Comparing and Combining Virtual Hand and Virtual Ray Pointer Interactions for Data Manipulation in Immersive Analytics

Jorge Wagner, Student Member, IEEE, Wolfgang Stuerzlinger, Member, IEEE, and Luciana Nedel, Member, IEEE

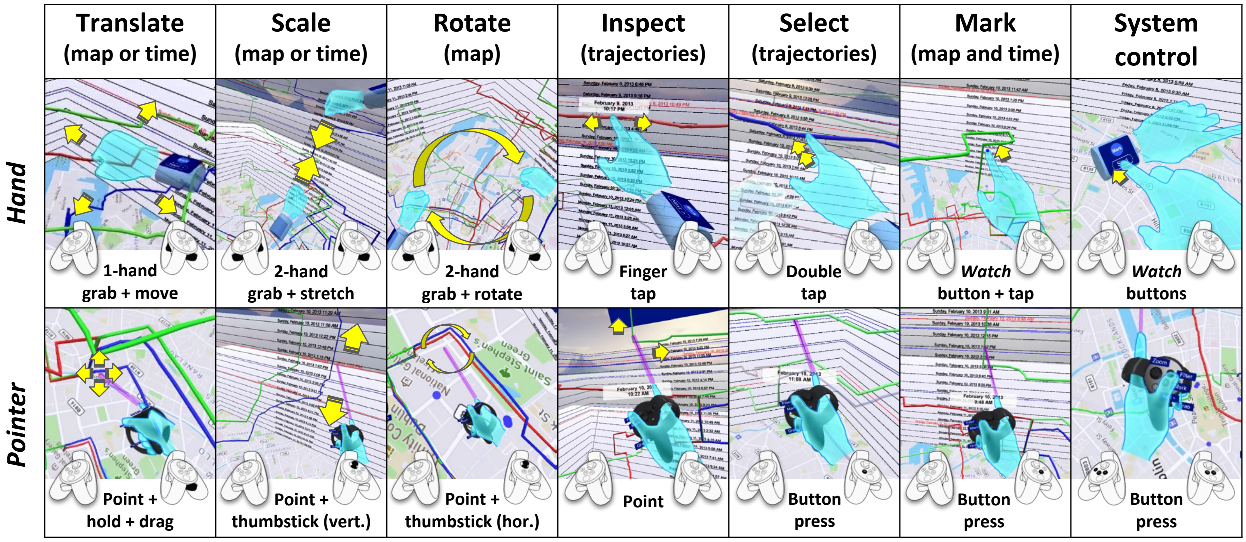

Fig. 1. Virtual hand and virtual ray pointer are the most common interaction modes in Immersive Analytics. With them, users perform data manipulations either with respect to their hands' positions or to remotely indicated points. In our study apparatus, all actions are supported by both modes, which correspond to different mappings for the hand controllers. *Virtual Smartwatches* replace controller buttons to offer additional functionality in the virtual hand approach. We investigate how each mode individually and a seamless combination of both affect user performance and preferences. Arrows added for illustration and hands and rays highlighted for clarity.

Abstract—In this work, we evaluate two standard interaction techniques for Immersive Analytics environments: virtual hands, with actions such as grabbing and stretching, and virtual ray pointers, with actions assigned to controller buttons. We also consider a third option: seamlessly integrating both modes and allowing the user to alternate between them without explicit mode switches. Easy-to-use interaction with data visualizations in Virtual Reality enables analysts to intuitively query or filter the data, in addition to the benefit of multiple perspectives and stereoscopic 3D display. While many VR-based Immersive Analytics systems employ one of the studied interaction modes, the effect of this choice is unknown. Considering that each has different advantages, we compared the three conditions through a controlled user study in the spatio-temporal data domain. We did not find significant differences between hands and ray-casting in task performance, workload, or interactivity patterns. Yet, 60% of the participants preferred the mixed mode and benefited from it by choosing the best alternative for each low-level task. This mode significantly reduced completion times by 23% for the most demanding task, at the cost of a 5% decrease in overall success rates.

Index Terms—Immersive analytics, interaction methods, virtual hand, virtual ray pointer, space-time cube

1 Introduction

Immersive Analytics (IA) investigates innovative Virtual Reality (VR) interfaces for data analysis tasks [32]. Previous work has shown that at least for some types of data, including spatio-temporal data, analysts are able to obtain a more accurate understanding through 3D representations in immersive environments [48, 51] and can gain new perspectives from navigating inside or around them. However, efficient interaction is essential to make it easy to explore the data, e.g., to analyze data points in detail or to minimize the need for navigation. Recently proposed IA systems often employ either a local mode of interaction through the virtual hand metaphor [14, 22, 49] or remote interaction through virtual ray pointing and controller buttons [17, 35, 50, 52] (see Sect. 2). In this work, we investigate whether these two modes affect user performance and subjective measures during IA sessions. Additionally, we investigate a seamless combination of both metaphors, which allows the user to select the best mode for each action or task.

Virtual hand actions afford an intuitive 3D user interface through a direct one-to-one mapping to real-world actions [11]. They can be used to, for example, move data representations with grabbing actions, obtain different points of view through hand rotations, or change the data scale with stretching actions. Further, selection and inspection can be mapped to finger taps. Such local interaction can also be used to manipulate data axes as primitives to generate new representations [13]. However, virtual hand actions incur additional effort when the user needs to interact with elements beyond arm's reach, as navigation is then necessary. They can also be less accurate for the selection of specific points, particularly in cluttered scenarios [48], where it often

- *Jorge Wagner and Luciana Nedel are with the Federal University of Rio Grande do Sul. E-mail: jawfilho@inf.ufrgs.br, nedel@inf.ufrgs.br.*
- *Wolfgang Stuerzlinger is with Simon Fraser University. E-mail: w.s@sfu.ca.*

Manuscript received 9 Sept. 2020; revised 15 Dec. 2020; accepted 8 Jan. 2021. Date of publication 22 Mar. 2021; date of current version 7 Apr. 2021. Digital Object Identifier no. 10.1109/TVCG.2021.3067759

becomes necessary to scale the data up and down repeatedly.

On the other hand, **virtual ray pointers** based on ray-casting enable easy remote selection of targets and afford accurate selection at shorter distances. Normally, controller buttons are then used to activate different actions, including accessing points of interest with *raycast-with-reeling* [7]. Such pointers can also be used to quickly select multiple targets using a brushing mechanism [23].

Building on previous work that demonstrated the benefits of IA for this domain [51], we implemented a Space-Time Cube visualization [29] use case (Sect. 3) to carry out our investigation. Inspired by the interaction design of recent studies and commercial VR applications, we implemented virtual hand and virtual ray pointer modes, with both supporting the same functionality. A virtual hand can be used to touch data points or buttons on a *Virtual Smartwatch*, or to grab and manipulate the dataset (by pressing triggers on the non-visible hand controller). In the pointing mode, virtual controller models are displayed, and virtual rays start from the tips of the index fingers. Pressing triggers then allows the user to remotely hold and drag surfaces and data trajectories, while the controller buttons and joysticks are mapped to selection and manipulation actions.

Finally, we also added another mode that enables users to seamlessly switch between both forms of interaction, in a **mixed interaction** mode. This mode selects actions based on the user interaction with the hand controllers, an approach that some recent VR systems adopted but which has not yet been explored in IA applications (see Sect. 2.3).

In a controlled user study, we invited 15 participants to perform data analysis tasks using all three interaction modes in a room-scale environment (Sect. 4). Here, we analyze the results in terms of performance, workload, and interactivity metrics (Sect. 5) and discuss how the outcomes inform other work (Sect. 6). We also explore how the different modes were used in combination in the mixed mode and how user performance and feedback correlate with characteristics such as prior VR experience and gaming habits.

The main contributions of this paper are:

- A controlled user study investigating the effect of different interaction metaphors in Immersive Analytics, comparing virtual hand, virtual ray pointer, and mixed modes.
- A discussion on how different forms of interaction (or combinations thereof) support data analysis tasks.
- Design recommendations for future IA systems.

2 Related Work

This section presents an overview of prior research on forms of interaction for immersive virtual environments (IVEs) and IA applications.

2.1 Interaction in IVEs

Interaction methods and metaphors have been an important research topic for the design of IVEs. Among others, Poupyrev et al. [42], Mine [33], and Bowman [6] discussed the importance of efficient and enjoyable interfaces for the acceptance of IVEs.

Mine [33] categorized fundamental forms of interaction and described local and at-a-distance as the two primary selection techniques. He also discussed the advantages of alternative manipulation forms. While hand actions, more related to the manipulation of objects in the real world, were described as more intuitive to change position, orientation, and rotation, physical or virtual controls allow more precise positioning and rotation. In our work, we investigate the effect of this difference for IA environments.

A study by Poupyrev et al. [42] compared two techniques for remote selection and manipulation of objects in IVEs: a virtual ray pointer and a virtual hand technique with a non-linear mapping of the real hand position (called *Go-Go*) [41]. While *Go-Go* was better for remote selection of small points, both were comparable for local selection and object repositioning at constant distances. Since altering the length of the ray, i.e., *raycast-with-reeling* [7], was not enabled, the pointer was less effective for manipulating the distance to an object.

Bowman [6] presented a series of interaction guidelines, including the option of redundant techniques for a single task since individual differences played a large role in user performance in his studies. He also recommended the use of ray-casting when the speed of remote selection was important, as well as allowing direct manipulations with a virtual hand. In this spirit, Bowman and Hodges [7] designed the *HOMER* (Hand-centered Object Manipulation Extending Ray-Casting) hybrid technique, which combined the best features of six others. In their approach, after grabbing an object with the virtual ray pointer, a virtual hand moves to the object position to allow hand-centered manipulation. While we do not assess this specific technique (see Sect. 2.3), our mixed mode follows a similar rationale.

