Scene Responsiveness for Visuotactile Illusions in Mixed Reality

Mohamed Kari
Reality Labs Research
Meta Platforms, Inc.
Redmond, Washington, USA

Reinhard Schütte
Institute for Computer Science
University of Duisburg-Essen
Essen, Germany

Raj Sodhi
Reality Labs Research
Meta Platforms, Inc.
Redmond, Washington, USA

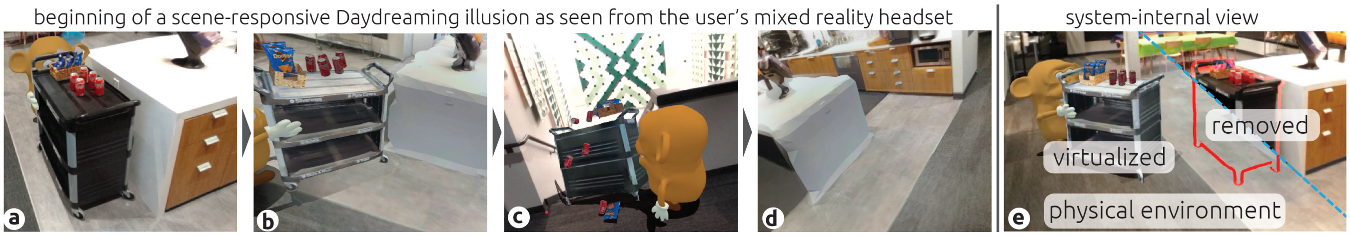

Figure 1: We present Scene Responsiveness, the visual illusion that virtual actions affect the physical scene. ⓐ Wearing a video-passthrough Mixed Reality (MR) headset, a user sees an authentic video view of their physical environment, captured with the externally-facing headset cameras. A virtual monkey character is situated and coherently occluded in the otherwise unmodified physical space, grabbing the PHYSICAL cart. ⓑ As the monkey starts dragging the cart, the object toggles its reality state to VIRTUALIZED just-in-time. Just-in-time virtualization is *coherence-preserving*, so the cart still throws its shadow onto the physical scene and the now seemingly empty surface. ⓒ To the user, it appears as if the monkey drags the physical cart and throws it down the staircase. Virtual shadows and collisions render coherently. ⓓ It appears as if the physical cart is gone. ⓔ Of course, what the user sees is just an illusion. The physical object did not move but was just masked from the user's view, blending seamlessly with the video passthrough. Everything seen in the debug capture, incl. the red object guardian outline and the revelation lens in the top-right corner is part of the system debug view. *The figure gives an authentic impression of the visual fidelity in the headset. However, please watch the accompanying video for a full impression of the visual experience.*

ABSTRACT

Manipulating their environment is one of the fundamental actions that humans, and actors more generally, perform. Yet, today's mixed reality systems enable us to situate virtual content in the physical scene but fall short of expanding the visual illusion to believable environment manipulations. In this paper, we present the concept and system of Scene Responsiveness, the visual illusion that virtual actions affect the physical scene. Using co-aligned digital twins for coherence-preserving just-in-time virtualization of physical objects in the environment, Scene Responsiveness allows actors to seemingly manipulate physical objects as if they were virtual. Based on Scene Responsiveness, we propose two general types of end to-end illusionary experiences that ensure visuotactile consistency through the presented techniques of object elusiveness and object rephysicalization. We demonstrate how our Daydreaming illusion enables virtual characters to enter the scene through a physically closed door and vandalize the physical scene, or users to enchant and summon far-away physical objects. In a user evaluation of our Copperfield illusion, we found that Scene Responsiveness can be rendered so convincingly that it lends itself to magic tricks. We present our system architecture and conclude by discussing the implications of scene-responsive mixed reality for gaming and telepresence.

UIST '23, October 29-November 1, 2023, San Francisco, CA, USA

ACM ISBN 979-8-4007-0132-0/23/10…$15.00
https://doi.org/10.1145/3586183.3606825

CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; **Collaborative interaction**; • **Computing methodologies → Procedural animation**.

KEYWORDS

Mixed reality; situated computing; spatial computing;

ACM Reference Format:
Mohamed Kari, Reinhard Schütte, and Raj Sodhi. 2023. Scene Responsiveness for Visuotactile Illusions in Mixed Reality. In *The 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23), October 29-November 1, 2023, San Francisco, CA, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3586183.3606825

1 INTRODUCTION

In most recent research on adaptive Mixed Reality (MR) interfaces, the physical scene dynamically influences the placement of virtual content [12, 54, 71, 78]. In this paper, we aim to invert the direction of influence between virtuality and physicality, asking "How can virtuality affect the physical world?"

As an answer, we propose *Scene Responsiveness*, the illusion that virtual actions affect the physical world. By altering the visual signal in a video-passthrough MR headset, Scene Responsiveness allows actors to seemingly manipulate their environment, such as a bi-ped character to open the next door for entering or exiting the scene, or a black hole to absorb physical objects from the scene. Also, the user themselves can cast magic enchantments to summon far-away objects. The visual illusion of state responses, shape responses, and pose responses starts by *toggling the reality state* of the manipulated object from PHYSICAL to VIRTUALIZED *just-in-time* as the user or virtual actor begins their manipulation. The response is spatially contained at the involved object, seamlessly blending with the video-passthrough anywhere else and thus maintaining the sensation of being in physical space.

However, the illusionary experience enabled by Scene Responsiveness is not limited to the visually seamless response of physical objects to user input or situated character animations. Instead, we develop and propose *Daydreaming* and *Copperfield illusionary episodes* as self-contained interaction-centered experiences. Daydreaming episodes start with a virtually triggered scene response, and then employ *object elusiveness* and *object rephysicalization* to prevent tactile disillusion: Influenced by the user's behavior, virtualized objects avoid collision with the user by diegetically eluding from the user's body or rephysicalizing in diegetic ways at the pose of their physical counterpart. Copperfield episodes, named after the eponymous vanishing-and-reappearance magician, expand on this by purposefully toggling reality states to make physical objects appear exactly in the pose where a virtual interaction seemingly transported them, thus deepening the illusionary experience of virtual control over physical space.

In addition to the conceptual contribution of Scene Responsiveness, we provide a system *prototype implementation* that draws all required spatial, visual, and semantic information from a co-aligned and unified digital representation of the space and its objects. First, our prototype implementation enables to render virtual content geometrically and physically integrated into the scene respectful of occlusions, collisions, and shadows, across all virtual, physical, and blended areas of the space. Second, the system with its Spielberg component, named after the director and pioneer of character animation, situates virtual characters and their animations as well as gestural user input semantically integrated into space, guided by character and user affordances. Third, apart from the former two system integration efforts, we contribute an integrated spatial computing and shading architecture for *coherence-preserving object virtualization* in world space on video-passthrough MR devices, not yet achievable with the image-space techniques of 2D inpainting in conventional Diminished Reality [64]. Through spatial arrangement of 3D occluders, and subsequent dynamic assignment of camera layers, manipulation of the rendering order, and purposeful writes to and reads from the depth buffer, we visually remove physical objects by masking them with the environment background, yet render all other virtual content coherently into the scene.

We conducted a user evaluation in two different spaces with 20 participants, 10 participants per space, evaluating the rendering coherence as well as the illusion fidelity in a Copperfield episode. We found out that 18 out of 20 participants were surprised they were able to take a seat on VIRTUALIZED chair that has been dropped by a virtual character in seemingly empty space.

Considering the upcoming releases of video-passthrough MR headsets [2, 58] and the expected industry focus on such devices for the years to follow, we argue that Scene Responsiveness can create captivating MR experiences not only for situated gaming but as a general means for situated MR, such as scene-responsive telepresence or virtual assistants.

Contributions

In summary, we contribute

- the *novel concept of Scene Responsiveness* for high-fidelity illusions in video-passthrough Mixed Reality,
- a *design space of end-to-end illusionary experiences* that prevent disillusion through object elusiveness and rephysicalization to maintain visuotactile consistency, along with the two design samples of *Daydreaming and Copperfield-type epsisodes*,
- the algorithmic *spatial computing and shading architecture RealityToggle* for *coherence-preserving just-in-time virtualization* in 3D world space on video-passthrough Mixed Reality devices,
- an integrated *system architecture and implementation*, comprising our *TwinBuilder Unity plug-in* to obtain, process, and annotate digital representations, as well as our *Spielberg component* for situated character-centric stories and animations,
- a *user evaluation* with 20 participants of the visuotactile Copperfield illusion,
- the applications of *scene-responsive gaming*, *scene-responsive telepresence*, and *scene-responsive television*.

2 BACKGROUND AND RELATED WORK

2.1 Scene Coherence in Mixed Reality

Under the term of *scene coherence*, foundational work in AR and MR has been dedicated to the graphical techniques needed to render geometrically and physically coherent occlusions [4, 8, 35], collisions, shadows, and reflectance [17, 40, 43, 74], as well as to the question of how to reconstruct the necessary depth [42, 67, 73, 82, 87, 89] and lighting [26, 45, 56, 90] information from the physical scene. Once virtual content can be coherently integrated into the scene, the question arises as to where to render it. FLARE [20] automatically layouts AR content on vertical and horizontal planes, guided by developer-defined rules. SnapToReality [70] automatically aligns virtual content, guided by planes and edges detected in the physical geometry. Lages et al. [44] investigate how to adapt UI elements in AR to vertical planes. DepthLab [17] not only proposes different depth representations for scene coherence but also considers geometrically coherent path planning for virtual characters.

To ensure illusion fidelity, we integrate occlusion, collision, and shadow coherence through digital space twins, created with our TwinBuilder Unity plug-in. However, our work most importantly differs from previous research in that it must also maintain *manipulation coherence* after the scene has been visually manipulated. To this end, we present a spatial computing and shading architecture with dynamic camera layer assignment, render ordering, and depth buffer access which preserves scene coherence after visually removing physical and inserting virtual objects (Fig. 2).

2.2 Scene Situatedness in Mixed Reality

Scene Responsiveness not only demands scene coherence but also situatedness [9, 18, 25, 74, 91, 93], i.e., a semantic relationship between virtual and physical content.

Digital counterparts co-aligned with physicality through markers or other means have been proposed early on to situate virtual content. A variety of visualization paradigms [34, 35, 51, 51, 61, 85, 100] for situated annotations [31], ghost views [36], virtual paths [55], or magic lenses [5–7, 52, 53, 88] have since emerged for use in navigation, commerce [30], maintainenance, education [24, 29], tutorial instructions [14, 60], etc. One closely related use of co-aligned representations is found in situated gaming [55, 72]. RoomAlive by Jones et al. [33] allows to create a spatial scan, author experiences therein, and then deploy this experience to a room-scale projection mapping system. We borrow the idea of digital representations that guide content placement, however, use this representation not only for situation, but also for *virtualization* of objects just-in-time by visually removing their physical counterpart first and then inserting a digital object in their place to make them interactable for virtuality.

