Spacetime: Enabling Fluid Individual and Collaborative Editing in Virtual Reality

Haijun Xia[1], Sebastian Herscher[2], Ken Perlin[2], Daniel Wigdor[1]

[1]University of Toronto
{haijunxia|daniel}@dgp.toronto.edu

[2]New York University
herscher@ nyu.edu, perlin@cs.nyu.edu

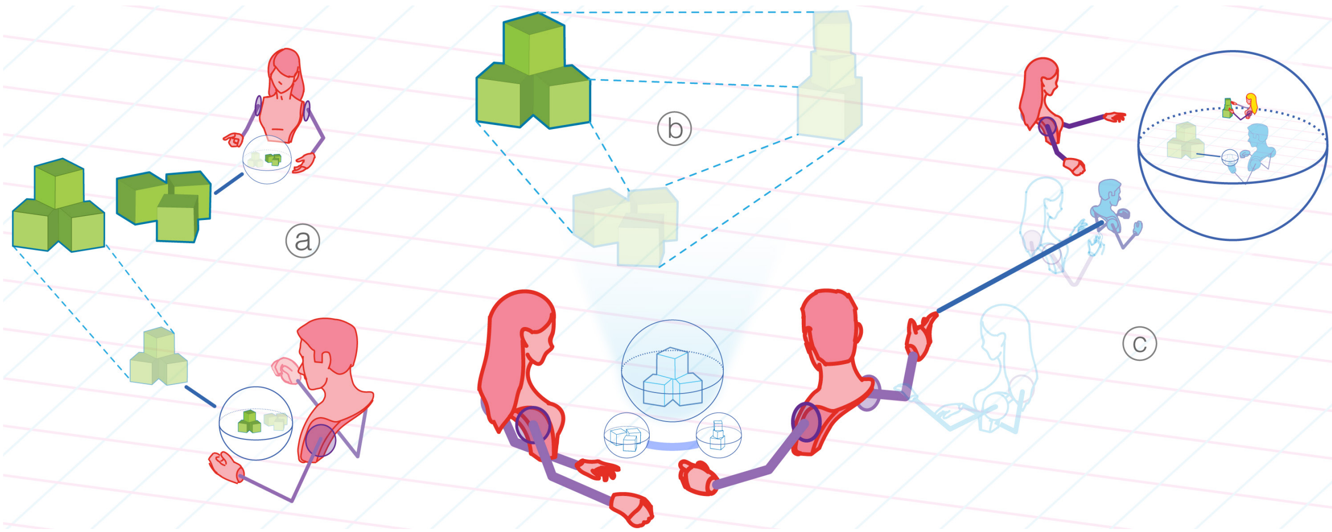

Figure 1 a) Using *Containers* to select and manipulate objects with the spatial context. b) Comparing different designs with *Parallel Objects*, c) Pulling off and Placing a Parallel *Avatar Object* to share views.

ABSTRACT
Virtual Reality enables users to explore content whose physics are only limited by our creativity. Such limitless environments provide us with many opportunities to explore innovative ways to support productivity and collaboration. We present Spacetime, a scene editing tool built from the ground up to explore the novel interaction techniques that empower single user interaction while maintaining fluid multi-user collaboration in immersive virtual environments. We achieve this by introducing three novel interaction concepts: *the Container*, a new interaction primitive that supports a rich set of manipulation and navigation techniques, *Parallel Objects*, which enables parallel manipulation of objects to resolve interaction conflicts and support design workflows, and *Avatar Objects*, which supports interaction among multiple users while maintaining an individual user's agency. Evaluated by professional Virtual Reality designers, Spacetime supports powerful individual interaction and fluid collaborative workflows.

UIST '18, October 14–17, 2018, Berlin, Germany.

ACM ISBN 978-1-4503-5948-1/18/10…$15.00
DOI: https://doi.org/10.1145/3242587.3242597

INTRODUCTION
Virtual reality (VR) has long been an area of exploration in HCI. VR is often used to enable users to experience immersive, 3D environments free from the constraints of the real world. Providing user input in these environments has long been a tenant of VR [59]. Recent technical advances allow for high-fidelity tracking such that users can physically walk around and interact with an environment using their hands [46]. This freedom enables users to transfer their rich experiences within the physical world to enhance and ease their interactions with the digital world. Even with this new-found freedom, users' perception and input are still, however, bounded by the limitations of the physical world, while the breadth of experience available in a VR environment is limited only by the imagination of its creator. Bridging the constrained input capabilities and the breadth of possible experiences has been the focus of much recent work. Significant efforts have focused on improving the techniques used to navigate virtual environments and manipulate objects [12, 29, 38, 51, 52, 57]. Less studied, however, are the higher-level interaction techniques that enable fluid workflows in VR, such as for content creation and creative exploration.

A key problem of content creation is that once a designer places objects into an environment (e.g., pen strokes and 3D models), manipulating multiple objects requires many steps, which can severely impact and lengthen their workflow. As such, selecting and manipulating multiple objects has been a key research challenge in HCI [17, 29, 64, 65]. While research on immersive virtual environments has explored expanding a user's abilities to interact with objects at different scales and distances, most proposed techniques have focused on interacting with a single object [51, 52]. It is unclear how well existing techniques can be applied to workflows that involve interacting with multiple objects.

Prior techniques also fall short on their applicability to multi-user settings. For example, teleportation allows users to quickly navigate an environment by pointing to a location and instantly moving to it [12]. While this may be straightforward for a single user, it becomes jarring when multiple users are co-present, as from their perspectives, another user may suddenly disappear from a conversation or intrude into their view.

The above problem relates to the challenges of collaborative work in immersive virtual environments, where our spatial awareness and social protocols must to be reconstructed, protected, and maintained. However, due to the lack of sensory feedback, natural human-human interaction (e.g., lifting another person into the air) is discouraged due to the risk of motion sickness, thus limiting the users' natural interactions. A multi-user virtual environment also inherits the challenges from Computer-Supported Collaborative Work (CSCW), where an individual user's actions may be further constrained to ensure consistent states of the content and overall coherence, thus severely impacting the individual user's experience.

We thus seek to enable fluid collaborative editing in a VR environment. We achieve this vision by introducing three novel interaction concepts:

- We propose the *Container* as a unifying primitive which objectifies space and time to ground interaction in context. The Container consistently encapsulates a rich set of navigation, selection, and manipulation techniques that can all be accessed via direct physical manipulation to provide fluid interaction with objects and navigation of the environment.
- We further support interaction with *Parallel Objects*. Instead of constraining an object to one location at a time, we allow an object to parallelly exist in multiple states that can be manipulated by multiple users simultaneously. This allows for parallel manipulation of the same object without constraining users' interaction, multiple parallel versions for design comparisons, and collaborative commenting that protects users' authorship.
- Finally, we unify the concepts of user avatar and object into *Avatar Objects*. Instead of treating a user as a separate concept, we treat users as objects with dynamic behaviors that can be directly manipulated like other objects. This allows for consistent and advanced multi-user interaction without adding complexity to the user interface. The combination of Avatar Objects and Parallel Objects further enables rich social interaction without impacting users' agency.

The interweaving of these concepts further leads to a rich set of interactions for powerful individual interaction and fluid collaboration. We demonstrate the interactions within Spacetime, a multi-user scene editing tool. That said, it is our intention that the proposed interaction techniques can be applied to other applications in virtual environments. We also report on the results of an evaluation of these concepts with professional virtual reality application designers.

RELATED WORK

The present work draws on prior literature focused on interaction techniques for direct physical manipulation, interaction within immersive virtual environments, CSCW, and collaboration in immersive virtual environments.

Direct Physical Manipulation

Leveraging commonly held physical manipulation skills to enable rich and intuitive human-computer interaction has long been a research theme [7, 10, 58]. Extensive work has explored rich interaction techniques that employ the hands [12, 62], fingers [38, 60], and graspable objects such as a pen [44, 58]. Built upon observations of natural human behavior [26], recent research has also investigated the complementary roles of hands [11, 51] and different input modalities [29] for bimanual interaction. These projects explored a rich set of compelling interaction techniques for metaphorical physical manipulation and demonstrated the power of direct physical manipulation.

