SpatialTouch: Exploring Spatial Data Visualizations in Cross-reality

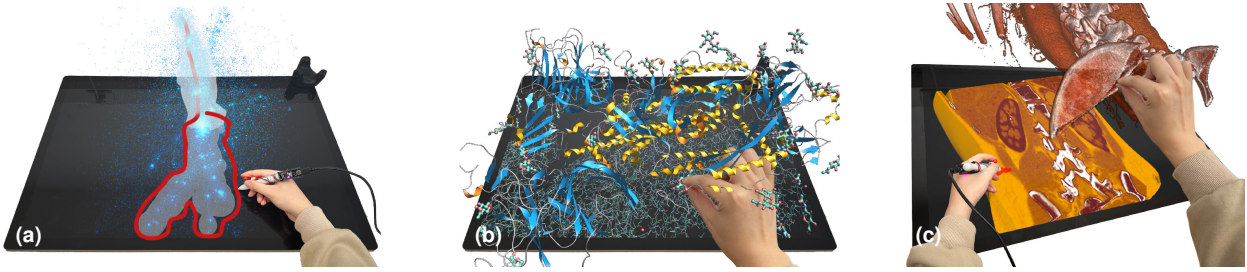Lixiang Zhao (ID), Tobias Isenberg (ID), Fuqi Xie (ID), Hai-Ning Liang (ID), Lingyun Yu (ID)

Fig. 1: Data exploration in SpatialTouch: (a) select astronomical points, (b) navigate molecular visualization, (c) annotate medical data.

Abstract—We propose and study a novel cross-reality environment that seamlessly integrates a monoscopic 2D surface (an interactive screen with touch and pen input) with a stereoscopic 3D space (an augmented reality HMD) to jointly host spatial data visualizations. This innovative approach combines the best of two conventional methods of displaying and manipulating spatial 3D data, enabling users to fluidly explore diverse visual forms using tailored interaction techniques. Providing such effective 3D data exploration techniques is pivotal for conveying its intricate spatial structures—often at multiple spatial or semantic scales—across various application domains and requiring diverse visual representations for effective visualization. To understand user reactions to our new environment, we began with an elicitation user study, in which we captured their responses and interactions. We observed that users adapted their interaction approaches based on perceived visual representations, with natural transitions in spatial awareness and actions while navigating across the physical surface. Our findings then informed the development of a design space for spatial data exploration in cross-reality. We thus developed cross-reality environments tailored to three distinct domains: for 3D molecular structure data, for 3D point cloud data, and for 3D anatomical data. In particular, we designed interaction techniques that account for the inherent features of interactions in both spaces, facilitating various forms of interaction, including mid-air gestures, touch interactions, pen interactions, and combinations thereof, to enhance the users' sense of presence and engagement. We assessed the usability of our environment with biologists, focusing on its use for domain research. In addition, we evaluated our interaction transition designs with virtual and mixed-reality experts to gather further insights. As a result, we provide our design suggestions for the cross-reality environment, emphasizing the interaction with diverse visual representations and seamless interaction transitions between 2D and 3D spaces.

Index Terms—Spatial data, immersive visualization, cross reality, interaction techniques.

---

1 INTRODUCTION

As computing and simulation technologies have advanced, the complexity of 3D structures within scientific datasets, alongside their contextual information, has grown substantially. Approaches to interactively explore 3D data have emerged to help users understand complex spatial structures across various disciplines. These datasets often encompass elaborate details, comprising millions of intricate components such as unstructured points in astronomical simulations [64, 65], flood simulation [8] or biological structures spanning multiple scales [24]. To gain a deeper understanding of such complex data, researchers often need to selectively focus on different regions. They may adjust visualization parameters to filter the data, emphasize key features, or alter data representations to highlight specific data attributes. This exploration can involve switching between 2D and 3D representations or displaying a combination of both for an in-depth analysis of data structures. Abstractocyte [43], e. g., is an interactive system designed to assist biologists in exploring the morphology of astrocytes using various levels of visual abstraction, while simultaneously analyzing neighboring neurons and

their connectivity. This tool allows users to manipulate a visualization widget of 2D abstraction space to smoothly transition between the different 3D and 2D abstraction levels, to identify morphological features for cells of interest. Furthermore, various visual representations, such as ball-and-stick models, are employed in molecular visualization to display 3D structures with different data features [31, 46, 67].

A common approach for supporting researchers to observe diverse combinations of data features, whether in 2D [4, 9, 75] or in 3D [41, 55], is through multiple linked views, facilitating connections among different representations. An alternative approach is to integrate 2D and 3D visual representations within a unified view [26]. Achieving a seamless interpretation of visual transitions and interaction with the data in such combined representations, however, is challenging—especially when dealing with complex data that contains a lot of detail. These challenges prompted us to explore a different display environment for presenting complex 3D data, one that integrates various visual representations and enables users to work with and seamlessly switch between different representations in a single setting.

To this end, we introduce SpatialTouch—a cross-reality (CR) environment [52] that merges a monoscopic interactive 2D surface (an interactive screen with touch and pen input) with a stereoscopic 3D space (an augmented reality HMD). We can visualize data simultaneously on both the 2D surface and in the 3D space such that the depiction on both displays remains in sync. Rather than imposing restrictions on which visual representations should be displayed in each space, we approach our setup as an integrated system in which the user decides where to display the data and how to interact with it. The integration of both views facilitates smooth transitions for data visualization between the 2D surface and the 3D space, ensuring a continuous situational awareness. With our CR SpatialTouch environment, we establish a platform that is capable of accommodating a wide range of data types that

- L. Zhao (赵锂想), F. Xie (谢馥琪), and L. Yu (俞凌云) are with Xi'an Jiaotong-Liverpool University, China. E-mail:{lixiang.zhao17@student | fuqi.xie20 @student | lingyun.yu@}.xjtlu.edu.cn. L. Yu is the corresponding author.
- T. Isenberg is with Université Paris-Saclay, CNRS, Inria, LISN, France. E-mail: given_name.family_name@inria.fr.
- H.-N. Liang (梁海宁) is with Hong Kong University of Science and Technology (Guangzhou), China. E-mail: hainingliang@hkust-gz.edu.cn.

can be displayed and interactively explored as needed. While creating many possibilities for visualization and interaction design, our setup also poses questions on how to best organize the data exploration.

A central question when hosting data visualizations within Spatial-Touch is how interaction techniques should be designed, ensuring that users can seamlessly continue their tasks, without having to consider how interactions should be performed in the respective other display space. Touch-enabled screens facilitate direct input on the 2D screen that typically shows projected views of the spatial 3D data; and many interaction techniques have been developed for such settings (e. g., for navigation [40, 62, 72, 81], selection [47, 78, 79], positioning [10, 66]). While the exploration of 3D spatial visualizations via a 2D view can pose challenges (e. g., it often requires users to rotate the data to explore its 3D aspects), due to the ubiquity of monoscopic screens, people have already developed a solid mental concept of how interactions are performed on the 2D screen. At the same time, however, they have also established a solid understanding of how gesture interactions are performed in the 3D space. When a visualization spans both of these spaces as with SpatialTouch, the question arises whether people can maintain their spatial awareness and are able to seamlessly transition between the different interaction techniques in the hybrid environment. To clarify, "spatial awareness" in our context refers to users' understanding of both the spatial position of interactions within the CR environment and the spatial status of the data, including its structure, orientation, and position, which guides them in performing specific actions.