Adaptive MR in contrast does not rely on a space-specific model but asserts the claim to generalize to different spaces. SemanticAdapt [12] and the approach by Luo et al. [54] investigate adaptive situation of 2D content. Previous work also considers the adaptive situation of virtual 3D actors specifically. Retargetable AR [84] situates virtual characters in a physical scene for situated storytelling. Li et al. [46] build on this concept of situated storytelling, but introduce user interactivity to influence story playback. Liang et al. [47] consider dynamically situating and controlling a virtual pet in MR. SpaceTime [32], Kim et al. [41] and Grønbæk et al. [22] situate avatars based on remote user activities in local space. Schmidt et al. [75] consider physical object manipulation by virtual agents through actuation. Story CreatAR [80], ScalAR [71], and Ng et al. [68] provide frameworks for authoring situated AR experiences. ARAnimator [97] situates animation sequences in space. Shin et al. [77, 78] study the effect of game adaption to different spaces. Scene Responsiveness also situates characters in space guided by objects in the physical scene. However, Scene Responsiveness differs from the above approaches in inverting the order of influence: Not only does the physical scene influence the virtual scene, but also virtual actors and actions can influence the presented physical scene through manipulation of the visual signal in a video-passthrough headset (Fig. 2).

Blending between physical scene and virtual scenes has been explored by Blending Spaces [15] and RealityCheck [27]. However, physical and virtual scenes are assumed to be structurally and functionally different. In contrast, Scene Responsiveness blends seamlessly between structurally and functionally identical scenes for object virtualization.

Situated VR co-aligns the virtual scene with the physical scene, either through alignment before the experience as in Reality Skins [76], Oasis [81], Substitutional Reality [79], Tailored Reality [16], or Scenograph [57], through procedural generation as in DreamWalker [96] or VRoamer [11], or through assisted annotation as in ARchitect [48]. While Scene Responsiveness shares the idea of paired physical-virtual object counterparts, it does not pursue showing a functionally different space in VR. Instead, Scene Responsiveness aims to replace a physically captured object with its virtually rendered counterpart as part of the object virtualization step for subsequent interaction. Technically fundamentally different, we target video-passthrough MR, where the user's own hands and body remain normally visible, and they can have face-to-face conversations with other humans in their space, even after manipulating the scene.

Figure 2: Levels of scene integration in MR. The concept of Scene Responsiveness aims to add an exciting new level of scene integration beyond coherence and situatedness to MR.

2.3 Scene Editing in Mixed Reality

Diminished Reality (DR) [13, 23, 39, 62, 63, 65, 66] offers to remove areas from the shown frame. Technically, Scene Responsiveness differs from conventional DR in that it operates through masking in 3D world space, rather than operating in 2D image space, to maintain depth-related scene coherence. Pragmatically, camera frames are not accessible on consumer-grade passthrough headsets to ensure privacy. Conceptually, Scene Responsiveness differs in that it virtualizes objects, i.e., inserts an object replica rather than just removing it. TransforMR [38] is also concerned with inserting virtual counterparts after removing physical objects to produce semantically coherent scenes. However, virtual objects directly follow the pose and articulation of physical objects, prohibitive of independent control over virtual objects in world space. Overlay-based AR such as Annexing Reality [28] also disallows object displacement.

SceneCtrl [98] offers to select, move, delete, and copy objects in an MR scene. It performs editing operations in image space and renders the results in the HoloLens optical see-through display. Remixed Reality [49] shows a rerendered voxel-based representation of the user's environment in VR, captured through Kinect cameras. However, Scene Responsiveness differs from Remixed Reality and SceneCtrl in three fundamental ways.

First, Scene Responsiveness aims for the *imperceptibilty of manipulation* in our Copperfield illusion. Our evaluation indicates that we accomplish this with our spatial computing and shading pipeline for seamlessly blending between co-aligned physical and virtual spaces while maintaining full scene coherence concerning occlusions, collisions, and shadows. In addition, the ability of illusion-quality scene manipulation asks for additional interaction concepts to maintain the illusion. Thus, we contribute the concepts of object elusiveness and rephysicalization as part of our different illusionary episodes, thereby ensuring visuotactile consistency. These considerations were out of the question in previous research, given the absence of illusion-quality visual manipulation.

Second, Scene Responsiveness aims for *semantic situation and manipulation* through afforded interaction, semantically integrated with the scene and meaningfully related to specific objects. Thus, Scene Responsiveness enables situated character-environment interactions, such as opening an elevator by a button press, or dragging a heavy cart differently than carrying a lighter chair, rather than providing universal yet generic geometric operations on images or voxel grids. A semantic rather than geometric consideration then also allows for operations that target the scene graph such as decomposing an object group to individually apply physics as seen with the coke cans, etc., in Fig. 1.

Third, Scene Responsiveness aims for a different *interaction paradigm*. Rather than focusing on user input for manipulation only, our concepts of receptive and responsive affordances in space enable story and telepresence modes.

2.4 Illusions in Mixed Reality

Actuation to provide haptics in VR has been used for tactile illusions [1, 83]. The dominance of vision over proprioception has also been taken advantage of for perceptual manipulation in VR [69, 86], in particular for haptic retargeting [3, 99]. Scene Responsiveness targets MR rather than VR.

3 SCENE RESPONSIVENESS FOR VISUOTACTILE ILLUSIONS

In the following, we first define the terms central to Scene Responsiveness, and then show how scene-responsive illusions can be maintained and completed through end-to-end *illusionary episodes*.

3.1 Scene Responsiveness in Mixed Reality

In situated MR, the physical scene affords the meaningful *placement* of virtual elements. We introduce *Scene Responsiveness* as the illusion that the physical scene *responds* to virtual actions.

Physical and Virtual Actors. Actions are performed by actors. We differentiate between *the user* as a physical actor in local space, *other physical actors* in local space or remote spaces, and *virtual actors* in local space. *Virtual actors* encompass *agents* such as non-player characters (NPCs) in gaming that pursue their own goals, *assistants* that follow the user's goals or instructions, and *avatars* that mimic the behavior of a physical remote actor. In the following, we use the terms "virtual actor" and "character" synonymously.

Physical and Virtual Actions. We further distinguish between *virtual actions* and *physical actions*. The actions of virtual actors are always virtual. In contrast, actions of the user can be either virtual or physical. When the user takes a seat on a physical chair, their action is fully contained in physical space and thus is physical. However, when the user lights a virtual fire that seemingly burns physical objects, swings a virtual lightsaber to seemingly cut a physical object in half, or uses a physical hand gesture for telekinesis to seemingly summon a remote physical object, we classify these actions as *virtual* because the action's effect is meant to be contained in virtuality and does not affect the physical world *in esse*. However, Scene Responsiveness can create instead an illusion *as if* the virtual action had affected the physical world. This is achieved by manipulating the visual signal that reaches the user's eyes and then maintaining this illusion diegetically by ensuring consistency between the tactile and the visual signal.

Receptive and Responsive Affordances. The meaningful action possibilities that a physical scene offers to an actor can be referred to as affordances [37]. While some actions, such as grabbing and moving an object, modify the physical scene, other actions simply "occupy space" without visibly modifying the physical scene itself, e.g., walking on the floor. To distinguish between these different types of "using" the physical world, we introduce the notion of *receptive affordances* and *responsive affordances* in the MR context.

Receptive affordances represent the meaningful *augmentation possibilities* offered by the physical scene without entailing a modification thereof. Such receptive affordances may invite the simple augmentation of a virtual object on the empty area of a desk. A more involved way of augmentation is situating a virtual avatar during a telepresence session on the spatially most appropriate and available seating accommodation given the local user's current pose and gaze in space.

In contrast, *responsive affordances* refer to the meaningful *manipulation possibilities* offered by the physical scene, entailing modification thereof. For example, a chair might not only offer the receptive affordance of taking a seat but also the responsive affordance of moving it, e.g., pulling it out from under the table.

Affordance Features. Both receptive and responsive affordances exhibit *affordance features* that describe the spatial and operational details of the afforded actor-object interaction, similar to the notion of affordance features in robotics [95]. The receptive affordance of "sitting", provided by a chair, might be described by features such as where to stand immediately before taking the seat, in which direction to look at the time, at which height the seating surface is located, whether there are armrests, and whether the seat is fixed, rotating, or mobile. The responsive affordance of "moving the chair" might be described by features such as the hand and body pose relative to the object when grasping for it and whether the object is held at a close body position or dragged over the floor while walking backward when moving it.

Situated Animations. These *affordance features* directly parameterize *situated animations* when a character acts upon the affordance. Continuing the example of a responsive affordance for grasping, in our prototype, we use path planning to situate the character at the desired body pose by means of a navigation mesh before grasping Then, we use inverse kinematics to control the hand pose for the actual grasping motion.

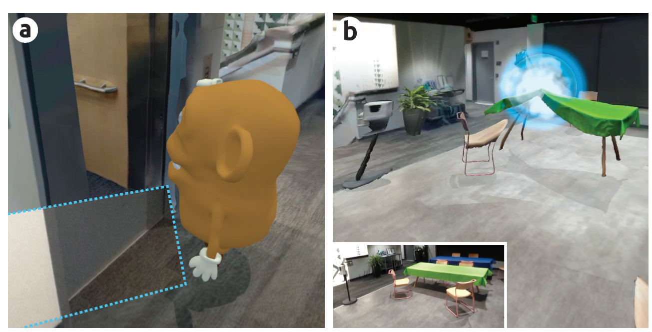

Figure 3: State and shape responsiveness. ⓐ Pressing the elevator's call button entails the state-changing response of sliding the elevator door open. ⓑ As the abstract black hole character approaches, objects in its vicinity are deformed, attracted, and even absorbed. Note that the user is still mostly seeing video-passthrough MR and only the visual areas involved along with a small blended neighborhood are visually manipulated. *Please refer to the accompanying video for a more immersive impression.*

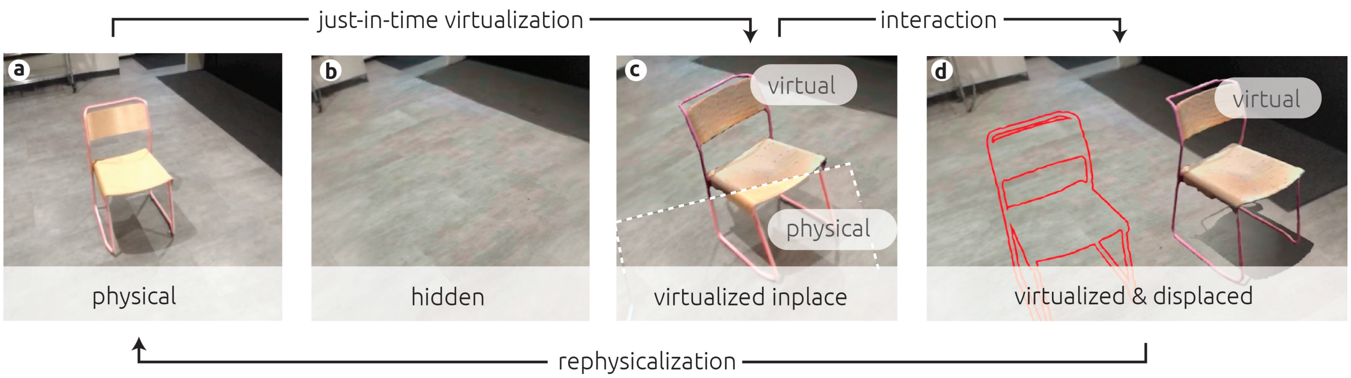

Figure 4: Object virtualization. Scene Responsiveness is fundamentally enabled by toggling an object reality state to VIRTUALIZED. ⓐ At first, the object is in PHYSICAL reality state. ⓑ Upon virtualization, we first switch it to HIDDEN reality state, using visual information from a co-aligned digital space twin. We employ alpha blending with a radial gradient for compositing the masking geometry and the passthrough background to obtain a seamless removal effect. ⓒ Then, we insert a digital object twin in the same pose as its physical counterpart, putting it to VIRTUALIZED reality state, but *in-place* manipulation state. Ideally, the visual signals in VIRTUALIZED and PHYSICAL reality state were identical. Therefore, we also preserve scene coherence in VIRTUALIZED reality state, meaning that the virtual counterpart casts a shadow and the masked area receives a shadow. ⓓ A virtual action by an actor leads to a scene response, putting it to *displacing* manipulation state in the process and to *displaced* once the response's target state is reached. ⓐ Rephysicalization in some diegetic way first returns the object to the pose of the physical counterpart and then toggles its reality state back to PHYSICAL.