Research has found that of all the commands in graphical user interfaces, users can only reliably agree on a single type of gesture: direct physical manipulation [61]. A common approach to apply gestural interaction to complex functionality is to define an application-specific input vocabulary [2] assisted with gesture teaching system [19]. Alternatively, object-oriented interaction extends the scope of direct physical manipulation by reifying abstract primitives, such as attributes [63] and selection [64] as objects, to serve as the building blocks of an interface that is entirely directly manipulable.

This work follows in this spirit in that it focuses on new interaction primitives, i.e., the Container, Parallel Objects, and Avatar Objects, the direct manipulation of which supports rich and intuitive navigation and interaction within a virtual environment.

Interaction in Immersive Virtual Environments

An immersive virtual environment allows users to navigate and interact with objects without being limited by real world physics [59]. One key challenge, however, is how to employ direct manipulation input, which is constrained by the limits of the human body, to interact with virtual content of different scales and distances. Various input control metaphors such as the eyeball-in-hand, scene-in-hand, the flying vehicle, and ray casting have been proposed to overcome issues with scale and distance [29]. Mackinlay et al. proposed the use of teleportation to enable rapid movement in an immersive environment [38]. Pausch et al. provided users with miniature representations of the content, which could be held by the non-dominant hand as reference [26], so that the dominant hand can reach in to manipulate the miniature, whose effects apply to the object it represents. A World in Miniature (WIM) represents the entire virtual environment, with which users can select new locations for an avatar, by moving the avatar miniature within the WIM [57]. Similarly, Voodoo Dolls enables users to manipulate remote objects of varying sizes [51]. Go-Go interaction uses the metaphor of interactively growing a user's arms to reach distant objects in a virtual environment [52].

Interactive portals have also been proposed to assist with navigation and manipulation between multiple locations in a virtual environment [50]. In the Zooming User Interface, a user can reach into a portal to relocate a remote object into the current view [50]. Kunert et al. enabled user navigation in space and time by using virtual photos of a scene as spatial and temporal references [37].

Research has also explored dynamically changing content to enable context-aware interaction. Kopper et al. proposed automatically changing the scale of content based on users' interests to allow for comfortable content viewing [36]. Zhang et al. further proposed adjusting the representation of content based on a user's scale [66] to provide semantically meaningful visual representations [50].

The present work is complementary to such prior work in that we present an interaction mechanism that encapsulates many of the above techniques and affords new ones, thus enabling fluid and expressive interaction with content.

Computer-Supported Collaborative Work

Significant work has strived to harness the potential of virtual environments to support collaborative activities in domains including document editing [21], video conferencing [28], video editing [45], data analysis [16], and information gathering [43].

In the earlier days, research on CSCW focused on ensuring consistent interaction with objects. For example, Greenberg et al. explored different methods to prevent concurrent manipulation of objects or user conflicts in real-time groupware [22], such as locking, serialization, and the degree of optimism.

Later work has focused on leveraging social protocols [43, 49, 54] in the physical world to avoid and resolve conflicts. Significant work explored their usage in real-world co-located settings, as such environments allow users to directly apply their rich spatial awareness and social knowledge. For example, large tabletops provide a shared 2D surface for collaborative interactions, while allowing face-to-face communication of collocated users. WeSearch enabled a group of users to conduct collaborative web search on a tabletop display [43]. Collocated spatial augmented reality enables users to interact with projected dynamic 3D content in the physical world [1, 33, 37]. Room2Room recreated face-to-face conversations by capturing the color and depth information of a remote user and projecting a life-size virtual copy into a local user's space [49].

Solely relying on social protocols to prevent or resolve conflicts, however, has been shown to be insufficient in many situations [22, 54]. Even in the physical world, conflicts occur when multiple people try to interact with the same objects or when one user's interactions interfere with another's, due to the unawareness of other users' presence or intention. As such, research has explored additional coordination policies in CSCW systems [38, 42]. NetSketch, for example, creates personal offline copies when conflicts occur, which will later be merged with other versions [38].

Instead of applying restrictive coordination policies, we augment the virtual world's physics to maintain a fluid collaboration experience even when social protocols fail to invoke. We enable the coexistence of parallel versions of objects in the shared virtual environment, which can be created implicitly and explicitly, to prevent interference of individual's workflow as well as enable flexible and powerful parallel collaborative exploration of multiple design alternatives.

Collaboration in Immersive Virtual Environments

Key design challenges to support collaboration in virtual environments lie in providing sufficient awareness of other users, enabling flexible and convenient sharing of views and contexts, and protecting users' agency [14, 15, 18, 27].

A common approach to sharing views and context in desktop-based collaborative environments is to provide additional views [18, 20, 66]. This is challenging in an immersive multi-scale collaborative virtual environment, as users dynamically change their own scales and locations. Gutwin and Greenberg suggested that virtual systems should allow users to gather awareness information in ways familiar to the physical world, e.g. through consequential communication, feedthrough, and intentional communication [27]. Fraser et al. explored the use of additional scene views and visualizing collaborators' field-of-view and grasping behaviors to increase awareness [18]. Zhang and Furnas explored using different avatar representations to maintain proper social presence when avatars were at different scales [66].

In the physical world, we are often willing to temporarily share agency with others to collaboratively achieve tasks. Such interaction is strongly discouraged in immersive virtual environments, as it often causes significant motion sickness due to the mismatch between visual and other sensory feedback [53]. While research has begun to explore haptic feedback for multi-user scenarios [38], a common design choice is to disable direct contact among users.

The present work seeks to address the unique challenges found in multi-user interaction in immersive environments by consistently expanding the availability of interaction techniques for general multi-user interaction without increasing the complexity of the system. The result is a set of interaction techniques that support rich multi-user interaction, from view sharing to collaborative design processes, while protecting users' agency.

SPACETIME

To explore the rich multi-user interaction in immersive 3D environments, we implemented Spacetime. Spacetime is a scene editing tool that enables a small group of users (2-3) to collaboratively populate and edit a scene with pre-made 3D models.

As the current implementation of Spacetime uses an Oculus VR headset and Touch controllers, a user can grab an object with either hand. While holding the object with one hand, a user can grab the object with index triggers to scale and rotate it. Releasing the trigger places the object into the scene.

A user can quickly navigate large distances by teleportation. Pushing the joystick forward shoots a ray out from the user's hand, which can intersect with objects in the environment. Releasing the joystick teleports the user to the intersection point. All teleportation motion is stored in a stack, so that the user can teleport back to previous positions by pulling the joystick backward.

Navigation by translating, scaling, or rotating the whole environment is also possible. Pressing either grip button and moving the associated hand translates the space. Pressing both grip buttons and moving both hands scales and rotates the environment. Multiple users can adjust their position in the environment independently. A user can scale up the environment to work on small objects, which will make him appear smaller from other users' perspectives. This allows multiple users to coexist in the same environment at their preferred positions and scales, while maintaining their relative spatial relationships (Figure 2).

Note that the focus of Spacetime was to explore the usage and implications of the proposed interaction concepts rather than on the specific input mechanisms used to achieve the necessary interaction. While the interaction techniques are coupled to Oculus Touch controllers, it is our intention that the interaction modality could be generalized to other input devices including direct gestural input.

SINGLE USER EDITING

One unique challenge of interaction in an immersive virtual environment is that it offers vast space beyond our reach. This creates challenges for the navigation of the environment and interaction with objects. While significant work has explored various navigation and object manipulation techniques, the focus has been on the exploration of physiologically acceptable navigation [40, 57] and the application of direct object manipulation [51, 52]. Less studied, however, is the combination of manipulation and navigation to support higher-level workflows. A traditional design approach of content creation applications focuses on the functionalities, providing discrete and commonly used modes, commands, or tools (e.g. select, copy, teleport). A unique challenge in immersive virtual environment, we have observed, is that users frequently revisit and reuse previous designs through the entire process. Our key insight, thus, is that, ***interaction requires context and creates contexts, spatially and temporally. Users should be able to interact based on the context, which should be easily revisited for reuse and iteration.***

Figure 2. Two users of different scales in Spacetime.

Containers

We unify navigation and object manipulation with Containers. Instead of focusing on discrete operations and commands, we focus on situating interactions in spatial and temporal context. We see Containers as a ***projection of space, similar to how slide projectors project small slides onto larger screens; they can also go back and forth in time, therefore objectifying space and time.*** This offers a new mental model for the interaction and navigation in immersive virtual environments, but also further expands the interaction vocabulary, supporting fluid workflow. Represented as a transparent sphere with a volumetric rendering inside, the direct manipulation of the Containers affords a range of interactions to assist with environment navigation and object manipulation (Figure 3).