2.2 Interaction in Immersive Analytics

Current IA systems use a wide variety of input methods, as identified by a recent survey [18]. Büschel et al. [9] also reviewed interaction methods for IA applications, focusing on *natural* user interfaces. They championed direct manipulation of information displays as a way of increasing the sense of immersion and reducing Norman's Gulf of Execution [36], noting that interactions at-a-distance are limited by tracking accuracy and hand stability. They also pointed out that no single input method is suitable for all tasks and that integrating different techniques is a key challenge, which our work investigates.

One common interaction metaphor in IA is the use of virtual ray pointers, typically originating from virtual controller models. The flexibility of this approach, which resembles a 3D version of the familiar mouse cursor metaphor, allows quick selection and inspection of points at arbitrary distances, as well as operation of menus. Holding a trigger or button is often mapped to dragging data objects or surfaces. Examples of IA applications supporting ray-pointing include systems for the visualization of geo-located data points [35], maps and globes [26, 52], and 3D scatterplots [17, 50]. In *FiberClay* [23], the pointer was extended to enable the composition of more complex visual queries for large volumes of 3D movement trajectories, allowing brushing with either a single ray or two rays simultaneously—to identify trajectories that start in one specific place and end in another specific one.

The other option, i.e., the virtual hand metaphor, is also frequently implemented for local interaction in IA applications, either with controllers or with hands-free methods. An early example by Osawa et al. [37] for CAVE-like systems allowed users to touch and select nodes of a graph, pinch and move nodes, and grab the complete graph. In this context, some work has investigated efficient gesture interfaces for IA. Huang et al. [22] used actions such as grabbing and pinching to move, rotate, highlight, and group elements in immersive graphs. Some of their actions were specifically designed to be easier to recognize by the hands-free tracking system (Leap Motion) but were less intuitive to users. Still, gestures were overall more efficient than mouse input for the manipulation of complex graphs. Austin et al. [2] recently conducted an elicitation study to identify intuitive gestures for interaction with maps in Augmented Reality, interviewing both experts and regular users. Their participants created mostly single-handed gestures for panning and selections and two-handed ones for zooming and height change. We used some of their results in our virtual hands mode.

In the *VirtualDesk* metaphor [49], virtual hands were used to intuitively manipulate 3D data representations displayed above a tangible work desk, employing gestures such as grabbing and finger tapping. Even though hand controllers were used to improve tracking accuracy, system features were not assigned to specific buttons so that users did not have to remember button mappings, a decision we replicate in our virtual hands mode. Compared to a desktop system, this approach resulted in higher usability and lower mental workload scores. Yet, the selection of points was slower and less accurate, particularly in cluttered areas. This suggests a need for integrating more than one interaction metaphor in the same application.

Examples of local interaction without virtual hands include *ImAxes* [13], which explored embodied bi-manual interactions as a way of manipulating data dimensions (represented by 3D axis elements) and composing multiple visualizations by bringing these axes together in different ways. Virtual controller models were used to facilitate the learning of button and trigger actions, and sphere cursors attached to the controllers for selections. Cordeil et al. [14] used small wands to highlight nodes in graphs, both physically in a CAVE environment and virtually in a head-mounted display (HMD)-based one.

Finally, there are also examples of IA work that use other, less common interaction approaches, which are outside the scope of our investigation. For instance, Drouhard et al. [15] used a gamepad and gaze interaction to point at objects in a system to visualize 3D structures in materials science. Kwon et al. [30] proposed a system for visualization of spherically laid-out graphs with both gaze and mouse input.

2.3 Mixed Interaction Modes

Previous work has established that different input modalities can be more appropriate for different low-level tasks in the analysis workflow, but also that mixing modalities in a way that combines their strengths while avoiding their weaknesses is an open challenge [3,9]. Building on virtual hand and virtual ray pointer interactions, we decided to also evaluate their combination, enabling both local and remote interactions. A similar intent led to the design of classical techniques such as *Go-Go* [41] and *HOMER* [7] in the past. However, as recognized by Bowman and Hodges [7], such arm-extension techniques improve remote object manipulation but are more demanding for remote object selection. In our IA use case, the user manipulates only the data representation as a whole, and thus hand-centered manipulations typically do not require non-linear mappings. Therefore, we opted to combine standard hand and pointer interactions, integrating support for *raycast-with-reeling* [7] to increase the comparability between the different modes.

A core component of our implementation is that we enable a seamless transition between both modes, without an explicit mode switch. Some recent commercial VR applications already use similar approaches, and some of their design choices informed the design of our implementation. Specifically, we build on the mappings used by *Oculus Home* and *SteamVR Home*. We note, however, that the interaction requirements for IA are typically more complex than many other VR applications. IA applications often use many views/graphs and/or show many data points, resulting in cluttered environments where one needs the ability to accurately select and inspect individual data points, while still being able to manipulate the representation as a whole.

Additionally, we applied Mine et al.'s [34] physical mnemonics concept of storing system controls relative to the user's body in our virtual hand interactions, through the metaphor of a *Virtual Smartwatch*. Mine et al. [34] reported that handheld widgets, a kind of physical mnemonic, are easier to interact with than object-bound widgets.

3 Interaction Modes

In this section, we present the prototype system we implemented as apparatus for our investigation, as well as its three interaction modes: ***Virtual Hands*** (*Hand*), ***Virtual Ray Pointers*** (*Pointer*), and ***Mixed***.

Before implementing this system, we first selected a specific visualization use case that represents IA applications well. We decided to adopt the 3D Space-Time Cube (STC) for movement trajectories. In the STC, the horizontal plane depicts movement across a map, while the vertical axis depicts the same movement over time. This approach, used in recent IA work [21,46,51], requires a good integration of all three spatial dimensions for every task, which is well-supported by immersive systems. Our visualization design choices followed recommendations from relevant work [1,2,24,51]. For example, we kept the map plane fixed at the bottom of the STC at a constant viewing angle, with trajectories being movable. We rendered trajectories as 3D tube meshes and colored them by individual ID. They started at the top of the representation and progressed towards the bottom. By default, the trajectory diameter was 1cm but dynamically adjusted according to the map scale. We obtained maps from the Mapbox Unity SDK.

We built our apparatus in the Unity3D engine and displayed the IVE in an Oculus Rift CV1 HMD. A room-scale setup using three Oculus sensors allowed us to accurately track movements in a $2.5\text{m} \times 2.5\text{m} \times 2\text{m}$ area in which participants stood, enabling real-walking egocentric exploration of the STC, with the base map coupled to the ground. We deemed this setup an adequate compromise between large-scale and smaller, desk-scale environments since it provides opportunities for the use of both local and at-a-distance interactions. Moreover, real-walking exploration of room-scale environments is a common approach in recent IA applications [13,31,45]. We employed adequate computing hardware to ensure a frame rate above 80 FPS and dynamically reduced the field-of-view while the user moved the data [16,26], to minimize potential discomfort.

As controllers are tracked more precisely than current hands-free alternatives, and to avoid introducing a confounding variable, users interacted with all system modes through two *Oculus Touch* hand controllers. They were held as seen in Fig. 1, and each used the following mapping: a frontal trigger, to be pressed with the user's index finger, an inner trigger, to be pressed with the middle finger, and, a thumbstick and three physical buttons on the top, to be operated with the thumb. The thumbstick and buttons are equipped with sensors that inform the system when they are pressed or just touched.

3.1 Virtual Hand Interaction (*Hand*)

Due to its widespread use in interaction studies and recent IA work, we chose to use virtual hand actions as one of our modes. Despite a recent trend of using only virtual controller models with 3D cursors [12,13], we believe that using virtual hands is a more promising direction. Hand metaphors are prevalent in Mixed and Augmented Reality systems, which are becoming more common, and we believe that IA should share interaction idioms with this trend. At the same time, hands-free tracking technologies are quickly becoming more accurate [19], which will eventually enable unconstrained intuitive gestural interaction.

For efficient virtual-hand-style interaction, we follow the design choices employed by *VirtualDesk* [48], as this work demonstrated high usability and low workload in two controlled studies. All actions employed to move, rotate, and scale the data also figured among the most common gestures spontaneously reported by users in a recent elicitation study [2]. We display only the virtual hands in the IVE and map no actions to the controller and its buttons except for performing the hand actions (see Fig. 1). This way, the controllers can be (mostly) ignored by the users, resulting in a device-agnostic approach.

The main afforded actions are: grabbing the dataset with one hand and moving it across time or space, grabbing and scaling it with two hands by changing the distance between them in the horizontal or vertical direction, and grabbing and rotating it around the time (z) axis with two hands by rotating them in the horizontal plane. The grabbing action is intuitively performed by pressing both frontal and inner triggers simultaneously, forming a fist. While possible, we disabled rotations around the other axes since we preferred to keep the base map attached to the floor. We introduced additional constraints to optimize the recognition of actions and avoid unintended results that might either slow down the system or disturb the user's workflow. In particular, the system requires a horizontal distance of at least 20cm between hands to perform rotations, a vertical distance less than 5cm to perform spatial scaling, and a vertical distance larger than 5cm to perform temporal scaling. We defined these parameters empirically based on observations of typical unintended actions, e.g., rotating the data when planning to scale or scaling along the wrong axis.

