Multi-Window 3D Interaction for Collaborative Virtual Reality

Andre Kunert, Tim Weissker, Bernd Froehlich, and Alexander Kulik

Abstract—We present a novel collaborative virtual reality system that offers multiple immersive 3D views at large 3D scenes. The physical setup consists of two synchronized multi-user 3D displays: a tabletop and a large vertical projection screen. These displays afford different presentations of the shared 3D scene. The wall display lends itself to the egocentric exploration at 1:1 scale, while the tabletop affords an allocentric overview. Additionally, handheld 3D portals facilitate the personal exploration of the scene, the comparison of views, and the exchange with others. Our developments enable seamless 3D interaction across these independent 3D views. This requires the simultaneous representation of user input in the different viewing contexts. However, the resulting interactions cannot be executed independently. The application must coordinate the interactions and resolve potential ambiguities to provide plausible effects. We analyze and document the challenges of seamless 3D interaction across multiple independent viewing windows, propose a high-level software design to realize the necessary functionality, and apply the design to a set of interaction tools. Our setup was tested in a formal user study, which revealed general advantages of collaborative 3D data exploration with multiple views in terms of user preference, comfort, and task performance.

Index Terms—Collaborative virtual environment, multi-display setups, 3D interaction techniques, input disambiguation

1 Introduction

Multi-user 3D displays enable co-located collaborative interaction in and with shared virtual environments [1], [2]. However, collaborating groups do not continuously focus on the same aspects of a shared task or the environment, but recurring phases of independent interaction can be beneficial. For object manipulation tasks, the necessary workspace separation can emerge spontaneously in a sufficiently large shared interaction space [3]. However, if navigating a large virtual environment is a primary subtask, groups may benefit from separate navigation capabilities for individual users that enable the parallel exploration of a virtual environment without mutual interference [4].

We developed a multi-window Virtual Reality (VR) system to support close and loose coupling during the the collaborative exploration and analysis of large-scale 3D data sets. Our hardware setup consists of two immersive multi-user 3D displays (Fig. 1): a large vertical projection screen and a smaller tabletop display. Both displays are synchronized and each provides three users with stereoscopic image pairs according to their individual viewing positions. They serve as independent viewing windows into the 3D scene and provide separate navigation capabilities. The tabletop is most suitable for allocentric overviews of the scene or the close-up analysis of surface details. The wall display, instead, facilitates egocentric scene exploration at 1:1 scale.

Despite the functional separation of both displays to serve as independent viewing windows, their physical co-location affords a continuous interaction space for all user input, where the same input devices with their virtual representations and application states (e.g., an object-dragging 3D pick ray) can be used consistently. While this appears similar to the operation of a mouse pointer across several independent 2D windows, it requires a more complex software architecture to enable similar behavior across discontinuous 3D views. We describe the implementation challenges and suggest a high-level software design to realize the required functionality. On this basis, we adapted common 3D manipulation techniques and visualization tools. This includes a set of heuristics to solve inherent ambiguities of multi-window 3D interaction for fundamental types of direct 3D user input.

Furthermore, we integrated handheld 3D viewing windows that can be used for private preparations as well as for public information exchange. These virtual displays constitute additional 3D interaction contexts, in which user input must also be represented. At the same time, they are virtual interaction tools that can be used consistently across all physical displays in the workspace. Consequently, they can be carried around between the wall and the tabletop display to compare and exchange views (Fig. 7b).

We tested our multi-display setup in a formal user study on the collaborative exploration of a 3D city model. Our user study compared the combination of both 3D displays with independent virtual navigation capabilities to a baseline condition using only the wall display. The results revealed benefits of multi-window 3D interaction in terms of usability, task performance, and user preference.

- *The authors are with the Virtual Reality and Visualization Research Group, Bauhaus-Universität Weimar, Weimar 99423, Germany. E-mail: {andre.kunert, tim.weissker, bernd.froehlich, kulik}@uni-weimar.de.*

Manuscript received 4 Sept. 2018; revised 9 Apr. 2019; accepted 10 Apr. 2019. Date of publication 3 May 2019; date of current version 6 Oct. 2020. (Corresponding author: Andre Kunert.) Recommended for acceptance by A. Lecuyer. Digital Object Identifier no. 10.1109/TVCG.2019.2914677

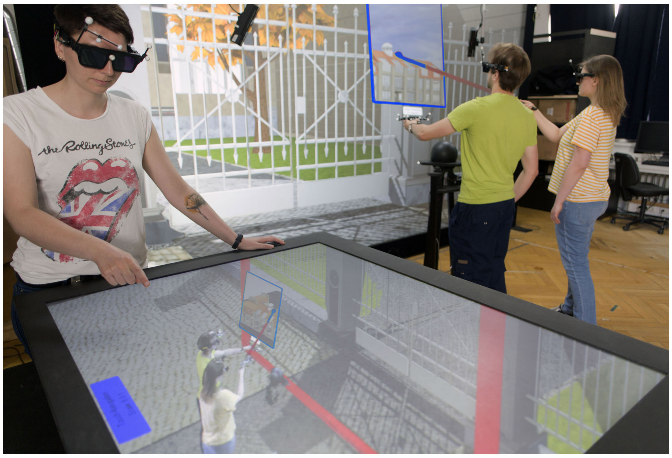


Fig. 1. Three users collaboratively explore a 3D building model in a co-located multi-user multi-display environment. The physical displays (wall and tabletop) and additional virtual views (portals) facilitate the simultaneous exploration of the virtual scene from different viewpoints. User input (e.g., a 3D pick ray) can be consistently applied across these independent viewing windows. 3D video avatars represent the users in the virtual environment (see tabletop view).

In summary, the main contributions of this paper are:

- the presentation of a collaborative multi-display 3D interaction environment building on two co-located multi-user 3D projection displays,
- the identification of challenges to realize seamless 3D interaction across multiple independent but co-located 3D displays and virtual viewing windows,
- a high-level software design for the implementation of the necessary functionality,
- implementation examples of seamless 3D interaction techniques across independent viewing windows,
- the results of a formal user study that demonstrate the benefits of multi-window 3D environments and 3D interaction for collaborative work.

2 Related Work

The work presented in this paper builds on a large body of prior research concerning the design of collaborative user interfaces. We paid particular attention to create a shared workspace that facilitates implicit awareness cues, while supporting both closely and loosely coupled cooperation. To this end, we followed the well established approach of multitasking support through multiple independent application windows. Our work extends and generalizes interface developments concerned with the continuous representation of user input across multiple displays and windows to solve the particular challenges involved in the collaborative interaction with multiple immersive 3D viewing windows.

2.1 Workspace Requirements for Collaboration

One of the primary collaboration requirements is the generation and maintenance of mutual awareness among users in a shared interaction space. Gutwin and Greenberg [5] emphasized that such workspace awareness in co-located settings largely builds on implicit information exchange through the observation of each other's activity (consequential communication) and the sensory perception of involved artifacts (feedthrough). In remote settings, such implicit cues are often missing and workspace awareness must be realized by more abstract notifications [6], [7], [8].

Many researchers also stressed the need of collaborating users to temporarily separate their activities and work on independent subtasks [9], [10], [11], [12]. Scott et al. [3] observed that people in physical workplaces tend to establish different interaction areas (territories) to serve for private usage and group exchange. Similar behavior was found during collaboration on visual analytics tasks with large computer displays, although territories were found to be more transient if the interface design promoted user mobility [11], [12], [13]. During such phases of loose coupling, maintaining awareness of collaborators and their activities, is a prerequisite for frequent transitions back to closely-coupled collaboration [5], [11], [12].

Collaborative applications that primarily require selection and manipulation input may suffice with a large enough interaction space to support territoriality behavior [11], [12], [13]. Virtual navigation on a shared display, on the other hand, affects all involved users. Chen et al. [4] suggested to exploit multi-user 3D displays for the presentation of completely different content according to the different subtasks and interests of involved users and thus provide each user with individual virtual navigation capabilities. They also observed, however, that the resulting incoherence of the shared workspace hampers mutual awareness and can result in perceptual conflicts [14]. As an alternative, we suggest the combination of multiple shared 3D views to maintain workspace coherence, while enabling independent 3D navigation and transient territoriality.

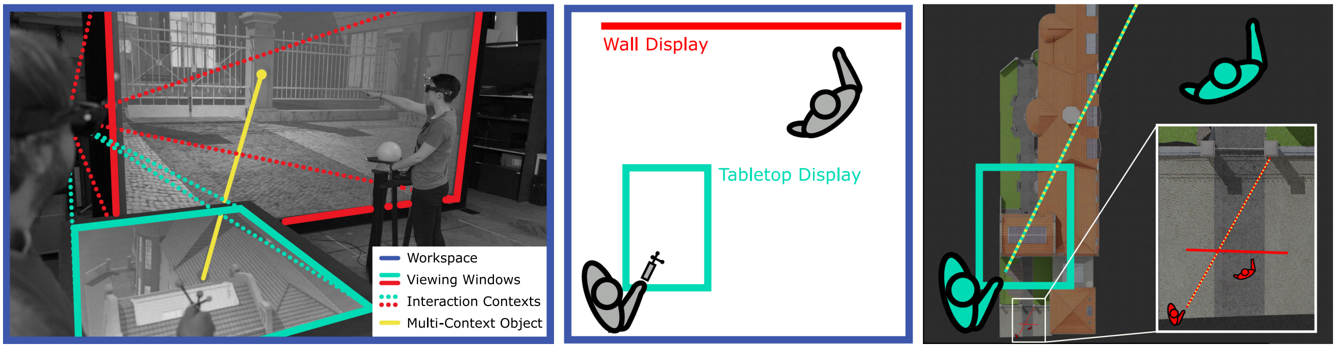
2.2 Multi-Window Workspaces

Desktop workplaces with multiple displays and/or viewing windows are well established to support multitasking. In collaborative systems, multiple windows or portals have also been suggested to provide personal interaction territories for individual users in a larger shared workspace [15], [16], [17], [18], [19]. Similarly, multi-display environments support loosely-coupled collaboration with individual users focusing on separate subtasks [20], [21], [22], [23], [24], [25], [26]. Our system combines both approaches with virtual 3D viewing windows that can be moved across independent multi-user 3D displays.