- ***Containers store the selection of objects***. A user can create a Container to easily compose complex selections.
- ***Containers act as proxies,*** allowing for the manipulation of multiple objects of arbitrary scales and distances, such as for transformation and copy/paste.
- ***Containers illuminate the spatial context*** of selected objects by visualizing unselected objects in proximity to selected ones.
- ***Containers act as viewports*** to the spaces where selections have been made and the locations a user has been to. As previously created Containers are always accessible by the user, a user can view, update, or reuse a previous selection in their current context, without having to navigate back to previous locations.
- ***Containers act as portals,*** when a user needs to go back to a previous context. The user can teleport into the Container to instantly move to the space where it was created.
- ***Containers reveal temporal context*** as a user spin the Container to go through the history of selected content.
- ***Containers impose structure.*** Smaller Containers can be nested inside larger ones, forming hierarchical structure.

Select with Containers

A Container is created when the index trigger is pressed with the hand in empty space. The Container is then held by the hand (Figure 3a). A ray shoots out from it, so the user can point at an object and use the thumb stick to select it. A proxy of that object is created and summoned into the Container. The user can continue pointing and adding objects to the Container. When multiple objects are selected, their proxies

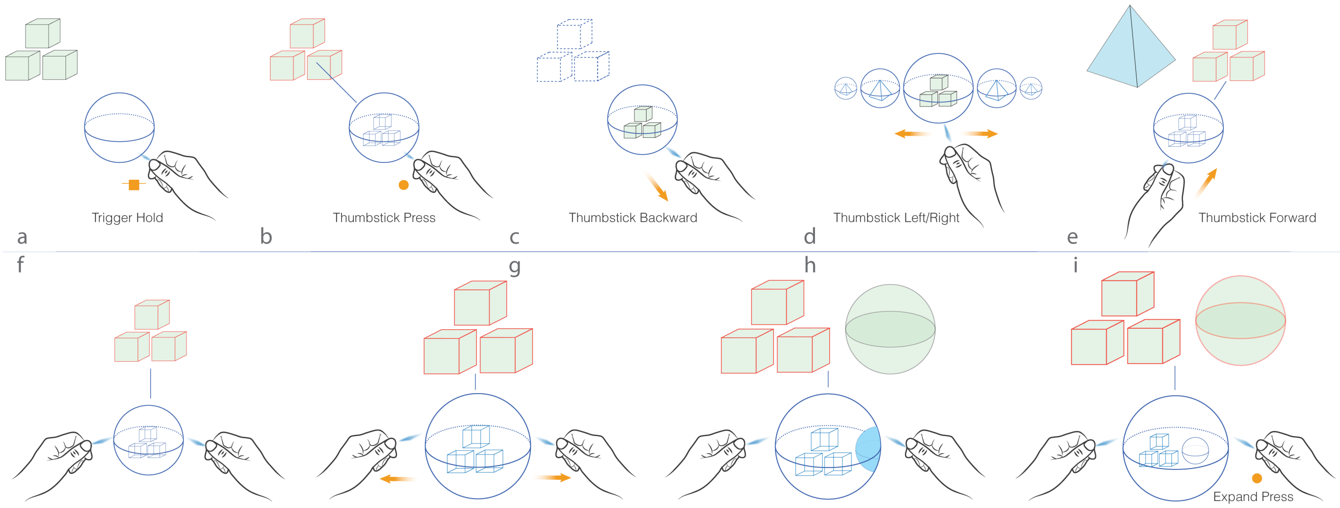

Figure 3. Interaction with Containers. a) Creating a Container, b) Selecting objects with a Container, c) Cutting / Deleting objects from the scene, d) Browsing previously created Containers e) Pasting contained objects into the scene, f) Grabbing the Container with another hand to manipulate the objects using it as proxy, g) Transforming the selected Objects, h) Viewing the surrounding space with Container, i) Including unselected objects in the surrounding space into the Container.

maintain the relative spatial relationships inside the Container. In a crowded setting where many objects need to be selected, the user can quickly brush through the objects while pressing the thumb stick to select all the objects the ray crosses.

Change the Projection

A Container can be used as a proxy; the manipulation of the Container will transform all associated selected objects. While holding the Container with one hand, the user can grab it with their other hand to bimanually translate, scale, and rotate it. The manipulation of the Container is applied to the selected objects proportionally (Figure 3 c-d). Such manipulation can be also seen as a filter applied to the projection, similar to the lenses on physical projectors.

Project or Not Project – Copy/Cut/Paste/Delete Objects

While holding the Container, the user can push the thumb stick forward to paste a new copy of the selection into the space (Figure 3e), or pull the thumb stick backward to cut the selected objects from the scene into the Container (Figure 3c) The Container thus contains the actual objects rather than their proxies. Pulling the thumb stick back again deletes the objects from the Container. Seeing the Container as projection of space, a user can capture a volumetric photo of the space (select and copy), project it again at a different space (paste), stop the projection (cut), and then throw away the slide (delete).

Viewports to the Space

In addition to showing the proxies of selected objects, the Container also visualizes the objects that are in proximity to the selected ones (Figure 3h). Therefore, the user can understand the context of the selected objects by looking into Containers. The user can also add the objects in proximity into the selection with *Expand* command accessible to the hand, which enables for the fast selection of a group of aggregated objects.

Portals to Context

A Container captures the space, thus recording its position. Spacetime allows users to browse previously made Containers by flicking the thumb stick left or right when not holding any object (Figure 3d). The user can view any updates to the space where the Container was created. They can also teleport to the Container to go back to that location. This enable the users to quickly travel among several contexts for design iterations without tedious navigation.

Replay Motion and History

Beyond recording and projecting static objects, the capture and projection of dynamic objects over time is also supported. Similar to recording and replay the motion of objects on film, a user can replay the motion of by spinning a Container. (Figure 4). The Containers also track the history of the contained objects. They can thus be spun to allow users to go back and forth in time to see how the selected objects and the space evolve over time.

Give a Presentation!

One natural and direct extension of the concept of projection is to allow a user to give a 3D presentation by gradually projecting out a list of prepared Containers in sequence, like giving a presentation using a slide projector.

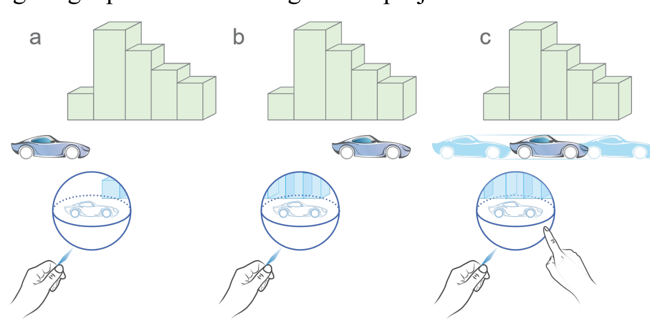

Figure 4. Replay motion with Container. a) Selecting a moving car, b) the car moves in the scene. c) The other hand can spin the Container to go back and forth in time.

Summary of The Container

The Container is a unifying interaction metaphor to support common operations (selection, copy, paste, transformation, etc.) and enable new ones – the Containers preserve spatial and temporal context of the selection to improve awareness and support fast reuse and navigation, enabling fluid design workflows. Because this integration is entirely guided by seeing the Container as the objectification of space and time, the rich functionalities all collapse into one object, which can all be accessed via direct bimanual manipulation.

COLLABORATIVE EDITING

The key challenge in supporting a fluid multi-user system lies in balancing the power each individual user holds, while also maintaining the collaborative experience. CSCW research has explored different ways of enabling fluid collaboration, such as by preventing inconsistent states, maintaining object concurrency, leveraging existing social protocols, and providing additional coordination techniques [22, 33, 42, 54]. These approaches, however, either increase the complexity of a system, as they require that users learn new protocols/rules, or result in restrictive systems, which constrain users' actions and impact user experiences.