Scene Response. Receptive affordances only require the dynamic playback of a situated animation to begin passive object interaction. Responsive affordances are additionally associated with the induced *scene response*. The scene response describes the transformation that the corresponding object undergoes after the user or character acts upon the affordance. This transformation might change spatial properties (Fig. 1b) or structural properties (Fig. 3a) or both.

The affordances and therefore the responses in a physical scene are as rich as the objects contained in it and also differ in their specificity. Some affordances such as moving an object, or kicking it over with the foot, are generic and provided by many objects. Other affordances such as opening a fridge by use of its handle or two-handed typing on a keyboard are much more specific to the object. However, all responsive affordances share the general method used to render the scene response as follows.

Toggling the Reality State of an Object. By default, we pass through a physical object to the headset displays as captured by the externally facing headset cameras. As soon as a scene response in the scene is triggered, the core illusion of Scene Responsiveness *pretends* that the virtual action stimulates a response in the physical object, however, *actually* we first replace the physical object with a virtual counterpart and then apply the response to the virtual counterpart instead. We refer to this as *virtualization*, i.e., toggling the object's *reality state* from PHYSICAL to VIRTUALIZED. The VIRTUALIZED object can then be spatially detached from the pose of its physical counterpart and transformed in any way virtuality affords. To toggle an object's reality state, we exploit the capacity provided by a video-passthrough MR headset to fully control every light ray that reaches the user's eyes when presenting the scene. Fig. 4 gives an overview of this process.

Self-contained Illusionary Episodes. We propose two types of self-contained *illusionary episodes* that are based on the core illusion of Scene Responsiveness. Both types follow the idea of “What you see is what you feel” to ensure visuotactile consistency, however, toggle reality states according to different rationales. Fig. 7 provides an overview of the different types.

3.2 Scene-Responsive Daydreaming Episodes

A Daydreaming episode aims to unfold more or less surreal happenings while preventing disillusion until completion. It begins with a scene response to a virtual action but then aims to maintain visuotactile consistency to the degree that users are left wondering whether what they saw really happened or if it was just a product of imagination, inspired by *Alice's Adventures in Wonderland* and *Inception*. In the following section, we present the steps that make up a Daydreaming episode and describe the causes and remedies for disillusion in detail.

3.2.1 Step 1: Initiating the Illusion. A Daydreaming episode begins as soon as a virtual action triggers a response in a PHYSICAL object. The object's reality state is toggled from PHYSICAL to VIRTUALIZED. Then, the response is applied to the VIRTUALIZED object, as described in Fig. 4. Fig. 1 demonstrates how an affordance provided by a single object–the cart–can trigger virtualization of the dependent objects–the coke cans and other items on top–as well.

Within the VIRTUALIZED reality state, we distinguish three *manipulation sub-states* in an object. An object is virtualized *in-place*, i.e., the VIRTUALIZED object's pose, shape, and state initially match those of the PHYSICAL counterpart. A triggered scene response then starts *displacing* the VIRTUALIZED object from its original source pose, shape, or state. Once the direct response ends, the object becomes *displaced* and remains so until changed again. Based on the manipulation state, the object's physics simulation can be intercepted as needed. For example, we override gravity and collision simulation while a character is carrying an object.

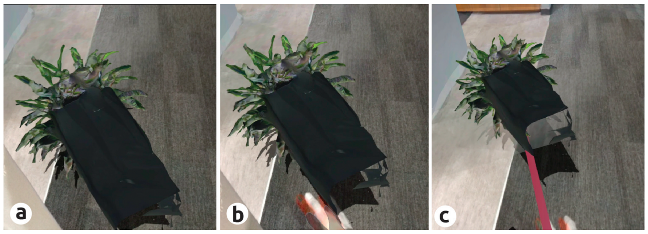

Figure 5: Object elusiveness prevents disillusion from visual collisions, where the user perceives the collision visually but not tactilely. In this example, we reuse the same navigation mesh that is used by the character for elusion through object agency.

3.2.2 Step 2: Maintaining the Illusion. Displacing a VIRTUALIZED object elicits a discrepancy between the perceived and the physical reality. This mismatch can cause disillusion when any part of the user's body enters into a *visual collision* with the VIRTUALIZED object: The user perceives the collision visually but not tactilely, providing the user with adamant evidence that they are being tricked. To prevent such disillusion, we must keep the visual and the tactile signal consistent. Tactility is a hard constraint, so visibility must be changed to fit tactility. Therefore, we propose the concept of *virtual object elusiveness* (Fig. 5): Whenever the user approaches a VIRTUALIZED object too closely, its spatial distance from the user is increased.

We trigger an *elusion event* in an object when the distance between either hand or the headset from the VIRTUALIZED object falls below a specified threshold. To ensure *the lack of tactile feedback*, the VIRTUALIZED object can be rendered *anywhere but* in the colliding volume. Therefore, elusion offers two degrees of freedom: *the elusion target* describing where to elude the object to, and the *elusion mechanism* describing how to get it there. The *elusion mechanism* could make the VIRTUALIZED object disappear, disintegrate, or melt, in front of the user's eyes and re-appear, re-integrate, or re-freeze on top of the closest table or shelf. Or the elusion mechanism could give the object some sense of agency, allowing it to innocently slide away a couple of inches as showcased in Fig. 5c, to jump away, or even, to grow legs and run away. Elusion could also be character-driven. For example, a character might run or jump toward the object and "snatch it away from under the user's nose" in the last moment to then position it somewhere else. Such elusion mechanisms could be chosen depending on the characters and rules of the presented fictional world.

3.2.3 Step 3: Completing the Illusion. A second cause of disillusion from visuotactile inconsistency arises from the *tactile collision* between any part of the user's body and the visually hidden, but physically present object: The user perceives the collision tactilely but not visually. Such a collision might even make the user tumble and thus constitutes a safety risk. As a hard constraint, the tactile signal is again determinative. Therefore, we complete the illusion by rephysicalizing the object just-in-time, as either hand or a radius around the headset come close to the hidden object (Fig. 6). Rephysicaliation brings the VIRTUALIZED object back in-place so that it matches the pose and shape of its physical counterpart by means of a smooth *rephysicalization mechanism* and then toggles the reality state from VIRTUALIZED to PHYSICAL again.

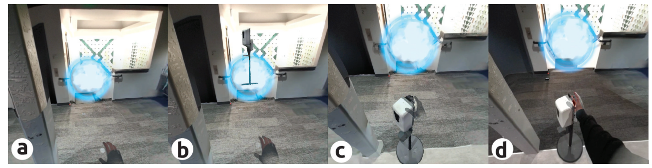

Figure 6: Just-in-time rephysicalization in Daydreaming prevents disillusion from tactile collisions. When a tactile collision is imminent, we rephysicalize the object just-in-time to maintain visuotactile consistency. Diagetic in-betweening maintains the illusion.

We trigger rephysicalization just-in-time, i.e., when a tactile collision is imminent as estimated by distance. Because the rephysicalization target is defined by the physical object, we can only decide on the mechanism, which–analogously to elusiveness–might again include de- and re-materialization, object agency, character-driven animations. A red object guardian outline (Fig. 4d) glows up if the distance to the user becomes critically small before the rephysicalization animation is finished.

3.3 Scene-Responsive Copperfield Episodes

We design Daydreaming episodes so that users have *no evidence against the illusion*. In a Copperfield episode, we aim to advance the experience by *providing evidence in favor of the illusion* through additional touch points with the physical world, similar to the magician who lets the audience stroke the elephant he seemingly made appear "out of thin air".

3.3.1 User Experience of Copperfield Episodes. Consider a user wearing an MR headset entering a room that shows multiple chairs. The user can walk up to these chairs and physically interact with

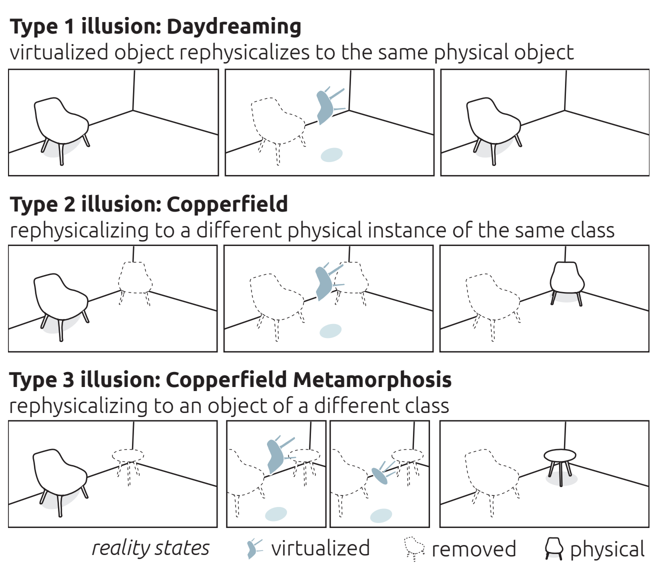

Figure 7: Illusion types. In Daydreaming episodes, a one-to-one relationship between the virtual and the physical object is retained. In Copperfield illusions, a single virtual object animates between multiple visually hidden, physically present objects.

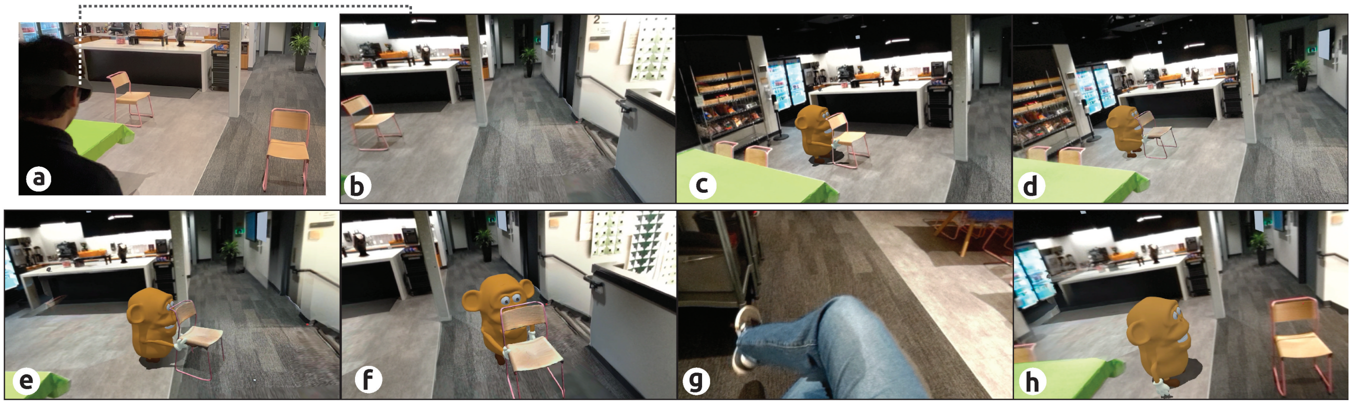

Figure 8: Copperfield rephysicalization. ⓐ A physical scene is observed through a video-passthrough MR headset. It features two physical chairs. ⓑ However, the headset only shows the chair *at source* in PHYSICAL reality state while the chair *at target pose* is in a HIDDEN reality state (i.e., diminished from the scene). Even at high zoom in this figure, the removal process is barely noticeable as visual artifacts are drowned out by other artifacts of the video passthrough system such as motion blur. ⓒ Pre-response, a character walks up to the PHYSICAL chair and reaches out to it to grab it. ⓓ As the character reaches the grab pose, the chair toggles from PHYSICAL to VIRTUALIZED reality state and attaches to the character's hand. ⓔ The character starts carrying the chair away. At this point, both the physical source and the physical target chair are hidden from view. ⓕ Controlling the character's hand through inverse kinematics, the character aligns the VIRTUALIZED chair at the target pose, where the still HIDDEN chair is located. ⓖ The user can touch the VIRTUALIZED chair and even sit on it. ⓗ The system toggles the chair's reality state from VIRTUALIZED to PHYSICAL outside the user's view. The chair at the source pose remains HIDDEN.