So far, research on collaboration in multi-window workspaces had been primarily concerned with 2D display systems. Combinations of multiple 3D displays are most established to extend a single view and gain a larger field of view (e.g., tiled displays and CAVE setups (see [27]).

The benefits of multiple 3D views at the same scene have also been explored in various forms, e.g., secondary scene representations (e.g., WIM [28], and Voodoo Dolls [29]), 3D portals [19], [30], [31], and independent 3D displays [20], [32]. Stürzlinger et al. [20] and Kunert et al. [19] considered also collaborative settings, but none of both groups proved the expected benefits. Also, the implementation challenges to realize consistent user input across independent 3D windows had not been addressed before.

The most similar prior workspace development was presented by Hua et al. [33]. They used head-mounted projectors in combination with retroreflective surfaces to build a



(a) Situation from a user's perspective (b) Situation in physical workspace (c) Situation in the virtual scene

Fig. 2. The wall and the tabletop display offer independent 3D views of the same virtual scene, hence also two different *Interaction Contexts* (a). The central illustration (b) shows of the spatial configuration of users, displays, and a 3D pointer in the physical workspace. Both users and their interaction tools are represented twice in the scene according to both displayed locations (c).

combination of a large 3D wall display, a 3D tabletop and handheld lenses for collaborative use. Our technical setup constitutes a very similar configuration of multi-user 3D displays, but with higher image quality, better ergonomics and advanced input opportunities. It was applied in several experimental settings, for which we developed suitable interaction tools and techniques. Our interface design supporting continuous user input across the separate views extends beyond the pioneering work of Hua et al. We also contribute a detailed description of novel interaction opportunities and their implementation challenges.

2.3 Continuous User Input Across Independent Views

Workspaces consisting of multiple independent displays and viewing windows have become a commonplace in the realm of 2D user interfaces. Users expect that their input can be continuously applied in all visible contexts. Prior research showed that adherence to geometrical consistency across all displays in the same workspace improves usability [34], [35], [36], [37]. In case of larger gaps between the displays, low-resolution projections have been suggested to show the moving cursor between the involved displays [38], [39].

The implementation of continuous user input across multiple independent 3D views or scene representations did not yet receive as much attention as multi-surface interaction with 2D user interfaces. In the latter case, input can always be precisely assigned to a single surface. Multi-window 3D interaction, instead, implies multiple simultaneous representations of user input which must be disambiguated according to the respective functionality. Benko et al. [40] suggested *cross-dimensional gestures* to realize transitions between collocated 2D and 3D displays. However, since only a single 3D view was provided, input could be clearly associated with either the 2D or the 3D context. The closest description of multi-window 3D interaction techniques in prior work are "through-the-lens techniques" by Stoev and Schmalstieg [30]. They suggested disambiguation based on the visibility of a 3D cursor (tip of a stylus) in a single nested 3D view (lens). We present a more generic approach that considers the visibility of potential input effects and their distance to the effector. This allows us to support various types of 3D user input across an unlimited number of concurrent 3D views that are not necessarily nested.

3 A Multi-User, Multi-Display Workspace for Collaborative 3D Data Exploration

We present a novel multi-user, multi-display infrastructure for the collaborative exploration and analysis of large 3D data sets. More specifically, we combined a multi-user 3D wall display and a 3D tabletop display in a joint workspace.

3.1 Technical Setup

Both displays are based on 360 Hz projection technology that provides individual stereoscopic image pairs with 60 Hz update rate for three users on a shared screen (similar to [2]). Shutter glasses separate the images in front of the users' eyes. As a result, the users perceive a shared 3D scene, undistorted, at the same location, and corresponding to their individual tracked viewpoints. The wall display dimensions are 4.2m in width and 2.6m in height with a pixel resolution of 1920×1200. The multi-user tabletop display offers an image size of $1.14m \times 0.85m$ with a resolution of 1400×1050 pixels. The projection hardware of both displays as well as the shutter glasses are synchronized. The latter are optically tracked at 150 Hz across the whole workspace.

3.2 Terminology

We use the following terms and definitions to describe the basic attributes of our multi-display VR setup (Fig. 2a): The *Workspace* sets the stage for user interaction with the application content. It involves the available displays and interaction devices. Multiple 3D *Viewing Windows* implement independent 3D Views and *Interaction Contexts* in terms of different spatial relations between users in the *Workspace* and the perceived virtual environment. Separate *Viewing Windows* can be tied to a single physical display or span across multiple ones. Our *Multi-Context Objects* ensure consistent input functionality across these concurrent *Interaction Contexts* and resolve inherent ambiguities. We suggest the visibility of input effects to the operating user as the main factor for heuristic selection among *Interaction Contexts*. The spatial boundaries of the latter are thus defined by the viewing frusta per user and *Viewing Window*.

3.3 Interaction Opportunities

In contrast to CAVE setups, the wall and the tabletop display in our setup serve as separate viewing windows into the virtual 3D environment. Through each of these windows, users can see individual, but corresponding, 3D perspectives of different scene locations (Fig. 1). Navigating through the virtual environment using any of the 3D windows does not affect other views. The virtual navigation techniques implemented for the separate displays differ in their modes of operation according to the physical affordances and intended use of the respective display. However, all of them support full 3D navigation and uniform scaling of the perceived virtual environment. A shared group navigation device for steering-based travel at the wall display is placed centrally accessible in the workspace. The tabletop display supports 3D navigation with multitouch input [41].

In addition to both physical displays, we provide virtual viewing windows to prepare, navigate, and share interesting views of the environment with a handheld input device (similar to [19]). The tracked device is equipped with several buttons that support the capturing and adaptation of views. The perceived position, orientation, and scale of the captured location can be freely adjusted to prepare any desired view of the scene. It allows users to individually explore data sets from various perspectives without mutual interference. If interesting features have been found, the handheld view can be easily shown or handed over to others (blue display frame in Figs. 1, 7a, 7b).

Various VR application scenarios benefit from direct 3D interaction capabilities. Commonly, direct 3D manipulation follows the metaphor of either a virtual hand or a 3D pick ray. A main contribution of our work is the adaptation of these basic interaction tools to enable their continuous and conflict-free operation within our multi-window environment in order to create new interaction and collaboration opportunities. Not least, our system represents users as live 3D video avatars (similar to [42]) to increase workspace awareness and facilitate mutual support. These avatars can be seen in different views of the virtual scene, for instance in the miniature overview on the tabletop display (Fig. 1).

4 Implementation Challenges

An immersive 3D display can be understood as a viewing window into a virtual 3D world, in which a section of this world becomes visible. However, the corresponding interaction space is not restricted to the surface, as in the case of monoscopic displays. It extends in front and behind the screen plane into the workspace. Multiple 3D displays or viewing windows in a shared workspace, thus result in a spatial overlap of the corresponding interaction volumes. Consequently, user input cannot simply be associated with a single viewing window at a time, but interactions with content shown on different displays must be considered concurrently. The main challenge of multi-window 3D interaction, therefore, consists in the combination of separate viewing capabilities per display and a continuous interaction space. Individually, each requirement is trivial to comply with. CAVE setups and tiled displays enable continuous interaction across multiple physical display areas but provide only a single view into the scene. Separate views, instead, can be realized with independent view transformations per display, but this also implies discontinuous spatial relationships between user input and the 3D content.

4.1 Concurrent Spatial Relationships

Multiple independent 3D viewing windows, imply concurrent spatial relations of users and their input tools and the displayed virtual environment. These must be modeled and processed separately by the application. The two users and the tracked pointer in Fig. 2, for instance, are represented twice in the virtual scene: once according to the egocentric view on the wall display (Fig. 2c red figures in the white-framed callout), but also as interacting with a miniature overview of the same scene shown on the tabletop. (Fig. 2c turquoise figures). Both transformations must be considered for visualization and interaction. From a user perspective, however, the resulting visual input representations are perceived to be integral, e.g., a single pointing ray spanning across both spatial contexts (Fig. 2a).

4.2 Dynamic Context Selection for User Input

Interaction tools must be adapted to cope with concurrent spatial transformations. Depending on the functionalities, input can be globally applied across multiple displays or it may affect only a single selected view. Operating a virtual flashlight in the workspace is an example for a global input scope. Continuous illumination across the different scene sections of all displays appears reasonable in most cases and does not introduce conflicts (Fig. 7a). Object manipulation in the virtual environment, on the other hand, requires the meaningful selection of only one active spatial transformation between user input and the virtual scene. Without further coordination, multiple pick ray transformations could independently affect different scene parts. This is generally undesired and can result in conflicts. Fig. 5 illustrates a situation where the transformations of a 3D pick ray into two views could simultaneously move two different objects in the scene. In other situations, they might even attempt to move the same object in different directions. In any case, the user interface must derive a conflict-free, consistent, and comprehensible solution. If users want to move virtual objects between the displays (e.g., drag and drop transitions), the active relation must be determined and continuously updated during dragging. Also, the tool states must be kept consistent among its different representations.

2D interfaces with multiple interaction windows generally use the window geometries as proxies for the selection of the corresponding input transformations. A 2D cursor, for instance, can be unambiguously associated with a single 2D viewing window. In case of overlaps, the frontmost window is selected. For user input in 3D space, the presence of the input device in the respective viewing frustum must be considered as suggested by Stoev and Schmalstieg: "The user can manipulate remote objects by reaching with the stylus into the frustum volume defined by the lens and the current viewpoint" [30, p.63]. If the interaction tools extend in 3D space (e.g., a 3D ray pointer or a flashlight), a proxy geometry of the tool can be used for the intersection test with the viewing frusta instead of the position of the physical input handle. For example, the pick ray in Fig. 3a can operate inside the virtual viewing window although the location of the pointing device lies outside.

Intersection tests of input proxies and the viewing frusta, however, may not always suffice for the disambiguation between different contexts. If the operated tool is at least

(a) Operator View (b) Observer View

Fig. 3. The visibility of input effects for the operating user in different viewing windows implies the reference frame for interaction. Seen from the operator's perspective (a), the 3D ray intersects with content visible inside the virtual viewing window (blue dot on the tower). However, for an observer (b), the ray does not intersect with any visible scene parts.