Another crucial question is, when users explore spatial data, how they react as they engage with diverse data representations within a CR environment. We can assume that people decide how interactions should be executed based on their perception and understanding of the specific data structures. When selecting subsets of unstructured point cloud data, e. g., individuals may circle around them to define a range, whereas when selecting string-like structures they may brush along the structure. Any interaction pattern, however, may undergo additional alterations when the data transitions from 3D space to the 2D surface or vice versa. When selecting point cloud data which is rendered partially on the 2D screen and partially in 3D space, for instance, people may realize at some point during the selection action they can no longer circle around the target points in the depth direction. Consequently, the question arises if people maintain or change their interaction strategies when a visual representation is used to depict data across both spaces.

Other important considerations when merging 2D and 3D display spaces and constructing a CR environment include the challenge of effectively presenting 2D/3D visualizations and determining the appropriate level of detail. We can often choose from a diverse set of visual representations that each focus on a particular set of features or contain various scales or visual abstractions, which in our case may be partially rendered stereoscopically in 3D space and partially projected on the 2D screen. We thus ask: how should we represent the data such that users gain a comprehensive understanding of it? Moreover, understanding the best specific setup of our CR environment (in particular, the optimal angle or angle range for the 2D surface) is crucial to facilitate the user's interaction experience. In summary, we contribute

- a CR environment that combines a monoscopic 2D surface with a stereoscopic 3D space to facilitate joint spatial data exploration;
- a study on user actions and interaction strategies while interacting with spatial data;
- a design space of how CR supports 3D spatial data analysis across various visual representations, scales, and data abstractions; and
- three uses cases for demonstrating data manipulation, selection, annotation and measurements in three distinct domains.

2 RELATED WORK

Our work focuses on three key aspects of CR environments: the environment itself, data visualization, and interaction techniques employed within these CR settings. We review related work on these topics next.

2.1 Cross-reality environments

The reality-virtuality continuum, as conceptualized by Milgram and Kishino [42], provides a framework for data analysis and visualization techniques. It spans from visual analytics on 2D displays, to stereoscopic visualizations with immersive technologies, including augmented reality (AR), augmented virtuality (AV), and virtual reality (VR). Cross-reality [52] emphasizes seamless integrations and transitions among these visualization environments, offering users effective visual and algorithmic assistance tailored for maximal cognitive and perceptual suitability based on data, tasks, and user requirements.

Recent surveys have extensively explored cross-reality/virtuality environments from various perspectives, such as design and human factors [16], visualization and interaction techniques [7] and collaborative analysis [60]. Fröhler et al. [22] and Auda et al. [2] conducted comprehensive reviews of existing work, classifying the literature regarding different stages in the reality-virtuality continuum, visualization and view transition techniques, collaboration, visualization and visual analytics techniques, evaluation methods, and application domains. These opportunities and challenges have been recognized by many researchers and extensively discussed in conference workshops [34, 35, 52].

The main purposes of CR environments for domain research span from observation to collaborative analysis. An initial CR prototype was created by Kijima and Ojika [29] based on a projective see-through HMD (PHMD) and 2D monitor, demonstrating how **object manipulation** can be performed using keyboard and mouse. The most obvious benefit of combined 2D and 3D environments is the inherent display environments for **observing 2D and 3D data representations**. Seraji and Stuerzlinger [57] introduced an immersive visual analytics tool that allows users to conduct analysis at either end of the reality-virtuality continuum. It demonstrates that users can experience a lower cognitive load while viewing both 2D and 3D representations from two environments and task-switching between these virtuality modes. CR environments are also employed for **displaying and connecting different views**. Reipschläger et al. [51] proposed the combination of large interactive displays with personal head-mounted AR for displaying information in various views to facilitate data analysis. Their approach demonstrates how CR can be designed to address challenges encountered in solely large displays, such as perception, multi-user support, and managing data density and complexity. Such environments also show great benefits in making the best use of 2D and 3D interfaces and providing users with familiar ways **perceiving, creating and manipulating 3D contents**. Reipschläger and Dachselt presented an augmented design workstation [50], which seamlessly integrated an interactive surface displaying 2D views with a stereoscopic AR HMD, demonstrating how this combined space facilitates 3D model creation. For facilitating complex data exploration, decision making, and **collaborative analysis**, Butscher et al. [12] investigated the combination of AR and touch-sensitive tabletops for collaborative multi-dimensional data analysis through 3D parallel coordinates, using established touch interactions with visualizations anchored to the tabletop.

All these purposes for using CR heavily inspired the design of our own CR environment, including the selection of devices, layout, and setup. Typical such devices encompass 2D desktops, tablet/mobile devices, tabletop, wall displays, CAVEs, as well as AR and VR HMDs. We discuss these devices and the implication of their use in detail related to our intended visualization tasks and domains in Appx. E.

2.2 Visualization and its tasks in immersive environments

Immersive environments provide a stereoscopic view to convey complex 3D structural arrangements, with great potential in exploring many types of scientific data. Molecular data, e. g., often consists of densely packed 3D structures with intricate internal detail that spans multiple scales—making it challenging to comprehend the contextual information of the data. Alharbi et al. [1], e. g., introduced a guided-tour generator for immersive environments to navigate and communicate multi-scale, crowded, scientifically accurate 3D models. Similarly, point clouds such as astronomical simulations typically exhibit complexities: 3D occlusions, non-uniform feature density, or intricate data shapes. Zhao et al. [82] proposed target- and context-aware selection techniques for users to select sub-regions based on their understanding of the 3D structures yet without the need for high input precision.

A key aspect of these visualizations lies in their emphasis on spatial

attributes and their positions within the data context. When exploring or creating data of this nature, researchers often consider these spatial features displayed in different views or using different representations. That is where CR comes into play. In medical imaging, e. g., detailed views that focus on key regions are crucial while maintaining the overall data context. Coffey et al. [15] addressed this issue by presenting 3D medical data in the air alongside a detailed stereoscopic 3D view. This 3D data was complemented, moreover, by a 2D overview presented on a table, enabling users to interact with the data using familiar 2D touch gestures. Furthermore, Sereno et al. [59] proposed a spatial selection technique for 3D point data, wherein selections are performed on a tablet, while a stereoscopic view is provided by an AR HMD. With AR showing an overview of the point distribution, users can perform selections on the tablet and precisely control the selection in depth.

When CR is used to expand the limited display space, various visual representations can overlay the presented information to indicate connections or provide additional context. Langner et al. [32], e. g., introduced a conceptual framework that extends a 2D scatterplot displayed on a mobile device with superimposed 3D trajectories shown in AR. Similarly, Reipschläger et al. [51] proposed a system that augments data on large displays via AR. Satriadi et al. [53, 54] envisioned how to present multivariate data around physical scale models such as tangible globes, with relevant data attributes being displayed on and around the display or tangible interfaces. There are usually no strict constraints on how data should be displayed within and across multiple views, as different views may show different aspects. The crucial issue is that the provided information should be spatially linked to facilitate easy interpretation. This point is particularly important in our context. When using multiple visual representations or abstractions to illustrate complex spatial data, it is crucial to ensure that users consistently perceive and understand the spatial relationships and features of the data.

When transitioning a data visualization across various spaces, a central design principle is to ensure that users understand this transformation and can seamlessly continue their tasks, without losing focus. For this purpose, Schwajda et al. [56] developed and evaluated different variants of transformation to seamlessly transition graph visualizations from 2D to 3D representations and from a 2D surface to 3D AR space, facilitating the development of a mental model in both environments. Fröhler et al.'s survey [22] introduced various visual/view transition techniques, including portal, fade, and off-screen transition. These techniques guide users when a visualization shifts along the reality-virtuality continuum, such as moving from reality to AR. Lee et al. [33] presented a design space for data visualization transformations between a 2D screen and 3D AR, along with the interactions that facilitate this transformation. Their focus primarily lies on abstract data, such as transforming between 2D and 3D scatterplots, histograms, and parallel coordinate plot extrusions. To the best of our knowledge, there has been limited research addressing visual transitions for spatial data visualization. In our work we thus target smooth visual transitions of spatial data visualizations between 2D and 3D spaces, with an emphasis on preserving the intrinsic 3D structures and spatial distribution to maintain continuous situational awareness in CR environments.