them. The user is asked to throw an explosive into the scene by means of a gesture. The explosive destroys some of the chairs while lifting others into the air and tossing them to places distributed across the room. All chairs are still burning a bit. The user walks up to one of the tossed-around but still burning chairs, taking a look at it (*situation* 1). As the user reaches out to touch it, the fire flares up, burning the chair to the ground, thus eluding the user's touch. The user walks up to a different chair tossed somewhere else into the scene by the explosive. The user approaches it and takes a look (*situation* 2). The fire extinguishes. They reach out and find that they can touch the chair despite the fact that it looks slightly virtual. The user takes a seat on the chair. After standing up again, the slightly virtual look is gone. It seems as if the explosive had thrown the physical chair around.

3.3.2 Undecidability of Reality States. In the above scenario, the user considers VIRTUALIZED chairs in *situation* 1 and *situation* 2. In *situation* 1, the VIRTUALIZED chair is placed in physically empty space. In *situation* 2, the VIRTUALIZED chair is placed in the same pose and shape as a HIDDEN physical chair. Assuming the lack of visible artifacts, the user has no information to decide from vision only whether the virtual chair will provide tactile feedback or not. The only way of knowing is by trying to touch it. We hypothesize that undecidability may provide an engaging and immersive experience for the following reason.

As a result of undecidability, there is no sense for users in reasoning about its physical existence, because they simply lack the information that would allow them to decide whether an object is physical or not. Instead, they have to trust the system. From this need to develop trust in the system in order to make sense of the tactile world, we hypothesize immersion increases over time, making users forget what they see is a visuotactile illusion rather than actual physics.

3.3.3 Steps in a Copperfield Episode. Fig. 8 provides an in-depth description of the steps in a Copperfield episode. Initially, all instances of the same object class are HIDDEN except for one PHYSICAL object. A virtual action triggers virtualization of the PHYSICAL object. Now, however, instead of waiting for just-in-time rephysicalization to its source pose, Copperfield employs story-driven *cross-object rephysicalization*, transporting the VIRTUALIZED object to seemingly empty space, however, in fact aligning it with a visually hidden, but physically present object, the user does not yet know about. Once the user approaches the still VIRTUALIZED, but physically aligned object, they find it will not escape from them.

3.3.4 Additional Considerations on Copperfield Rephysicalization.

Pseudo-random rephysicalization. Story-driven rephysicalization in Copperfield is a chance to double down on the illusion and provide faked evidence that virtuality affects physicality. This offers a chance for *pseudo-randomness in the rephysicalization.* Users are conditioned from physics, gaming, and movies that explosives toss around objects at random. Using an animation that looks like a physics simulation but is actually a deterministic process deliberately transporting the object to a defined target position might reinforce the sensation of randomness. The subsequent ability to physically interact with a seemingly randomly placed object might advance the believability of the illusion even further.

Premature and ultimate rephysicalization. A Copperfield episode must deviate from an ongoing and longer-running story-narrated rephysicalization, and instead rephysicalize just-in-time if a user is about to run into a HIDDEN object. Given the user has never seen the HIDDEN targets, this is much more likely to happen than in Daydreaming. Just-in-time blocking the seemingly empty, but physically dangerous space, e.g., through a virtual fire or NPCs might avoid premature rephysicalization. To *ultimately conclude* any Copperfield episode, new instances of the virtual object must enter the scene through different diegetic ways, re-introducing a one-to-one relationship between physical objects and virtual counterparts.

Metamorphosis rephysicalization. The above-presented Copperfield episode hinges on the availability of multiple instances of the same object class in the physical scene. By virtually morphing the VIRTUALIZED object "on its way" (Fig. 7 bottom), we can broaden the applicability of the Copperfield episode for more diverse scenes. *Morphing mechanisms* could range from a literal mesh-deforming morph operation to computationally less demanding operations like shrinking the object mid-air to a scale of zero and growing it back as a different object, or even abstract ways of changing an object's type, e.g., where the character leaves the room with the object of the source type and comes back with an object of the target type. In particular, the latter approaches alleviate the need for complex, potentially object-specific animation work.

Multi-user rephysicalization. In any Copperfield episode, physical objects are HIDDEN before the user sees the scene for the first time. This limits its applicability for single-user home-usage applications because users probably will know their physical space well. However, we imagine the possibility of *multi-user setups* where co-present users manipulate the physical scene, thus making any individual user lose overview of the physicality.

4 ARCHITECTURE AND IMPLEMENTATION

In the following, we show how we obtain a rich digital representation of a space and its objects using our *TwinBuilder* component. Then, we present our spatial computing and shading algorithm which makes use of the produced twins to toggle between object reality states and ensures scene coherence, implemented in the *RealityToggle* component. Finally, we present our *Spielberg* component which makes use of RealityToggle in its application flow to control Daydreaming and Copperfield episodes by starting Scene Responsiveness, ensuring elusiveness, and triggering rephysicalizations while controlling the character and providing user interactivity with objects.

4.1 *TwinBuilder* Component: Co-aligned and Semantically Rich Space and Object Twins

4.1.1 Step 1: Capture Space and Objects in Individual Scans. We obtain a 3D mesh model with textures of the space after moving relevant interactable objects out using Polycam on an iPhone with a LIDAR sensor. We also obtain the models for interactable objects.

4.1.2 Step 2: Integrate and Annotate in a Unified Twin Representation. After cropping and converting the models, we import them into Unity and use our custom Unity plugin (Fig. 9) for further processing. In particular, we integrate objects in space in a single twin by positioning and orienting the objects faithfully. The mesh area navigable by the character is automatically derived based on the mesh faces' relative orientation to the floor plane using Unity.

Afterward, we annotate character and user affordances. We implemented an exemplary set of receptive character affordances (sit, lie, hide, climb) and responsive affordances (drag/push, carry, press for the bi-ped character, absorbable by the Black Hole). Selecting

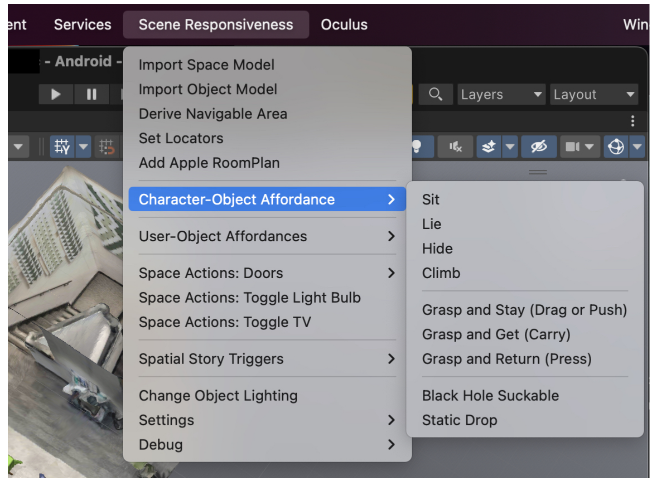

Figure 9: Our custom Unity plugin allows semantic annotation of the scanned space, needed to enable Scene Responsiveness.

them in the plug-in adds the needed Unity components and allows parametrization for the associated situated animation. We implemented summoning, disintegration, and repulsion interactions as responsive user affordances. For responsive affordances, the `RealityToggle` Unity component is automatically added. At this point, the resulting digital scene can be executed like a game on the computer or deployed in *interactive mode* to the headset. Alternatively, in addition, affordances and spatial triggers can be connected in a sequence to compose longer-running `Spielberg` stories for *story mode*. This architecture generalizes easily to new scenes, is cleanly extensible, and coherently fits into the 3D application development process.

The resulting digital twin contains all information needed to render coherent, situated, and responsive MR in the passthrough view. Labels in the figure (Fig. 10) indicate how we use this twin.

4.1.3 Step 3: Co-align Digital Twin with Physical Space. We also specify two easily identifiable and stable floor locations in the twin, e.g., wall-wall-floor corners, used for co-alignment in the headset.

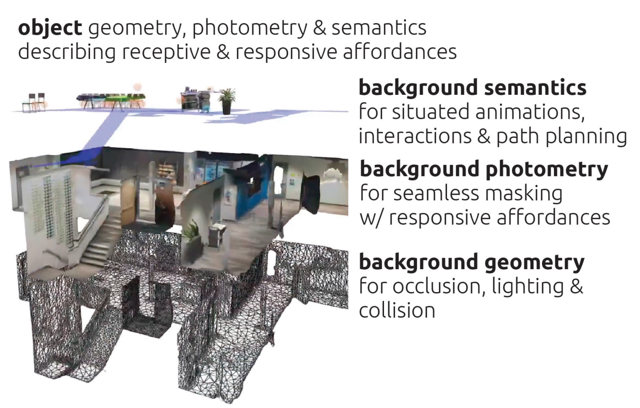

Figure 10: Digital twins provide the geometric, visual, and semantic space and object information, needed for Scene Responsiveness.

4.2 *RealityToggle* Component: Spatial Computing and Shading

RealityToggle is our spatial computing and shading component which enables toggling an object's reality state, i.e., virtualizing it for our core illusion of Scene Responsiveness (Fig. 4), by making use of the previously built twin.

4.2.1 Masking Pipeline. First, we *pose a masking occluder* in the 3D scene (Fig. 11 left). We deliberately do not use a pixel-perfect mask but use a quad as a larger and simpler masking geometry to avoid complicated edges and enable smooth alpha-blending towards the edges. We position the occluder quad on the ray from the headset to the object center, oriented orthogonal to the ray, and tangent with the object's bounding sphere. Heuristically, we set the quad's edge length large enough to fully enclose the object in the conical frustum behind the quad's opaque center area.

Second, we *texture and shade* the masking occluder (Fig. 11 mid). The quad is textured with a stereoscopic render texture, produced from two additional *masking cameras* with the same calibration as the eye cameras, but rendering the background mesh. We implemented a custom *stereoscopic masking shader* that samples the eye-specific render texture at the corresponding screen coordinates of the respective eye camera. We apply a radial two-step gradient texture map for alpha blending to achieve a smooth fade toward the edges. We render the masking quad with standard depth testing and writing, thus integrating it coherently into the scene.

Finally, we composite (Fig. 11 right) the passthrough layer with the rendered quad (`Blend SrcAlpha OneMinusSrcAlpha, One OneMinusSrcAlpha`). The masking cameras render in a black skybox to maintain the passthrough color faithfully.