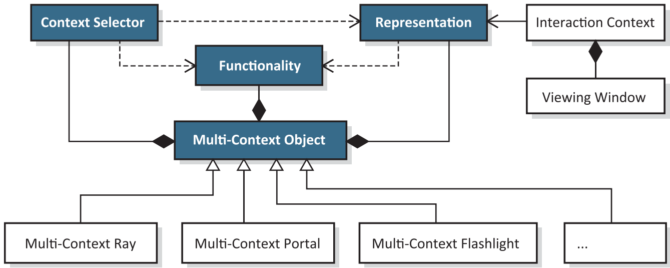


Fig. 4. Class diagram of an abstract *Multi-Context Object* (blue) and its association with *Interaction Contexts*. The latter specify the spatial transformations between the workspace and the virtual scene and thus imply separate, *functionality*-specific *Representations* of user input at the specified scene locations. *Context Selectors* determine which of these *Representations* will be used for the execution of one or more *Functionalities*. For similar *Functionalities*, the same *Representations* and *Context Selectors* can be reused. Concrete multi-context 3D interaction techniques are implemented by inheritance.

partially visible in a particular view, we must also consider its functionality, compute potential effects in the scene, and check their visibility. In case of the 3D pick ray the intersection point with the scene geometry must be computed and confirmed to be visible. Fig. 3a shows the pointing device outside of the frustum of the virtual window, but the operated pick ray intersects visibly with the model inside. If input effects occur in multiple views simultaneously, but only one effect can be applied without conflict, further effect parameters such as the distance to the tool handle or its size in the different views can be considered to select a single interaction context.

In the case of multi-user systems, the visibility of input effects may differ between users. Fig. 3 illustrates such an ambiguous situation. The operator of a virtual pointer perceives the ray intersecting with scene content through a virtual viewing window (Fig. 3a). For another user, this intersection may be invisible (Fig. 3b). Visibility-based context selection, thus, requires the association of all input events to a unique operating user. It may be predefined or dynamically derived from the spatial relationships between users and input devices in the workspace.

4.3 General Approach

Multi-window 3D interaction can be realized with multiple scene representations (with individual transformations) in the users' interaction space (e.g., a WIM [28]). User input must be dynamically associated with one of these representations. To maintain a consistent world state, all scene representations have to be synchronized. We follow the opposite approach and represent users and their inputs multiple times in the same virtual scene at different locations. This implicitly enables the visualization of users and interaction effects in different spatial contexts (Fig. 1).

Specifically, we propose *Multi-Context Objects* (see Section 5) to model and disambiguate multiple concurrent relations between user input and the displayed 3D content. They were designed to maintain the semantic consistency of interaction tools and other user-space objects across all views in a workspace. This is achieved through multiple representations in the scene and rules selecting the most suitable ones for the application of each functionality.

Many software architecture models have been introduced in the past to meet the increasing complexity of interactive systems (e.g., MVC [43], PAC [44], PAC-C3D [45], PAC* [46]). They all build upon the functional separation of concerns in the software structure targeting various asects such as parallelism, reusability, portability, distributed applications, and groupware support. The implementation pattern of *Multi-Context Objects* facilitates the coordination of concurrent spatial input mappings. To our knowledge, this complementary design challenge was not addressed by prior work. However, the suggested software structure can be integrated with existing models, for instance as an extended *Controller* acccording the Model-View-Controller (MVC) paradigm [43]. In our case, the *Model* is a virtual 3D environment that can be perceived through different *Views* as provided by multiple 3D viewing windows. Users operate *Controller* entities to manipulate the *Model* according to the given functionality in the spatial contexts of the selected *Views*.

5 Multi-Context Objects

Multi-Context Objects implement seamless 3D interaction across independent 3D viewing windows and resolve potential conflicts. We suggest the separation of responsibilities among three components (Fig. 4): 1. unique input *Functionalities* for consistent interface behavior with 2. multiple input *Representations* at various locations in a scene and 3. functionality-specific *Context Selectors* that select one or multiple *Interaction Contexts* based on the visibility of the potential interaction effects.

5.1 Functionality

The *Functionality* component represents a particular interaction method in our model. Despite its interaction with the *Context Selector*, it is equivalent to the corresponding implementation of the same method for single-display systems. It implements the functional behavior of *Multi-Context Objects*, e.g., an object dragging method for a Multi-Context Pick Ray. Based on its current state, the *Functionality* component calls its associated *Context Selector* to receive one or several *Representations* through which it then executes the functionality at the respective scene location(s). A *Multi-Context Object* can have multiple *Functionalities*, e.g., a pick ray tool may support target selection, object dragging, path tracing, and many other interaction methods.

5.2 Representation

Representations are the virtual embodiments of user input (e.g., a ray or a 3D cursor) at a specific location in the 3D scene. Their appearance may vary, like the ray colors in Figs. 1, 3, and 6. Showing all of these concurrent representations in all views can be confusing. Therefore, their visibility and further visualization parameters are defined per *Interaction Context*. Each *Representation* of an interaction tool is typically only visible in the *Interaction Context* it was created for, but it can be exposed to other contexts on demand, e.g., to increase mutual awareness of activities (see Sections 6.1 and 6.3). The *Context Selector* decides which of these *Representations* are applicable for a specific *Functionality* and enables or disables them on demand.

5.3 Context Selector

The Context Selector implements *Functionality*-specific heuristics for the selection among input *Representations* in different *Interaction Contexts*. Most tools for direct 3D input (e.g., a 3D cursor or pick ray) require the unambiguous selection of a single *Interaction Context* and its corresponding *Representation*. Other *Functionalities* (e.g., a virtual flashlight or lens visualization) can be simultaneously applied in multiple *Interaction Contexts*. If a tool *Functionality* requests the selection of valid *Interaction Context(s)*, the following disambiguation steps are performed subsequently:

1) *Window Visibility.* First of all, the visibility of all viewing windows related to a particular *Interaction Context* is tested. If the operating user cannot see any of these windows, i.e., they are out of his field of view, the corresponding *Representation* is rejected.
2) *Object Visibility.* Thereafter, the visibility of the remaining *Representations* is tested. We use geometric abstractions of users and their interaction tools for the validation of their visibility, e.g., a 3D point (3D cursor or virtual hand), a line segment (ray-based tools), a plane (lenses), or volumetric shapes (flashlight). 3D point abstractions must be inside one of the operator's viewing frusta of an *Interaction Context*. All other geometric abstractions are tested for intersections with these. They may thus be partially outside of the viewing frustum.
3) *Effect Visibility.* Several manipulation tools like a ray pointer require the determination of an exact effect location, e.g., an intersection point with the scene geometry. If an effect can be computed and is visible to the operating user, the corresponding input *Representation* remains a candidate.
4) *Effect Weighting.* Input that can only be applied in a single spatial context may require a final disambiguation step. This final selection could either involve explicit user control or further implicit selection rules. So far, we follow the latter approach and evaluate further aspects of the input effects such as the distance to the user or the visible size.

The described selection process depends on the viewpoint and viewing direction of the operating user. In multi-user settings, the ownership of input tools must be predefined or determined dynamically. Currently, we identify the operating user based on the shortest distance between the input device and lines representing the users' upright positions in the workspace. This simple calculation is sufficient for most cases since people usually follow proxemic constraints and do not enter private spaces of others. To avoid involuntary reevaluation during an operation, ownership is only updated in phases of tool inactivity. During a dragging operation, for instance, the user assignment is not changed.

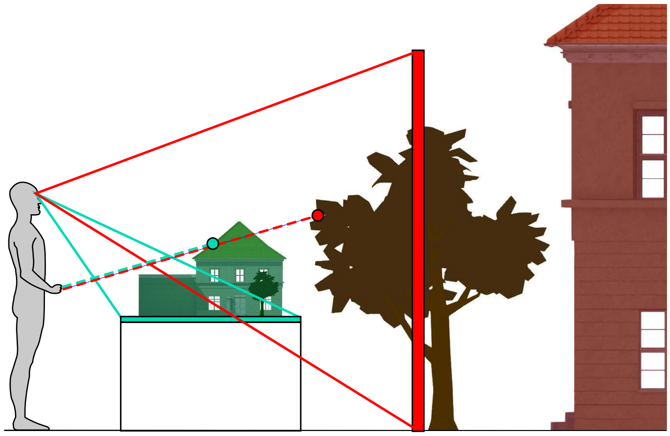


Fig. 5. The user's pick ray is represented twice in the same scene. One *Representation* relates to the wall display view (red parts), the other one to the tabletop (turquoise parts). In this situation, both *Representations* intersect with scene objects. This ambiguity must be resolved. We apply the ray input in the *Interaction Context* with the closest geometry intersection (colored dots), that is visible for the operator. Here, the *Interaction Context* of the wall display is chosen, since the intersection with the miniature scene on the tabletop is not visible for the operator.

6 Pattern Application

We adapted several 3D interaction techniques (e.g., 3D cursor, 3D pick ray, virtual flashlight, and 3D portals) according to the proposed pattern. In the following, we describe three implementation examples of *Multi-Context Objects* with a particular focus on the *Context Selection* heuristics.

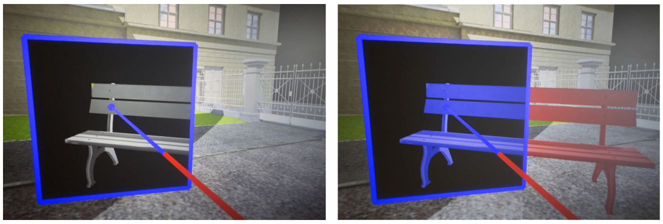
6.1 Multi-Context 3D Pick Ray

Object dragging with the pick ray can only be applied in a single *Interaction Context* without inducing conflicts. Below, we explain the disambiguation process of the *Context Selector* exemplarily for the situation illustrated in Fig. 5.

In a first step, the *Interaction Contexts* of the wall and the tabletop are both confirmed to be valid since both displays are visible for the operating user. The ray geometry also intersects with the viewing frusta of both displays, hence, also the second disambiguation step confirms the validity of both potential *Interaction Contexts*. In the third step, the *Representation* related to the tabletop view is rejected because the corresponding intersection point with the scene (turquoise dot in Fig. 5) is outside of the respective viewing frustum. Thereafter, only one spatial context remains valid. The final disambiguation step, thus, can be skipped. If both ray intersections with scene geometry were visible for the operator (red and turquoise dot in Fig. 5), the *Effect Weighting* step would have favored the closer one (relative to the tracked input handle) in the spatial context of the tabletop display.