2.3 Spatial/touch interaction in cross-reality

Drawing from the CR environments we discussed, typical input techniques include touch input, mid-air gestures, pen input, tangible and haptic interactions, as well as input through mobile devices or HMD controllers. This input predominantly occurs "inside" the stereoscopic view of 3D data presented in the air, on the 2D visualization displayed on a 2D interface, or directly with the mobile devices. Jackson et al. [27], e. g., proposed a prop-based **tangible interface** to control visualizations of thin 3D fiber structures. Fröhlich et al. [23] created a cubic tangible input device to precisely manipulate the slice with data visualization on a stereoscopic display. In the realm of model design or 3D content creation, multiple interactions facilitate the checking a 3D design, while creating and revising on 2D surfaces. DesignAR [50], e. g., allowed users to create and refine 3D models on a 2D surface with **touch and pen input**, while manipulating the virtual model with **mid-air gestures** in 3D. Similarly, Mockup [17] used sketching tools
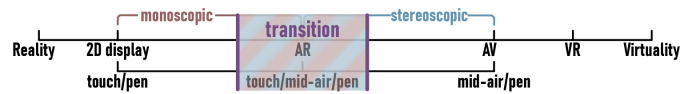

Fig. 2: SpatialTouch's placement within the reality-virtuality continuum.

to construct models on a tabletop, extruding the sketches from the 2D surface to 3D space using mid-air gestures. Hybridaxes [57] demonstrated how to transition 2D or 3D data from a display to AR, allowing users to switch interfaces using **free-hand interaction or controllers**. Wu et al. [77] proposed interactions that harnesses physical affordances to assist digital interaction in AR with hand gestures.

Interestingly, regardless of where these interactions occur, people typically have a solid mental concept of how interactions should be performed on a selected interface. On mobile/touch interfaces, e. g., users usually use two touches for scaling, and one touch for either x- and y-translation or trackball rotation [81]. Inspired by touch-based interaction, users tend to use two pinch gestures in the air to zoom content in or out [5]. As users perform a pinch gesture, their fingers naturally come into physical contact, creating explicit haptic feedback to reinforce the virtual action [77]. Thus, Lubos et al. [37] allow users to touch a 3D point cloud in mid-air and transform it with pinch gestures.

To maximize the benefits of CR environments, interactions can be performed on familiar interfaces or interfaces best suited for the task. The Interactive slice WIM [15], for instance, projected a data overview on the table and allows users to interact with the 3D data through familiar touch interaction. Similarly, López et al. [36] allowed viewers to use touch-based interactions to navigate and control the visualization on a monoscopic tablet, while observing data on a large stereoscopic display. In these cases, however, it is crucial to define how 2D interactions can be effectively mapped to the 3D visualization tasks.

In conclusion, we saw that most studies typically regard the CR environment as comprising distinct environments, each with its inherent and tailored interaction paradigm. It is thus interesting to investigate seamless interactions across different levels of virtuality—which is what we do. If users perceive the entire CR as an integrated environment, there is likely a substantial potential for interactions across the diverse interfaces to facilitate continuous actions, in a way that eases also the mental connection between the different output spaces.

3 SPATIALTOUCH: A CROSS-REALITY ENVIRONMENT

Before we detail our experiments, we first introduce our CR environment SpatialTouch (for its placement in the reality-virtuality continuum see Fig. 2) with its camera settings, input techniques, interaction devices, and general implementation. As we do so, we also discuss the key considerations drove the design. Our focus at this stage lies solely on creating an integrated visualization environment for displaying spatial data, without yet considering specific interaction designs.

SpatialTouch comprises two dedicated display areas: a monoscopic 2D interactive surface (Microsoft Surface Studio) and a stereoscopic 3D space (Microsoft HoloLens). We designed it to accommodate 3D spatial data and its associated information. As an integrated (hybrid) visualization space, the data can be positioned and manipulated anywhere within the CR environment—above, on, or below the surface, or spanning across it. To emphasize 3D structures with different data features, we want the visual representations that we use to show it to be able to take various forms, including 2D or 3D representations, various scales and forms of visual abstraction [68, 69], or combined forms such as the ball-and-stick representation in molecular visualization [31]. These requirements of having multiple forms of representation—later to be adjusted to the specific depiction location—set our approach apart from others as most existing CR applications fully transition data from one visual format to its 3D counterpart when moving it across different spaces (e. g., converting a 2D node-link diagram into a 3D visualization [56]). In addition, we not only focus on how data or views should be presented, arranged, and transformed in CR but we also provide a perspective view of the spatial data within the whole environment to ensure that viewers correctly perceive the 3D structures and their spatial distribution, to be able to maintain a continuous situational awareness while transitioning the data between both display spaces.
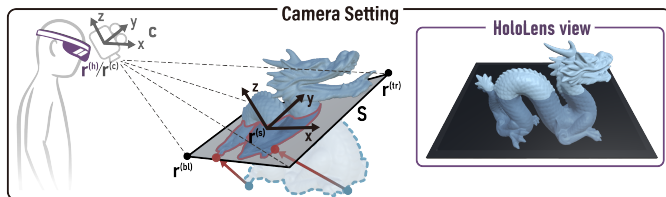
Fig. 3: The configurations of the AR camera and the Surface camera **c** are depicted, illustrating how the virtual content below the Surface (marked by a blue dotted line) is projected onto the Surface (a red solid line). The HoloLens view shows what users perceive in SpatialTouch.

Camera settings. In our design, regardless of the viewing angle, the data rendered on the surface and through the Hololens seamlessly combines into a cohesive 3D representation. To achieve this effect, we developed a rendering algorithm that aligns the visualization content displayed on the 2D surface with the AR visualization. We set two virtual cameras in our system, (1) an AR HMD camera rendering the 3D view and (2) a Surface camera **c** for rendering 2D view (Fig. 3). For the AR HMD camera, we employ a perspective projection, a standard method used in AR/VR applications. For the Surface camera **c**, inspired by the Fishtank VR concept [73], we first align **c** to the HMD's position and then adopt an oblique perspective projection. This way we dynamically calculate **c**'s parameters—position $\mathbf{r}^{(c)}$, orientation, and projection matrix $\mathbf{m}^{(c)}$—in each frame based on the AR HMD camera's position $\mathbf{r}^{(h)}$ and surface **s**'s center position $\mathbf{r}^{(s)}$ (we describe more detail on the camera configuration in Appx. C and also share an open-source simulator of the camera setup in our supplemental material).

Touch/Pen/Mid-air Interactions. Microsoft's API Surface Studio captures multi-touch input, while we can obtain mid-air input from the AR HMD. For precise interaction needs such as selection and annotation, however, we want to augment touch and mid-air input from the Surface Pen—both on the surface *and* in mid-air. While the Surface Pen input can easily be captured on the Surface, it is normally not detected when used in the air. To overcome this limitation, we attached an Arduino board to the pen, allowing us to detect events when users press the physical button on the pen without it resting on the surface.