4.2.2 Scene Coherence in Masked Frustums.

Depth coherence beyond the frustum. Next, we insert the virtual object twin. Because the masking quad is placed in 3D, a frustum emerges behind it. Virtual content inside the frustum gets culled by occlusion, however should render as if nothing changed. Therefore, we make sure that all virtual content, that shall be rendered in front, is captured by the render texture by dynamically manipulating the camera layering. In particular, the virtual counterpart, the character, and any other virtual object are included in the masking render texture.

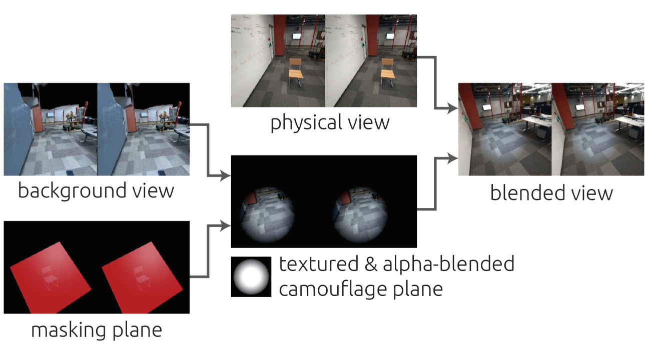

Figure 11: Passthrough compositing pipeline. Brightness increased for illustration. We use an architecture of virtual stereo-cameras with render textures to texture the masking occluder, Please refer to Fig. 13.6 for faithful colors.

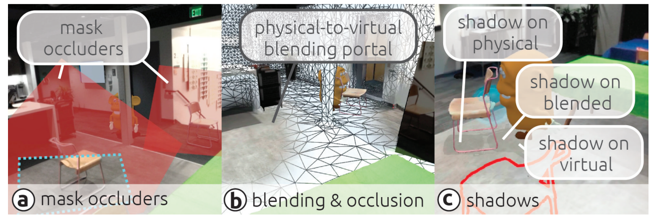

Figure 12: Spatial computing and shading architecture. ⓐ We can pose multiple masks in the scene, even in the same line of sight. ⓑ Dynamically assigning rendering layers, rendering queue positions, depth write, and depth test flags, ensure depth coherence for characters, objects, and hands. ⓒ A custom lighting model ensures that shadows are coherently cast onto physical and virtualized objects and surfaces as well as blended areas.

Lighting coherence for shape responsiveness. In Fig. 3, we have demonstrated shape responsiveness by means of our abstract Black Hole character and its ability to deform and absorb objects. We implemented the mesh deformations efficiently in a *Black Hole surface shader* with a vertex modifier. The use of a surface shader enables adding and even deforming shadows during the mesh deformation (`fullforwardshadows addshadow` in the surface shader pragma).

4.2.3 Scene Coherence in Unmasked Areas. In unmasked areas, virtual content shall be rendered coherently with the physical scene instead. Ensuring both coherence in masked and unmasked areas will elegantly produce the desired coherency in the alpha-blended regions of the view (Fig. 12).

To enable occlusions, we create a second but invisible instance of the background mesh and all the virtual objects with a custom two-pass "Physical" shader that ensures coherent lighting and occlusion. We ensure coherent *occlusion* in a first *fragment shader pass* by rendering the background mesh as a phantom (i.e., rendering early in the queue filling the depth buffer (`ZWrite On`) but without drawing (`Blend Zero One`)

We ensure coherent *lighting* in a second *surface shader pass*, implementing a custom *ShadowReceiverOnPassthrough* lighting model. This lighting model always renders black but redirects inverted attenuation into the alpha channel, thus allowing to blend shadows with the passthrough view (`Blend SrcAlpha OneMinusSrcAlpha, One OneMinusSrcAlpha` with `keepalpha` surface shader pragma). To ensure shadows cast onto the background mesh are also occluded by the background mesh itself, we exploit that shadows can only exist on a surface (ZTest Equal).

We ensure coherent *collision* with standard physics simulation between any virtual objects and the invisible background mesh.

4.2.4 Additional Tools. For illustration in this paper, we implemented a *revelation lens* that can be pulled up on a controller. Used throughout the figures in this paper, it renders as an invisible occluder thus revealing the passthrough layer. *Object guardian* as a safety fallback to just-in-time virtualization is implemented as a red outliner where width scales with by distance between user and object. We implemented a variety of *debug shaders*, e.g., to render the mesh's wireframe.

4.3 *Spielberg* Component: Character and User Environment Interaction

Our event-driven *Spielberg* component takes care of directing the characters, handling user input, and controlling the behavior, pose, shape, and state of the objects involved.

4.3.1 *Character-Environment Interaction.*

Character Model, Rig, and Animations Design. We modeled, rigged, and animated our *Bi-Ped character* from scratch [19] in Blender and used Unity's Mecanim animation system. We switch between in-place animations for NavMesh navigation and tree-based blending with root motion for situated character animations.

Control. We implemented three ways of controlling the character. In *interactive mode*, the user can raycast onto affordances, visualized through spheres, and character ghost previews upon hovering. In *story mode*, the action target is updated automatically, based on a scripted story and spatial triggers. In *telepresence mode*, the action target is updated by the remote actions of a user. Details on the implementation of the underlying state machines for character control can be found in the supplementary material.

4.3.2 User-Environment Interaction. We build on the Quest Hand Tracking and the Oculus Interaction SDK to implement hand tracking, gesture detection, and interaction patterns such as remote object selection and summoning. Hand tracking allowed us to include hand occlusion over virtual content, in particular over masked areas.

4.4 Equipment

Design time equipment. We use Polycam 3.1 on an iPhone 13 Pro for scanning and Blender 3.3.1 for model conversion. Our annotation plugin runs in Unity 2022.1.18f (Oculus Integration SDK v49.0, OVRPlugin 1.81.0 for OpenXR) on both Windows and Mac.

Run time equipment for single-user experiences. From Unity, we deploy the system to a Meta Quest Pro (build v49.0), which offers colored video passthrough. Note that our system works on a standard Quest Pro without the need for jailbreaking or attaching additional custom cameras, because our object masking pipeline does not require raw video frames.

5 PRELIMINARY EVALUATION

Participants and Apparatus. To gain insight into how users perceive the Copperfield illusion, we conducted a user evaluation with 20 participants from our institution. Users wore the Quest Pro and used our app in interactive mode, instructed by the experimenter what to select or do next. We set up an evaluation course, leading through various stations that involved receptive and responsive affordances and ending with the user taking a seat on a previously hidden chair that seemingly has been placed there by the character as part of a Copperfield episode. We adapted the same evaluation course to two spaces: a CORRIDOR space (Fig. 13) and a LIBRARY space (see supplementary material). We performed a Wilcoxon rank-sum test [92] on our between-subjects, post-evaluation Likert questionnaire (Fig. 14).

Rationale. Participants were shortly briefed on the abstract capability of the system to "explore new ways with respect to perception of and interaction between users, virtual characters and their environment".

However, we did not provide specific details on the Scene Responsiveness and Copperfield illusions to prevent tendencies *in favor of the illusion* through social-desirability bias in participant responses, or tendencies *at the expense of the illusion* through attentional bias in participant perception. Participants were fully guided through the evaluation course verbally by the experimenter, gathering qualitative information in a semi-structured fashion and asking for perceptual feedback at specific moments along the course. We debriefed participants at the end of the procedure.

5.1 Procedure and Evaluation Course

Because timing, user agency, user knowledge, and user expectation are pivotal to understanding the expressiveness of our illusion perceptibility evaluation, we describe our procedure in detail. Please refer to Fig. 13 for a visual walk-through.

First phase: On Boarding. ① After putting on the headset, the participant learned the basic ray-casting interactions for hovering and selecting affordance visualizers by the example of a receptive hiding affordance. They were trained to control character navigation by pointing and selecting on the ray-interactable nav mesh on the floor. ② We asked them to sit down on a bench. Using a spatial action trigger, the character automatically took a seat next to them, so they learned about its ability to semantically interact with the world on its own.

Second phase: Scene Responsiveness. In ③, they had physical contact with a book: They were asked to take it off the shelf and read the title out loud. In ④, they were asked to take a seat, point toward the book, and summon it by selecting it. The book was VIRTUALIZED just-in-time and levitated towards the participant's selecting hand. They were asked to read out the author.

Third phase: Copperfield illusion. Because the participant took a seat, step ④ also represents the first physical contact with the chair as the Copperfield source object. ⑤ The participant was asked to stand up and turn around to face the chair, which featured a visualizer for a responsive affordance at its side. The participant was instructed to select the responsive affordance in order to "command the character to grab the chair". The participant maintained visual contact during the character's grasp and thus observed the just-in-time virtualization from a distance of approximately 1m. Note that the character grasps the chair without moving it yet so that virtualization and displacing are *two fully decoupled steps* with a pause in-between until the user selects the next affordance. This allows us to analyze the perceptibility of virtualization before participants were made aware that the chair is suddenly movable and thus *must* be virtual. Before, they might not even notice the virtualization. Once the character grasped the chair, the participant was asked to describe what happened. Then, they were asked to make the character walk away while grabbing the chair, thus now inevitably exposed to the fact that we substantially manipulate the visual signal.

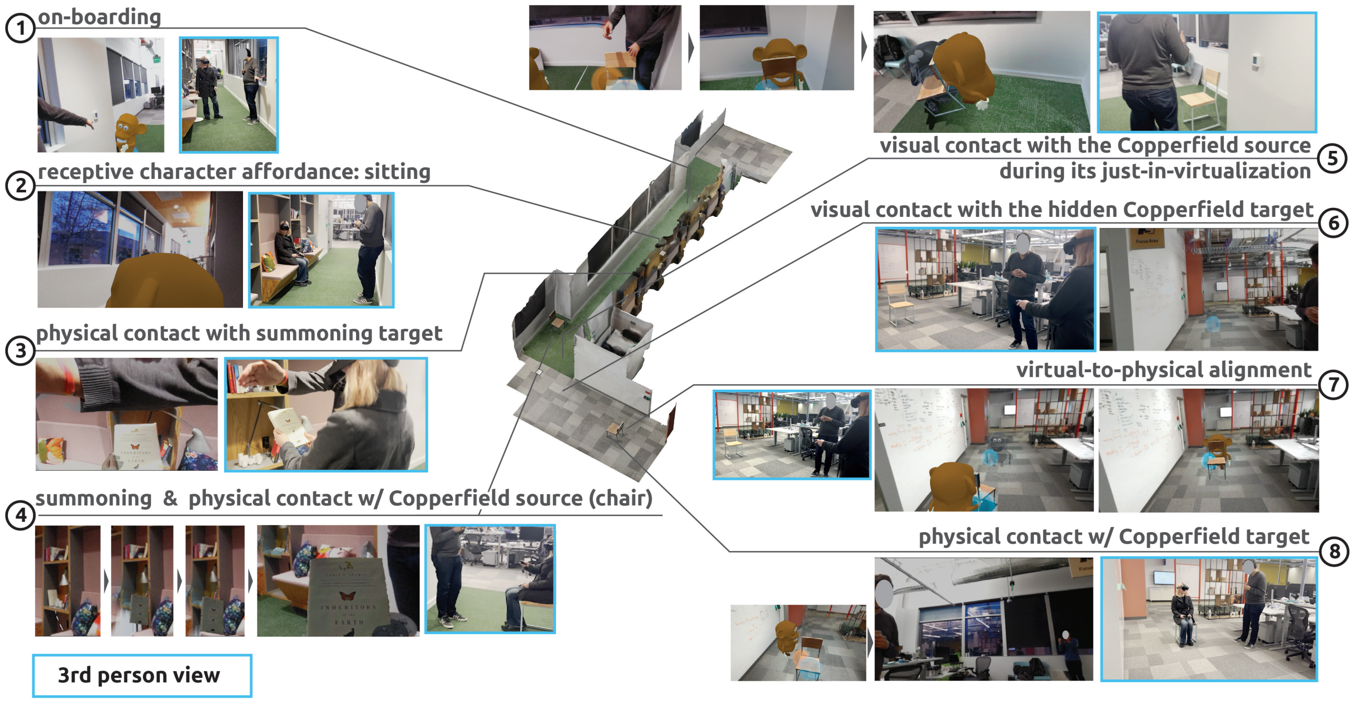

Figure 13: Evaluation course, adapted to our *Corridor* space. The core moments are seen in ⑤ and ⑥: ⑤ The participant faces the chair they just sat on as the character grabs it, thus triggering the *just-in-time virtualization*. Notice how the character runs away with it while the physical chair is masked coherently. ⑥-right: As the participant walks around the corner, all they see is empty space and a transparent blue sphere indicating a selectable drop affordance. At that point in the evaluation course (analogous spot for the Library space included), all participants were asked to describe what they saw. All participants described the blue sphere. None of the 20 participants described the existence of anything suspicious at that point in time yet. Notice how the passthrough view still shows the other person in their view but does not show the physical chair visible in the 3rd person view (⑥-left).