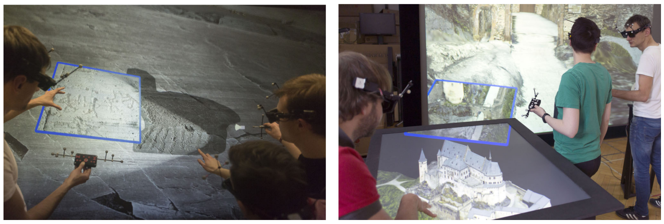
By default, the selected ray *Representation* is only visible in its associated *Interaction Context*, while the dragged object can be seen in all views. Without an obvious effector, the object movement can be irritating. We thus dynamically adapt the visualization settings of the selected ray *Representation* to appear in all views during dragging operations.

Similar to most 2D graphical user interfaces, our *Multi-Context Pick Ray* supports object dragging across the different



(a) Object appears cropped (b) Seamless object visualization

Fig. 6. Objects can appear cropped at the display borders (a). During dragging operations, we show copies of the dragged object in all adjacent Interaction Contexts (b) to realize visually seamless and predictable transitions. Here, the bench inside the virtual viewing window is dragged (blue) and duplicated at the location of the main display context (red).



(a) Comparing parts of a scene (b) Portal transition btw. displays

Fig. 7. Handheld 3D portals provide additional views to other locations in the same scene. On the left (a), 3D scans of rock engravings at two different places are compared under the influence of a virtual flashlight affecting both views. Portals are also *Multi-Context Objects* that can be used across different physical displays (b).

Interaction Contexts. If an object is dragged beyond the boundaries of the currently active *Interaction Context* (i.e., outside the corresponding viewing frustum), a new spatial context for the dragged object is identified by the *Context Selector*. If the intersection point of the pick ray is not located in any of the operator's viewing frusta in the different *Interaction Contexts*, object dragging is continued in the prior one. In case of multiple candidate contexts, we choose the closest one visible from the operator's viewpoint.

If the dragged object is simply removed from one *Interaction Context* and inserted to another one, the transition is not always comprehensible due to cropping at the borders of both involved contexts (Fig. 6a). To realize visually seamless transitions (Fig. 6b), we show copies (multiple *Representations*) of the dragged object in all adjacent *Interaction Contexts* during the dragging operation. If these show the scene at different scale levels, the object copies must be scaled inversely to appear consistent in size for the user.

At object release in the new context, the adapted size can be maintained for scale consistency in the users' interaction space. This might be useful to examine size variations of objects at a specific location (e.g., furniture models). Alternatively, the object's scale can be reset to the initial value to maintain scale consistency relative to the virtual scene. We usually prefer the latter, thus the bench in Fig. 6 is automatically resized to its original dimensions after release.

6.2 Multi-Context 3D Portals

Our virtual viewing windows are similar to earlier examples of 3D portals [19], [30], [31], [47], [48]. The implementation in a multi-display setup, however, is slightly more complex. On the one hand, portals serve as independent views and *Interaction Contexts*. Therefore, other user input such as a 3D pointer (Figs. 1 and 6) or a virtual flashlight (Fig. 7a) can be directly represented at the shown portal locations. On the other hand, they are also input tools that must be represented in multiple other *Interaction Contexts*, e.g., on both physical displays in our setup. A tracked handle allows users to move them in the physical workspace, which implies transitions between other *Interaction Contexts* (Fig. 7b).

In contrast to the *Interaction Contexts* of physical displays, the portal windows are more dynamic. This means that they are activated and deactivated based on the visibility of their *Representations* in other active *Interaction Contexts*, which is evaluated by the respective *Context Selector*. For the continuous portal visualization across *Interaction Contexts*, multiple *Representations* can be active at the same time (Fig. 7b). In case that the views of these *Interaction Contexts* have different scale levels, the respective *Representations* have to be scaled inversely to appear seamless for the users (similar to object copies during dragging across contexts).

For the specification of the perspectives shown on these portals, we follow the earlier suggested metaphor of virtual photography [19]. Mimicking the operation of a photo camera, users can directly capture views from any other *Interaction Context*. The described context selection process (see Section 5.3) defines a single spatial context to that end. In case the portal is present in multiple contexts, the final *Effect Weighting* step in the *Context Selector* takes the visible size of the portal window into account. The *Interaction Context* with the largest visible portal section will be chosen.

6.3 Multi-Context 3D Avatars

In the simplest case, *Multi-Context Objects* have only a single *Functionality* that adjusts the visualization rules of the corresponding *Representations*. *Multi-Context Avatars* are such an example using the suggested software structure to coordinate the rendering of multiple virtual user *Representations* at different scene locations. Their *Context Selector* follows the general selection scheme, but instead of testing and weighting functional effects as required for interaction tools, visibility filters are applied.

In our projection based setup, users can see their real bodies. Other than the virtual representations of interaction tools, their avatars thus do not need to be rendered in the *Interaction Contexts* that define their location in the scene. Instead, showing these avatars in other *Interaction Contexts* allows users to observe themselves in the virtual environment, e.g., as miniatures in the tabletop view (Fig. 1). We also decided to show only avatars that appear smaller or approximately life-sized to avoid awe-inspiring giants in the perceived virtual environment.

6.4 Generalization

We discussed that multiple independent 3D views in the same workspace result in concurrent transformations between user space and the displayed virtual environment. These need to be coordinated for visualization and interaction purposes. *Multi-Context Objects* therefore associate multiple virtual *Representations* of user-space objects with a shared set of *Functionalities*. Functionality-specific *Context Selectors* resolve the resulting ambiguities. As shown with the examples above, this general implementation pattern applies to a variety of 3D interaction tools and techniques with very different *Funtionalities* and *Representations*. Also the *Context Selection* rules are application specific, but we note that the suggested sequence of visibility tests with geometric primitives can often be reused.

Predefined selection and visualization rules, as suggested so far, may not always comply with the user's expectations. The same is true for transitional effects like the scaling of dragged objects described in Section 6.1. Towards more explicit user control, effective strategies for manual interference must be identified in future work and carefully integrated into the interaction sequences.

Multi-Context Objects are also directly applicable to HMD-based multi-window environments. Users, then, would meet in a virtual workspace where virtual viewing windows as described in Section 6.2 provide different *Interaction Contexts*. These virtual windows can be operated by handheld interfaces or they can be placed at fixed positions in the virtual workspace, for instance to mimic the tabletop/wall configuration in our laboratory.

7 User Study

We used our multi-display infrastructure and our set of multi-context tools in an archaeological application focusing on the visual analysis of 3D scanned prehistoric rock engravings (Fig. 7a) and their surrounding environment [49]. Archaeological experts provided a positive high-level evaluation of our system and the multi-context interaction techniques. This feedback was motivating, but it did not allow us to evaluate in depth, how groups of collaborating users can take advantage of the proposed multi-display environment. We therefore devised a more formal user study.

7.1 Hypotheses

Our developments were motivated by the assumption that separate viewing and interaction windows in a coherent interaction space facilitate more effective multi-user collaboration. We aimed to simultaneously support both independent subtask execution (loose coupling) and joint decision making (close coupling). A single shared viewing window enables the latter, but it does not allow two users to perform different subtasks if these involve independent view navigation. We therefore expected general benefits of multi-window 3D interaction based on a higher parallelization of subtasks and increased flexibility. Furthermore, we expected an increase in user comfort regarding the amount of the perceived viewpoint navigation. With a second display at disposal, a user can always change over to the other display to avoid being passively moved around by others.

Specifically, we tested the following hypotheses:

- H1: The combination of two shared displays enables more effective collaborative performance.
- H2: Group navigation with a single display increases symptoms of cybersickness.
- H3: Collaborating users prefer using both displays.

7.2 Apparatus

Our experimental setup consisted of the multi-display environment described in Section 3.1, but with slightly constrained navigation facilities to reduce confounding variables. The tabletop only offered a common 2D navigation interface with four degrees of freedom and also navigation on the wall display favored ground-following locomotion. All demonstrators and the user study were implemented using the VR framework Avango-Guacamole [50]. The scene was continuously updated and rendered at 60 Hz. We measured an end-to-end latency of about 100 ms.

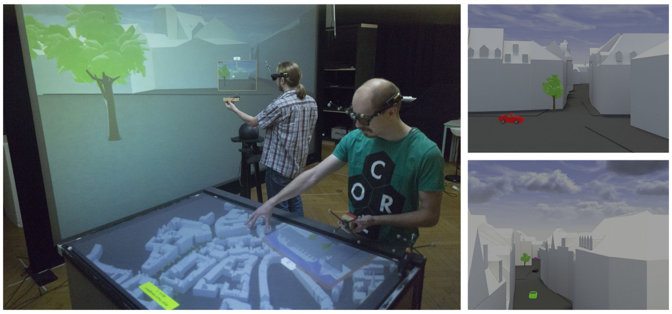


Fig. 8. Left: A participant pair performing the experimental task in the *multi-display* condition. The user at the wall display is aligning the picture frame with the assumed capturing position of the shown perspective. The other user is simultaneously searching for the next target image using a top-down miniature view of the scene on the tabletop. Right: Two exemplary views that had to be found and aligned at original scale.

7.3 Experimental Task

The study compared collaborative visual search performance in a virtual 3D city model. A low fidelity city model was used to control the recognizability of search targets in terms of their visual features. The building models featured roofs and pediments but no windows, textures, or other facade details. We captured a number of views in this city model as 2D pictures. All of these target images showed the scene at 1:1 scale. They were unique but not recognizable without visual comparison to the 3D city model (Fig. 8).

We placed additional assets, such as trees and car models, in the scene to create target perspectives with five levels of the difficulty as defined by the number, recognizability, and uniqueness of features. Two sets of ten images with overall comparable difficulty were compiled. The images in each set were sorted with increasing difficulty: 1. *very easy*, 2. and 3. easy, 4-6. *medium*, 7-9. *hard*, 10. *very hard*. Another set of six different images was prepared for training of the participants on the task and the operation of the user interfaces. We paid attention that all target locations were recognizable from an egocentric view at the life-size model as well as from an allocentric view at its miniature.