Two additional actions are performed with the user's extended index finger, i.e., when the frontal trigger is not being pressed. Tapping any point along a trajectory inspects it, displaying a timestamp label. A quick double-tap with the virtual finger, emulating a double-click, selects that trajectory, filtering out all others. Although the double-tap action is not intrinsically direct, prior work on gestures has reported that it became intuitive to users due to the influence of existing UIs [40]. The two taps need to happen within 100 to 400 ms to be interpreted as a double-tap. For robustness, we disabled this action when the same hand was being used to move the data and when the pointing finger was outside of the view frustum. Red markers on the map and red lines on the wall grid continuously show the projections of the finger positions.

To support easily-accessible additional controls without resorting to using controller buttons or obstructing the view of the data, we adopted the concept of ***Virtual Smartwatches***. This is a physical mnemonic variant [34] that should be familiar to many users due to the popularity of such devices in the real world. Our design displays semi-transparent cylinders at the user's wrists (based on the hands' positions), coupled with a square blue panel on the upper side (see Fig. 3 – left). We opted

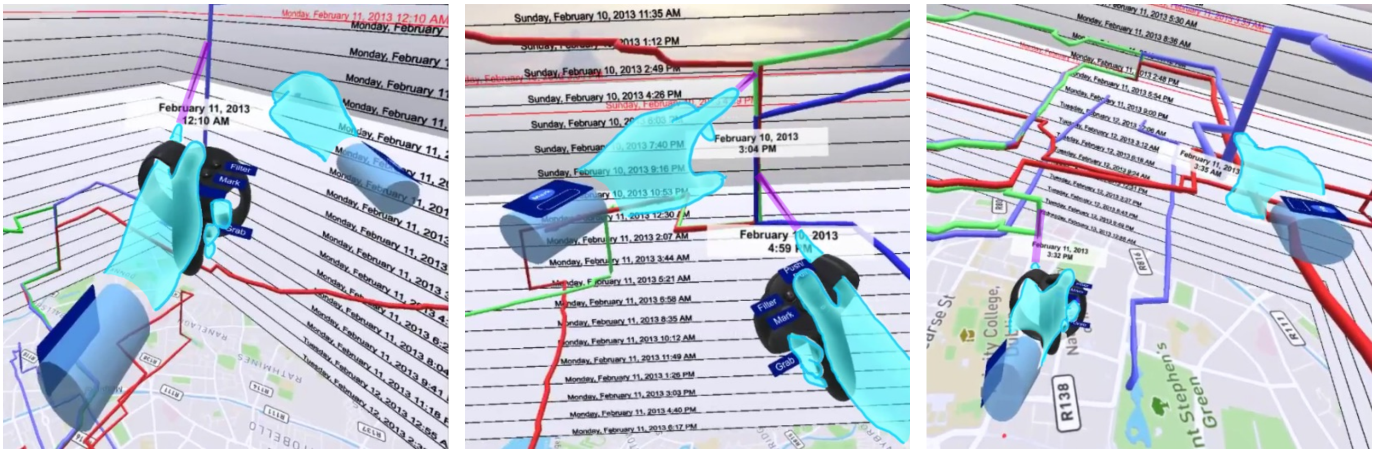

Fig. 2. Mixed interaction allows the use of different metaphors for different low-level tasks. In this mode, a permanent ray pointer originates from the index finger to allow easy transition between modes. All actions from Fig. 1 can be executed at any time. Controller models only become visible upon touching their buttons or performing the hold action. In the left image, the user is grabbing the data with the right hand and inspecting it with the left pointer. Meanwhile, in the center, they hold the data with the right pointer and inspect it with the left finger. On the right, the user is inspecting a closer data point with the finger while comparing it to a more distant one with the pointer. Hands and pointers highlighted for clarity.

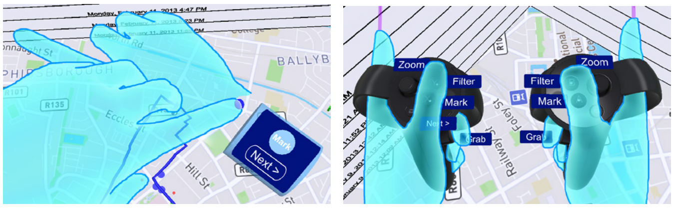

Fig. 3. In *Hand* (left), we designed *Virtual Smartwatches* to support system controls such as activating the annotation feature. We attach a 3D cursor to the fingertip to visually indicate when this feature is enabled. In *Pointer* (right), we mapped controls to labeled controller buttons.

to provide identical watches for both arms, so that users could use the one they preferred. Through these watches, the user can easily access controls that would otherwise need to be assigned to controller buttons. In our case, this included starting and finishing a task (*Start*/*Submit*) and activating the annotation feature (*Mark*)—which attaches a 3D "ink" cursor to the index finger. We believe the smartwatches are a more efficient solution than controller buttons. Besides being controller-agnostic, they do not require users to memorize button assignments, allow the display of free virtual hands, motivate gesture interaction, and are easily scalable to a larger number of features.

3.2 Virtual Ray Pointer Interaction (*Pointer*)

For the virtual (ray) pointer interactions, we show virtual models of the hand controllers and virtual rays start at the tip of each index finger. The interaction design and the mapping of system actions were influenced by the popular *Oculus Home* and *Google Earth VR* [25, 26] applications and adapted to the specific requirements of our use case. All the same actions as in the virtual hand interaction scheme (Sect. 3.1) are supported but were modified to always work with reference to a remotely indicated point (see Fig. 1).

In *Pointer*, the index finger typically stays pointed. By pressing the inner trigger while pointing at the map or walls, the user can hold and drag the spatial or temporal dimensions. To ensure maximum comparability with *Hand*, users can also hold any given point in a trajectory and drag it in all three dimensions simultaneously, as well as bring it closer or further away with a controller thumbstick, through *raycast-with-reeling* [7]. While pointing at the map or the walls, the thumbstick's vertical axis can be used for spatial and temporal scaling around the indicated point, while the horizontal axis can be used to rotate the map around it. We again disabled rotations around arbitrary axes, but this could be easily implemented by using the plane perpendicular to the ray orientation as a reference.

Users can inspect the trajectories by pointing the ray at arbitrary points, emulating a mouse-over. Red marks on the map and red lines on the wall grids follow the tip of the rays to help with remote inspections. We assigned all remaining actions to controller buttons, labeling them to indicate their function (see Fig. 3 – right): *Mark*, *Filter*, and *Start*/*Submit*. The visible controllers do not occlude the data as users can change their hand positions without changing the focus of selection. Even though the pointing rays effectively extend the user's reach, selecting small targets at large distances can be difficult due to the needed degree of hand stability. Thus, navigation and/or dataset manipulation are still needed to approach the data. However, the frequency of such actions is smaller compared to *Hand*.

3.3 Mixed Interaction (*Mixed*)

As previously stated, we believe that each of the two just introduced approaches has its inherent advantages depending on the low-level tasks being performed or on the user's immediate needs. Thus, we wanted to evaluate both conditions individually but also their combination. To support a seamless and reasonably intuitive transition between the two modes, without explicit mode switches, we took inspiration from the existing *Oculus Home* and *SteamVR Home* applications, adapting and extending them as necessary for our IA environment.

In the *Mixed* mode, permanently visible virtual ray pointers originate at the tip of the user's index fingers for remote inspection, as in *Pointer*. The virtual controller models are only shown in certain cases: when the user touches one of the buttons, indicating the intent of using them, or when the user presses (only) the inner trigger (the action typically used to hold data or surfaces with the pointer). In all other situations, *Hand* operations are used: the virtual finger can be used for tapping, double-tapping, and interacting with watches, and simultaneously pressing all triggers grabs, stretches, or rotates the data.

Thus, *Mixed* enables alternate workflows (see Fig. 2). For example, one can grab and move the dataset up and down with one hand while using the ray in the other to inspect the time of trajectory points at-a-distance. Or one can rotate or scale the data around a distant or hard-to-access point but still use finger gestures to inspect and filter. Similarly, one can use hand gestures for manipulation, but the pointers and controller buttons for quicker marking and filtering as desired.

Compared to the systems mentioned above, a challenge here is that buttons need to be available even when an object is not being grabbed with the ray (as they are used to filter or mark static trajectories). The hand grabbing action needs to be available even when the hand is not intersecting a specific object because it is used to translate, scale, and rotate the whole dataset. These differences make mixed interaction with IA data more complex and prone to erroneous activation.

4 User Study

Here we present the design decisions and rationale behind our comparative study to evaluate the interaction modes introduced above. Within a controlled experimental framework, we defined study hypotheses and user tasks to be performed in a spatio-temporal dataset.

4.1 Dataset and Tasks

For this study, we selected the 3-day simulated GPS-trajectory dataset created by Amini et al. [1] for STC visualization, which had already been used in immersive settings [51]. To balance visual clutter with information density, we selected a subset with 12 trajectories, which is complex enough that different forms of interaction are both needed and usable to identify desired information in the data (see Fig. 4). We picked another subset with only 3 trajectories (disjoint from the above subset) for the tutorial phases.

To maximize system usage and interaction and minimize the need for domain knowledge, we opted to employ multiple subsequent trials involving simple information-seeking tasks. With this, we covered the three basic kinds of questions in Peuquet's [38] *Triad* framework, as shown in Table 1. We selected three different task stimuli for each task trial, one in each day in the dataset, i.e., at different heights in the STC. We rotated these different stimuli between different modes for each participant to counterbalance task difficulties. Considering that a participant never received the same task stimulus more than once and the large information density of the evaluation environment (see Fig. 4), we largely minimized potential data learning effects.

Participants addressed tasks by either adding marks (T1 and T3) or filters (T2) to the data. The tasks were doable with reasonable effort, and their purpose was to encourage the users to access different regions of the dataset, motivating them to perform various forms of manipulation, including translation, rotation, scaling, and filtering.