Devices and Implementations. A 28-inch Microsoft Surface Studio (637 mm × 438 mm; 4,500 × 3,000 px) serves as the 2D surface of our SpatialTouch CR setup, which captures both pen and multi-touch input. It can be adjusted smoothly from a vertical position to a nearly horizontal orientation, the latter resulting in a slight inclination of $\approx 20°$ from the horizontal. As AR HMD we used Microsoft's HoloLens 2 (2,048 × 1,080 px per eye, 52° FoV), equipped with spatial tracking and gesture recognition. We connected both to a PC (Intel Core™ i9, 3.5 GHz, 64 GB RAM, GeForce RTX3090, 24 GB video memory), the AR HMD via holographic remoting. We attached a physical button to the Surface pen, along with the Arduino nano board on the back so that users could initiate actions both in 3D space and on the 2D surface. When the Arduino detects a button press, we compute the pen position based on the user's index finger position—tracked by the HoloLens using computer vision—, to which we add a constant offset to arrive at the pen tip. The resulting pen precision was sufficient for our prototype implementation. For more precise input, however, we could explore advanced technologies such as Logitech's VR Ink Pilot Edition.

We developed our prototype implementations with C# in the Unity 3D engine and ran on both the Surface and the HoloLens. We realized the communication between both display spaces through Universal Windows Platform sockets. After starting the program on the Hololens, we calibrate the setup by aligning the two coordinate systems via QR code tracking and manually manipulating the anchor point, as in past work [50]. After calibration, we employ HTC Vive trackers to follow the position and orientation of the Surface Studio.

4 ELICITATION STUDY

To explore potential interaction designs for SpatialTouch, we conducted an elicitation study to gauge users' reactions to this new environment. Similar to Wang et al. [70], our study encouraged participants to propose any interactions they could imagine. We captured their responses

and interactions, subsequently incorporating this feedback into our final interaction techniques design.

4.1 Study setup

Based on SpatialTouch that we just introduced in Sec. 3, we had the following goals: (G1) to determine whether an optimal position (or range) exists for the 2D surface to facilitate the observation of 3D data and to enhance user interaction; (G2) to understand how users perceive information within a CR environment; and (G3) to observe the participants' interactions and strategies while they complete exploration tasks with 3D spatial data, as well as to identify the physical locations of where the interactions occur. We pre-registered our study (`osf.io/avxr9`) and received IRB approval for the protocol (XJTLU University Research Ethics Review Panel, № 20240201174957).

Participants. We recruited eight voluntary participants from the local university, with ages ranging from 24 to 41 years old (mean 26.5, SD 5.5). The participants' past VR/AR experience included weekly use (3×), annual use (3×), and no past experience (2×). On average, they had 5.3 years of experience in the visualization or human-computer interaction fields. One participant was left-handed. All participants had normal or corrected-to-normal vision with no color deficiency, ensuring a clear ability to perceive the data visualization and colors in our study.

Datasets. To represent a variety of application domains we used five datasets with a diverse set of features (more detail in Appx. A):
Structured molecular data: a spike protein from the SARS-CoV-2 virus [45], reconstructed from the electron microscopy images.
Volume visualization data: MRI volume data with multiple 2D slices.
Unstructured point cloud data: three point datasets, each with different features: (1) a synthetic semi-spherical shell of particles; (2) an N-body simulation [64]; and (3) a Millennium-II data subset [65].

Task and Procedure. After providing their written informed consent, we asked participants to comfortably sit on a chair ≈ 0.5 m above the floor. We placed the surface screen on the table, ≈ 0.75 m from the floor, with an initial angle of $\approx 20°$ to the horizontal. Participants could adjust the chair height if desired. We then asked them to complete the three tasks described below. Importantly, for both manipulation and selection tasks, we did not show any feedback to the participants on their proposed interactions. We only provided them with a data context with highlighted (exploration or selection) targets and asked them to imagine how actions should be performed according to the given tasks.

Task 1. We instructed the participants to adjust the screen angle to achieve a suitable position for an optimal observation of the 3D structures. In addition, we reminded them of the importance of considering interaction on both the 2D Surface and in 3D space for a comprehensive understanding. We also informed the participants that they could re-adjust the screen angle whenever needed, throughout the study.

Task 2. We asked the participants to manipulate the five spatial datasets. In all manipulation tasks, we provided them with a target object and a target position. Specifically, we asked them to: (1) translate data displayed below and above the surface within each display space; (2) translate data displayed below and above the surface to the other display space; (3) translate data displayed partially in both spaces to a target position above or below the surface; and (4) rotate and scale data located in 3D space, within a 2D surface, or partially in both spaces.

Task 3. We asked the participants to select a given highlighted target object or highlighted regions of interest.

We recorded the screen angle chosen by participants, their actions (incl. head/hand position, orientation), and device input events for our analysis (which we make available at `osf.io/avxr9`). After completing the three tasks, we conducted semi-structured interviews with them to discuss their thoughts and suggestions on the interaction.

4.2 Findings

Our participants demonstrated that the integrated 3D data visualization in our environment allowed them to gain a comprehensive understanding of the 3D content. They mainly concentrated on the 3D visualization in the AR space and attempted to view it from various angles. Their decision on where to interact with the data was influenced by the location of key information such as target data or target location.
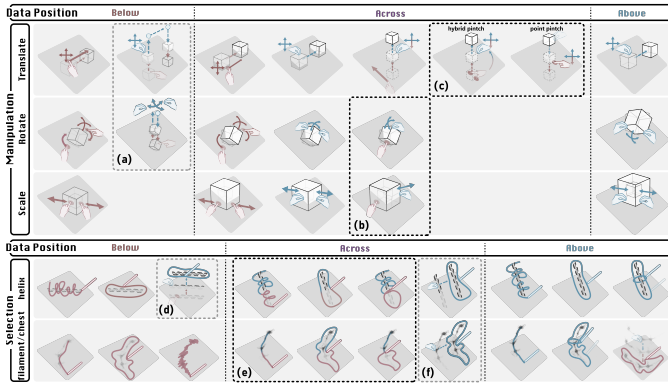
Fig. 4: Design space for interaction techniques for two visualization tasks: data manipulation and selection. Red: interactions on 2D surface; Blue: interactions in 3D space. **Below**, **Across**, and **Above**: positions of the target data/location. (a), (d), (f): move data above the surface for interaction. (b) (c) and (e): interaction transitions across both spaces.

Their preference for direct manipulation methods remained consistent, however, such as touching the data on the surface or brushing it in the air. Below we present our findings on surface angle, visualization techniques, and interaction techniques tailored to our CR environment.

4.2.1 Surface angle

From the initial $\approx 20°$ incline w.r.t. the horizontal table, some participants made slight adjustments at the beginning of the study, with angles ranging from 18.2° to 22.6° (mean 21.0°). After the initial setup, however, they did not make any further adjustments, despite being informed that they could do so. During the interviews, they said that the initial setting provided adequate space for observing stereoscopic 3D data through the AR HMD, while also offering a convenient position and orientation for interacting with data presented on the surface.

We also found that, when comparing our setup with a past stereoscopic multi-touch system [10, 11], a notable difference in display orientation: Butkiewicz et al.'s screen was nearly perpendicular to the table. This difference initially seems strange because Butkiewicz et al. [10, 11] had also relied on innovative interaction techniques using two-finger pinch gestures on the touch screen for tasks such as positioning the cursor under the display for 3D exploration and selection. The only difference appears to be that they had employed an auto-stereoscopic touch display, while we used an AR HMD. A main reason for the difference then appears to be the additional 3D space visible through the AR HMD, *decoupled from the touchscreen*, which substantially reduced people's reliance on the 2D surface. In the recorded data, in tasks where the visualization spanned both two spaces we saw that $\geq 67.5\%$ of the time was spent on observing 3D data in AR space. Users apparently preferred the stereoscopic AR space and only referred to the 2D surface when needed. Our environment also encouraged users to actively move around to find a good view, rather than adjusting the surface angle. All these factors diminished the importance and constraints of the surface angle. Moreover, the $\approx 20°$ angle is also appropriate for design work [50] as it provides stability and physical support from the base, enabling users to interact confidently and securely.