The 2D target images were shown on handheld virtual picture frames, a simplified version of the portal windows described in Section 6.2. These frames could display two different images at the same time, one on the front and one on the backside. We provided each user with such a picture frame and a 3D pick ray to facilitate parallel activities. Both frames showed the same two images. The pick rays and the picture frames were implemented as *Multi-Context Objects*. As such, they could be carried around between both displays and were automatically associated with the *Interaction Contexts* in which they were visible.

The participants were organized in pairs and had to solve the search task collaboratively (Fig. 8). Specifically, they were asked to 1. find the target views in the city model, 2. navigate to the shown locations at 1:1 scale, and 3. confirm the found

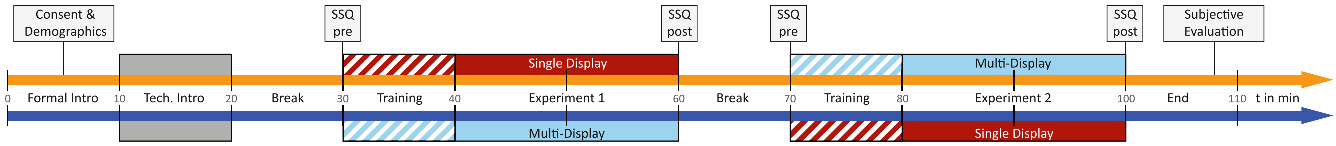


Fig. 9. Timeline of the experimental procedure. Subsequent to an introduction phase, the two display conditions (brown and light blue color) were tested in consecutive experiments. The order of conditions (orange and dark blue color) was balanced between the participant pairs.

perspectives. When a target perspective had been located, the 3D pick rays could be used to place a visual marker on the ground at this position (a large upright cylinder), which helped to reach it at 1:1 scale. Eventually, the picture frame with the corresponding view had to be placed precisely at the assumed capturing position and then confirmed with the pick ray. The position and orientation of the picture frame relative to the original capturing pose of the image was logged as a measure of accuracy. We asked our participants to agree on each adjusted view before confirmation. If a user group deemed a particular view too difficult to find, they could skip over to the next one.

7.4 Conditions

We tested the *multi-display* condition against a *single-display* baseline. The *single-display* condition involved only the wall display, while the tabletop offered an additional 3D interaction window in the *multi-display* condition. We chose the larger and more versatile wall display for the *single-display* condition, since the tasks could always be solved using the wall display alone. It supports the exploration of virtual environments from an egocentric perspective at 1:1 scale, but through scaling, users can also create a miniature of the scene for a better overview or fast travel between locations. The *multi-display* condition offered the benefits of two simultaneously available views and more freedom for the parallelization of activities. Both display conditions were compared using a within-subjects design. The order of conditions and the order of image sets was balanced between four experimental groups of participant pairs.

7.5 Procedure

The study was structured in three parts. An introduction and training session was followed by two recorded experiments, one for each condition (Fig. 9). Each part took about 30 minutes and we devised breaks of 10 minutes in between. The whole study took about 100 to 120 minutes.

Initially, the participants were asked to provide their consent on data recording and they had to fill a self-assessment questionnaire on skills and experiences we considered relevant to the task. Thereafter, they were introduced to the technical setup with an emphasis on 3D tracked stereo viewing, the virtual navigation techniques, and the 3D input devices. Both participants of each pair were introduced explicitly to all interaction tools. They could spend up to ten minutes to learn the system operation with available assistance until they felt confident. Eventually, the experimental task was explained in detail, demonstrated by the experimenter, and performed at least one time by the participants to prove their understanding.

Each test of a display condition started with a formal training of 10 minutes during which the experimenter encouraged the participants to explore the available options for subtask distribution in the respective condition. Thereafter, the recorded task required to find and align 10 target views with the assumed capturing position in the scene as fast and precisely as possible. They were given 20 minutes to perform all search tasks. If this time elapsed during the alignment phase of an identified perspective, we allowed to complete it within a tolerance of two extra minutes.

Before and after testing each display condition, participants filled a Simulator Sickness Questionnaire (SSQ) [51]. We also asked for subjective feedback after each condition, which included a quantitative assessment using the System Usability Scale (SUS) [52]. A final questionnaire captured subjective preferences with respect to task efficiency, collaboration support, and user satisfaction.

7.6 Participants

Our system was tested by 40 participants (11 female, 29 male) in groups of two. The participant's age ranged from 19 to 36 years ($M = 24.5$, $SD = 3.68$). They were recruited from our university campus, received 10 Euros allowance and were motivated to win a 80 Euro restaurant voucher if they found the most views in the shortest time.

Skills and prior experiences were captured with Likert-scales ranging from 1 (very bad or very little) to 5 (very good or very much). Most participants claimed to have good or very good spatial perception and orientation skills ($M = 3.94$, $SD = 0.65$). Reports on earlier experience with interactive 3D graphics applications were more diverse, ranging from 1 to 5 ($M = 2.8$, $SD = 1.34$). Most participants stated to know their partner well ($M = 4.03$, $SD = 0.92$), also from working or playing together ($M = 3.9$, $SD = 1.17$). One group reported to know each other very little and three others noted that had no or little experience of acting together.

7.7 Performance Results

Task performance was captured in terms of the number of perspectives that were found by each participant pair, their placement accuracy, and the total task completion time. We identified seven cases of erroneous image placement with a rotation error larger than 30° or a translation error larger than 50 m. These cases were not counted as successfully solved search tasks. In total, 311 successful subtasks were performed by the 20 participant pairs. Statistical results for both conditions are presented in the following with subscript-M (*multi-display*) and subscript-S (*single display*).

Despite the mentioned outliers, all participant pairs achieved very similar accuracy in both conditions (Rotation error: $M_S = 7.40°$, $SD_S = 5.74$ versus M_M $7.03°$, $SD_M = 4.61$; Translation error $M_S = 9.49$ m, $SD_S = 7.97$ versus M_M 9.53 m, $SD_M = 8.05$). In both conditions the translation error was most pronounced on the depth axis ($M = 8.71$m, $SD = 8.42$).

As expected in H1, our participants found and confirmed slightly less images in the *single-display* compared to the

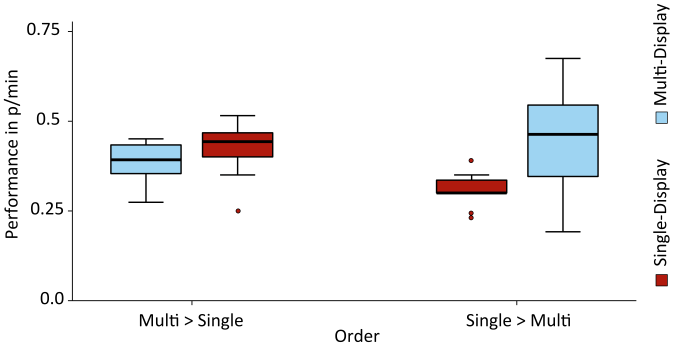


Fig. 10. Visual search performance (perspectives per minute) plotted against order of conditions and display conditions (brown and light blue color), the latter sorted by sequence.

multi-display condition (M_S = 7.40, SD_S = 1.70 versus M_M = 8.20, SD_M = 1.70). For more accurate comparisons, we computed a performance score, which expressed the ratio of perspectives found by a group and their task completion time. If all ten perspectives were found in the available 20 minutes, the resulting performance score would be 0.5 $^p/_{min}$ (perspectives per minute). According to a Shapiro-Wilk test, the resulting performance scores in all display conditions and with both image sets were normally distributed. The average performance score was 0.39 $^p/_{min}$ (SD = 0.10). The difference between both image sets was marginal (M_1 = 0.39 $^p/_{min}$, SD_1 = 0.11 versus M_2 = 0.38 $^p/_{min}$, SD_2 = 0.09) and not significant according to a t-test ($t_{(38)}$ = 0.330, p = 0.74, Cohens d = 0.10).

The performance difference between both display conditions was more pronounced (M_S = 0.36 $^p/_{min}$, SD_S = 0.08 versus M_M = 0.42 $^p/_{min}$, SD_M = 0.11). An ANOVA with display conditions as a within-subjects factor and order of conditions as a between-subjects factor revealed a significant effect for display condition ($F_{(1,18)}$ = 6.06, p = 0.024) with a large effect size (η_p^2 = 0.25). The order of conditions had no significant effect ($F_{(1,18)}$ = 0.52, p = 0.48, η_p^2 = 0.028), but we found a significant interaction of order with display condition that also had a large effect size ($F_{(1,18)}$ = 16.258, p = 0.001, η_p^2 = 0.48). This indicates that both experimental groups with different order of conditions performed on a comparable level overall, but training effects between the subsequent conditions differed significantly depending on the order. Using the *multi-display* condition after the *single-display* condition allowed a mean performance improvement of 44.5 percent (from 0.31 $^p/_{min}$ to 0.45 $^p/_{min}$), while the participants could only improve about 8.6 percent (from 0.39 $^p/_{min}$ to 0.42 $^p/_{min}$), when using both display setups in reverse order (Fig. 10).

7.8 Detailed Analysis of User Activities

During all experiments, the application state and all user input were logged at 60 Hz. Noise and jitter were removed with a moving average filter of 200 ms. The head-tracking data of the users was related to that of other input devices as well as the known positions and dimensions of the displays and the group navigation device in the workspace. This allowed us to derive more meaningful information about each participant's activities and display usage. Data on virtual view navigation was recorded for both displays individually. During each experiment, we accumulated deltas of the view orientation, translation and the scaling factor between subsequent frames. The resulting translation distance was scale-corrected to represent the visually perceived motion flow in the interaction space of the users. Our logs for the wall display revealed about twice as much view translation in the *multi-display* condition, but 27 percent less rotation and 45 percent less scaling. At the tabletop, if available, users extensively scaled the shown city model and changed its orientation. The applied view translations, instead, were comparably small. Miniature scale levels were used here most of the time, which implies short perceived translation distances.