4.2 User Study Hypotheses

Despite the highly investigative nature of our user study, we defined three expected outcomes for the *Mixed* mode.

- H1 Mixed interaction will be more efficient in time, as it combines the strengths of both other modes.
- H2 The seamless transition between modes will not increase the mental workload nor decrease system usability, since users will be free to perform the actions the way they prefer.
- H3 Users will combine virtual hand and virtual pointer interactions, choosing the metaphor according to the nature of the action.

4.3 Participants

We recruited 15 undergraduate and graduate students from our university campus. Their mean age was 26.6 years (SD 5), with 13 males and 2 females. Our population was rather diverse in terms of prior familiarity with VR (6 none, 1 low, 5 average, 3 high) and motion controllers (4 none, 2 low, 7 average, 1 high, 1 very high), but familiar with 3D computer games (1 none, 4 average, 4 high, 6 very high) and gamepads (5 average, 6 high, 4 very high). When asked to rate their gaming frequencies from 1 (never) to 7 (everyday) [31], the average rating was 4.2 (median = 5, sd = 1.6). Most participants reported low familiarity with the specific controllers used in the experiment (7 none, 3 low, 2 average, 3 high). All participants in the study were right-handed.

4.4 Experiment Design

We used a within-subjects design to minimize the effects of personal differences. The three modes (*Hand*, *Pointer*, and *Mixed*) were our within-subjects factor. Each participant performed 3 trials/repetitions of each of the 3 tasks in each of the 3 modes, which adds up to 27 evaluated tasks per participant. We asked participants to balance accuracy and speed. We always counterbalanced the order of the three within-subjects conditions but always preserved the task order to avoid confusion. Even though the mixed mode should ideally be the last, i.e., used only after participants had a better understanding of their own preferences, we still counterbalanced its order to minimize learning effects.

Table 1. We used multiple trials based on the three basic spatio-temporal information-seeking tasks [38] to encourage interaction with all areas of the dataset and minimize learning effects.

	Category [38]	Example
T1	*when + what → **where***	*Where was RED at 8pm on the 6th?*
T2	*when + where → **what***	*Who was at the place marked in red at 7am on the 6th?*
T3	*where + what → **when***	*When did GREEN arrive at the place marked in red?*

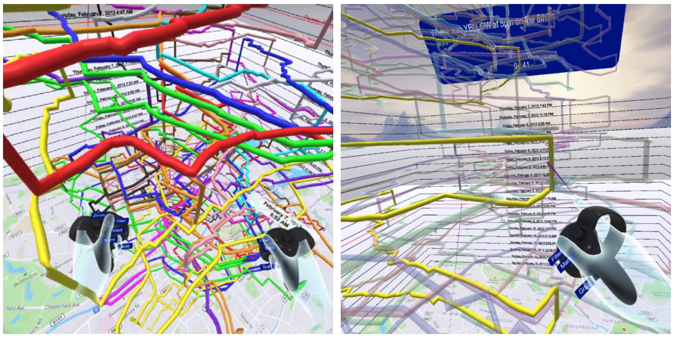

Fig. 4. We used a dense dataset with 12 trajectories in the evaluated tasks, demanding various forms of interaction to seek information. *Selection* commands make irrelevant trajectories semitransparent (right).

Before starting the tasks, participants filled consent and demographic forms, as well as a pre-exposure Simulator Sickness Questionnaire (SSQ) [28]. Then, they experienced a detailed system tutorial phase, which briefly introduced the STC visualization and guided them through all system features and interaction modes. The experimenter also assisted the participants in this tutorial, whenever appropriate. During the tutorial phase, we counterbalanced the introduction of *Hand* and *Pointer* interaction conditions to avoid influencing user preferences, but always introduced the possibility of combining both last. The tutorial also included three example tasks with the reduced dataset, requiring participants to find the correct answer before moving on to ensure that they understood how to seek the different kinds of information.

During the experiment, we showed task instructions on a large blue panel. Users could read them and ask questions before starting (by pressing the *Start* button on the controller or smartwatch). After 4 minutes and 30 seconds had passed on a task, a 30-second countdown started to signal that the time limit was approaching. As the tasks were not that complex, this happened only in less than 1% of all trials. After marking a selected answer, users pressed the same button on the controller or watch, now labeled *Submit*, to finish, and then answered a Single Ease Question (SEQ) [43] inside the virtual environment.

After completing the 9 trials in each mode, we asked participants to take off the HMD and fill questionnaires on a computer: a post-exposure SSQ, the Raw NASA TLX [20], the System Usability Questionnaire (SUS) [8], and the single general item from the Igroup Presence Questionnaire (IPQ) [44]. This also served as VR exposure breaks. After the last round of tasks, we asked participants questions regarding their preferred mode of interaction for a series of actions.

5 Results

In this section, we report the results obtained by our user study for task performance, workload, usability, and comfort, as well as user preferences and interactivity patterns. Since a large majority of variables did not present normal distributions, we assessed statistical significance ($\alpha = .05$) with the non-parametric Friedman test followed by paired Wilcoxon signed-rank post-hoc tests with Bonferroni-Holm correction.

We indicate significant effects on figures using asterisks ($*$ for $p < .05$, $**$ for .01, and $***$ for .001). We also report effect sizes for the Wilcoxon and Friedman tests. Following Tomczak and Tomczak [47], we estimate effect sizes using the correlation coefficient r, for the former, and Kendall's W, for the latter test. We classify effect sizes into small ($> .1$), moderate ($> .3$), and large ($> .5$) [10].

5.1 Task Performance

As expected based on our pilot studies, all tasks were easily answerable. Participants made only 18 errors in all 405 trials (95.6% success rate). We accepted answers as correct within a threshold of 2cm for *Where* and 45 minutes for *When* tasks (approximately 2cm at the default time resolution). Detailed rates for each mode and task are shown in Fig. 5 – bottom. We were not able to find any significant effects in *Where* ($\chi^2(2) = 4.8$, $p = .090$, $W = .16$), *What* ($\chi^2(2) = 2$, $p = .368$, $W = .06$), and *When* ($\chi^2(2) = 1.1$, $p = .565$, $W = .04$) tasks individually. However, when aggregating all tasks to investigate more general differences, a small effect of mode emerged ($\chi^2(2) = 6.7$, $p = .035$, $W = .22$). In this case, *Hand* performed significantly better ($p = .044$, $r = .68 = large$) than *Mixed*: 97.8% vs. 92.6%, a 5.2% absolute gain.

In terms of completion times (Fig. 5 – top), we found a small effect of interaction mode for the *Where* task only ($\chi^2(2) = 6.5$, $p = .038$, $W = .21$). *Mixed* interaction was significantly quicker than *Hand* ($p = .020$, $r = .67 = large$): 53.8s vs. 70.3s average time, a 23.5% relative gain. Tests for the *What* ($\chi^2(2) = 3.7$, $p = .155$, $W = .12$) and *When* ($\chi^2(2) = 0.1$, $p = .936$, $W = .004$) tasks yielded non-significant results.

5.2 Workload

Besides task performance itself, we were interested in investigating possible differences in the perception of workload across interaction modes. Fig. 6 presents scores obtained from the Raw NASA Task Load Index (TLX) questionnaires filled after each condition. We did not find statistically significant differences in terms of the TLX score ($\chi^2(2) = 1.8$, $p = .397$, $W = .06$) or of any of its subcomponents—Mental ($\chi^2(2) = 0.2$, $p = .903$, $W = .006$), Physical ($\chi^2(2) = 4.4$, $p = .112$, $W = .15$), Temporal ($\chi^2(2) = 0.1$, $p = .956$, $W = .003$), Performance ($\chi^2(2) = 0.9$, $p = .639$, $W = .03$), Effort ($\chi^2(2) = 3.8$, $p = .150$, $W = .13$), and Frustration ($\chi^2(2) = 2.5$, $p = .289$, $W = .08$).

In terms of ease-of-use, data from the Single Ease Questions (SEQ) answered after each trial indicate that all approaches enabled easy task execution. We did not find significant differences either per task or by aggregating all tasks per mode ($\chi^2(2) = 0.4$, $p = .819$, $W = .01$). Using a Likert-scale from 1 to 7 (7 being the easiest), *Hand* averaged 5.55 (sd 0.9), compared to 5.49 (sd 0.8) for *Pointer*, and 5.42 (sd 1) for *Mixed*.

5.3 Usability and Comfort

According to the System Usability Scale (SUS) questionnaires filled after each mode, participants generally liked all interaction modes, with no significant difference ($\chi^2(2) = 5.5$, $p = .064$, $W = .18$). The score given to *Hand* was slightly higher (81.8, sd 11.8), while *Pointer* received 77.6 (sd 11), and *Mixed* 77 (sd 12.5).