Spacetime address this problem with the following insight: ***we are so deeply bounded by the physics of the physical world, where an object can only exist at one state at one moment*, *that we have continued to maintain it in the digital world.*** In typical multi-user applications, an object being manipulated by a user is considered locked to that user. While this can prevent concurrent manipulation of object from different users which may result in inconsistent states, it requires users to wait for each other to finish their actions, therefore, making the multi-user interaction ***sequential***.

Parallel Objects

We propose that the physics of virtual worlds should be augmented to allow multiple versions of one object to coexist at the same time. To do this, virtual environments should allow for the ***parallel*** manipulation of objects, by creating a Parallel Object whenever a conflict occurs, or exploring multiple versions of the object would be helpful.

Creating Parallel Objects to Resolve Conflicts

Parallel Objects can be created when a user wants to manipulate an object that is already being manipulated. Different from the general approach, if an object is already held by a user, a user can pull off a Parallel Object and manipulate it as they wish.

A general challenge of multi-user environments is that one user's action may interfere with the actions of another user. For example, when mapping out a city scene, a user may want to select a block of buildings and duplicate them in another empty area. As the user is about to select the buildings, another user may have already started to manipulate the objects, unaware (e.g. at a smaller scale, occluded, or simply out of the view) that the first user is also interested in manipulating the same set of objects. The first user would have wait for the second user to finish their operation before the first user can start their own operation.

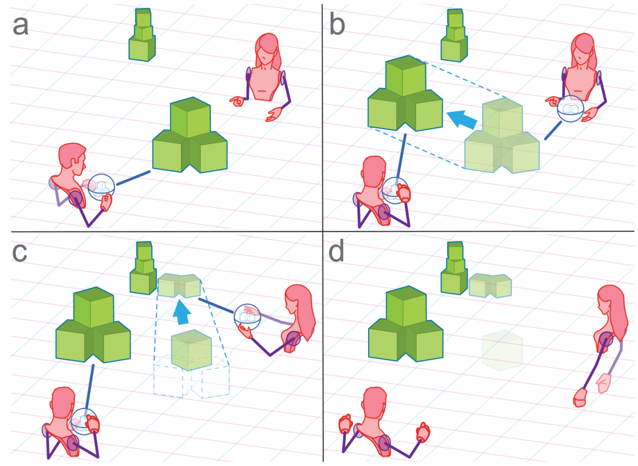

Figure 5. Resolving conflicts with Parallel Objects. a) A user seeks to manipulate objects that are being manipulated, b) Creating Parallel Objects to enable parallel manipulation, c) Manipulating a subset of Parallel Objects, d) Unselected Parallel Objects fade away as no conflict has occurred.

In Spacetime, when a set of objects are being selected and manipulated by a user, the others can still select and manipulate the Parallel Objects with the Container. When a user creates a Container and points to objects they want to select, if the objects are being manipulated by others, the system automatically creates Parallel Objects of the manipulated ones to facilitate the selection. The selected Parallel Objects will be preserved as the user's selection confirms the design conflicts. The unselected Parallel Objects will gradually disappear as no conflicts have been established. This enables the users to express their intentions in parallel, without waiting for others to finish their tasks.

For a set of Parallel Objects, the chosen version is shown at full opacity, while the rest of objects in the set are shown in 30% opacity in our system to reduce their visual impact.

Creating Parallel Objects for Design Exploration

When setting up a scene with models, users often lay out the same set of objects in different ways to compare different design choices. In Spacetime, a user can explicitly create several sets of Parallel Objects to compare different designs.

The user can create Parallel Containers, which contain Parallel Objects of the selected objects. When holding a Container, a user can press the *Parallel* command on the controller to create the Parallel Objects of the contained objects at the same position as the original objects. While the user continues to manipulate the original objects, other users can simultaneously manipulate the parallel ones to experiment with different designs.

Switching Between and Deleting Parallel Objects

When resolving conflicts and comparing design choices, it is important to visualize different versions in a scene one by one to see which one fits the entire scene, or visualize all the options at once to compare them against each other. Spacetime enables both workflow by adjusting the opacity – the projection strength – of the Parallel Objects.

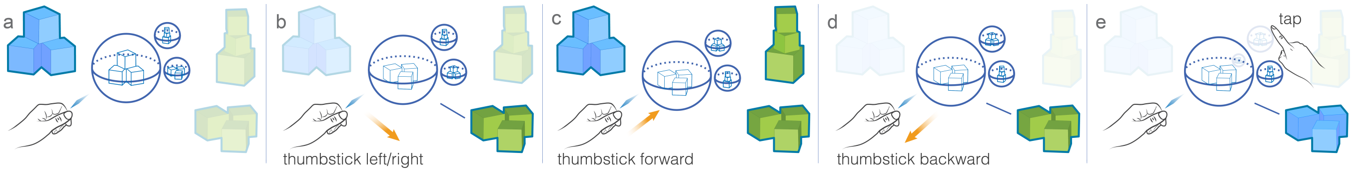

Figure 6. Manipulating Parallel Containers. a) Parallel Containers, b) Switching parallel version, c) Highlighting all the versions, d) Hiding all the versions except the chosen one, e) Merging a parallel version into the chosen one.

By default, Spacetime only shows the chosen version at full opacity. When Parallel Containers are created, they are arranged around the main one, like a round-tray slide projector (Figure 6). The user can flick the thumb stick left or right to switch the main Container and thus the projection into space. When switched, the objects smoothly animate between the parallel configurations. The user can quickly delete different versions of the Parallel Objects one by one by flicking the thumb stick back. If the Parallel Objects being removed are the chosen ones, the system automatically chooses the next version.

When holding a Parallel Container, the user can push and hold the thumb stick forward to gradually increase the projection strength to project all parallel versions into the scene or pull backward to hide all the unchosen versions. This allows the user to compare the different versions when viewing their full visual style or see how one fits in the entire scene. A common approach in existing applications to support such comparisons is to make several copies, which clutters a scene. Furthermore, these duplicated objects are independent, requiring users to manually manage them. Parallel Objects, however, allow users to easily change the ways of comparison and preserve the versions they are interested in and revisit them anytime in the design process.

Merging Parallel Objects

Another advantage Parallel Objects afford is the ability to allow different users to work on different aspects of a design and then blend the visual style. For example, while one user is configuring the layout of an object, another user can simultaneously adjust the color of the parallel version. The user can choose to add the style of the parallel version to the current version by tapping on a Parallel Container. The Parallel Container then collapses into the main one, merging the updated style. Conflicted attributes will be prompted and resolved by explicitly choosing from the conflicted attributes.

Summary of Parallel Objects

Different from traditional git-like conflict management whose goal is to always ensure one shared version [38], Parallel Objects enables multiple users to dynamically create multiple shared versions to collaboratively configure, compare, and merge different designs of the same objects without interfering each other's workflow.

Avatar Objects

One key characteristic of collaboration in the physical world is that collaborators are willing to temporarily cede agency to collaboratively achieve tasks. For example, one person can lift another person in the air to get objects beyond their reach or grab another person's arm to get their attention and then lead them to another place. Despite these interactions are natural in the physical world, they are extremely risky in virtual reality, as they will suddenly change others' views and cause motion sickness due to the lack of haptic and proprioceptive feedback. As such, the manipulation of other users strongly discouraged in virtual reality applications. Our key design goal, on the other hand, is to ***permit such manipulation but prevent its negative impact***.

We see avatars as objects with dynamic behaviors, just like common objects in real world, which are directly manipulable. Spacetime ***allows for the direct manipulation of avatars but protect a manipulated user's agency by decoupling and redirecting the effects of the manipulation to their Parallel Avatars.*** This brings back the rich interaction we are familiar with the real world into the virtual environment and enables a fluid collaborative experience.

Creating Parallel Avatars

While objects can be temporarily held and owned by other users, an avatar is permanently owned by its host. Therefore, instead of directly manipulating an original avatar, a user can pull off a Parallel Avatar of the original. The Parallel Avatar can be translated, scaled, or rotated to suggest a different view without affecting the original avatar. When a Parallel Avatar is created from an original, an unobtrusive vibration – a virtual shoulder tap – is generated on the controller of the grabbed side to notify the user that a parallel version of has been created by another user.

View Sharing

A user can switch to the parallel version of themselves by pointing at the parallel version and teleporting to it. This lets the user easily experience the views that the other users have created (Figure 7). A user can also record the manipulation of a a Parallel Avatar, defining a continuous viewing path.