7.8.1 Display Usage in the Multi-Display Condition

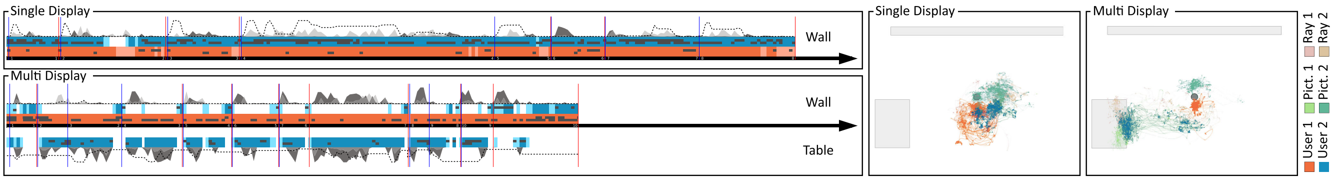
In the *multi-display* condition, both participants could decide which display to use and how closely they collaborate. Most groups used the wall display almost constantly during the task (M = 1148 sec. SD = 108) and they also used it together for about half of that time (M = 562 sec., SD = 224). The mean tabletop usage duration was shorter (M = 620 sec., SD=189) and it was only rarely used together (M=33 sec., SD = 67). This means that our participants focused their joint activities at the wall display, which was left occasionally by one of them to work on alone at the tabletop.

We found large differences between participant pairs in their strategy of activity distribution. While some sticked together most of the time, others preferred to work separately. For pairs who worked with both displays in their second test condition, the ratio of parallel display usage seems to correlate with their task performance (Fig. 12, orange data points). Those who started with the *multi-display* conditions did not become proficient enough with the overall task to take advantage of more parallel work (Fig. 12, blue data points). Three of the four participant pairs who reported little experience of mutual collaboration also showed the lowest performance (see dotted lines in Fig. 12).

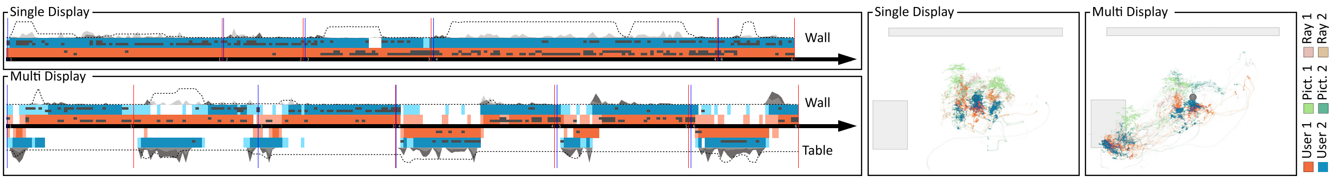
7.8.2 Exemplary Activity Visualizations

We created timeline visualizations and motion maps of all recorded user activities during the experimental tasks (Fig. 11). The typical pattern of workload distribution in the *single-display* condition was that one user primarily controlled the navigation, while the other operated the handheld input devices. Some participants swapped roles during the task. This type of subtask distribution had not much potential for increased efficiency since the bulk of interaction was view navigation to locate and reach the target perspectives. In the *multi-display* condition, we observed different usage patterns of the two displays. Some groups used the wall and the tabletop displays together most of the time, while others exploited the possibility of parallel scene navigation at different displays. Fig. 11 shows two representative participant groups for the mentioned strategies.

Also, the aforementioned differences in the amount of virtual viewpoint navigation between both conditions can be seen in these diagrams. In the *single-display* condition, both groups frequently altered the scaling between 1:1 and miniaturization (see dashed scale-level lines in Figs. 11a and 11b). In the *multi-display* condition, instead, less scaling occurred at the wall display. Most often it was used at 1:1 scale, while the tabletop was preferred for visual search in miniature scene views.



(a) Activity visualization of participant pair *3*. In the *single-display* condition (upper timeline), the blue user primarily operated the handheld input devices, while the orange user was navigating. Eight perspectives were found. In the *multi-display* condition (lower timeline), a strategy of highly parallel display usage was applied. The blue user searched for perspectives at the tabletop display and joined her partner at the wall display in the final picture alignment phases. The group found the maximum number of 10 perspectives in less than 15 min.



(b) Activity visualization of participant pair *11*. This group was working closely together in both conditions. Roles (navigator and tool operator) were swapped after half the experiment duration in the *single-display* condition (upper timeline). In the *multi-display* condition (lower timeline), both displays were used, but mostly together. Phases of parallel display usage are rarely found. In both conditions, 5 perspectives were found.

Fig. 11. Activity visualization for two participant pairs (both started with the *single-display* condition). The timelines on the left hand side (range: 0-22 min) illustrate user activities over the duration of an experiment (above: *single-display* condition (S), below: *multi-display* condition (M)). Activities associated to the wall display (W) are shown above the separating black timeline, while those related to the tabletop display (T) are shown below in mirrored ordering. Blue and orange areas illustrate the display visibility for the respective user over the course of the experiment (less saturated if navigation input is out of reach). Gray spots within these areas illustrate motion of the handheld picture frames and pointers associated with the corresponding user and display. Layered area plots above and below the visibility graphs show normalized view rotation (light gray) and translation (dark gray). The dashed line shows the absolute scale between level 1:1 and the maximum scale level per display (W=1:300, T=1:1000). The vertical lines indicate the start (blue) and end (red) of subtasks. The two motion maps on the right hand side illustrate the same activities for both conditions in a top-down view of the physical workspace. The gray rectangles in these maps represent the two displays.

7.9 Simulator Sickness

Participants filled in a simulator sickness questionnaire before and after their experience with our system in both display conditions. This resulted in four subsequent measures of simulator sickness (*S-pre* and *S-post* for the *single-display* condition, *M-pre* and *M-post* for the *multi-display* condition). The computed SSQ scores (N-ausea, O-culomotor disturbance, D-isorientation, and T-otal; see [51]) were not normally distributed but highly right-skewed, hence, non-parametric tests were applied for statistical analysis. A Mann-Whitney U-test revealed small but significant effects of order of conditions on nausea symptoms (N) ($z = 2.42$, $p = 0.016$, $r = 0.27$) and total severity (T) ($z = 2.31$), $p = 0.021$, $r = 0.26$). All further tests were thus performed separately for both experimental groups with different order of conditions.

Friedman tests revealed significant differences within the four subsequent measures for both experimental groups and in all four categories (all $p < 0.05$, most $p < 0.01$). The three more specific scores (N,O, and D) followed the general trend of total severity scores (T). Post-hoc Wilcoxon signed rank tests were only performed on the latter. We compared *pre* and *post* measures of both display conditions for both experimental groups with different order of conditions independently (Bonferroni-adjusted $\alpha = 0.0125$). For both experimental groups, we found significant differences between *pre* and *post* measures in the *single-display* condition with medium effect sizes ($z = 2.99$, $p = 0.003$, $r = 0.33$ and $z = 3.74$, $p < 0.001$, $R = 042$). The differences between *pre* and *post* measures in the *multi-display* condition, instead, were not significant (Fig. 13).

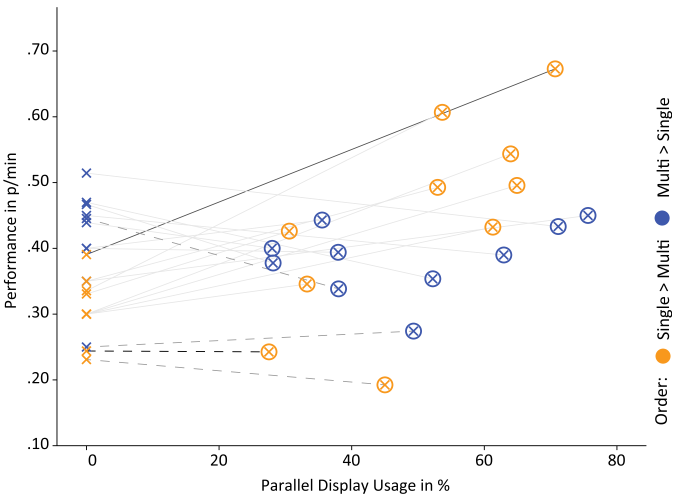


Fig. 12. Visual search performance of each participant pair with both display conditions (linked by grey lines) and plotted against the ratio of parallel display usage. The *single-display* condition (crosses) did not enable parallel display usage, while in the *multi-display* condition (encircled crosses), this possibility was exploited to different degrees. Learning effects between subsequent experiments depended on the order of conditions (color coded). The dotted lines represent the four groups, who knew each other only briefly. The exemplary groups 3 and 11 in Fig. 11 are highlighted with a darker line between conditions.

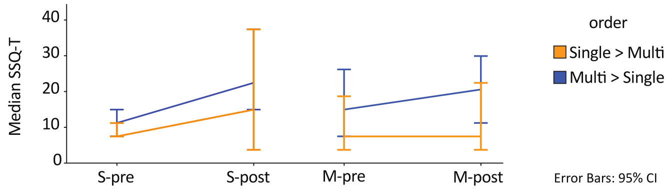


Fig. 13. Increase of the simulator sickness symptoms (median) during the experiments plotted against both display conditions (left: *single-display* condition, right: *multi-display* condition) and order of conditions (color coded). Working in the *single-display* setup resulted in a large increase of sickness symptoms. No significant increase of symptoms was reported in the *multi-display* condition.

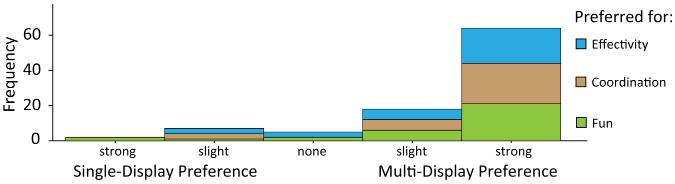


Fig. 14. Stacked histogram of subjective preference ratings. The majority of the users reported a strong preference for the *multi-display* setup regarding efficiency, coordination, and fun.

7.10 Subjective Usability Evaluation

Both conditions scored almost equally well on the SUS with 80.13 (SD = 11.52) for the *multi-display* setup and 74.94 (SD = 13.62) for the *single-display* condition. We also asked participants to express on five-point scales 1. which condition better supported their work tasks, 2. which better supported their collaborative coordination, and 3. which of both they considered to be more fun to use (1 = strong *single-display* tendency, 5 = strong *multi-display* tendency) (Fig. 14).