We applied the Simulator Sickness Questionnaire four times to each participant, at the beginning of the experiment and after concluding each mode. We did not find any difference between modes ($\chi^2(2) = 2.8$, $p = .241$, $W = .09$), and all stayed within the negligible (< 5) score range [27]. In fact, 7 out of 15 participants had a negative delta, possibly due to getting distracted with the experiment and forgetting about symptoms they had felt before. Average scores were -0.5 for *Hand* (sd 14.7), 2.7 for *Mixed* (sd 6.3), and -2.2 for *Pointer* (sd 11.1). Even if we constrain the deltas to a minimum of 0, scores remain in the same range, averaging 4.2 for *Hand* (sd 7.1), 3.7 for *Mixed* (sd 5.3) and 2.2 for *Pointer* (sd 5.8). This was particularly interesting considering that the VR exposure time in this experiment was relatively high. The average period spent inside the IVE was 46 minutes (sd 11), with a minimum of 29 and a maximum of 75 minutes. The relatively long tutorial needed to explain the 3 different modes notably contributed to this. The average duration of the tutorial phase was 19 minutes (sd 4), with a minimum of 13 and a maximum of 31 minutes. The dynamic reduction of the

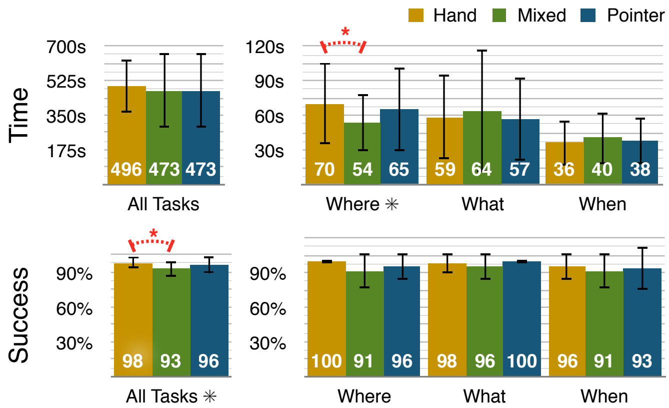

Fig. 5. Mixed interaction significantly reduced completion times by 23.5% in the most time-intensive task, *Where* (top right). Yet, it also significantly decreased the success rate by 5.2% across all tasks (bottom left). Error bars indicate standard deviations and value labels are rounded.

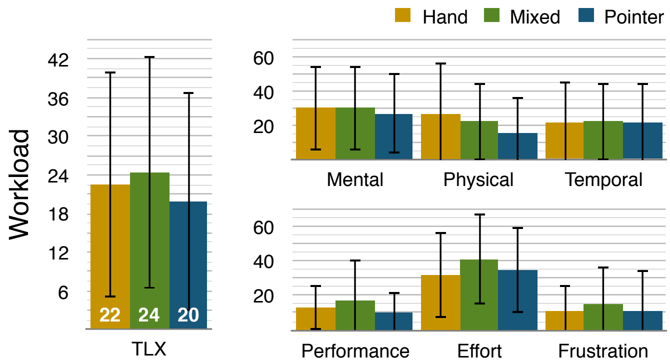

Fig. 6. We did not find significant differences between modes in terms of the NASA TLX workload factors (lower is better). Effort was the highest component across all modes, while perceived performance and frustration were the lowest. Error bars indicate standard deviations.

field-of-view [16], the adoption of real-walking navigation, and the high frame rate likely contributed to a comfortable experience.

Interestingly, the *Hand* technique scored higher in the General Item (*In the computer generated world I had a sense of "being there"*) extracted from the Igroup Presence Questionnaire (IPQ), obtaining 5.13 (sd 0.74) out of 6. The *Mixed* mode received 4.73 (sd 1.33) and *Pointer* received the lowest score, 4.66 (sd 1.23). However, this difference was not significant ($\chi^2(2) = 5.5$, $p = .064$, $W = .18$).

5.4 User Preferences and Comments

All participants quickly learned how to interact with the controller triggers to execute the grabbing actions in all conditions. Fig. 7 shows participants' responses for their preferred ways of performing each action, as well as their overall preferred modes. This feedback confirms our initial speculation that both virtual hand and virtual pointer would be preferred for different low-level tasks. Intuitive manipulations with the virtual hands were vastly preferred over their virtual controller counterparts for rotation, translation, and scaling. A close majority also preferred inspecting data points with the virtual fingers over inspecting them with the virtual rays, and the same applied for answering the SEQ questions mid-air. On the other hand, 13 out of 15 preferred filtering trajectories by pressing a controller button over performing the double tapping action—this is due in large part to the difficulty faced by multiple participants in mastering this action. The same applied to starting/ending tasks and adding marks, actions that in the *Hand* approach demanded interaction with the *Virtual Smartwatches*, but this was not a strong majority. We observed the same pattern when analyzing the responses from the three participants that reported high

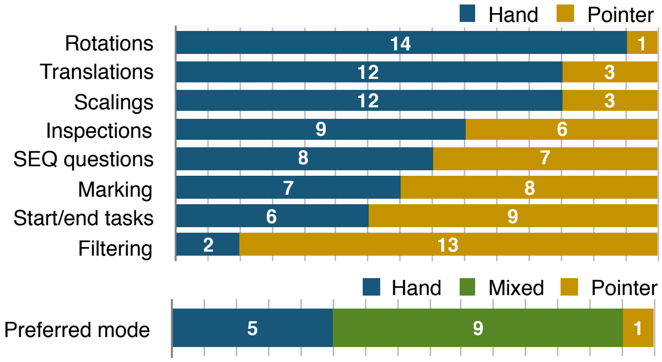

Fig. 7. Participants preferred intuitive *Virtual Hand* actions for rotating, moving, and scaling the dataset. Controller buttons in *Pointer* made it easier to filter the data and start tasks. 60% would prefer using the *Mixed* mode over any of the other two separately.

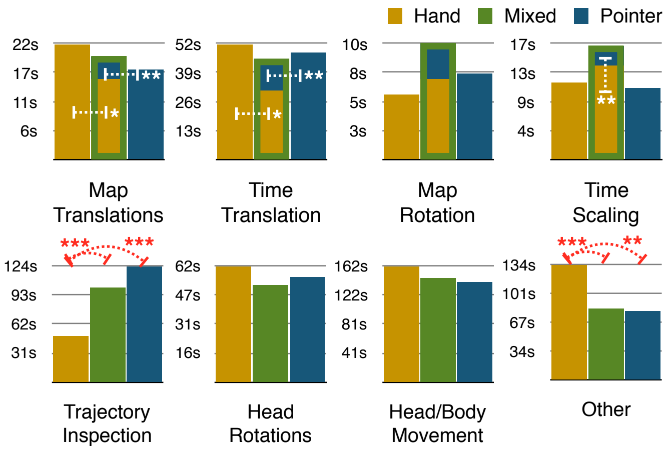

Fig. 8. The average distribution of the total task time across low-level activities shows that manipulation times were very similar with *Hand* and *Pointer*. The decomposition of *Mixed* manipulation times (upper row) show that, in this mode, participants indeed combined the two metaphors, but relied mostly on hand actions over controller actions.

prior VR experience. They unanimously preferred gestures for scaling and rotating and pointers for marking and filtering.

Asked about their overall preferred modes, 9 out of 15 participants chose being able to use both virtual hand gestures and the virtual pointers. Five said they would rather use only the hands, while a single user preferred the pointers. The 6 participants that did not prefer *Mixed* stated that it was confusing and ultimately did more harm than good.

Comments from the free-form section confirmed these observations. One participant mentioned that *"movements were more fluid with the virtual hands, but selections were much quicker and easier with the rays"*, while another stated that *"manipulating with gestures was more intuitive, natural, and easy to learn"* but pointed out that *"if the user is very familiar with the system, manipulating with the pointing rays could be quicker and demand less effort"*. One suggested that, instead of the automatic approach we adopted in *Mixed*, there could be an explicit option for alternating between modes.

Multiple users mentioned the double-tapping action as the most difficult. Even though we had tuned the recognition method, it was often necessary to attempt it multiple times before being able to apply a filter, while at other times, some unintended selections occurred. On the other hand, marking with a single tap (after selecting this option on the watch) required considerable attention to avoid unintended annotations. Finally, some also suggested new features, such as cutting planes for time comparisons and an option to reset the original data configuration.

5.5 Interactivity Patterns

Fig. 8 shows the average distribution of time in terms of different low-level tasks across all users and tasks for all modes. This analysis aims to objectively verify if the users' behaviors while interacting with the system match their stated preferences. Data manipulation tasks included translation (spatial or temporal), rotation (spatial), and scaling (spatial or temporal). Due to a logging issue, it was not possible to specify times for map scalings. Thus, we included this component in *Other*. Yet, we estimate this component to be similar in behavior to the time scaling one, since both were triggered in very similar ways.

Our analysis indicates that the different modes did not affect the time dedicated for each low-level manipulation. We did not find a significant effect on the total manipulation time ($\chi^2(2) = 1.7$, $p = .420$, $W = .05$), and pairwise comparisons between equivalent actions in *Hand* and *Pointer* were always non-significant ($p = .18$ for map translations, .93 for time translations, .83 for map rotations, and 1 for time scaling).

Considering *Mixed*, users consistently combined both forms of interaction. Pointer manipulations were significantly less used in this mode than in *Pointer* ($p < .001$, $r = .84 =$ *large*), while hand actions were non-significantly less used than in *Hand* ($p = .055$, $r = .5 =$ *large*). Data from Fig. 8 suggests that, when able to do either, participants opted to rely on virtual hand actions relative to the center of the hands more than on virtual pointers relative to a remote location. Pairwise comparisons show that the use of both in *Mixed* was indeed significantly different for scaling ($p = .002$). Differences in map translations ($p = .051$), rotations ($p = .119$), and time translations ($p = .167$) were not significant.

Another noticeable difference is that when virtual pointers are available (*Mixed* and *Pointer*), we see a large increase in the number of trajectory inspections and a decrease in *Other*. This is understandable considering that in *Hand* the user has to move their finger closer to the data to inspect (acting as a 3D cursor). With pointing rays, they are permanently inspecting intersected trajectories, often unintentionally. For this reason, we do not consider this difference particularly insightful.