Next ⑥, the participant was asked to walk around the corner. At this time, they first made visual contact with the HIDDEN Copperfield target chair, more precisely they made visual contact with the background they are presented with. Generally, we expected the participant not to suspect the existence of a second chair.

At the position of the Copperfield target ⑦, participants were shown a blue sphere visualizing an affordance which they were instructed to select to make the character drop the chair. The character walked up to the HIDDEN chair and aligned the VIRTUALIZED chair it was carrying with the visually hidden, but physically present chair. Participants were asked to follow the character.

Once the character dropped the chair ⑧, participants were asked to take a seat on the still VIRTUALIZED chair.

5.2 Results and Discussion

5.2.1 Perception of Copperfield Just-in-Time Virtualization. Initiating a Copperfield episode comprises at least three facets, interesting for evaluation.

Our *object masking coherence* question (Fig. 14 ⑤ top) shows balanced responses. 50% of participants agreed slightly, mostly, or strongly that they noticed the masking whereas 50% disagreed. Such a balanced result is quite interesting as it certainly indicates masking fidelity can be improved, yet might hint at the fact that personal and situational factors, such as the current focus of attention or immersion might play a role.

The *object insertion coherence* question (Fig. 14 ⑤ mid) is the only one where answers significantly differed between spaces ($p < 0.1$) as revealed by the Wilcoxon rank-sum test. Indeed, considering the scanned object model of the chair in the Library space, its textures are susceptibly brighter than the physical object that appears in the passthrough view. This hints at the potential for either including a way of adapting the texture brightness in the twin-building procedure or a way for real-time adaptivity of the system.

Finally, the *coherence after displacing* question (Fig. 14 ⑤ bottom) shows that–once the physical object is masked–14 out of 20 participants consider the mixed-reality scene strongly visually coherent, and 5 mostly visually coherent.

5.2.2 Perception of Copperfield Target Masking and Rephysicalization. As seen in Fig. 14 ⑥–⑧, the existence of the physical target chair was surprising to 18 out of 20 participants. A pilot user initially even refused to sit down when asked. We also asked the question "How did you become aware of the physical chair in the final interaction?" offering participants to choose all factors of disillusion. In the following, we only report the answer corresponding to each participant's earliest moment of disillusion. 2 out of 20 participants

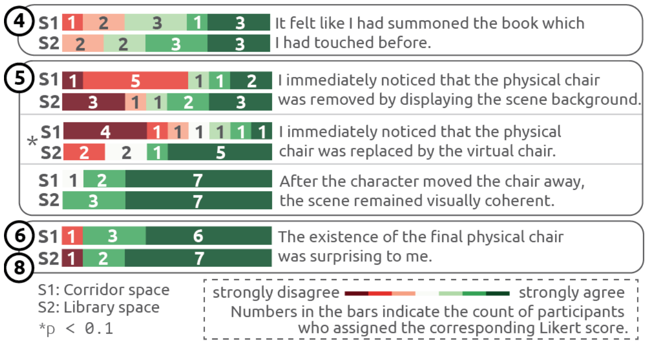

Figure 14: Evaluation of a subset of questions from our post-evaluation questionnaire. Numbers correspond to the phases in Fig. 13.

responded "When I walked up to it, I noticed visual artifacts which gave it away early". 3 out of 20 participants responded "I discovered it through the headset's peripheral gap before the experimenter told me to sit down." 8 out of 20 participants responded "When I walked up to it, but before being asked to sit down, I suspected the existence of the physical chair from the story design." 7 out of 20 participants responded "I was still skeptical after being asked to sit down and only believed the physical chair's existence after touching it or sitting down".

The fact that visual artifacts were selected only twice is promising. In contrast, in the previous paragraph, discussing just-in-time virtualization, 50% of the participants agreed slightly, mostly, or strongly, that they immediately noticed the scene's background was blended in to visually remove the chair. The perceptual difference between the two is observing the *transition from physical to blended scene* versus only observing the *already blended* scene. It seems natural that the eye can pick up abrupt changes easier than finding artifacts in a static signal (similar to the example of Gestalt emergence in James' optical illusion of a Dalmatian, popularized by Gregory [21]). Such a perceptual phenomenon seems to contribute to the convincing results of the Copperfield illusion.

5.2.3 Relevance of Visual Fidelity. The unexpected inflection point in the Copperfield episode hinges on visual fidelity. Interestingly, it seems to be of secondary importance to maintain suspension of disbelief and ensure consistency with the rules of the presented fictional universe in other parts of the experience. In particular, nearly half of the participants expressed unsolicitedly that the character felt "real" (P2, P15), "here" (P2), that they felt a connection to it (P19), that the character seemed aware of the user (P3, P4, P7, P12), that they felt alone after the character left the scene (P18), or similar. These statements indicate presence or connectedness despite a simple character design. In summary, *semantic fidelity*–i.e., awareness of objects and their affordances–drives these experiences, allowing meaningful situated user interactions and character behaviors in the physical scene while *geometric fidelity* ensures virtual content looks as if it is located in the user's space.

6 LIMITATIONS AND OUTLOOK

Increasing visual fidelity. The use of 3D scans entails challenges for the representations' visual fidelity.

From a *static* perspective, while the results of LIDAR-based scanning with today's consumer-grade technology are already astonishing, slight geometric distortions across larger spaces, tesselation artifacts at edges, the lack of finer geometrical structures, or mesh holes due to feature sparsity remain perceivable in rendered surfaces and objects. Thus, the use of representations that offer higher visual fidelity such as Neural Radiance Fields, in particular those that aim to infer missing regions [50, 59], can be of interest. This may become more important as the video-passthrough quality increases in the next generations of headsets.

From a *dynamic* perspective, fidelity suffers from changes to the physical scene occurring between scan and use. Moving any object, or even adding clutter to the scene, leads to misalignment. Thus, employing recent methods for 6DoF object tracking is of interest to maintain a real-time understanding. Similarly, adapting light intensity or balancing might help ensure photometric fidelity. Also, except for hands, our system prototype does not take into account dynamic occluders in front of a visually removed object such as humans crossing the line of sight. Adding body detection can help mitigate this.

Generalizing across spaces and objects. However, a technical long-term vision at play might consider the more abstract problem of generalizing from static twins to procedural scene understanding and reconstruction for real-time twin generation. Here, we used a static yet integrated representation of geometric, photometric, and semantic layers for space and objects. With this paper, we hope to provide another reason for advancing efforts in computer perception to decompose reality into logical constituents, thus potentially even allowing to provide MR experiences that are perceived as passthrough [10, 94], however actually result from generatively re-rendering reality. Apart from this generalization on the *perception* side, it is equally thought-provoking to generalize the *rendering* side through procedurally generating animations of characters.

Applications. However, we believe that Scene Responsiveness can already enable a variety of captivating applications today. As insinuated throughout the paper, situated gaming is a natural fit for Scene Responsiveness. Gameplays that take advantage of Scene Responsiveness could revolve around a wave-like invasion–say, of spiders– which open closed cupboards or press out electrical sockets to enter the scene, and then try to steal physical objects–such as the user's computer to gather intel. Players defend themselves and the object of interest by summoning and throwing objects, blocking the way by enlarging them, closing virtually opened cupboards again, or destroying the cupboards entirely to slow down the invasion. They collect points to make stuffed animals or other objects spring to life and help them in defense. We imagine the possibility of an ecosystem around scanned objects, modeled characters, situated animations, and designed gameplays and see the opportunity for sharing a scanned space and its objects with users in the same space, not only but also for a co-presence multi-player experience.

More broadly, we understand Scene Responsiveness as a general concept with the potential to enable or enhance arbitrary domains of mixed reality. In scene-responsive *telepresence*, activities of remote users can be semantically retargeted onto a virtual avatar interacting with physical objects in local space. This enables avatars to take a seat on a physical chair, even if the chair is pushed under the table and otherwise would not be available for the avatar. A proof-of-concept approach and video, implemented in our system with WebRTC, can be found in the supplementary materials. For *health*, we imagine as the user reaches out to a physical unhealthy chocolate bar, it morphs into something less appealing such as a spider. Or, the chocolate bar grows legs, runs away, morphs into a banana while running, and rephysicalizes at the location of a physical banana. For *learning*, a digital workout coach might use the physically available rowing machine in space to demonstrate correct usage. For *movie entertainment*, 2D content shown in a 2D panel in the headset may semantically affect the physical space, e.g., physical objects in the user's living room start floating when watching a movie situated in outer space. Characters such as a Minion might even step out of the 2D screen and seemingly steal a physical object before stepping back into the frame.

7 CONCLUSION

We presented Scene Responsiveness as a novel concept to increase integration between the virtual and the physical world through high-fidelity illusions. We unfolded the end-to-end illusionary experiences of Daydreaming and Copperfield that maintain visuo-tactile consistency through elusiveness and rephysicalization. Our evaluation with 20 users suggests that our coherence-preserving spatial computing and shading implementation for just-in-time virtualization enables highly believable visuotactile illusions across different spaces. Considering the increasing industry focus on video-passthrough MR, we believe that Scene Responsiveness can not only become a concept for exciting gaming experiences but for the MR field in general.

ACKNOWLEDGMENTS

We thank Hrvoje Benko and Wolf Kienzle for the opportunity to work on this project. We thank Andreas Rene Fender, David Bethge, Michael Nebeling, and Rishi Hazra for discussions. We thank Michael Abrash, Andrew "Boz" Bosworth, Gabriel "Gabe" Aul, and Sean Keller for their feedback. We thank Roko Parać for exploring neural methods of scene reconstruction with us.

REFERENCES

[1] Parastoo Abtahi, Benoit Landry, Jackie Yang, Marco Pavone, Sean Follmer, and James A Landay. 2019. Beyond the Force: Using Quadcopters to Appropriate Objects and the Environment for Haptics in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[2] Apple. 2023. Introducing Apple Vision Pro: Apple's First Spatial Computer. https://www.apple.com/newsroom/2023/06/introducing-apple-vision-pro/

[3] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D Wilson. 2016. Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1968–1979.

