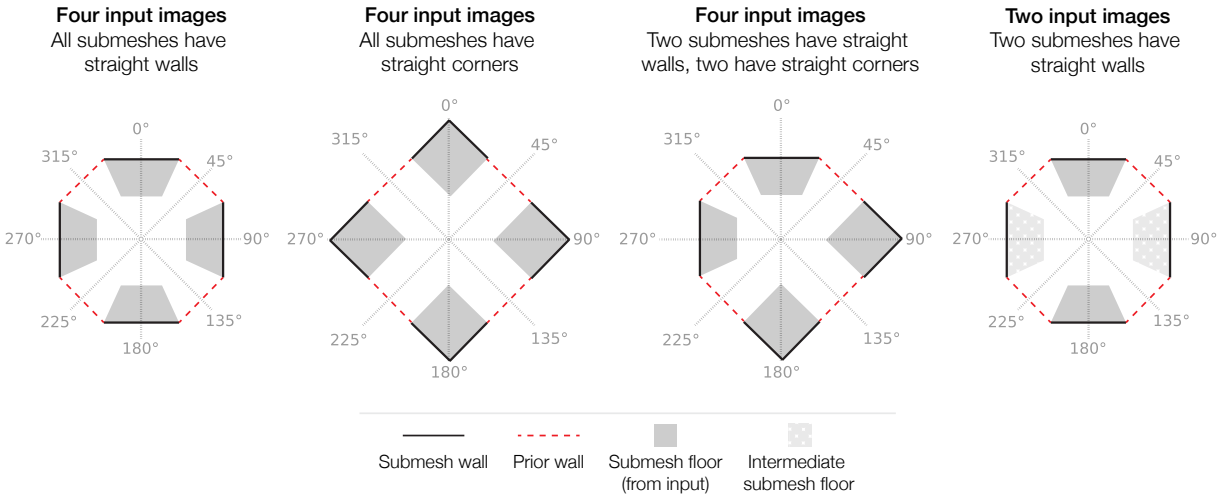


**Figure A.3: Visual explanation of the impact of submesh count and shape on the shape of the geometric prior mesh and final blended space.**


**Figure A.4: Examples of spaces generated with various numbers of submeshes. Left: a SpaceBlender mesh based on four input images, all featuring corners. Right: a SpaceBlender mesh based on five input images featuring a mixture of room shapes.**

## A.4 Preliminary User Study: Questionnaire Items

| **Please indicate your level of agreement with the following statements**<br>Do not agree at all; Disagree; Neutral; Agree; Fully Agree | **Category** |
|---|---|
| I felt like I was actually there in the environment of the presentation. | Self-Location |
| It seemed as though I actually took part in the action of the presentation. | Self-Location |
| It was as though my true location had shifted into the environment in the presentation. | Self-Location |
| I felt as though I was physically present in the environment of the presentation. | Self-Location |
| The objects in the presentation gave me the feeling that I could do things with them. | Possible Actions |
| I had the impression that I could be active in the environment of the presentation. | Possible Actions |
| I felt like I could move around among the objects in the presentation. | Possible Actions |
| It seemed to me that I could do whatever I wanted in the environment of the presentation. | Possible Actions |
| I noticed (my partner). | Co-presence |
| (My partner) noticed me. | Co-presence |
| (My partner's) presence was obvious to me. | Co-presence |
| My presence was obvious to (my partner). | Co-presence |
| (My partner) caught my attention. | Co-presence |
| I caught (my partner's) attention. | Co-presence |

**Table A.1: Post-task questionnaire for measuring Self-Location and Possible Actions from The Spatial Presence Experience Scale questionnaire by Hartmann et al. [28] and Co-presence from the Networked Minds Measure of Social Presence questionnaire by Harms and Biocca [27].**

## A.5 Preliminary User Study: Walkthrough Scenarios


**Figure A.5: Overview of environments used in the walkthrough segment of the semi-structured interview, including a *collaborative study session* scenario (left) and a *cooking class with friends* scenario (right).**

## A.6 Preliminary User Study: Semi-Structured Interview Questions

**Questions involving clustering task:**

(1) You mentioned you preferred *condition*, over the others. Could you elaborate on your preference?
(2) Describe the strategies you used to perform the affinity diagramming task. Did you adopt a consistent strategy in all three environments, or did you use different strategies?