Finally, we also measured the frequency of use of both modes for applying filters (double-tap vs. controller button) and adding marks (button on the *Virtual Smartwatch* followed by tapping vs. controller button) in the *Mixed* mode. Participants chose to use the virtual controller buttons 72.4% of the time for filtering and 73.4% of the time for marking, confirming the preferences stated in the previous section.

5.6 User, Hand, and Data Movements

Fig. 9 shows how the different modes affected user, data, and hand movement. Here, we considered the avatar *base* object in Unity for the user body movement metrics, to only account for larger displacements such as walking. Despite the fact that the virtual ray pointers could potentially afford at-a-distance interactions, we observed a very similar amount of user translation ($\chi^2(2) = 3.7$, $p = .155$, $W = .12$) and rotation ($\chi^2(2) = 2.8$, $p = .247$, $W = .09$) across modes. This is likely a result of IA tasks requiring local inspections regardless of the interaction mode.

Nonetheless, we found significant differences in terms of data translations ($\chi^2(2) = 6.5$, $p = .038$, $W = .22 =$ *small*) and rotations ($\chi^2(2) = 10.5$, $p = .005$, $W = .35 =$ *moderate*). Users moved the dataset (in all directions) significantly more with *Pointer* (avg 40.7 m during all tasks) than with *Hand* (26.8 m) ($p = .02$, $r = .67 =$ *large*), possibly due to larger translations requiring less effort. On the other hand, they rotated it (around the z axis) significantly less with *Pointer* (avg 33.2°) than with both *Hand* ($p = .01$, $r = .78 =$ *large*, avg 214°) and *Mixed* ($p = .042$, $r = .59 =$ *large*, avg 255.6°). This was likely due to the "steering wheel" hand metaphor being more intuitive than using the controller joystick. Note, however, that data rotations (always lower than 200°) are negligible compared to user body rotations (over 5000°).

Finally, we also looked into how the different modes affect the amount of hand movements. By avoiding gestures, *Pointer* significantly reduced the demand for translating and rotating both hands, while measures for *Mixed* were close to those for *Hand* (see Fig. 9 – bottom).

5.7 Correlations with User Characteristics

We computed Spearman rank correlations between 32 logged and questionnaire-based metrics and two user characteristics: prior VR

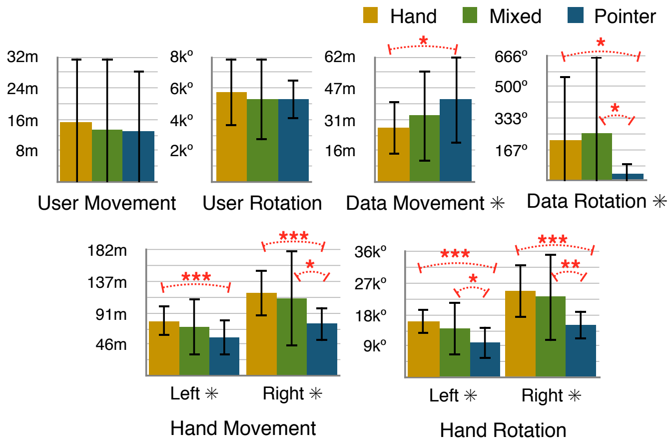

Fig. 9. The different modes did not affect how much the users moved around the virtual environment to accomplish the visualization tasks. However, they affected data movement and hand movement demand. *Pointer* made data translations easier but rotations less intuitive, and required less hand translations and rotations by avoiding gestures. All participants were right-handed. Error bars indicate standard deviations.

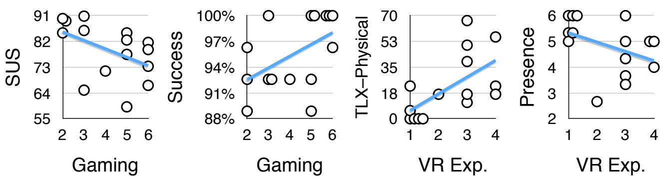

Fig. 10. Correlations between variables and user characteristics might indicate directions for further study. Each point represents a participant (average of all modes). Gamers gave the system lower usability ratings but were more successful in the tasks. Users without prior VR experience rated physical demand very low and presence very high.

experience (from 1, *none*, to 5, *very high*) and gaming frequency (from 1, *never*, to 7, *everyday*). We considered both global averages per participant and their results in each mode. Here, we highlight cases with statistical significance [5]. Even though we cannot establish causality, we believe this data can suggest directions for further study.

Gaming significantly correlated with four variables. We observed a negative correlation with the average SUS usability scores (ρ = -.55, p = .03), also seen without significance in both *Hand* (ρ = -.48, p = .07) and *Pointer* (ρ = -.50, p = .056). Most non-gamers rated the system high on usability, while gamers provided more distributed ratings (see Fig. 10 – left). Thus, one hypothesis is that gamers are already used to alternative forms of 3D interaction. On the other hand, gaming positively correlated with success rate overall (ρ = .53, p = .04) and in the *Mixed* mode (ρ = .58, p = .02). Unlike other participants, most gamers answered all trials correctly (see Fig. 10 – center left), possibly because those participants saw the tasks as *missions* and dedicated more effort. Gaming also positively correlated with time spent using hands for both time (overall: ρ = .65, p = .007 / *Hand*: ρ = .53, p = .04) and map translations (overall: ρ = .5, p = .056 / *Hand*: ρ = .65, p = .007).

VR experience correlated mostly with the workload scores. In particular, it significantly correlated with reported physical workload (see Fig. 10 – center right) in all modes: *Hand* (ρ = .68, p = .005), *Pointer* (ρ = .60, p = .018), and *Mixed* (ρ = .68, p = .005). In the case of *Hand*, it also correlated with mental workload (ρ = .57, p = .026 / all modes: ρ = .46, p = .083) and the overall TLX score (ρ = .55, p = .031 / all modes: ρ = .47, p = .073). At the same time, it correlated negatively with the presence score (all modes: ρ = -.52, p = .046 / *Hand*: ρ = -.53, p = .04 / *Mixed*: ρ = -.50, p = .058) (see Fig. 10 – right). One hypothesis is that a "novelty effect" led less experienced participants to underestimate the workload and overestimate presence.

6 Discussion

In this section, we discuss the takeaways from our study and potential questions regarding the generalizability of the findings and our expectations for other approaches.

6.1 Findings and Takeaways

We did not identify significant differences between *Hand* and *Pointer* in terms of task performance, workload, interactivity patterns, or user movement. Results indicate that both designs, despite their considerable differences, do not affect results of IA studies. In fact, even the time spent with each low-level manipulation was very similar in both. Nonetheless, our results do show differences in terms of user preferences, the use of data translations and rotations, and hand movement demand. Depending on which tasks will be most frequently executed in an IA system, designers may prefer to favor one mode or the other. Standard deviations were high for most metrics due to large interpersonal variability in terms of abilities, preferences, and strategies and also due to varying levels of difficulty among the trials.

Mixed interaction significantly reduced completion times over *Hand* by 23% for one investigated task, but also slightly reduced overall success rates by 5%. This time-efficiency trade-off partially confirmed our hypothesis H1. The *Where* task was the most demanding since it required searching the whole STC, and identifying the time period of interest in the vertical grid was more challenging than identifying a specific point on the map. The drop in success rates was most likely due to unintended selections or marking done by participants unaccustomed to the combination of techniques. Although designers should consider this issue when implementing such solutions, we believe that future refinements of the *Mixed* technique might also overcome this disadvantage. Note that neither time nor success rate differed significantly relative to the commonly-used *Pointer* mode.

Participants preferred to perform different low-level tasks in different modes. This confirmed hypothesis H3 and, interestingly, contradicted the notion that *Hand* would be more suited for local interactions and *Pointer* for remote ones. According to their stated preferences and recorded interactivity behaviors, users relied on intuitive *Hand* interactions for manipulations such as rotation, translation, and scaling, which could easily be performed remotely by pointing at the walls or floor. Meanwhile, they preferred *Pointer* for filters, marking, and system control due to its larger precision and easier access through buttons, despite the first two actions often demanding proximity for easier interaction and visual inspection. Participants were divided for inspecting trajectories and answering mid-air questions.

Most users enjoyed the ability to change between modes depending on their needs. 9 out of 15 participants stated that they would rather use the mixed mode than any of the other two individually. This feedback is likely related to the different preferences discussed in the previous topic and the trade-off between somewhat higher usability but significantly larger hand movements in *Hand*. Interestingly, only one participant preferred *Pointer* alone, even though this very common approach afforded similar performance, workload, and usability.

Mixed interaction did not significantly increase workload or decrease system usability and task ease. Despite potentially increasing system complexity by integrating two redundant forms of interaction, we did not find significant differences for these metrics, confirming H2. However, usability ratings were marginally-significantly higher for *Hand*, which should be further investigated in future studies.

Participants reported only negligible sickness symptoms. Despite an average of 46 minutes of VR exposure, all interaction modes resulted in very low SSQ scores. The use of dynamic FOV reduction during any data movements and the high frame rate likely contributed to this result, which is very promising for IA applications.

Participants preferred activating features on the *Virtual Smartwatches* over learning more complex gestures. Only 2 participants preferred filtering with double-taps (considered hard to execute by most) over the controller buttons. Meanwhile, 7 out of 15 preferred selecting an "ink" button on their smartwatches and then tapping a data point over using a controller button—these were participants with no to medium prior VR experience. This suggests that adding a similar watch button for filtering could be a promising direction (possibly as an alternate for the double-tapping action, which some enjoyed). Moreover, the watches could also be used to activate additional functionality, such as categorical, brush, or volume filters. However, it is important to avoid overloading the watches, as deep menus and many alternatives would inevitably slow down interaction. Even though we positioned the smartwatches relative to the hands' positions, without tracking wrists, participants quickly learned how to operate them without hitting their real arms due to misalignment, likely assisted by proprioception.