[4] Ronald T Azuma. 1997. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments* 6, 4 (1997), 355–385.

[5] Domagoj Baričević, Cha Lee, Matthew Turk, Tobias Höllerer, and Doug A Bowman. 2012. A Hand-held AR Magic Lens with User-Perspective Rendering. In *2012 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 197–206.

[6] Peter Barnum, Yaser Sheikh, Ankur Datta, and Takeo Kanade. 2009. Dynamic Seethroughs: Synthesizing Hidden Views of Moving Objects. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 111–114.

[7] Eric A Bier, Maureen C Stone, Ken Pier, William Buxton, and Tony D DeRose. 1993. ToolGlass and magic lenses: the see-through interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. 73–80.

[8] David E. Breen, Ross T. Whitaker, Eric Rose, and Mihran Tuceryan. 1996. Interactive Occlusion and Automatic Object Placement for Augmented Reality. *Computer Graphics Forum* 15, 3 (1996), 11–22. https://doi.org/10.1111/1467-8659.1530011 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.1530011

[9] Nathalie Bressa, Henrik Korsgaard, Aurelien Tabard, Steven Houben, and Jo Vermeulen. 2022. What's the Situation with Situated Visualization? A Survey and Perspectives on Situatedness. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (1 Jan. 2022), 107–117. https://doi.org/10.1109/TVCG.2021.3114835

[10] Gaurav Chaurasia, Arthur Nieuwoudt, Alexandru-Eugen Ichim, Richard Szeliski, and Alexander Sorkine-Hornung. 2020. Passthrough+: Real-Time Stereoscopic View Synthesis for Mobile Mixed Reality. *Proc. ACM Comput. Graph. Interact. Tech.* 3, 1, Article 7 (may 2020), 17 pages. https://doi.org/10.1145/3384540

[11] 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, 359–366.

[12] Yifei Cheng, Yukang Yan, Xin Yi, Yuanchun Shi, and David Lindlbauer. 2021. SemanticAdapt: Optimization-based Adaptation of Mixed Reality Layouts Leveraging Virtual-Physical Semantic Connections. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 282–297.

[13] Yi Fei Cheng, Hang Yin, Yukang Yan, Jan Gugenheimer, and David Lindlbauer. 2022. Towards Understanding Diminished Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–16.

[14] Subramanian Chidambaram, Sai Swarup Reddy, Matthew Rumple, Ananya Ipsita, Ana Villanueva, Thomas Redick, Wolfgang Stuerzlinger, and Karthik Ramani. 2022. EditAR: A Digital Twin Authoring Environment for Creation of AR/VR and Video Instructions from a Single Demonstration. In *2022 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 326–335.

[15] Robbe Cools, Augusto Esteves, and Adalberto L Simeone. 2022. Blending Spaces: Cross-Reality Interaction Techniques for Object Transitions Between Distinct Virtual and Augmented Realities. In *2022 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 528–537.

[16] Zhi-Chao Dong, Wenming Wu, Zenghao Xu, Qi Sun, Guanjie Yuan, Ligang Liu, and Xiao-Ming Fu. 2021. Tailored Reality: Perception-aware Scene Restructuring for Adaptive VR Navigation. *ACM Transactions on Graphics (TOG)* 40, 5 (2021), 1–15.

[17] Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, et al. 2020. DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 829–843.

[18] Philipp Fleck, Aimée Sousa Calepso, Sebastian Hubenschmid, Michael Sedlmair, and Dieter Schmalstieg. 2022. RagRug : A Toolkit for Situated Analytics. *IEEE Transactions on Visualization and Computer Graphics* (2022). https://doi.org/10.1109/TVCG.2022.3157058

[19] Raymond Gabriel. 2020. 3D Character Creation in Blender. https://www.youtube.com/playlist?list=PL5TzNMWlAlkqm0aVC7teQsI9B8LG6Upf5

[20] Ran Gal, Lior Shapira, Eyal Ofek, and Pushmeet Kohli. 2014. FLARE: Fast Layout for Augmented Reality Applications. In *2014 IEEE international Symposium on Mixed and Augmented Reality*. IEEE, 207–212.

[21] Richard Langton Gregory. 1970. The Intelligent Eye. (1970).

[22] Jens Emil Grønbæk, Ken Pfeuffer, Eduardo Velloso, and Hans Gellersen. 2023. Partially Blended Realities: Aligning Dissimilar Spaces for Distributed Mixed Reality Meetings. (2023).

[23] Jake Guida and Misha Sra. 2020. Augmented Reality World Editor. In *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology*. 1–2.

[24] Sinem Guven and Steven Feiner. 2006. Interaction Techniques for Exploring Historic Sites through Situated Media. In *2006 IEEE Symposium on 3D User Interfaces*. Ieee, 111–118.

[25] Sinem Guven, Steven Feiner, and Ohan Oda. 2006. Mobile Augmented Reality Interaction Techniques for Authoring Situated Media On-Site. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*. 235–236. https://doi.org/10.1109/ISMAR.2006.297821

[26] Bjoern Haefner, Simon Green, Alan Oursland, Daniel Andersen, Michael Goesele, Daniel Cremers, Richard Newcombe, and Thomas Whelan. 2021. Recovering Real-World Reflectance Properties and Shading from HDR Imagery. In *2021 International Conference on 3D Vision (3DV)*. IEEE, 1075–1084.

[27] Jeremy Hartmann, Christian Holz, Eyal Ofek, and Andrew D Wilson. 2019. RealityCheck: Blending Virtual Environments with Situated Physical Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.

[28] Anuruddha Hettiarachchi and Daniel Wigdor. 2016. Annexing Reality: Enabling Opportunistic Use of Everyday Objects as Tangible Proxies in Augmented Reality. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1957–1967.

[29] Tobias Hollerer, Steven Feiner, and John Pavlik. 1999. Situated Documentaries: Embedding Multimedia Presentations in the Real World. In *Digest of Papers. Third International Symposium on Wearable Computers*. IEEE, 79–86.

[30] Brandon Huynh, Adam Ibrahim, Yun Suk Chang, Tobias Höllerer, and John O'Donovan. 2019. User Perception of Situated Product Recommendations in Augmented Reality. *International Journal of Semantic Computing* 13, 03 (2019), 289–310.

[31] Brandon Huynh, Jason Orlosky, and Tobias Höllerer. 2020. Designing a Multitasking Interface for Object-aware AR applications. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 39–40.

[32] Dongsik Jo, Ki-Hong Kim, and Gerard Jounghyun Kim. 2015. SpaceTime: Adaptive Control of the Teleported Avatar for Improved AR Tele-conference Experience. *Computer Animation and Virtual Worlds* 26, 3-4 (2015), 259–269.

[33] Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. 2014. RoomAlive: Magical Experiences Enabled by Scalable, Adaptive Projector-Camera Units. In *Proceedings of the 27th annual ACM symposium on User Interface Software and Technology*. 637–644.

[34] Denis Kalkofen, Erick Mendez, and Dieter Schmalstieg. 2007. Interactive Focus and Context Visualization for Augmented Reality. In *2007 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 191–201.

[35] Denis Kalkofen, Christian Sandor, Sean White, and Dieter Schmalstieg. 2011. *Handbook of Augmented Reality*. Springer, Chapter Visualization Techniques for Augmented Reality.

[36] Denis Kalkofen, Eduardo Veas, Stefanie Zollmann, Markus Steinberger, and Dieter Schmalstieg. 2013. Adaptive Ghosted Views for Augmented Reality. In *2013 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 1–9.

[37] Victor Kaptelinin and Bonnie Nardi. 2012. Affordances in HCI: Toward a Mediated Action Perspective. In *Proceedings of the 2012 CHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 967–976. https://doi.org/10.1145/2207676.2208541

[38] Mohamed Kari, Tobias Grosse-Puppendahl, Luis Falconeri Coelho, Andreas Rene Fender, David Bethge, Reinhard Schütte, and Christian Holz. 2021. TransforMR: Pose-Aware Object Substitution for Composing Alternate Mixed Realities. In *2021 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 69–79.

[39] Hanseob Kim, TaeHyung Kim, Myungho Lee, Gerard Jounghyun Kim, and Jae-In Hwang. 2020. Don't Bother Me: How to Handle Content-Irrelevant Objects in Handheld Augmented Reality. In *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology*. 1–5.

[40] Kangsoo Kim, Mark Billinghurst, Gerd Bruder, Henry Been-Lirn Duh, and Gregory F Welch. 2018. Revisiting Trends in Augmented Reality Research: A Review of the 2nd Decade of ISMAR (2008–2017). *IEEE transactions on visualization and computer graphics* 24, 11 (2018), 2947–2962.

[41] Kangsoo Kim, Divine Maloney, Gerd Bruder, Jeremy N Bailenson, and Gregory F Welch. 2017. The Effects of Virtual Human's Spatial and Behavioral Coherence with Physical Objects on Social Presence in AR. *Computer Animation and Virtual Worlds* 28, 3-4 (2017), e1771.

[42] Georg Klein and David Murray. 2007. Parallel Tracking and Mapping for Small AR workspaces. In *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 225–234.

[43] Ernst Kruijff, J Edward Swan, and Steven Feiner. 2010. Perceptual issues in Augmented Reality Revisited. In *2010 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 3–12.

[44] Wallace S Lages and Doug A Bowman. 2019. Walking with Adaptive Augmented Reality Workspaces: Design and Usage Patterns. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*. 356–366.

[45] Chloe LeGendre, Wan-Chun Ma, Graham Fyffe, John Flynn, Laurent Charbonnel, Jay Busch, and Paul Debevec. 2019. DeepLight: Learning Illumination for Unconstrained Mobile Mixed Reality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5918–5928.

[46] Changyang Li, Wanwan Li, Haikun Huang, and Lap-Fai Yu. 2022. Interactive Augmented Reality Storytelling Guided by Scene Semantics. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–15.

[47] Wei Liang, Xinzhe Yu, Rawan Alghofaili, Yining Lang, and Lap-Fai Yu. 2021. Scene-aware Behavior Synthesis for Virtual Pets in Mixed Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–12.

[48] Chuan-en Lin, Ta Ying Cheng, and Xiaojuan Ma. 2020. Architect: Building Interactive Virtual Experiences from Physical Affordances by Bringing Human-in-the-Loop. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[49] 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*. 1–13.

[50] Hao-Kang Liu, I Shen, Bing-Yu Chen, et al. 2022. NeRF-In: Free-form NeRF Inpainting with RGB-D Priors. *arXiv preprint arXiv:2206.04901* (2022).

[51] Mark A Livingston, J Edward Swan, Joseph L Gabbard, Tobias H Hollerer, Deborah Hix, Simon J Julier, Yohan Baillot, and Dennis Brown. 2003. Resolving Multiple Occluded Layers in Augmented Reality. In *2003 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 56–65.

[52] Julian Looser, Mark Billinghurst, and Andy Cockburn. 2004. Through the Looking Glass: The Use of Lenses as an Interface Tool for Augmented Reality Interfaces. In *Proceedings of the 2nd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*. 204–211.

[53] Julian Looser, Raphael Grasset, and Mark Billinghurst. 2007. A 3D Flexible and Tangible Magic Lens in Augmented Reality. In *2007 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 51–54.

[54] Weizhou Luo, Anke Lehmann, Hjalmar Widengren, and Raimund Dachselt. 2022. Where Should We Put It? Layout and Placement Strategies of Documents in Augmented Reality for Collaborative Sensemaking. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–16.