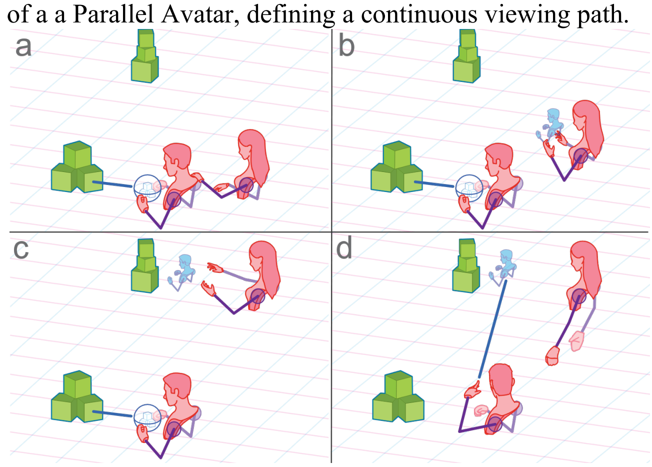

Figure 7 Parallel Avatars a) Tapping on the shoulder, b) Pulling off and resizing a Parallel Avtar, c) Placing a Parallel Avatar, d) Teleporting to the Parallel Avatar to get the view.

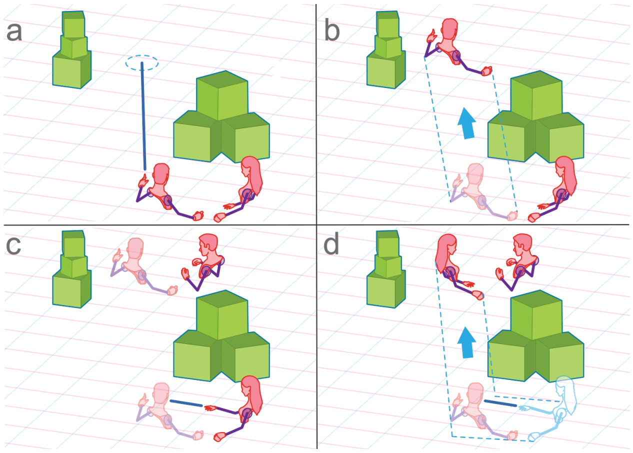

Figure 8 Teleportation. a) A user teleports, b) leaving motion lines and two Parallel Avatars. c) The other user teleport to a Parallel Avatar d) to reuse the teleportation.

Teleportation

In most systems, teleportation is an instant action, which causes jarring experiences in multi-user scenarios. In Spacetime, each teleportation creates two Parallel Avatars at both ends of the teleportation, with motion lines connecting them. This informs the others of the action a user has taken. Other users can teleport to the beginning Parallel Avatar and then teleport automatically to the location of the end Parallel Avatar, reusing the teleportation operation. As the user is not aware of destination of the teleportation, Spacetime uses a blink teleportation, where the screen fades to black and back, to reduce motion sickness.

Dynamic Parallel Avatars

Parallel Avatars created for view sharing and teleportation are designed to be static to preserve the views. A fully parallel version of an avatar, however, should not only share the appearance but also the dynamic behavior.

A user can create a dynamic Parallel Avatar of any user with both the body postures and user actions synchronized. As such, any object the original avatar creates in his own scene will automatically have a parallel version. Multiple dynamic Parallel Avatars can be created and adjusted to different sizes to reuse one user's action at different scales.

For example, when collaborating on a multi-level game, two users can first discuss the style and design. After reaching an agreement, they can branch off to work on different levels of the game. Before one leaves the scene, they can select and store the other avatars in a Container just like any other object. The user can then travel to a new scene and project the Parallel Avatar into the environment.

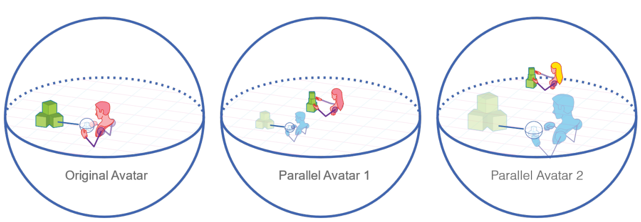

Figure 9 Dynamic Parallel Avatars. Avatars' poster and action are synchronized with respect to their scales.

In addition to reusing the content created by others, the Parallel Avatar enables the users to see the progress of collaborators in context and in real-time while content is being created. This improves awareness and allows for timely discussion, avoiding excessive iterations on the final results.

Summary of Avatar Objects

The Avatar Objects allow users to easily share the views with other without breaking their workflow. Note that, this is done by consistently extending the Parallel Objects beyond common objects to the avatar. The result is a rich set of interaction techniques without introducing any additional user interface components.

EXPERT EVALUATION

We conducted an expert evaluation to gain feedback about the effectiveness and usefulness of the proposed interaction techniques. We are particularly interested in whether the interaction concepts can afford continuous and fluid individual and collaborative workflows.

Participants

Six professional Virtual Reality developers (aged 22 to 41) were recruited to evaluate the interaction techniques. Each participant has more than 3 years' experience developing Virtual Reality content. All except P4 had developed and published virtual reality applications. Participants were compensated $50 for an 80-minute session.

Apparatus

Spacetime was implemented using Unity with Oculus Virtual Reality SDK on desktop computers with Nvidia GTX 1060. We used Oculus Rift CV1 with two Oculus Touch controllers tracked by two Oculus Sensors for both computers. The synchronization of the two setups was performed using Photon Unity Networking SDK.

Procedure

The expert review session consisted of the following stages.

Introduction and Training (15-20 minutes)

Participants were first given an introduction to the study. They then joined the experimenter in a office scene to begin the training. The experimenter described the basic manipulation and navigation techniques, the concept and interaction of the Container, Parallel Objects, and Avatar Objects. During training, the experimenter described the interactions verbally and asked the participants to perform the actions. The experimenter would demonstrate an interaction if a participant had difficulty replicating it. The training continued until the participant could interact with the system without help.

Exercise and Exploration (30-40 minutes)

The experimenter and the participants then switched to another scene to collaborate on a task to populate an empty city map with 3D models (cars, buildings, and trees). After completing the task, participants were introduced to additional advanced functionalities that had not been demonstrated during the study but were included in the post-task interview.

Questionnaire and Interview (15-20 minutes)

The study concluded with post-task questionnaire and interview. The questionnaire consisted of 7-point Likert scale questions (1 – Strongly disagree, 7 – Strongly agree) to collect participants' feedback on both the usefulness and usability of the techniques. The experts were then interviewed with open-ended questions to further gain their feedback on how well the system supported the individual and collaborative workflow and how they foresaw the system supporting their workflows in virtual reality.

Results

All participants successfully completed the tasks and responded positively to the system. "*I want it now. This is something that I want to spend all day in*." (P2)

Containers

Participants responded positively to the Container as an interaction primitive for navigation and object manipulation. They found using the Container to select and manipulate objects intuitive (4/6 strongly agree, 2/6 agree). The Container allowed them to manipulate objects "*in a relaxing way"* (P4), as *"if you want to work in VR for a while, you want to be able to do things in the easiest possible way"* (P6). Being able to "*just sit there and select the stuff, and then grab with another hand and use the proxy to easily move them around is fundamental [to the experience]"* (P2). Participants found it beneficial over existing approaches as "*you don't have to move objects one by one"* (P3), or "*first get in front of the objects, find the selection tool, select, sometimes you have to move around to see if you have missed anything, and then switch mode, and then manipulate them"* (P5). The Container, on the contrary, was found to be "*really fast and fluid"* (P2), and acted as a viewport, "*the little hologram" (P2)* allows users to quickly know "*what's selected and what's not"* (P3).

The Container as the projection of space and time was reported as easy to understand (2/6 strongly agree, 3/6 agree, 1/6 neutral). All participants, except P1, found the projection concept helpful for them to "*connect the features"* (P2), and see "*the Container more as a viewport and projection rather than just the selection"* (P6). Using the Container as *a space anchor* (P2) to quickly teleport back is strongly favored by participants as *"it lets you directly go back to the things you have worked on."* (P4), as *"that's what you do most of the times"* (P3) Participants also found the projection concept "*resonated very well with the Parallel Objects"* (P5). P1, who rated neutral, however, didn't "*buy it for the copy paste stuff, but I think it makes total sense for the Parallel Objects*."