User performance and feedback may correlate with specific individual characteristics. An interesting side result of our study was the observation of moderate but significant correlations between characteristics, such as gaming habits and prior VR experience, with logged and questionnaire-based metrics such as success rates, usability, workload, and presence. We believe further study is warranted to clarify this relationship since, if confirmed, participant backgrounds could significantly affect future study results. One limitation to be considered is that our user group heavily skewed male, with only two female participants.

6.2 Generalizability and Perspectives

Are our findings generalizable to other data domains? We believe that our STC environment is representative of many IA applications, as it requires perceptual integration of the three data dimensions to extract information and allows both ego- and exocentric perspectives. The evaluated tasks, while specific to the spatio-temporal domain, correspond to simple information-seeking activities along all data dimensions. These are necessary for any 3D visualization that uses spatial dimensions to encode information. Moreover, for any 3D visualization, the required manipulations to accomplish information-seeking tasks are similar to those needed in the STC. Even though we restricted rotations around arbitrary axes, these can be easily added to both *Hand* (using the vector between both hands as reference) and *Pointer* (using the plane perpendicular to the ray as reference). Since 14 out of 15 participants already preferred virtual hand rotations in our study, there is no reason to expect that this would be different in this more complex scenario, where the use of the virtual hands would likely be even more intuitive.

It could also be argued that movement trajectories are difficult to select with a virtual finger or ray pointer, due to their small diameters or difficulties in depth perception. Nonetheless, we believe this is a common characteristic of most IA applications, including graphs, scatterplots, and parallel coordinates, either due to the scale of visualized data or the characteristics of the visual representation, especially as the data density increases. Yet, we acknowledge that each data domain has particular analysis requirements. We believe that representations without reference axes or planes, e.g., 3D scatterplots and graphs, would need even more local manipulations. However, even in that case, pointer interaction would still assist quick and accurate selection and to manage multiple representations in the same IVE.

Are our findings generalizable to higher volumes of data? For higher data volumes, navigation through translating and rotating to obtain different points of view or scaling around regions of interest continues to be necessary. Though individual filters and marks would be less likely in such a scenario, all approaches could be extended with local and at-a-distance brushing mechanisms.

Could other variables not included in our experiment design lead to differences between interaction modes? The main conceptual difference between virtual hand and pointer is that, in the former, interactions are performed locally with respect to the hands' positions, while in the latter, they are performed with respect to a remotely indicated point. For this reason, interaction distance is a variable that might influence the choice of mode. We decided not to investigate it in our study, e.g., by forcing participants to stay at a given distance, because we believe that looking at data details in IA inherently demands local inspection for most tasks. Moreover, fixing distances would result in an unrealistic scenario, while we intended to observe user preferences and performance during realistic information-seeking tasks.

Other potential variables include the environment size and the user posture (e.g., seated instead of walking), which can make physically moving around uncomfortable or impossible. Even though manipulation could be used to bring the data closer to the user, these differences could potentially favor *Hand* or *Pointer*, depending on the circumstances, which should be verified in a follow-up.

Could observed differences be due to implementation choices? Our interaction design was based on common VR systems, such as *Oculus Home*, and prior IA research [2, 49, 52]. We believe it was successful in reproducing typical virtual hand and pointer approaches and in combining them. The high usability scores for the three environments support this argument. Also, we spent significant effort to ensure that they were comparable by always allowing equivalent manipulations. The only interaction which might have skewed user preferences was the double-tap, which required precision and was challenging for many.

How would a controller visible for local interaction compare? Some of the user-preferred actions in *Pointer* were based on the use of controller buttons, e.g., to filter or mark data points, which was not possible in *Hand*. The approach adopted by *ImAxes* [13] benefits from having all controller features permanently visible for local bimanual interaction. We could enable rotation and scaling actions by simultaneously pressing triggers on both controllers. Whether the intuitiveness and usability of such actions would be similar to virtual hands is a topic for future work. Another alternative is a seamless approach to hide and show the controllers without virtual pointers, but this would not afford the combined workflows mentioned above.

How can future technological advancements change our results? More accurate tracking of user hands and arms could potentially help the *Hand* mode, e.g., for more precise positioning of the *Virtual Smartwatches*. A hands-free mixed interaction framework would require defining specific actions to grab remote points, likely through a closed fist with a pointed finger. Even though the watches were well-received and afford easy access to system controls, other approaches would be required for selecting targets, e.g., pointing and voice input.

Would preference trends change with more training or constant usage? We believe it is reasonable to assume that virtual pointer manipulation could become more frequently used, given its lower hand movement demand and the fact that the participants would become more familiar with its controls. Moreover, our data suggest that more experienced participants tend to notice the physical workload more. In fact, this is one of the strengths of the mixed approach since it enables both forms of interaction, depending on convenience.

7 Conclusion

In this work, we demonstrated that both virtual hand and virtual pointer are efficient forms of interaction in IA without significant differences in performance, workload, or interactivity patterns. This suggests that current results in IA are not significantly affected by the specific metaphor used. Furthermore, as users preferred different modes for different low-level tasks, designers can choose interactions that favor tasks they prioritize. Our concept of *Virtual Smartwatches* can help users easily access relevant features without obstructing the view of the data and without memorizing controller-specific button assignments.

Users preferred a mixed mode over the other two individually, and results indicate it can help lower task times without increasing workload or decreasing usability. We believe this shows that integrating different forms of interaction is not only helpful but necessary for IA to overcome the limitations of specific input methods. Being able to seamlessly switch between local and remote manipulations in hands-free contexts will be a particular challenge as new tracking technologies emerge.

Future work includes evaluating other selection metaphors, e.g., virtual controllers for local interaction, or pen controllers [4, 39], and evaluating the same approach in different usage environments, e.g., seated instead of walking or in other data domains.

Acknowledgments

The authors thank the user study participants for their feedback and the reviewers for their insightful contributions. They acknowledge financial support from CNPq–Brazil, NSERC, and Global Affairs Canada. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

[1] F. Amini, S. Rufiange, Z. Hossain, Q. Ventura, P. Irani, and M. J. McGuffin. The impact of interactivity on comprehending 2D and 3D visualizations of movement data. *IEEE Transactions on Visualization and Computer Graphics*, 21(1):122–135, 2015. doi: 10.1109/TVCG.2014.2329308

[2] C. R. Austin, B. Ens, K. A. Satriadi, and B. Jenny. Elicitation study investigating hand and foot gesture interaction for immersive maps in augmented reality. *Cartography and Geographic Information Science*, 47(3):214–228, 2020. doi: 10.1080/15230406.2019.1696232

[3] S. K. Badam, A. Srinivasan, N. Elmqvist, and J. Stasko. Affordances of input modalities for visual data exploration in immersive environments. In *IEEE VIS Workshop on Immersive Analytics: Exploring Future Interaction and Visualization Technologies for Data Analytics*, 2017.

[4] A. U. Batmaz, A. K. Mutasim, and W. Stuerzlinger. Precision vs. power grip: A comparison of pen grip styles for selection in virtual reality. In *IEEE VR Workshop on Novel Input Devices and Interaction Techniques at VR*, NIDIT '20, pp. 23–28. IEEE, 2020. doi: 10.1109/VRW50115.2020.00012

[5] D. J. Best and D. E. Roberts. Algorithm AS 89: The upper tail probabilities of Spearman's rho. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(3):377–379, 1975. doi: 10.2307/2347111

[6] D. A. Bowman. *Interaction Techniques for Common Tasks in Immersive Virtual Environments: Design, Evaluation, and Application*. PhD thesis, Atlanta, GA, USA, 1999. AAI9953819.

[7] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, p. 35–ff. ACM, 1997. doi: 10.1145/253284.253301

[8] J. Brooke. SUS – A quick and dirty usability scale. In P. W. Jordan, B. Thomas, I. L. McClelland, and B. Weerdmeester, eds., *Usability Evaluation in Industry*. Taylor & Francis, 1996.

[9] W. Büschel, J. Chen, R. Dachselt, S. Drucker, T. Dwyer, C. Görg, T. Isenberg, A. Kerren, C. North, and W. Stuerzlinger. Interaction for immersive analytics. In K. Marriott, F. Schreiber, T. Dwyer, K. Klein, N. H. Riche, T. Itoh, W. Stuerzlinger, and B. H. Thomas, eds., *Immersive Analytics*, pp. 95–138. Springer, 2018. doi: 10.1007/978-3-030-01388-2_4

[10] J. Cohen. *Statistical power analysis for the behavioral sciences*. Routledge, 2013. doi: 10.4324/9780203771587

[11] M. Cordeil, B. Bach, Y. Li, E. Wilson, and T. Dwyer. Design space for spatio-data coordination: Tangible interaction devices for immersive information visualisation. In *Proceedings of IEEE Pacific Visualization Symposium (PacificVis)*, pp. 46–50. IEEE, 2017. doi: 10.1109/PACIFICVIS.2017.8031578

[12] M. Cordeil, A. Cunningham, B. Bach, C. Hurter, B. H. Thomas, K. Marriott, and T. Dwyer. IATK: An immersive analytics toolkit. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 200–209. IEEE, 2019. doi: 10.1109/VR.2019.8797978

[13] M. Cordeil, A. Cunningham, T. Dwyer, B. H. Thomas, and K. Marriott. ImAxes: Immersive axes as embodied affordances for interactive multivariate data visualisation. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pp. 71–83. ACM, 2017. doi: 10.1145/3126594.3126613

[14] M. Cordeil, T. Dwyer, K. Klein, B. Laha, K. Marriott, and B. H. Thomas. Immersive collaborative analysis of network connectivity: CAVE-style or head-mounted display? *IEEE Transactions on Visualization and Computer Graphics*, 23(1):441–450, 2017. doi: 10.1109/TVCG.2016.2599107

[15] M. Drouhard, C. A. Steed, S. Hahn, T. Proffen, J. Daniel, and M. Matheson. Immersive visualization for materials science data analysis using the Oculus Rift. In *IEEE International Conference on Big Data*, pp. 2453–2461. IEEE, 2015. doi: 10.1109/BigData.2015.7364040

[16] A. S. Fernandes and S. K. Feiner. Combating VR sickness through subtle dynamic field-of-view modification. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 201–210. IEEE, 2016. doi: 10.1109/3DUI.2016.7460053

[17] A. Fonnet, F. Melki, Y. Prié, F. Picarougne, and G. Cliquet. Immersive data exploration and analysis. In *Student Interaction Design Research Conference*, 2018.