Twenty-five of our 40 participants found the *multi-display* system much more effective (5). Eight of the remaining expressed a moderate preference for this condition (4), four were undecided (3) and three expressed a moderate preference for the *single-display* condition (1). Twenty-nine test users found that the *multi-display* system supported their coordination with their partners much better (5). Seven users expressed a moderate preference in that respect and the remaining four expressed a slight preference for the *single-display* system (2). Twenty-six users found the *multi-display* system to be much more fun to use (5), nine stated it was slightly more fun (4), two were undecided (3), one reported to have slightly more fun with the *single-display* system (2) and only two users considered the latter to be much more fun to use (1). The responses to these questions clearly support our hypothesis H3. Further user comments emphasized the particular importance of continuous interaction capabilities across both displays available with the *Multi-Context Picture Frame* and the *Multi-Context Pick Ray*.

7.11 Discussion

The study results confirmed our three hypotheses. On average, our participant pairs achieved significantly higher performance in a visual search task if they could use two independent 3D displays in parallel (H1). This *multi-display* condition also resulted in significantly less simulator sickness symptoms after extended use of the virtual reality setup (H2). Moreover, almost all participants of our study preferred the suggested combination of 3D displays compared to the *single-display* setup (H3).

The visual search performance of participant pairs in our study was also dependent on learning effects. Significant interaction effects between display condition and the order of conditions were observed. Apparently, it requires proficiency with the task and the system to take advantage of parallel activities at the different displays. The largest performance improvements were achieved by groups who used the *multi-display* after the *single-display* condition. Not much of a task improvement could be observed, instead, if the conditions were tested the other order. We also note that the necessary coordination to achieve benefits of cooperative interaction depends on the participants' familiarity with each other. Three of the four participant pairs, who stated to know each other only briefly, were those with the lowest visual search performance in both display conditions.

Significant differences of simulator sickness symptoms support our hypothesis H2 that collaborative visual search and virtual navigation with a single shared display increases the risk of such effects. However, the reasons may be other than expected. We identified the more passive and the more dominant users on the navigation controls during the *single-display* condition from our activity logs, but we could not find an interaction with the obtained SSQ scores. We also assumed that providing a secondary display, for an allocentric overview, would decrease the demand for egocentric viewpoint navigation on the wall display. The accumulated scale and orientation changes on the wall were indeed higher in the *single-display* condition, but the amount of translation was higher in the *multi-display* condition. Perhaps scaling and turning have a stronger impact on sickness symptoms than viewpoint translation, but without a measure of the perceived visual flow, this is mere speculation.

Nevertheless, we observed that some users who reported a notable increase of sickness symptoms in the *single-display* condition could avoid these issues by working primarily at the tabletop display in the *multi-display* condition.

8 Conclusion and Future Work

We presented a novel collaborative VR setup consisting of a 3D wall display, a 3D tabletop, and handheld 3D viewing windows in a shared workspace. The displays are synchronized and provide three users with individual stereoscopic views. The different affordances of the display types as well as the demand for closely and loosely coupled collaboration motivated us to use them as separate viewing windows into the same virtual scene. However, with respect to the physical co-location of the displays in the shared workspace, it is nonetheless desirable to support a coherent interaction space across the independent views.

The suggested design of *Multi-Context Objects* facilitates the implementation of a large variety of 3D interaction tools and techniques with functional and perceptual consistency in multi-window VR environments. It presents a solution to the problem of managing the seamless representation of users, input tools and manipulated objects across multiple independent 3D views.

A formal user study based on a collaborative visual search task revealed significant advantages of our multi-display environment compared to a single display setup. We will continue the analysis of behavioral patterns and the applied strategies for task distribution in multi-window environments to advance our collaborative 3D user interfaces.

We are also continuously extending the system's functionality with novel interaction tools which confirms the versatility of the *Multi-Context Objects* approach. Future work will include the design and smooth integration of interaction techniques that offer more explicit user control over context selection and behavior during transitions. In order to support a wider range of applications, we will add further displays with specific capabilities as well as extended support for remote collaboration in distributed setups across multiple independent workspaces.

We are convinced that the seamless operation of 3D interaction tools across real and virtual multi-window VR systems significantly improves the usability of such advanced collaborative applications and that its a key feature expected by most users.

ACKNOWLEDGMENTS

This research has received funding from the European Unions Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2) and from the German Federal Ministry of Education and Research (BMBF) under grant 03PSIPT5A (project Provenance Analytics).

REFERENCES

[1] M. Agrawala, A. C. Beers, I. McDowall, B. Fröhlich, M. Bolas, and P. Hanrahan, "The two-user responsive workbench: Support for collaboration through individual views of a shared space," in *Proc. SIGGRAPH*, 1997, pp. 327–332. [Online]. Available: http://dx.doi.org/10.1145/258734.258875

[2] A. Kulik, A. Kunert, S. Beck, R. Reichel, R. Blach, A. Zink, and B. Froehlich, "C1x6: A stereoscopic six-user display for co-located collaboration in shared virtual environments," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 188:1–188:12, Dec. 2011. [Online]. Available: http://doi.acm.org/10.1145/2070781.2024222

[3] S. D. Scott, M. Sheelagh, T. Carpendale, and K. M. Inkpen, "Territoriality in collaborative tabletop workspaces," in *Proc. Proc. ACM Conf. Comput. Supported Cooperative Work*, 2004, pp. 294–303. [Online]. Available: http://doi.acm.org/10.1145/1031607.1031655

[4] W. Chen, N. Ladeveze, C. Clavel, D. Mestre, and P. Bourdot, "User cohabitation in multi-stereoscopic immersive virtual environment for individual navigation tasks," in *Proc. Virtual Reality*, 2015, pp. 47–54. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/VR.2015.7223323

[5] C. Gutwin and S. Greenberg, "A descriptive framework of workspace awareness for real-time groupware," *Comput. Supported Cooperative Work*, vol. 11, no. 3, pp. 411–446, Nov. 2002. [Online]. Available: http://dx.doi.org/10.1023/A:1021271517844

[6] J. Bowers, J. O'Brien, and J. Pycock, "Practically accomplishing immersion: Cooperation in and for virtual environments," in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 1996, pp. 380–389. [Online]. Available: http://doi.acm.org/10.1145/240080.240349

[7] J. Hindmarsh, M. Fraser, C. Heath, S. Benford, and C. Greenhalgh, "Object-focused interaction in collaborative virtual environments," *ACM Trans. Comput.-Hum. Interact.*, vol. 7, no. 4, pp. 477–509, Dec. 2000. [Online]. Available: http://doi.acm.org/10.1145/365058.365088

[8] A. S. García, J. P. Molina, P. González, D. Martínez, and J. Martínez, "An experimental study of collaborative interaction tasks supported by awareness and multimodal feedback," in *Proc. 8th Int. Conf. Virtual Reality Continuum Appl. Ind.*, 2009, pp. 77–82. [Online]. Available: http://doi.acm.org/10.1145/1670252.1670270

[9] K. Baker, S. Greenberg, and C. Gutwin, "Empirical development of a heuristic evaluation methodology for shared workspace groupware," in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 2002, pp. 96–105. [Online]. Available: http://doi.acm.org/10.1145/587078.587093

[10] M. Tobiasz, P. Isenberg, and S. Carpendale, "Lark: Coordinating co-located collaboration with information visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 6, pp. 1065–1072, Dec. 2009.

[11] P. Isenberg, D. Fisher, S. A. Paul, M. R. Morris, K. Inkpen, and M. Czerwinski, "Co-located collaborative visual analytics around a tabletop display," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 5, pp. 689–702, May 2012. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/TVCG.2011.287

[12] M. R. Jakobsen and K. Hornbæk, "Up close and personal: Collaborative work on a high-resolution multitouch wall display," *ACM Trans. Comput.-Hum. Interact.*, vol. 21, no. 2, pp. 11:1–11:34, Feb. 2014. [Online]. Available: http://doi.acm.org/10.1145/2576099

[13] A. Tang, M. Tory, B. Po, P. Neumann, and S. Carpendale, "Collaborative coupling over tabletop displays," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2006, pp. 1181–1190. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124950

[14] W. Chen, C. Clavel, N. Férey, and P. Bourdot, "Perceptual conflicts in a multi-stereoscopic immersive virtual environment: Case study on face-to-face interaction through an Avatar," *Presence*, vol. 23, no. 4, pp. 410–429, Nov. 2014. [Online]. Available: http://doi.org/10.1162/PRES_a_00209

[15] E. Tse, J. Histon, S. D. Scott, and S. Greenberg, "Avoiding interference: How people use spatial separation and partitioning in SDG workspaces," in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 2004, pp. 252–261. [Online]. Available: http://doi.acm.org/10.1145/1031607.1031647

[16] C. Forlines and C. Shen, "DTLens: Multi-user tabletop spatial data exploration," in *Proc. 18th Annu. ACM Symp. User Interface Softw. Technol.*, 2005, pp. 119–122. [Online]. Available: http://doi.acm.org/10.1145/1095034.1095055

[17] P. Isenberg and S. Carpendale, "Interactive tree comparison for co-located collaborative information visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 6, pp. 1232–1239, Dec. 2007. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/TVCG.2007.70568

[18] D. Klinkhammer, M. Nitsche, M. Specht, and H. Reiterer, "Adaptive personal territories for co-located tabletop interaction in a museum setting," in *Proc. ACM Int. Conf. Interactive Tabletops Surfaces*, 2011, pp. 107–110. [Online]. Available: http://doi.acm.org/10.1145/2076354.2076375

[19] A. Kunert, A. Kulik, S. Beck, and B. Froehlich, "Photoportals: Shared references in space and time," in *Proc. 17th ACM Conf. Comput. Supported Cooperative Work Social Comput.*, 2014, pp. 1388–1399. [Online]. Available: http://doi.acm.org/10.1145/2531602.2531727

[20] W. Stuerzlinger, L. Zaman, A. Pavlovych, and J.-Y. Oh, "The design and realization of CoViD: A system for collaborative virtual 3D design," *Virtual Reality*, vol. 10, no. 2, pp. 135–147, Oct. 2006. [Online]. Available: https://doi.org/10.1007/s10055-006-0048-0