Parallel Objects

Participants respond positively to the utilities of Parallel Objects. All participants found the Parallel Objects allowed them to easily create and compare different designs (3/3 strongly agree, 3/3 agree) and easily discuss the design with their collaborators (3/6 strongly agree, 3/6 agree). Parallel Objects were also found useful in supporting individual workflows. As P5 noted, "*It's completely useful. Because when you are designing a whole bunch of stuff, you want to review things. And I like the idea that you can switch projection to choose the one you want to leave in the scene."*

Parallel Objects were also found to be beneficial for social interaction, "*social interaction is a really tricking thing in VR. you want to have as much personal control of what you are doing, while also giving the other people the ability to interact with the things they want to"* (P5). Participants found Parallel Objects "*resolve[d] the conflicts in a smart way"* (P5), and "*instead of blocking people, the system lets the conflicts happen so that they can do whatever they want ….. they are aware of the conflicts, and they can discuss to resolve them"* (P4). The advantage of the coexistence of the conflicted versions "*immediately let my collaborator know what I'm thinking and then discuss, rather than waiting for him to finish, and I come in to make changes, which by the way also destroys his design, it's just a lot of friction and redundant work"* (P3).

Higher level workflows were also found to be facilitated by Parallel Objects. As found by P2, being *"able to work off each other's schedule; each person can work on each other's model without blocking them; each person can inspire one another, is the most beneficial part to everyone"*. P4 found working on different aspects of the model in parallel, discuss, and combine them "*brings so much to the whole production"*.

Notably, using Parallel Objects for communication was a recurring topic amongst participants. Several participants found that Parallel Objects would be very useful to facilitate the communication and presentation to clients.

P2: "*You can almost make three versions of each. And then when you are doing a Q&A, you can mix and match the three version of each, and ask them which one they like."*

P5*: "Let's say you are meeting a client, you want to be like, hey, here are my three options, you want to be able to easily and quickly and quickly navigate to each option. And the clients can come in to make changes to things."*

P6 found the Parallel Objects "*similar to the layers in Photoshop"* and noted it "*absolutely should be a standard feature in VR. In Photoshop, I use multiple layers to save the different versions of things and toggle the layer on and off to see different designs. Anything that lets you explore different versions will be super helpful and necessary."*

Avatar Objects

Participants found pulling off and place Parallel Avatars to be useful to inform their collaborators where to go (2/6 strongly agree, 3/6 agree, 1/6 neutral) and alerted them as to where their collaborator wanted them to go (strongly agree 3/6, agree 3/6), compared to the alternative such as, "*do you see where I'm standing, come over, stand here, make yourself smaller, face that direction."* (P2). P1, who was neutral, expressed his concern that the "*grabbing operation is a bit too much, especially in modern times you don't really touch people, but knowing where to go is helpful, I prefer to use the Container to select and paste them into the scene and adjust the view for them."*

Being able to share views was also found useful to facilitate the communication especially for 3D content, "*where you need to guide the users to look at things from certain perspectives*" (P1) or demonstrate "*camera movement*" (P6). P3 found that it was useful to directly grab avatars to assign tasks "*when you get project supervisors, modelers, and animators. It's gonna be very very intuitive like hey, we finish these models here, you are gonna animate this one, and you are gonna animate that one. Depending on what role you have in your production, you are gonna use this very differently. I just see it in my head, very clearly, how it could integrate into production pipeline*".

Usability

All participants found the system overall was easy to use (3/6 strongly agree, 3/6 agree) and easy to learn (2/6 strongly agree, 4/6 agree). However, P3 and P6 reported they were a bit overwhelmed by the introduced features. P1 found rotating the world causing motion sickness for him and suggested either removing rotating the world or rotating and scaling in discrete steps to prevent motion sickness.

Summary

The results of the expert evaluation show that the integration of three concepts enables consistent interaction techniques to support powerful individual functionalities as well as fluid and parallel collaborative workflow.

DISCUSSION AND FUTURE WORK

Container, Sphere, and Volumetric Display

We have focused on exploring the novel interaction techniques afforded by objectifying space and time as Containers. While users should have the flexibility to use other 3D shapes to represent a selected space (e.g. cube, cone), a spherical representation was chosen to connect with prior work that explored interacting with spherical multitouch [5] or volumetric displays [22, 24]. On-surface multi-touch gestures and in-air hand gestures can not only eliminate the reliance on hand held controllers, but also further expand the Container's functionalities.

Parallel Objects vs Instancing

Sutherland pioneered the concept of instancing [58], where with graphical elements that are interconnected, the change of the master object is automatically applied to all the instanced objects. Parallel Objects are orthogonal to this in that they are fundamentally not about the relationships between multiple objects, but rather the multiple states of one object. To use the analogy of inheritance from computer programming, instancing allows the change of an inherited valued to propagate to all instanced objects. Parallel states, however, are not about inherence at all - they depict all the different sets of values that an object can have.

Interaction Between Parallel Objects

Future work will focus on the implications and extensions of Parallel Objects. For example, Parallel Objects can collapse into one, where consistent states are automatically merged, and inconsistent ones are resolved by the users. Comparison of different designs also be further facilitated by broadcasting a certain attribute of one parallel to all the parallel states to eliminate the factor from the comparison.

Landing on One or Keeping Them All There?

Parallel Objects allow users to compare different designs to choose the suitable one. But is it necessary to require the user to land on one final version? In the future, we will explore the co-existence of multiple design variations. For example, when making a video game with different levels, depending on players' skills, a user may create a scene with parallel versions and assign each version to different groups of users. The Parallel Object may also stay in Quantum states, so that different designs are chosen with probabilities. This may add dynamics and randomness in a generative design setting.

From Multi-User to Crowd

We explore the interaction techniques with the emphasis on supporting intimate collaborative sessions among 2-3 users. One future direction is to scale interaction to a crowd of users. This will allow us to evaluate the interaction techniques in a more realistic and complex setting and to explore interaction techniques that support the rich social interaction of a large crowd. For example, an owner of a scene may delegate tasks to a crowd using Parallel Avatar Objects; he can create Parallel Dimensions, where groups of users can work on the same scope without physically being co-located in the same location.

From Virtual Reality to Mixed Reality

We chose VR for the present exploration as it posed the least constrains. However, we believe the core interaction concepts can be applied to Mixed Reality as well. For example, to account for the potential motion sickness of moving other avatars and their users' view, we enable users to directly manipulate the parallel avatars to suggest alternative views to others. In an augmented reality setting, the risk of motion sickness is gone, but the risk of breaking others' workflows remains. The Parallel Avatar, therefore, could be useful to share or suggest context while protecting the social coherence. Instead of teleportation which is not feasible in the physical world, the system can generate navigation path to guide the user to the suggested view.

CONCLUSION

We have presented Spacetime, a scene editing tool, built from the ground up to explore three novel interaction concepts: the *Container,* which objectifies space and time to situate interaction in context; *Parallel Objects*, which reduces the friction of and enables powerful parallel workflows for multi-user interaction; and *Avatar Objects*, which support natural social interaction while preserving an individual user's sense of agency.

We demonstrated how simple alterations of the fundamental conventions could lead to new perspectives of context, objects, and users in the environment, each of which results in a set of novel and powerful interaction techniques. The interweaving of these concepts further establishes a new foundation that supports powerful individual interaction and fluid collaboration in immersive virtual environments.

RIGHTS FOR FIGURES

ACKNOWLEDGEMENT

We would like to thank members of DGP and FRL for their support and the reviewers for the valuable feedback. Special thanks to Peter Hamilton for the discussion and Chris De Paoli for support of illustrations.