[18] A. Fonnet and Y. Prié. Survey of immersive analytics. *IEEE Transactions on Visualization and Computer Graphics*, 2019. To appear. doi: 10.1109/TVCG.2019.2929033

[19] S. Han, B. Liu, R. Cabezas, C. D. Twigg, P. Zhang, J. Petkau, T.-H. Yu, C.-J. Tai, M. Akbay, Z. Wang, A. Nitzan, G. Dong, Y. Ye, L. Tao, C. Wan, and R. Wang. Megatrack: Monochrome egocentric articulated hand-tracking for virtual reality. *ACM Transactions on Graphics*, 39(4), July 2020. doi: 10.1145/3386569.3392452

[20] S. G. Hart. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, pp. 904–908. Sage Publications, 2006.

[21] F. Homps, Y. Beugin, and R. Vuillemot. ReViVD: Exploration and filtering of trajectories in an immersive environment using 3D shapes. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 729–737. IEEE, 2020. doi: 10.1109/VR46266.2020.00096

[22] Y.-J. Huang, T. Fujiwara, Y.-X. Lin, W.-C. Lin, and K.-L. Ma. A gesture system for graph visualization in virtual reality environments. In *IEEE Pacific Visualization Symposium (PacificVis)*, pp. 41–45. IEEE, 2017. doi: 10.1109/PACIFICVIS.2017.8031577

[23] C. Hurter, N. H. Riche, S. M. Drucker, M. Cordeil, R. Alligier, and R. Vuillemot. Fiberclay: Sculpting three dimensional trajectories to reveal structural insights. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):704–714, Jan 2019. doi: 10.1109/TVCG.2018.2865191

[24] T. Kapler and W. Wright. GeoTime information visualization. In *IEEE Symposium on Information Visualization*, pp. 25–32. IEEE, 2004. doi: 10.1109/INFVIS.2004.27

[25] D. Käser, E. Parker, and M. Bühlmann. Bringing Google Earth to virtual reality. In *ACM SIGGRAPH 2016 Talks*, pp. 78:1–78:1. ACM, 2016. doi: 10.1145/2897839.2927441

[26] D. P. Käser, E. Parker, A. Glazier, M. Podwal, M. Seegmiller, C.-P. Wang, P. Karlsson, N. Ashkenazi, J. Kim, A. Le, M. Bühlmann, and J. Moshier. The making of Google Earth VR. In *ACM SIGGRAPH 2017 Talks*, pp. 63:1–63:2. ACM, 2017. doi: 10.1145/3084363.3085094

[27] R. S. Kennedy, J. M. Drexler, D. E. Compton, K. M. Stanney, D. S. Lanham, and D. L. Harm. Configural scoring of simulator sickness, cybersickness and space adaptation syndrome: Similarities and differences. In L. J. Hettinger and M. W. Haas, eds., *Virtual and adaptive environments: Applications, implications, and human performance issues*, chap. 12, pp. 247–276. CRC Press, 2003.

[28] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The International Journal of Aviation Psychology*, 3(3):203–220, July 1993.

[29] M.-J. Kraak. The space-time cube revisited from a geovisualization perspective. In *Proceedings of the 21st International Cartographic Conference (ICC)*, pp. 1988–1996. International Cartographic Association, 2003.

[30] O.-H. Kwon, C. Muelder, K. Lee, and K.-L. Ma. A study of layout, rendering, and interaction methods for immersive graph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(7):1802–1815, July 2016. doi: 10.1109/TVCG.2016.2520921

[31] W. S. Lages and D. A. Bowman. Move the object or move myself? walking vs. manipulation for the examination of 3D scientific data. *Frontiers in ICT*, 5:15, 2018. doi: 10.3389/fict.2018.00015

[32] K. Marriott, F. Schreiber, T. Dwyer, K. Klein, N. H. Riche, T. Itoh, W. Stuerzlinger, and B. H. Thomas, eds. *Immersive Analytics*, vol. 11190 of *Lecture Notes in Computer Science*. Springer, 2018. doi: 10.1007/978-3-030-01388-2

[33] M. R. Mine. Virtual environment interaction techniques. Technical report, USA, 1995.

[34] M. R. Mine, F. P. Brooks Jr, and C. H. Sequin. Moving objects in space: Exploiting proprioception in virtual-environment interaction. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 19–26. ACM Press/Addison-Wesley Publishing Co., 1997. doi: 10.1145/258734.258747

[35] A. Moran, V. Gadepally, M. Hubbell, and J. Kepner. Improving big data visual analytics with interactive virtual reality. In *IEEE High Performance Extreme Computing Conference (HPEC)*, pp. 1–6. IEEE, 2015. doi: 10.1109/HPEC.2015.7322473

[36] D. A. Norman. *The psychology of everyday things*. Basic Books, 1988.

[37] N. Osawa, K. Asai, and Y. Y. Sugimoto. Immersive graph navigation using direct manipulation and gestures. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pp. 147–152. ACM, 2000. doi: 10.1145/502390.502418

[38] D. J. Peuquet. It's about time: A conceptual framework for the representation of temporal dynamics in geographic information systems. *Annals of the Association of American Geographers*, 84(3):441–461, 1994. doi: 10.1111/j.1467-8306.1994.tb01869.x

[39] D.-M. Pham and W. Stuerzlinger. Is the pen mightier than the controller? a comparison of input devices for selection in virtual and augmented reality. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, 2019. doi: 10.1145/3359996.3364264

[40] T. Piumsomboon, A. Clark, M. Billinghurst, and A. Cockburn. User-defined gestures for augmented reality. In *Human-Computer Interaction – INTERACT 2013*, pp. 282–299. Springer, 2013.

[41] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. 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*, pp. 79–80. ACM, 1996. doi: 10.1145/237091.237102

[42] I. Poupyrev, T. Ichikawa, S. Weghorst, and M. Billinghurst. Egocentric object manipulation in virtual environments: Empirical evaluation of interaction techniques. *Computer Graphics Forum*, 17(3):41–52, 1998. doi: 10.1111/1467-8659.00252

[43] J. Sauro and J. S. Dumas. Comparison of three one-question, post-task usability questionnaires. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1599–1608. ACM, 2009. doi: 10.1145/1518701.1518946

[44] T. Schubert, F. Friedmann, and H. Regenbrecht. The experience of presence: Factor analytic insights. *Presence*, 10(3):266–281, 2001. doi: 10.1162/105474601300343603

[45] M. Simpson, J. Zhao, and A. Klippel. Take a walk: Evaluating movement types for data visualization in immersive virtual reality. In *IEEE VIS Workshop on Immersive Analytics: Exploring Future Interaction and Visualization Technologies for Data Analytics*. IEEE, 2017.

[46] S. Y. Ssin, J. A. Walsh, R. T. Smith, A. Cunningham, and B. H. Thomas. GeoGate: Correlating geo-temporal datasets using an augmented reality space-time cube and tangible interactions. In *Proceedings of the 26th IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 210–219, 2019. doi: 10.1109/VR.2019.8797812

[47] M. Tomczak and E. Tomczak. The need to report effect size estimates revisited. An overview of some recommended measures of effect size. *Trends in Sport Sciences*, 21(1):19–25, 2014.

[48] J. A. Wagner Filho, C. M. Freitas, and L. Nedel. VirtualDesk: A comfortable and efficient immersive information visualization approach. *Computer Graphics Forum*, 37(3):415–426, 2018. doi: 10.1111/cgf.13430

[49] J. A. Wagner Filho, C. M. D. S. Freitas, and L. Nedel. Comfortable immersive analytics with the VirtualDesk metaphor. *IEEE Computer Graphics and Applications*, 39(3):41–53, May 2019. doi: 10.1109/MCG.2019.2898856

[50] J. A. Wagner Filho, M. F. Rey, C. M. D. S. Freitas, and L. Nedel. Immersive visualization of abstract information: An evaluation on dimensionally-reduced data scatterplots. In *Proceedings of the 25th IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 483–490. IEEE, March 2018. doi: 10.1109/VR.2018.8447558

[51] J. A. Wagner Filho, W. Stuerzlinger, and L. Nedel. Evaluating an immersive space-time cube geovisualization for intuitive trajectory data exploration. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):514–524, 2019. doi: 10.1109/TVCG.2019.2934415

[52] Y. Yang, B. Jenny, T. Dwyer, K. Marriott, H. Chen, and M. Cordeil. Maps and globes in virtual reality. *Computer Graphics Forum*, 37(3):427–438, 2018. doi: 10.1111/cgf.13431