4.2.2 Visualization in CR

In addition to the Fish Tank VR view [73] on the 2D surface, we also asked participants to compare it with a static orthogonal view and a 2D slice view as alternatives. All participants expressed a strong preference for the perspective Fish Tank VR view, highlighting its effectiveness in creating a seamless 3D experience. Several participants recognized the significance and necessity of 2D slice view on the surface, particularly for tasks such as adjusting 2D image slices or marking MRI volume data. Furthermore, they also noted that the see-through AR HMD allowed them to look through the stereoscopic rendering directly to the 2D surface visualization. The resulting experience thus combined a comprehensive 3D data understanding with the ability to observe projected information, and the environment is flexible in how and where the information should be presented.

4.2.3 User interaction in CR

Interactions in 2D or 3D display spaces. When observing our participants engaging in either of the two individual methods of displaying and manipulating spatial 3D data—projected on the 2D surface or stereoscopically in 3D space—we noticed that they have a well-established mental concept of how an interaction should be performed. As we show in Fig. 4 (Manipulation, **Below**), when participants manipulated data presented below the surface, they used one finger to pan the visualization and two fingers to rotate and scale it on the surface. Conversely, when they saw data rendered in the AR HMD above the surface (Fig. 4, Manipulation, **Above**), they used one or two hands to grab and manipulate the data directly. Similarly, for the selection tasks, when the data was presented below the surface (Fig. 4, Selection, **Below**), participants preferred to brush or circle data on the surface. When the data was entirely above the screen (Fig. 4, Selection, **Above**), however, brushing and circling interactions were performed in 3D space.

Interestingly, some participants also used the benefits of the environment to aid their understanding of the data structure or to make more precise inputs. When both the target data and target location were below the surface, for instance, some participants would grab it out and put it back underneath to position the target. They explained in the interview that this extra action would provide them with more precise control over the depth in the positioning. Another example is that, to make precise selections, some participants would pull the data entirely out of the surface so that they can see the whole dataset stereoscopically or, alternatively, push it entirely below the surface for precise selection input. We highlighted these interactions in Fig. 4(a) and Fig. 4(d,f).

Interactions across two display spaces. Two types of tasks also required interaction across both spaces: if the data was visualized across both spaces or if it needed to be translated from one to the other. We show these actions in **Across** in Fig. 4 (Manipulation, Selection).

In the rotation and scaling tasks with a visualization spanning both spaces, participants performed actions on either the surface, in 3D space, or using both spaces for different tasks. They mainly made this decision based on where they perceived the key information. In addition, 4 participants favored using both spaces, most of them with a notable strategy: they (3×) used one finger to designate a rotation/ scaling center on the surface, while using their other hand to execute the rotation or scaling actions around the selected center (Fig. 4(b)). In the interviews, these participants mentioned for them that this approach was a more precise and secure method of manipulation.

In the selection tasks in which we placed the data across both spaces, we also observed intriguing patterns. Most participants (6×) approached our environment as an integrated system and used consistent selection strategies across both spaces. For example, participants followed the 3D structure and brushed along the data in 3D space, then continued brushing the rest along the structure projected on the 2D surface (Fig. 4(e, left)). Or they drew a large lasso across both spaces to enclose the data (Fig. 4(e, middle)). These selection strategies remained consistent even when they physically reached the surface and noticed that they could no longer follow the spatial structure in the depth direction.

The most interesting finding is that participants maintained their mental model when interacting with data across both display spaces. Some employed a mixed approach, brushing data in 3D space and circling data on the 2D surface, or vice versa (Fig. 4(e, right)), which indicates that they maintained spatial awareness and seamlessly transitioned between different interaction techniques in the hybrid environment.

In tasks requiring transition across both spaces, we observed two distinct transition methods (Fig. 4(c)). First, when the data was situated below the surface and needed to be moved out, some participants initially employed 2D pinch gestures to pull it out. Once their two fingers naturally came into physical contact, they seamlessly transitioned to 3D pinch gestures to continue the action. They used the same method to put the data back in its original position. Second, some participants pointed at the data on the 2D surface, assuming that it would automatically approach their finger position. They then continued the manipulation using 3D pinch gestures to pull it out. In the AR space, all participants uniformly used 3D pinch gestures, which facilitated positioning of the data—whether in 2D or 3D space—in a rapid manner.

Interactions on visual representations. Participants employed a consistent approach in manipulating all three datasets. They exhibited, however, varied approaches for selecting data with different representations in our environment. For the structured surface data, participants selected it by adhering to its 3D structure, such as wrapping a protein helix with a helical stroke. Conversely, for the unstructured point cloud data, they attempted to enclose it with a freeform lasso, such as drawing a lasso around the astronomical point cloud data. We observed that participants relied on their established mental models to determine how the data should be selected, based on their perception and understanding of the specific data structures. Therefore, most participants did not alter their selection strategies when interacting across both spaces—even though their interactions were physically limited to reach the visualization below the surface: they believed that through continuous input they could successfully select data within the CR environment.

5 SPATIALTOUCH FOR DOMAIN-SPECIFIC USE

Based on these findings, we then focused on developing SpatialTouch sample applications tailored to three distinct domains: astronomical point cloud analysis, molecular visualization, and medical anatomy imaging—each with its unique challenges for interactive visualization. Based on the optimal angle for the 2D surface identified in the elicitation study (G1), we set the angle of the surface at a 21° incline w.r.t. the horizontal table. We use the Fishtank VR view on the Surface camera and a perspective projection in AR HMD camera to ensure a seamless visualization (G2). In the following description we highlight, in particular, specific realizations within SpatialTouch for the given type of data or domain. We encourage readers to watch more detailed figures in Appx. D and the supplemental video for a demonstration.

5.1 Astronomical point cloud visualization

Astronomical simulation datasets typically comprise billions of spatial points [64, 65]. Researchers often need to navigate in 3D to obtain a clear view of the structures to select and explore the regions of interest. The task of data selection becomes paramount in this context, serving as a critical step in data visualization and exploration [7, 76]. While much research has been dedicated to developing selection techniques for 2D surfaces [14, 38, 48, 78, 79] and in 3D space [21, 30, 44, 61, 82], these methods are predominantly designed for a single display. In 3D space, although users can clearly see the spatial data distribution, limitations persist in their ability to precisely delineate selection regions. Conversely, the 2D surface facilitates accurate input for target inclusion but struggles with depth prediction, posing a challenge for selecting specific data regions. SpatialTouch bridges this gap by supporting data observation in AR, affording users a comprehensive understanding of data density distribution and context, while also facilitating precise data selection on a 2D surface. As an integrated environment—users are empowered with the flexibility to decide both *where* the selection occurs (whether on the 2D surface, within 3D space, or spanning both) and *how* these selections are made (via a freeform lasso or a brush).