REFERENCES

1. Maneesh Agrawala, Andrew C. Beers, Ian McDowall, Bernd Fröhlich, Mark Bolas, and Pat Hanrahan. 1997. The two-user Responsive Workbench: support for collaboration through individual views of a shared space. In Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH '97). ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 327-332. DOI: http://dx.doi.org/10.1145/258734.258875
2. Thomas Baudel and Michel Beaudouin-Lafon. 1993. Charade: remote control of objects using free-hand gestures. Commun. ACM 36, 7 (July 1993), 28-35. DOI: http://dx.doi.org/10.1145/159544.159562
3. Steve Benford, Chris Greenhalgh, Tom Rodden, and James Pycock. 2001. Collaborative virtual environments. Commun. ACM 44, 7 (July 2001), 79-85. DOI: http://dx.doi.org/10.1145/379300.379322
4. Hrvoje Benko, Edward W Ishak, and Steven Feiner. 2004. Collaborative mixed reality visualization of an archaeological excavation. In Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on. IEEE, 132–140.
5. Hrvoje Benko, Andrew D. Wilson, and Ravin Balakrishnan. 2008. Sphere: multi-touch interactions on a spherical display. In Proceedings of the 21st annual ACM symposium on User interface software and technology (UIST '08). ACM, New York, NY, USA, 77-86. DOI: https://doi.org/10.1145/1449715.1449729
6. Anastasia Bezerianos and Ravin Balakrishnan. 2005. The vacuum: facilitating the manipulation of distant objects. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05), 361-370. http://dx.doi.org/10.1145/1054972.1055023.
7. Eric A. Bier, Maureen C. Stone, Ken Pier, Ken Fishkin, Thomas Baudel, Matt Conway, William Buxton, and Tony DeRose. 1994. Toolglass and magic lenses: the see-through interface. In Conference Companion on Human Factors in Computing Systems (CHI '94), Catherine Plaisant (Ed.). ACM, New York, NY, USA, 445-446. DOI: http://dx.doi.org/10.1145/259963.260447
8. Eric A. Bier and Steven Freeman. 1991. MMM: a user interface architecture for shared editors on a single screen. In Proceedings of the 4th annual ACM symposium on User interface software and technology (UIST '91). ACM, New York, NY, USA, 79-86. DOI: http://dx.doi.org/10.1145/120782.120791
9. Mark Billinghurst and Hirokazu Kato. 2002. Collaborative Augmented Reality. Commun. ACM 45, 7 (July 2002), 64–70. DOI: http://dx.doi.org/10.1145/514236.514265
10. Richard A. Bolt. 1980. “Put-that-there”: Voice and gesture at the graphics interface. In Proceedings of the 7th annual conference on Computer graphics and interactive techniques (SIGGRAPH '80). ACM, New York, NY, USA, 262-270. DOI: http://dx.doi.org/10.1145/800250.807503
11. Peter Brandl, Clifton Forlines, Daniel Wigdor, Michael Haller, and Chia Shen. 2008. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In Proceedings of the working conference on Advanced visual interfaces (AVI '08). ACM, New York, NY, USA, 154-161. DOI: https://doi.org/10.1145/1385569.1385595
12. Doug A. Bowman, David Koller, and Larry F. Hodges. Travel in immersive virtual environments: An evaluation of viewpoint motion control techniques. In Virtual Reality Annual International Symposium, 1997., IEEE 1997, pp. 45-52. IEEE, 1997.
13. Xiang Cao, Andrew D. Wilson, Ravin Balakrishnan, Ken Hinckley, and Scott E. Hudson. ShapeTouch: Leveraging contact shape on interactive surfaces. TABLETOP '08, 129–136. DOI: 10.1109/TABLETOP.2008.4660195
14. Christer Carlsson and Olof Hagsand. 1993. DIVE A multi-user virtual reality system. In Virtual Reality Annual International Symposium, 1993., 1993 IEEE. IEEE, 394–400.
15. Elizabeth F Churchill and Dave Snowdon. 1998. Collaborative virtual environments: an introductory review of issues and systems. Virtual Reality 3, 1 (1998), 3–15.
16. Maxime Cordeil, Tim Dwyer, Karsten Klein, Bireswar Laha, Kim Marriott, and Bruce H. Thomas. 2017. Immersive collaborative analysis of network connectivity: CAVE-style or head-mounted display? IEEE Transactions on Visualization and Computer Graphics 23, 1 (Jan. 2017), 441–450. DOI: http://dx.doi.org/10.1109/TVCG.2016.2599107
17. J. D. Foley, V. L. Wallace, P. Chan. 1984. The human factors of computer graphics interaction techniques. IEEE Computer Graphics and Applications 4, 11: 13-48.
18. Mike Fraser, Steve Benford, Jon Hindmarsh, and Christian Heath. 1999. Supporting awareness and interaction through collaborative virtual interfaces. In Proceedings of the 12th annual ACM symposium on User interface software and technology (UIST '99).

ACM, New York, NY, USA, 27-36. DOI: http://dx.doi.org/10.1145/320719.322580

19. Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. 2009. ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures. In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09). ACM, New York, NY, USA, 165-172. DOI: https://doi.org/10.1145/1731903.1731935
20. William W. Gaver, Abigail Sellen, Christian Heath, and Paul Luff. 1993. One is not enough: multiple views in a media space. In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93). ACM, New York, NY, USA, 335-341. DOI: https://doi.org/10.1145/169059.169268
21. Google Doc. https://www.google.com/docs/about/
22. Saul Greenberg and David Marwood. 1994. Real time groupware as a distributed system: concurrency control and its effect on the interface. In Proceedings of the 1994 ACM conference on Computer supported cooperative work (CSCW '94). ACM, New York, NY, USA, 207-217. DOI: http://dx.doi.org/10.1145/192844.193011
23. Tovi Grossman, Daniel Wigdor, and Ravin Balakrishnan. 2004. Multi-finger gestural interaction with 3d volumetric displays. In Proceedings of the 17th annual ACM symposium on User interface software and technology (UIST '04). ACM, New York, NY, USA, 61-70. DOI: http://dx.doi.org/10.1145/1029632.1029644
24. Tovi Grossman and Ravin Balakrishnan. 2008. Collaborative interaction with volumetric displays. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08). ACM, New York, NY, USA, 383-392. DOI: https://doi.org/10.1145/1357054.1357118
25. Jan Gugenheimer, Evgeny Stemasov, Julian Frommel, and Enrico Rukzio. 2017. ShareVR: Enabling Co-Located Experiences for Virtual Reality between HMD and Non-HMD Users. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 4021-4033. DOI: https://doi.org/10.1145/3025453.3025683
26. Yves Guiard. Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. In Journal of Motor Behavior 19 (1987), 486–517.
27. Carl Gutwin and Saul Greenberg. 2002. A Descriptive Framework of Workspace Awareness for Real-Time Groupware. Comput. Supported Coop. Work 11, 3 (November 2002), 411-446. DOI: https://doi.org/10.1023/A:1021271517844
28. Jefferson Han and Brian Smith. 1997. CU-SeeMe VR immersive desktop teleconferencing. In Proceedings of the fourth ACM international conference on Multimedia (MULTIMEDIA '96). ACM, New York, NY, USA, 199-207. DOI: http://dx.doi.org/10.1145/244130.244199
29. Ken Hinckley, Patrick Baudisch, Gonzalo Ramos, and Francois Guimbretiere. 2005. Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05). ACM, New York, NY, USA, 451-460. DOI: http://dx.doi.org/10.1145/1054972.1055035
30. Ken Hinckley, Randy Pausch, John C. Goble, and Neal F. Kassell. 1994. A survey of design issues in spatial input. In Proceedings of the 7th annual ACM symposium on User interface software and technology (UIST '94). ACM, New York, NY, USA, 213-222. DOI: http://dx.doi.org/10.1145/192426.192501
31. Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen + touch = new tools. In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 27-36. DOI: https://doi.org/10.1145/1866029.1866036
32. Roland Holm, Erwin Stauder, Roland Wagner, Markus Priglinger, and Jens Volkert. 2002. A combined immersive and desktop authoring tool for virtual environments. In Virtual Reality, 2002. Proceedings. IEEE. IEEE, 93–100.
33. Shahram Izadi, Harry Brignull, Tom Rodden, Yvonne Rogers, and Mia Underwood. 2003. Dynamo: a public interactive surface supporting the cooperative sharing and exchange of media. In Proceedings of the 16th annual ACM symposium on User interface software and technology (UIST '03). ACM, New York, NY, USA, 159-168. DOI: https://doi.org/10.1145/964696.964714
34. 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 (UIST '14). ACM, New York, NY, USA, 637–644. DOI: http://dx.doi.org/10.1145/2642918.2647383
35. Azam Khan, Justin Matejka, George Fitzmaurice, Gordon Kurtenbach. 2005. Spotlight: directing users' attention on large displays. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05), 791-798. http://dx.doi.org/10.1145/1054972.1055082

36. Regis Kopper, Tao Ni, Doug A. Bowman, and Marcio Pinho. 2006. Design and Evaluation of Navigation Techniques for Multiscale Virtual Environments. In Proceedings of the IEEE conference on Virtual Reality (VR '06). IEEE Computer Society, Washington, DC, USA, 175-182. DOI: http://dx.doi.org/10.1109/VR.2006.47

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

38. Joseph LaViola, Loring S. Holden, Andrew S. Forsberg, Dom S Bhuphaibool, and Robert C. Zeleznik, 1998. Collaborative conceptual modeling using the sketch framework. In Proceedings of the Third Conference on Computer-Aided Architectural Design Research in Asia, 154-158. CAADRIA. Osaka, Japan: Osaka University.