[21] D. Wigdor, H. Jiang, C. Forlines, M. Borkin, and C. Shen, "WeSpace: The design development and deployment of a walk-up and share multi-surface visual collaboration system," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2009, pp. 1237–1246. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518886

[22] C. Pirchheim, M. Waldner, and D. Schmalstieg, "Deskotheque: Improved spatial awareness in multi-display environments," in *Proc. Virtual Reality*, Mar. 2009, pp. 123 –126. [Online]. Available: http://dx.doi.org/10.1109/VR.2009.4811010

[23] W. McGrath, B. Bowman, D. McCallum, J. D. Hincapié-Ramos, N. Elmqvist, and P. Irani, "Branch-explore-merge: Facilitating real-time revision control in collaborative visual exploration," in *Proc. ACM Int. Conf. Interactive Tabletops Surfaces*, 2012, pp. 235–244. [Online]. Available: http://doi.acm.org/10.1145/2396636.2396673

[24] M. Spindler, W. Büschel, and R. Dachselt, "Use your head: Tangible windows for 3D information spaces in a tabletop environment," in *Proc. ACM Int. Conf. Interactive Tabletops Surfaces*, 2012, pp. 245–254. [Online]. Available: http://doi.acm.org/10.1145/2396636.2396674

[25] T. Seyed, M. Costa Sousa , F. Maurer, and A. Tang, "Skyhunter: A multi-surface environment for supporting oil and gas exploration," in *Proc. ACM Int. Conf. Interactive Tabletops Surfaces*, 2013, pp. 15–22. [Online]. Available: http://doi.acm.org/10.1145/2512349.2512798

[26] C. T. Dang and E. André, "A framework for the development of multi-display environment applications supporting interactive real-time portals," in *Proc. ACM SIGCHI Symp. Eng. Interactive Comput. Syst.*, 2014, pp. 45–54. [Online]. Available: http://doi.acm.org/10.1145/2607023.2607038

[27] T. Ni, G. S. Schmidt, O. G. Staadt, M. A. Livingston, R. Ball, and R. May, "A survey of large high-resolution display technologies, techniques, and applications," in *Proc. Virtual Reality*, Mar. 2006, pp. 223–236. [Online]. Available: https://doi.org/10.1109/VR.2006.20

[28] R. Stoakley, M. J. Conway, and R. Pausch, "Virtual reality on a WIM: Interactive worlds in miniature," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 1995, pp. 265–272. [Online]. Available: http://dx.doi.org/10.1145/223904.223938

[29] J. S. Pierce, B. C. Stearns, and R. Pausch, "Voodoo dolls: Seamless interaction at multiple scales in virtual environments," in *Proc. Symp. Interactive 3D Graph.*, 1999, pp. 141–145. [Online]. Available: http://doi.acm.org/10.1145/300523.300540

[30] S. L. Stoev and D. Schmalstieg, "Application and taxonomy of through-the-lens techniques," in *Proc. Proc. ACM Symp. Virtual Reality Softw. Technol.*, 2002, pp. 57–64. [Online]. Available: http://doi.acm.org/10.1145/585740.585751

[31] P. B. Raghupathy and C. W. Borst, "Investigation of secondary views in a multimodal VR environment: 3D lenses, windows, and mirrors," in *Proc. Int. Symp. Visual Comput.*, 2011, pp. 180–189. [Online]. Available: https://doi.org/10.1007/978-3-642-24031-7_18

[32] D. Coffey, N. Malbraaten, T. B. Le, I. Borazjani, F. Sotiropoulos, A. G. Erdman, and D. F. Keefe, "Interactive slice WIM: Navigating and interrogating volume data sets using a multisurface, multi-touch VR interface," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 10, pp. 1614–1626, Oct. 2012.

[33] H. Hua, L. Brown, and C. Gao, "Scape: Supporting stereoscopic collaboration in augmented and projective environments," *IEEE Comput. Graph. Appl.*, vol. 24, no. 1, pp. 66–75, Jan./Feb. 2004. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/MCG.2004.10004

[34] J. Rekimoto and M. Saitoh, "Augmented surfaces: A spatially continuous work space for hybrid computing environments," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 1999, pp. 378–385. [Online]. Available: http://doi.acm.org/10.1145/302979.303113

[35] P. Baudisch, E. Cutrell, K. Hinckley, and R. Gruen, "Mouse ether: Accelerating the acquisition of targets across multi-monitor displays," in *Proc. Extended Abstracts Hum. Factors Comput. Syst.*, 2004, pp. 1379–1382. [Online]. Available: http://doi.acm.org/10.1145/985921.986069

[36] K. Hinckley, G. Ramos, F. Guimbretiere, P. Baudisch, and M. Smith, "Stitching: Pen gestures that span multiple displays," in *Proc. Work. Conf. Adv. Visual Interfaces*, 2004, pp. 23–31. [Online]. Available: http://doi.acm.org/10.1145/989863.989866

[37] M. A. Nacenta, S. Sallam, B. Champoux, S. Subramanian, and C. Gutwin, "Perspective cursor: Perspective-based interaction for multi-display environments," in *Proc. Conf. Human Factors Comput. Syst.*, 2006, pp. 289–298. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124817

[38] A. D. Wilson and H. Benko, "Combining multiple depth cameras and projectors for interactions on, above and between surfaces," in *Proc. 23nd Annu. ACM Symp. User Interface Softw. Technol.*, 2010, pp. 273–282. [Online]. Available: http://doi.acm.org/10.1145/1866029.1866073

[39] R. Xiao, M. A. Nacenta, R. L. Mandryk, A. Cockburn, and C. Gutwin, "Ubiquitous cursor: A comparison of direct and indirect pointing feedback in multi-display environments," in *Proc. Graph. Interface*, 2011, pp. 135–142. [Online]. Available: http://dl.acm.org/citation.cfm?id=1992917.1992939

[40] H. Benko, E. W. Ishak, and S. Feiner, "Cross-dimensional gestural interaction techniques for hybrid immersive environments," in *Proc. Virtual Reality*, Mar. 2005, pp. 209–216. [Online]. Available: https://doi.org/10.1109/VR.2005.1492776

[41] A. Kulik, A. Kunert, M. Keil, and B. Froehlich, "RST 3D: A comprehensive gesture set for multitouch 3D navigation," in *Proc. Virtual Reality*, Mar. 2018, pp. 363–370. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/VR.2018.8447554

[42] S. Beck, A. Kunert, A. Kulik, and B. Froehlich, "Immersive group-to-group telepresence," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 4, pp. 616–625, Apr. 2013. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2013.33

[43] T. M. H. Reenskaug, "The original MVC reports," 1979. [Online]. Available: http://urn.nb.no/URN:NBN:no-14314

[44] J. Coutaz, "PAC: An object oriented model for implementing user interfaces," *SIGCHI Bull.*, vol. 19, no. 2, pp. 37–41, Oct. 1987. [Online]. Available: http://doi.acm.org/10.1145/36111.1045592

[45] T. Duval and C. Fleury, "PAC-C3D: A new software architectural model for designing 3D collaborative virtual environments," in *Proc. ICAT*, 2011, pp. 53–60. [Online]. Available: https://hal.inria.fr/inria-00636143

[46] G. Calvary, J. Coutaz, and L. Nigay, "From single-user architectural design to PAC*: A generic software architecture model for CSCW," in *Proc. ACM SIGCHI Conf. Hum. Factors Comput. Syst.*, 1997, pp. 242–249. [Online]. Available: http://doi.acm.org/10.1145/258549.258717

[47] J. S. Pierce, A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M. R. Mine, "Image plane interaction techniques in 3D immersive environments," in *Proc. Symp. Interactive 3D Graph.*, 1997, pp. 39–ff. [Online]. Available: http://doi.acm.org/10.1145/253284.253303

[48] K. Kiyokawa and H. Takemura, "A tunnel window and its variations: Seamless teleportation techniques in a virtual environment," *HCI Int.*, 2005.

[49] A. Kulik, A. Kunert, S. Beck, C.-F. Matthes, A. Schollmeyer, A. Kreskowski, B. Fröhlich, S. Cobb, and M. D'Cruz, "Virtual valcamonica: Collaborative exploration of prehistoric petroglyphs and their surrounding environment in multi-user virtual reality," *Presence: Teleoper. Virtual Environ.*, vol. 26, no. 3, pp. 297–321, May 2018. [Online]. Available: https://doi.org/10.1162/pres_a_00297

[50] S. Schneegans, F. Lauer, A. Bernstein, A. Schollmeyer, and B. Froehlich, "guacamole - An extensible scene graph and rendering framework based on deferred shading," in *Proc. IEEE 7th Workshop Softw. Eng. Archit. Realtime Interactive Syst.*, 2014, pp. 35–42. [Online]. Available: doi.ieeecomputersociety.org/10.1109/SEARIS.2014.7152799

[51] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness," *Int. J. Aviation Psychology*, vol. 3, no. 3, pp. 203–220, 1993. [Online]. Available: https://doi.org/10.1207/s15327108ijap0303_3

[52] J. Brooke, "SUS-a quick and dirty usability scale," *Usability Eval. Ind.*, vol. 189, no. 194, pp. 4–7, 1996.

Andre Kunert is working toward the PhD degree in the Virtual Reality and Visualization Research Group, Bauhaus-Universität Weimar. His research interests include 3D interaction techniques, collaboration in co-located and distributed VE, and software structures for user interfaces.

Tim Weissker is working toward the PhD degree in the Virtual Reality and Visualization Research Group, Bauhaus-Universität Weimar. His research interests include interaction techniques for projection-based multi-user VR systems, head-mounted display setups for multiple collocated and remote user groups, and group navigation techniques in social VR.

Bernd Froehlich is a chair of the Virtual Reality and Visualization Research Group at Bauhaus-Universität Weimar. His research interests include real-time rendering, visualization, 2D and 3D input devices, 3D interaction techniques, display technology, and support for collaboration in co-located and distributed VE.

Alexander Kulik is a postdoctoral researcher with the Virtual Reality and Visualization Research Group, Bauhaus-Universität Weimar. He is enthusiastic about user interfaces that leverage human cognitive, manual, and social skills for expressive interactions. His current research focuses on interactive systems for bimanual and social cooperation.