Based on the interaction techniques detected from the elicitation study (G3, Fig. 4), we developed two new spatial selection techniques to facilitate a seamless transition from 3D selection to 2D selection, and vice versa. These techniques are based on the context- and target-aware selection metaphors CloudLasso [78], WYSIWYP [74], and MeTABrush [82]. These methods analyze the density distribution and data features within the local area of user interaction so that users can identify and select critical features of interest, without the need for precise input. Our first new technique, *BrushWYP*, draws inspiration from the user interactions illustrated in Fig. 4(e, left), MetaBrush [82], and WYSIWYP [74]. It enables users to trace the string-like shape of 3D point cloud data by brushing over these structures in the 3D space and continuing to brush the rest along the structure on the 2D surface. We designed our second new technique, *BrushLasso*, based on Fig. 4(e, right), MetaBrush [82], and CloudLasso [78]. This method allows users to begin their selection by brushing over the data in the 3D space and draw a lasso on the 2D surface to enclose the targeted points. Beyond the ability to transition between two spaces, both techniques are flexible in that they also allow users to brush the target points on the 2D
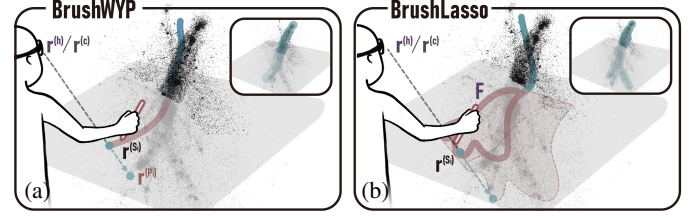

Fig. 5: Point cloud data selections: (a) BrushWYP, (b) BrushLasso.

surface and in 3D space, or encircle the target points on the 2D surface. Moreover, similar to our previous works [78, 79, 82], subtraction can be achieved using region-based techniques. A potential action to activate subtraction could be, for instance, to turn the Surface pen over and to rub the end of the pen over the selected data.

Technically similar to MeTACAST [82], we leveraged a continuous density field $\rho(\mathbf{r})$ to represent the particle density at location $\mathbf{r}$. We pre-compute the density of the field at all nodes i of the regular $128 \times 128 \times 128$ 3D grid box B that covers the dataset, denoted as $\rho(\mathbf{r}^{(i)})$, offline on GPU. We use the modified Breiman kernel density estimation with a finite-support adaptive Epanechnikov kernel [20]. This approach allows us to apply our selection not only to point clouds but also to volumetric data, which samples a scalar field that represents important data aspects in a visually salient way (not limited to density).

BrushWYP. We employ MeTABrush to allow users to select structures in 3D space by direct tracing along 3D structures. When the input transitions from the 3D space to 2D surface, however, the direct brushing on the target location is constrained. We address this issue with a modified WYSIWYP method, enabling users to select points below the surface by identifying the correct depth of the point of interest (POI).

Our algorithm initiates by sampling points along the input stroke on the surface, denoted as $\mathbf{r}^{(s)} = \{\mathbf{r}^{(s_0)}, \mathbf{r}^{(s_1)}, \ldots \mathbf{r}^{(s_n)}\}$, as well as above the surface, represented by $\mathbf{r}^{(a)} = \{\mathbf{r}^{(a_0)}, \mathbf{r}^{(a_1)}, \ldots \mathbf{r}^{(a_m)}\}$. For each sampled point $\mathbf{r}^{(s_i)}$ in $\mathbf{r}^{(s)}$, we project a ray from $\mathbf{r}^{(s_i)}$ toward the direction of $\mathbf{r}^{(s_i)} - \mathbf{r}^{(h)}$, where $\mathbf{r}^{(h)}$ denotes the position of the AR HMD (Fig. 5(a)). Then we traverse along each emitted ray in fixed steps and search for the POI $\mathbf{r}^{(p_i)}$ that exhibits the maximum density along the ray. This approach is different from the original WYSIWYP [74], which identifies the highest jump of accumulated scalar value along the ray. After traversing $\mathbf{r}^{(s)}$, we get a list of POIs, $\mathbf{r}^{(p)} = \{\mathbf{r}^{(p_0)}, \mathbf{r}^{(p_1)}, \ldots \mathbf{r}^{(p_n)}\}$. Subsequently, we obtain the input array of MeTABrush as $\mathbf{r}^{(i)} = \{\mathbf{r}^{(p)}, \mathbf{r}^{(a)}\}$ and select the target points with the MeTABrush method.

BrushLasso. We integrate MeTABrush and CloudLasso to provide users with the ability to brush target points in mid-air and encircle points on the surface through a single, seamless input. The original CloudLasso selects point cloud clusters of high density that fall within the input lasso. Directly merging both methods, however, could lead to disconnected selection volumes—one above the surface and several isolated ones below the surface—which might not align with the user's intention of making a continuous selection from the 3D space to the 2D surface. To solve this issue, we implemented a modified CloudLasso method that ensures that the selection volume is smooth and continuous.

This algorithm initiates by sampling points along the input stroke on the surface. For each sampled point $\mathbf{r}^{(a)}$ above the surface, we calculate an initial volume of interest (VOI) V_{init} with MeTABrush algorithm [82]. We then remove the parts below the screen and keep only the VOI above the surface $V_a = V_{\text{init}} \cap V_{3D}$, where V_{3D} is the space above the surface. We then connect the sampled points on the surface, $\mathbf{r}^{(s)}$, to form a lasso, $\mathbf{L}$ (Fig. 5(b)). We then map the particle coordinates to the view coordinates of the surface camera, $\mathbf{c}$, using the model-view transformation. Similar to CloudLasso [78], we then compute the lasso frustum $\mathbf{F}$ based on the first-level binding stage. We can then obtain the VOI below the surface $V_b = \mathbf{F} \cap V_{2D}$, where V_{2D} is the space below the surface. Finally, we combine both VOIs to get the interconnected VOI, $V_{CR} = V_b \cup V_a$. We compute the initial density threshold ρ_0 as:

$$\rho_0 = \frac{1}{N_{V_{CR}}} \sum_{n=1}^{N_{V_{CR}}} \rho(\mathbf{r}^{(n)}), \qquad (1)$$

where $N_{V_{CR}}$ is the number of grid-nodes, $\mathbf{r}^{(n)}$, inside of the combined VOI, V_{CR}. We select the volume V with density ρ above ρ_0 within V_{CR}:

$$V = \{\mathbf{r} \,|\, \mathbf{r} \in B, \mathbf{r} \in V_{CR}, \rho(\mathbf{r}) > \rho_0\}, \qquad (2)$$

We generate the iso-surface with Marching Cubes based on the density threshold ρ_0. Both resulting selection techniques facilitate a seamless and natural transition between 2D and 3D selections.

5.2 Molecular visualization

Molecular visualization is a field rich with diverse data representations, each designed to highlight various aspects of molecular structures and interactions. These include space-filling diagrams, ball-and-stick models, ribbon models, licorice visualization, backbone, and surface visualizations, etc. Given the complexity of molecular interactions, it is common to use these representations as a combination to provide a comprehensive view of the molecules. For biologists and chemists, gaining a deep understanding of how drug molecules interact within larger molecular structures is essential. They require insights into the precise location and distribution of these molecules to assess their interaction and affinity. Traditionally, they need to select specific features or regions of interest from biological sequences and link these on a 3D visualization rendered on a 2D display. The task of selecting spatial features and observing the dynamic behavior of molecules through 2D display, however, presents significant challenges. Even within a pure 3D environment, maintaining spatial awareness can be a struggle for users, given the inherent complexity of the involved datasets.

With SpatialTouch we address this need by merging the distinct display spaces to facilitate a concurrent visualization of different representations and abstractions. As we illustrate in Fig. 6, e.g., we can show the hyperball representation [13] stereoscopically in AR space for a detailed view of the molecule, while a ribbon model is rendered on the surface for context. Alternatively, we can put a ribbon diagram in AR space, complemented by a licorice diagram on the surface. SpatialTouch allows users to combine these representations based on their preferences and exploration needs. In addition, we enable them to adjust visual representations in the specific regions as needed, both in AR space and on the surface (Fig. 6(a)). Users can also directly grab the visualization to check details of local regions, as we show in Fig. 6(b).