[55] Carsten Magerkurth, Adrian David Cheok, Regan L Mandryk, and Trond Nilsen. 2005. Pervasive games: bringing computer entertainment back to the real world. *Computers in Entertainment (CIE)* 3, 3 (2005), 4–4.

[56] David Mandl, Kwang Moo Yi, Peter Mohr, Peter M Roth, Pascal Fua, Vincent Lepetit, Dieter Schmalstieg, and Denis Kalkofen. 2017. Learning Lightprobes for Mixed Reality Illumination. In *2017 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 82–89.

[57] Sebastian Marwecki and Patrick Baudisch. 2018. Scenograph: Fitting Real-Walking VR Experiences into Various Tracking Volumes. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 511–520. https://doi.org/10.1145/3242587.3242648

[58] Meta. 2022. Introducing Meta Reality - A Look at the Technologies Necessary to Convincingly Blend the Virtual and Physical Worlds. https://www.meta.com/blog/quest/mixed-reality-definition-passthrough-scene-understanding-spatial-anchors/

[59] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. 2022. SPIn-NeRF: Multiview Segmentation and Perceptual Inpainting with Neural Radiance Fields. *arXiv preprint arXiv:2211.12254* (2022).

[60] Peter Mohr, David Mandl, Markus Tatzgern, Eduardo Veas, Dieter Schmalstieg, and Denis Kalkofen. 2017. Retargeting Video Tutorials Showing Tools with Surface Contact to Augmented Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 6547–6558.

[61] Peter Mohr, Markus Tatzgern, Jens Grubert, Dieter Schmalstieg, and Denis Kalkofen. 2017. Adaptive User Perspective Rendering for Handheld Augmented Reality. In *2017 IEEE Symposium on 3D User Interfaces*. IEEE, 176–181.

[62] Shohei Mori, Okan Erat, Wolfgang Broll, Hideo Saito, Dieter Schmalstieg, and Denis Kalkofen. 2020. InpaintFusion: Incremental RGB-D Inpainting for 3D Scenes. *IEEE Transactions on Visualization and Computer Graphics* 26, 10 (2020), 2994–3007.

[63] Shohei Mori, Jan Herling, Wolfgang Broll, Norihiko Kawai, Hideo Saito, Dieter Schmalstieg, and Denis Kalkofen. 2018. 3D PixMix: Image Inpainting in 3D Environments. In *2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct*. IEEE, 1–2.

[64] Shohei Mori, Sei Ikeda, and Hideo Saito. 2017. A Survey of Diminished Reality: Techniques for Visually Concealing, Eliminating, and Seeing Through Real Objects. *IPSJ Transactions on Computer Vision and Applications* 9, 1 (2017), 1–14.

[65] Shohei Mori, Dieter Schmalstieg, and Denis Kalkofen. 2022. Good Keyframes to Inpaint. *IEEE Transactions on Visualization and Computer Graphics* (2022).

[66] Shohei Mori, Fumihisa Shibata, Asako Kimura, and Hideyuki Tamura. 2015. Efficient Use of Textured 3D Model for Pre-Observation-Based Diminished Reality. In *2015 IEEE International Symposium on Mixed and Augmented Reality Workshops*. IEEE, 32–39.

[67] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *2011 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 127–136.

[68] Gary Ng, Joon Gi Shin, Alexander Plopski, Christian Sandor, and Daniel Saakes. 2018. Situated Game Level Editing in Augmented Reality. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*. 409–418.

[69] Niels Christian Nilsson, André Zenner, Adalberto L Simeone, Donald Degraen, and Florian Daiber. 2021. Haptic Proxies for Virtual Reality: Success Criteria and Taxonomy. In *Proceedings of the 1st Workshop on Everyday Proxy Objects for Virtual Reality.(EPO4VR'21)*.

[70] Benjamin Nuernberger, Eyal Ofek, Hrvoje Benko, and Andrew D Wilson. 2016. SnapToReality: Aligning Augmented Reality to the Real World. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1233–1244.

[71] Xun Qian, Fengming He, Xiyun Hu, Tianyi Wang, Ananya Ipsita, and Karthik Ramani. 2022. Scalar: Authoring Semantically Adaptive Augmented Reality Experiences in Virtual Reality. In *Proceedings of the 2022 CHI Conference on*

Human Factors in Computing Systems. 1–18.

[72] Damien Rompapas, Christian Sandor, Alexander Plopski, Daniel Saakes, Dong Hyeok Yun, Takafumi Taketomi, and Hirokazu Kato. 2018. HoloRoyale: A Large Scale High Fidelity Augmented Reality Game. In *Adjunct Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 163–165.

[73] Menandro Roxas, Tomoki Hori, Taiki Fukiage, Yasuhide Okamoto, and Takeshi Oishi. 2018. Occlusion Handling Using Semantic Segmentation and Visibility-Based Rendering for Mixed Reality. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*. 1–8.

[74] D. Schmalstieg and T. Höllerer. 2016. *Augmented Reality: Principles and Practice*. Addison-Wesley. https://books.google.de/books?id=Y4r-ngEACAAJ

[75] Susanne Schmidt, Oscar Javier Ariza Nunez, and Frank Steinicke. 2019. Blended Agents: Manipulation of Physical Objects within Mixed Reality Environments and Beyond. In *Symposium on Spatial User Interaction*. 1–10.

[76] Lior Shapira and Daniel Freedman. 2016. Reality Skins: Creating Immersive and Tactile Virtual Environments. In *2016 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 115–124.

[77] Jae-eun Shin, Hayun Kim, Callum Parker, Hyung-il Kim, Seoyoung Oh, and Woontack Woo. 2019. Is Any Room Really OK? The Effect of Room Size and Furniture on Presence, Narrative Engagement, and Usability During a Space-Adaptive Augmented Reality Game. In *2019 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 135–144.

[78] Jae-eun Shin, Boram Yoon, Dooyoung Kim, and Woontack Woo. 2021. A User-Oriented Approach to Space-Adaptive Augmentation: The Effects of Spatial Affordance on Narrative Experience in an Augmented Reality Detective Game. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–13.

[79] Adalberto L Simeone, Eduardo Velloso, and Hans Gellersen. 2015. Substitutional Reality: Using the Physical Environment to Design Virtual Reality Experiences. In *Proceedings of the 2015 CHI Conference on Human Factors in Computing Systems*. 3307–3316.

[80] Abbey Singh, Ramanpreet Kaur, Peter Haltner, Matthew Peachey, Mar Gonzalez-Franco, Joseph Malloch, and Derek Reilly. 2021. Story CreatAR: a Toolkit for Spatially-Adaptive Augmented Reality Storytelling. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. IEEE, 713–722.

[81] Misha Sra, Sergio Garrido-Jurado, and Pattie Maes. 2017. Oasis: Procedurally Generated Social Virtual Spaces from 3D Scanned Real Spaces. *IEEE Transactions on Visualization and Computer Graphics* 24, 12 (2017), 3174–3187.

[82] Andrei State, Gentaro Hirota, David T Chen, William F Garrett, and Mark A Livingston. 1996. Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 429–438.

[83] Ryo Suzuki, Hooman Hedayati, Clement Zheng, James L Bohn, Daniel Szafir, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2020. RoomShift: Room-scale Dynamic Haptics for VR with Furniture-moving Swarm Robots. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–11.

[84] Tomu Tahara, Takashi Seno, Gaku Narita, and Tomoya Ishikawa. 2020. Retargetable AR: Context-aware Augmented Reality in Indoor Scenes Based on 3D Scene Graph. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 249–255.

[85] Markus Tatzgern, Raphael Grasset, Eduardo E Veas, Denis Kalkofen, Hartmut Seichter, and Dieter Schmalstieg. 2013. Exploring Distant Objects with Augmented Reality. In *EGVE/EuroVR*. 49–56.

[86] Wen-Jie Tseng, Elise Bonnail, Mark Mcgill, Mohamed Khamis, Eric Lecolinet, Samuel Huron, and Jan Gugenheimer. 2022. The Dark Side of Perceptual Manipulations in Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–15.

[87] Julien Valentin, Adarsh Kowdle, Jonathan T Barron, Neal Wadhwa, Max Dzitsiuk, Michael Schoenberg, Vivek Verma, Ambrus Csaszar, Eric Turner, Ivan Dryanovski, et al. 2018. Depth from Motion for Smartphone AR. *ACM Transactions on Graphics (ToG)* 37, 6 (2018), 1–19.

[88] John Viega, Matthew J Conway, George Williams, and Randy Pausch. 1996. 3D Magic Lenses. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*. 51–58.

[89] David R Walton and Anthony Steed. 2017. Accurate Real-Time Occlusion for Mixed Reality. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. 1–10.

[90] Thomas Whelan, Stefan Leutenegger, Renato Salas-Moreno, Ben Glocker, and Andrew Davison. 2015. ElasticFusion: Dense SLAM without a Pose Graph. Robotics: Science and Systems.

[91] Sean White and Steven Feiner. 2009. SiteLens: Situated Visualization Techniques for Urban Site Visits. In *Proceedings of the 2009 CHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1117–1120. https://doi.org/10.1145/1518701.1518871

[92] Frank Wilcoxon. 1992. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1992).

[93] Wesley Willett, Yvonne Jansen, and Pierre Dragicevic. 2017. Embedded Data Representations. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 461–470. https://doi.org/10.1109/TVCG.2016.2598608

[94] Lei Xiao, Salah Nouri, Joel Hegland, Alberto Garcia Garcia, and Douglas Lanman. 2022. NeuralPassthrough: Learned Real-Time View Synthesis for VR. In *ACM SIGGRAPH 2022 Conference Proceedings* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. Association for Computing Machinery, New York, NY, USA, Article 40, 9 pages. https://doi.org/10.1145/3528233.3530701

[95] Natsuki Yamanobe, Weiwei Wan, Ixchel G. Ramirez-Alpizar, Damien Petit, Tokuo Tsuji, Shuichi Akizuki, Manabu Hashimoto, Kazuyuki Nagata, and Kensuke Harada. 2017. A Brief Review of Affordance in Robotic Manipulation Research. *Advanced Robotics* 31, 19-20 (2017), 1086–1101. https://doi.org/10.1080/01691864.2017.1394912

[96] Jackie Yang, Christian Holz, Eyal Ofek, and Andrew D Wilson. 2019. DreamWalker: Substituting Real-World Walking Experiences with a Virtual Reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 1093–1107.

[97] Hui Ye, Kin Chung Kwan, Wanchao Su, and Hongbo Fu. 2020. ARAnimator: In-situ Character Animation in Mobile AR with User-Defined Motion Gestures. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 83–1.

[98] Ya-Ting Yue, Yong-Liang Yang, Gang Ren, and Wenping Wang. 2017. SceneCtrl: Mixed Reality Enhancement via Efficient Scene Editing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. 427–436.

[99] Yiwei Zhao and Sean Follmer. 2018. A Functional Optimization Based Approach for Continuous 3D Retargeted Touch of Arbitrary, Complex Boundaries in Haptic Virtual Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–12.

[100] Stefanie Zollmann, Tobias Langlotz, Raphael Grasset, Wei Hong Lo, Shohei Mori, and Holger Regenbrecht. 2020. Visualization Techniques in Augmented Reality: A Taxonomy, Methods and Patterns. *IEEE Transactions on Visualization and Computer Graphics* 27, 9 (2020), 3808–3825.