**Questions involving walkthrough of additional SpaceBlender environments:**

(1) If any, in what scenarios do you think integrating environmental context that you have a personal relation to in a virtual environment could be valuable or interesting?
(2) Is there anything you wished was different about the design, composition, or function of the blended spaces that you've seen that could make it more engaging, useful, or comfortable?

## A.7 Preliminary User Study: Full Self-Reported Questionnaire Response Analysis

We used the Friedman test for overall comparisons for our analysis, with subsequent post hoc analyses conducted via Wilcoxon signed-rank tests. All reported p-values for post hoc analyses have been adjusted using the Bonferroni correction.

Given that the assumptions requisite for parametric tests were not met for a majority of the results, and given the general recommendation to use non-parametric tests in studies with limited sample sizes, our analysis utilized the Friedman test for overall comparisons, with subsequent post hoc analyses conducted via Wilcoxon signed-rank tests. All reported p-values for post hoc analyses have been adjusted using the Bonferroni correction.

*A.7.1 Possible Actions.* The Friedman test revealed a statistically significant difference in the scores of Possible Actions across the three conditions ($\chi^2(2) = 14.81, p < 0.0001$), suggesting that participants' perceptions of possible actions varied significantly depending on the condition. Post hoc analyses were conducted to explore these differences further. Results showed a significant decrease (W = 5.0, $p = 0.0021$) in Possible Actions scores from Generic3D (M = 3.89, SD = 0.71) to Text2Room (M = 3.13, SD = 0.95), indicating a diminished perception of possible actions within the environment under Text2Room. However, the difference between Generic3D and SpaceBlender (W = 23.0, $p = 0.104$) and between Text2Room and SpaceBlender (W = 27.0, $p = 0.055$) did not reach statistical significance.

*A.7.2 Self-Location.* A similar analysis was conducted for Self-Location scores, with the Friedman test indicating a significant difference across conditions ($\chi^2(2) = 10.53, p = 0.0052$). This finding highlights that the different environments significantly affected the sense of being situated within the environment. After post hoc analysis, the comparison between Generic3D and Text2Room and between Generic3D and SpaceBlender did not show significant differences (W = 49.0, $p > 0.999$). However, a significant difference (W = 0.0, $p = 0.0039$) was found between Text2Room (M = 3.46, SD = 0.78) and SpaceBlender (M = 3.91, SD = 0.68), indicating a change in the sense of self-location between these two conditions.

*A.7.3 Co-Presence.* The analysis of co-presence scores using the Friedman test did not reveal a statistically significant difference across conditions ($\chi^2(2) = 1.59, p = 0.452$).

*A.7.4 Impact of Environmental Factors.* The assessment of environmental factors on task performance revealed several statistically significant differences across conditions.

The analysis for *Layout* showed a $\chi^2(2) = 20.22, p < 0.0001$, indicating significant variations in task conductance related to environmental layout. Post-hoc analysis indicated significant differences between Generic3D (M = 3.92, SD = 0.68) and Text2Room (M = 3.20, SD = 0.75) (W = 10.0, $p = 0.0043$). A significant difference was also observed between Text2Room and SpaceBlender (M = 3.85, SD = 0.70) (W = 4.0 and $p = 0.0036$). However, the difference between Generic3D and SpaceBlender did not reach statistical significance (W = 22.0, $p = 0.4842$).

For *Visual Quality*, a significant difference was found ($\chi^2(2) = 21.73, p < 0.0001$). There was a difference in perceived support for task execution between Generic3D (M = 4.00, SD = 0.00) and Text2Room (M = 3.17, SD = 0.75) (W = 0.0 and $p = 0.0010$), and between Text2Room and SpaceBlender (M = 3.90, SD = 0.30) (W = 3.5 and $p = 0.0049$). The comparison between Generic3D and SpaceBlender did not show a significant difference (W = 18.0, $p = 0.2121$).

*Familiarity* also demonstrated significant differences among conditions ($\chi^2(2) = 14.56, p = 0.00069$). The difference between Text2Room (M = 3.10, SD = 0.83) and SpaceBlender (M = 3.95, SD = 0.22) was significant (W = 0.0 and $p = 0.0039$). Comparisons between Generic3D and Text2Room, and between Generic3D and SpaceBlender did not achieve significance (W = 10.0, $p = 0.1566$; W = 12.0, $p = 0.0604$).