We enable users to interact with data rendered below the surface using familiar touch interactions or above the surface through gesture interactions, as illustrated in Fig. 4 (manipulation, **Below** and **Above**). In addition, inspired by the findings in the elicitation study as illustrated in Fig. 4(c), we developed two seamless interaction transition techniques to enable users to move data across two spaces. The first method supports users to employ a familiar 2D pinch gesture on the touch surface to "pull" the data to the surface level (akin to Hancock et al. [25]), then continue with a 3D pinch gesture to "grab" it further into the air above the surface, as illustrated in Fig. 4(c, left) and Fig. 6(b). Conversely, performing this gesture in reverse enables users to seamlessly return the data to below the surface. This approach supports users to directly interact with and manipulate regions of interest, whether transferring them from the 3D space directly onto the 2D surface for extended analysis or relocating them to any other area within the CR environment for closer inspection. It is worth noting, however, that 2D pinch gestures are typically associated with zooming interactions. The second method thus allows users to press the screen to "push" data further into the depth of the surface or to "pull" it to the outside. The 3D pinch interaction in AR HMD remains, as we illustrate in Fig. 4(c, right). While both methods provide seamless translations across two spaces, we also implemented scaling and rotation: users can point at a particular feature on the 2D surface and use 3D pinch gestures to rotate or scale the data around that chosen point, as illustrated in Fig. 4(b).

5.3 Medical anatomical visualization

Medical imaging plays a central role in many healthcare practices for diagnosis, treatment planning, and patient care. In particular, 3D anatomical visualization has been used for treatment planning for various medical procedures in radiology [28] as well as for teaching [80]. It often relies on volume rendering to show internal structures in detail. While 2D slices are commonly used and are particularly effective for many diagnostic purposes, 3D visualizations are often necessary for
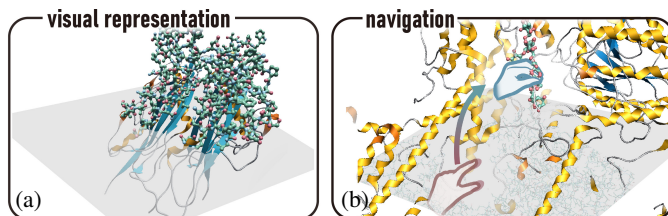

Fig. 6: Proteins visualized with UnityMol [18]. PDB ID: (a) 4fpq, (b) 8rfe. Users can "grab" the 3D visualization directly to check the local regions.
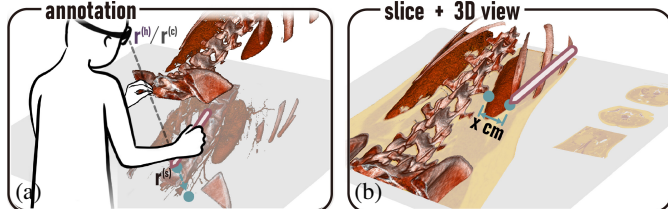

Fig. 7: (a) Annotation on 2D surface while grabbing 3D data; (b) 3D visualization superimposed on 2D slices to facilitate distance measurement.

addressing complex cases such as those involving intricate fractures. Physicians also frequently interact with both 2D slices and 3D visualizations for a range of tasks, including assessing injuries and devising treatment plans [49]. When planning surgical interventions for complicated orthopaedic injuries, for example, the integration of 2D and 3D visualizations allows surgeons to accurately visualize and navigate the affected areas [6]. This combined approach significantly enhances the precision of both planning and executing surgical procedures.

Prior work has explored the use of 2D displays [39] and 3D environments [63] for exploring 3D medical visualizations within clinical research tasks. Particularly relevant is Slice WIM [15], which uses a stereoscopic display and a multi-touch table to present both an overview and detailed views of 3D medical data, and projects 2D slices onto a wall or table display. For SpatialTouch, in contrast, we use an integrated environment that allows users to view 2D slices directly on the surface, while also observing stereoscopic renderings superimposed on these slices. Furthermore, users can use an interactive Surface pen for precise annotation and distance measurements. Moreover, with our method selected 2D slices can be saved and set aside on the surface, facilitating quick observation and navigating to specific local structures (Fig. 7(b)).

With this example we want to highlight that domain experts often need to engage in precise interactions, based on their observations of 3D structures—our environment being able to fulfill these requirements. Physicians can make annotations and measure distances directly on the 2D touch surface, while simultaneously viewing the 3D representation in AR space. To provide a clear view of the 2D slices and facilitate accurate interactions on the surface, we implement a "lifting" feature. It is activated with a 3D pinch gesture, which temporarily elevates the 3D volume visualization away from the 2D slice on the touch surface. This separation facilitates unobstructed annotation or marking. Once these tasks are complete, releasing the pinch gesture returns the 3D volume visualization to its original position on the screen, seamlessly integrating the new annotations with the remaining data.

In addition, with SpatialTouch interaction can happen anywhere within CR, including making annotations directly on the 2D slice (on the surface), marking features on the 3D visualization (above the surface), and selecting features that appear in depth but are rendered below the surface. We realized the latter interaction based on the WYSIWYP principle [74]. To identify the depth of a ROI, our method detects a significant change in the accumulated scalar value along the ray. This ray originates from the contact point on the surface, denoted as $r^{(s)}$, and projects it in the direction of $r^{(s)} - r^{(h)}$, as shown in Fig. 7(a). Our integrated environment thus offers users a seamless and precise interaction with medical 3D data for a detailed immersive examination.

6 EVALUATION

We ran two evaluations with domain experts, both pre-registered (osf.io/avxr9) and IRB-approved (same as before). First, we evaluated

SpatialTouch's usability with domain experts, focusing on its use in the domain. Second, we assessed the interaction with VR/AR/MR experts.

6.1 Evaluation with domain experts

We presented SpatialTouch to four domain experts: two biologists and one astronomer from the local university, and one doctor from a local hospital. All experts had >10 years of professional experience.

With the biologists, our demonstration focused on the SARS-CoV-2 virus spike protein [45] (Fig. 1(b)). The domain experts appreciated that the 2D and 3D perspective views offer a thorough understanding of the molecular data. They were highly interested in transitioning the visualization between 2D and 3D spaces, finding it intuitive to grab the visualization out from the 2D screen to check its 3D structure. They mentioned several scenarios where SpatialTouch could significantly benefit their research. First, the stereoscopic rendering in 3D space enhances molecular structure comprehension. It enables them to estimate distances between structures and gain a better understanding of molecular interactions and alignments. Second, 3D space provides a flexible interaction platform with more degrees of freedom. One expert noted that "this environment would be particularly useful for data analysis. I can analyze two molecular datasets on the surface and grab them to the 3D space to test various angles for fitting them together and observing their interaction. Afterward, I can return them back to the surface for further analysis." Third, the integration of 2D surface and 3D space combines their unique strengths, enhancing usability. Linking 2D representations on the surface with 3D visualizations in AR space, for instance, facilitates simultaneous editing of legends and observation of molecular interactions. Users can create and compare multiple copies within the environment. Moreover, the experts discussed how SpatialTouch could change their approach to data selection. Currently, they select data based on amino acid sequences. This process is cumbersome when the target region is spread over multiple sequence segments. To optimize the ligand affinity, for instance, they need to check the surrounding amino acids and select the pieces one by one from the sequence. Thus, they look forward to intelligent spatial selection techniques that facilitate efficient selections across both 2D and 3D representations to streamline this process.