39. Pedro Lopes, Alexandra Ion, and Patrick Baudisch. 2015. Impacto: Simulating Physical Impact by Combining Tactile Stimulation with Electrical Muscle Stimulation. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15). ACM, New York, NY, USA, 11-19. DOI: https://doi.org/10.1145/2807442.2807443

40. Jock D. Mackinlay, Stuart K. Card, and George G. Robertson. 1990. Rapid controlled movement through a virtual 3D workspace. In Proceedings of the 17th annual conference on Computer graphics and interactive techniques (SIGGRAPH '90). ACM, New York, NY, USA, 171-176. DOI: http://dx.doi.org/10.1145/97879.97898

41. P McAvinney. The Sensor Frame-A Gesture-Based Device for the Manipulation of Graphic Objects. 1986

42. Meredith Ringel Morris, Kathy Ryall, Chia Shen, Clifton Forlines, and Frederic Vernier. 2004. Beyond "social protocols": multi-user coordination policies for co-located groupware. In Proceedings of the 2004 ACM conference on Computer supported cooperative work (CSCW '04). ACM, New York, NY, USA, 262-265. DOI: http://dx.doi.org/10.1145/1031607.1031648

43. Meredith Ringel Morris, Jarrod Lombardo, and Daniel Wigdor. 2010. WeSearch: supporting collaborative search and sensemaking on a tabletop display. In Proceedings of the 2010 ACM conference on Computer supported cooperative work (CSCW '10). ACM, New York, NY, USA, 401-410. DOI: https://doi.org/10.1145/1718918.1718987

44. Thomas P. Moran, Patrick Chiu, and William van Melle. 1997. Pen-based interaction techniques for organizing material on an electronic whiteboard. In Proceedings of the 10th annual ACM symposium on User interface software and technology (UIST '97). ACM, New York, NY, USA, 45-54. DOI: http://dx.doi.org/10.1145/263407.263508

45. Cuong Nguyen, Stephen DiVerdi, Aaron Hertzmann, and Feng Liu. 2017. CollaVR: Collaborative In-Headset Review for VR Video. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17). ACM, New York, NY, USA, 267-277. DOI: https://doi.org/10.1145/3126594.3126659

46. Oculus Rift. https://www.oculus.com/rift/

47. Ohan Oda, Carmine Elvezio, Mengu Sukan, Steven Feiner, and Barbara Tversky. 2015. Virtual Replicas for Remote Assistance in Virtual and Augmented Reality. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15). ACM, New York, NY, USA, 405-415. DOI: https://doi.org/10.1145/2807442.2807497

48. Oliver Otto, Dave Roberts, and Robin Wolff. 2006. A review on effective closely-coupled collaboration using immersive CVE's. In Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications (VRCIA '06). ACM, New York, NY, USA, 145-154. DOI: http://dx.doi.org/10.1145/1128923.1128947

49. Tomislav Pejsa, Julian Kantor, Hrvoje Benko, Eyal Ofek, and Andrew Wilson. 2016. Room2Room: Enabling Life-Size Telepresence in a Projected Augmented Reality Environment. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW '16). ACM, New York, NY, USA, 1716-1725. DOI: https://doi.org/10.1145/2818048.2819965

50. Ken Perlin and David Fox. 1993. Pad: an alternative approach to the computer interface. SIGGRAPH '93, 57-64. DOI: http://dx.doi.org/10.1145/166117.166125

51. Jeffrey S. Pierce, Brian C. Stearns, and Randy Pausch. 1999. Voodoo dolls: seamless interaction at multiple scales in virtual environments. In Proceedings of the 1999 symposium on Interactive 3D graphics (I3D '99). ACM, New York, NY, USA, 141-145. DOI: http://dx.doi.org/10.1145/300523.300540

52. Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. The go-go interaction technique: non-linear mapping for direct manipulation in VR. In Proceedings of the 9th annual ACM symposium on User interface software and technology (UIST '96). ACM, New York, NY, USA, 79-80. DOI: http://dx.doi.org/10.1145/237091.237102

53. James T Reason and Joseph John Brand. 1975. Motion sickness. Academic press.

54. Randall B. Smith, Ranald Hixon, and Bernard Horan. 1998. Supporting flexible roles in a shared space. In Proceedings of the 1998 ACM conference on

Computer supported cooperative work (CSCW '98). ACM, New York, NY, USA, 197-206. DOI=http://dx.doi.org/10.1145/289444.289494

55. Dave Snowdon, Elizabeth F Churchill, and Alan J Munro. 2000. Collaborative virtual environments: digital spaces and places for CSCW. Collaborative Virtual Environments (2000), 1–34. DOI: http://dx.doi.org/10.1.1.114.9226

56. Aaron Stafford, Wayne Piekarski, and Bruce Thomas. 2006. Implementation of God-like Interaction Techniques for Supporting Collaboration Between Outdoor AR and Indoor Tabletop Users. In Proceedings of the 5th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR ’06). IEEE Computer Society, Washington, DC, USA, 165–172. DOI: http://dx.doi.org/10.1109/ISMAR.2006.297809

57. Richard Stoakley, Matthew J. Conway, and Randy Pausch. 1995. Virtual reality on a WIM: interactive worlds in miniature. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems CHI '95, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 265-272. DOI: http://dx.doi.org/10.1145/223904.223938

58. Ivan E. Sutherland. 1964. Sketch pad a man-machine graphical communication system. In Proceedings of the SHARE design automation workshop (DAC '64). ACM, New York, NY, USA, 6.329-6.346. DOI: http://dx.doi.org/10.1145/800265.810742

59. Ivan E. Sutherland. 1968. A head-mounted three dimensional display. In Proceedings of the December 9-11, 1968, fall joint computer conference, part I (AFIPS '68 (Fall, part I)). ACM, New York, NY, USA, 757-764. DOI: https://doi.org/10.1145/1476589.1476686

60. Pierre Wellner. 1991. The DigitalDesk calculator: tangible manipulation on a desk top display. In Proceedings of the 4th annual ACM symposium on User interface software and technology (UIST '91). ACM, New York, NY, USA, 27-33. DOI: http://dx.doi.org/10.1145/120782.120785

61. Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. 2009. User-defined gestures for surface computing. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 1083-1092. DOI: https://doi.org/10.1145/1518701.1518866

62. Mike Wu and Ravin Balakrishnan. 2003. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In Proceedings of the 16th annual ACM symposium on User interface software and technology (UIST '03). ACM, New York, NY, USA, 193-202. DOI: https://doi.org/10.1145/964696.964718

63. Haijun Xia, Bruno Araujo, Tovi Grossman, and Daniel Wigdor. 2016. Object-Oriented Drawing. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 4610-4621. DOI: https://doi.org/10.1145/2858036.2858075

64. Haijun Xia, Bruno Araujo, and Daniel Wigdor. 2017. Collection Objects: Enabling Fluid Formation and Manipulation of Aggregate Selections. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 5592-5604. DOI: https://doi.org/10.1145/3025453.3025554

65. Haijun Xia, Ken Hinckley, Michel Pahud, Xiao Tu, and Bill Buxton. 2017. WritLarge: Ink Unleashed by Unified Scope, Action, & Zoom. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 3227-3240. DOI: https://doi.org/10.1145/3025453.3025664

66. Xiaolong Zhang and George W. Furnas. 2002. Social interactions in multiscale CVEs. In Proceedings of the 4th international conference on Collaborative virtual environments (CVE '02). ACM, New York, NY, USA, 31-38. DOI: http://dx.doi.org/10.1145/571878.571884