The astronomer specializes in computational astrophysics, planetary system dynamics, few-body systems, and star cluster dynamics. After seeing the N-body simulation visualizations (Fig. 1(a)) in SpatialTouch, the expert mentioned that 3D stereoscopic rendering enhances the understanding of data structures, providing an intuitive grasp of spatial relationships and dynamic processes within the data. Its capability is particularly beneficial for exploring simulations of physical phenomena and identifying specific data patterns. Conversely, the 2D surface can be used for coding to perform complex data analysis tasks. The combination of both visual interfaces can allow users to leverage their existing data analysis practices, while motivating them to test their hypotheses and observe 3D simulations more easily. The expert did not show a strong preference for whether the selection interaction should occur on the 2D surface or in 3D space. Instead, they highly valued the flexibility to select based on where the data feature is perceived.

The doctor's research focuses on orthopaedics. After observing and manipulating the anatomical data visualization (Fig. 1(c)), he expressed strong interest in SpatialTouch and believed that it would play a significant role in surgical planning, where multi-views of 3D structures, 2D measurements, and precise interaction are required. The smooth and flexible transition between 2D and 3D allows doctors to view medical structures and the surgical plan from different perspectives. In addition, he mentioned that it is often difficult to control the depth at which a needle or a screw is inserted, which can harm neighboring tissues or organs. With SpatialTouch, they can view specific treatment locations on the 2D interface and test, compute, and plan where and how needles or screws should be inserted in 3D space.

6.2 Evaluation with VR/AR/MR experts

We then invited three VR/AR/MR experts with each over three years of experience and daily technology use. After obtaining their informed consent, we introduced them to SpatialTouch, in two sessions. First,

we focused on the manipulation tasks, including translation, rotation, and scaling. After a brief tutorial, we asked them to use our techniques to manipulate the molecular visualization from Sec. 5.2, comprising a combination of ribbon and licorice diagrams (Fig. 1(b)). We particularly evaluated four interaction transitions—two for translation (Fig. 4(c)) and rotation and scaling (Fig. 4(b))—and gathered their feedback in interviews. Our second session focused on our two selection techniques, BrushWYP and BrushLasso (Sec. 5.1). Following these two sessions, we asked them to fill in a questionnaire to assess and compare accuracy, efficacy, and user experience of both selection and manipulation techniques using 7-point Likert scales. In addition, throughout the study we encouraged them to share any feedback they may have.

The experts showed high engagement with our environment and its interactivity. They actively compared the visual effects and differences between the two visual representations across both spaces. All three experts expressed that they enjoyed the feeling of interacting with data seamlessly throughout the environment, appreciating the intuitive nature of the interactions without experiencing any confusion about how interactions should be performed. The experts particularly focused on the approaches for transferring data between the two displays. In their comparison of the two translation techniques, they mentioned that both methods provided a natural and fluid movement of data across spaces. They felt, however, the 2D pinch + 3D pinch gesture to be more precise than the 2D pointing + 3D pinch one. One reason was that the 2D pinch gave them precise control of data depth—allowing them to directly correlate the distance between their fingers with depth adjustment of the data below the surface. Depth control with a single finger-pointing gesture, in contrast, was challenging as it depended on visual feedback from below the 2D surface. An interesting observation was that all three experts did not realize that the familiar 2D pinch gesture had been repurposed from scaling to facilitate translation from **Below** to **Above** the surface. They seamlessly adapted to using an integrated 2D pinch + 3D pinch gesture to "grab" data across spaces. Moreover, while finding all interaction designs intuitive and easy to remember, the experts suggested that additional icons would be helpful visual cues for interaction, especially for transitions between the two distinct spaces. All experts also appreciated the design of both selection methods. Two of them favored BrushWYP for its uniform brushing approach from start to finish, aligning with their expectation of how data selection should work—even when transitioning across two distinct spaces. The other expert felt that either method could be effective. In comparing the accuracy of both, all experts felt both methods supported them in selecting the intended target data above and below the surface. Interestingly, we did not observe that using a lasso on the 2D screen would enhance their perception of accuracy. This may be because their main challenge was not to achieve precise 2D input but rather the inability to track the data below the surface, unlike in AR space. We solved this issue, however, with our target-aware selection technique.

7 SPATIALTOUCH AND CR VISUALIZATION AND INTERACTION

Based on the design considerations of SpatialTouch and other CR solutions (Appx. E), and their potential use in domain research, we can now revisit the initial questions we posed: (1) how to represent data to ensure users understand what they see and (2) how to design interaction techniques that allow users to seamlessly continue their tasks without the need to deliberate on how interactions translate between different display spaces. We experimented with a variety of data representation strategies to elucidate data attributes (e. g., ball-stick, surface, stripes). We also investigated different visualization placements, **Above** or **Below** the surface, employing various levels of detail and abstraction. We also explored the use of perspective views, orthographic views, or exclusively 2D cutting planes for presenting data content on 2D surfaces. One crucial lesson emerged from this work: it is not viable to impose strict limits on which visual representations should be displayed in each space. Instead, we need to treat CR environments as integrated systems, in which users can decide where data is displayed and how to interact with it. The focus of our design considerations thus shifted toward managing **transitions** effectively, breaking down into two essential facets: visualization transformation and interaction transition.

Visualization transformation involves the seamless migration of data representations between two distinct spaces, such as from a 2D display to 3D space. This process demands thoughtful design to ensure that the essence and clarity of the data are preserved during the transition to not break a user's mental model. Lee et al. [33] have thoroughly explored visualization transformation approaches that can be optimally designed to support visualization tasks. The transition of 3D spatial data from 2D to 3D is inherently facilitated by the data's existing 3D structure if perspective views are employed. The merging of 2D visualizations on flat surfaces with 3D visualizations in spatial environments to represent data features coherently, however, presents a considerable challenge: visual complexity. This complexity arises as users are presented with a blend of visual representations—2D or 3D, with varied visual elements—tailored to highlight distinct data features. For instance, when data moves from a 2D display to 3D space and the visual representations shift from surfaces and lines to stripes and curves, there is a concern about whether users can maintain their situational awareness. This challenge of visual complexity is not unique but is inherent across many environment designs within CR settings, especially when incorporating both situated and embedded visualizations to enhance user understanding of data features. After discussions with domain experts, we identified that a promising approach is to enable users to adjust visual representations locally—irrespective of being in 2D, 3D, or intermediate space—based on their specific needs. The shallow-depth area across two spaces, in particular, may be an appropriate place for this kind of adjustment. With SpatialTouch, users can effortlessly switch between different representations to dynamically examine data features. This flexibility in adapting visual representations aligns with Schwajada et al.'s [56] findings, who highlight the benefits of user-controlled transformation in enhancing task performance efficiency by catering to the users' requirements.

Interaction transition. To the best of our knowledge, there has not been a thorough exploration of interaction transition within CR environments. We thus introduce the notion of interaction transition as *the process in which interactions extend across various levels of virtuality within a CR environment, ensuring all actions maintain coherence, thus creating a continuous and seamless interactive experience.* Several factors influence this process, including the tactile feedback upon touching a physical surface, constraints in interacting with both depth and 2D surfaces, established mental models guiding 2D and 3D interactions, and how visual perception of data features impacts interaction.

Central to our understanding is the goal to preserve a user's **spatial awareness** to facilitate natural interaction within CR. This goal is twofold: first, we need to ensure that users maintain spatial awareness of their surrounding space (whether 2D or 3D), allowing them to seamlessly continue tasks without needing to reconsider interaction modalities for different display spaces. When using *BrushLasso* (Fig. 5(b)) in CR, e. g., users intuitively brush data in 3D and then draw a circle to enclose target points on the 2D surface. Second, it is crucial that users maintain spatial awareness of their data (structure, orientation, position). This awareness guides them in understanding where and how to perform specific actions. Users can trace, for instance, the 3D structure of data with *BrushWYP* (Fig. 5(a)) and seamlessly continue their interaction by following the structure onto the 2D surface.

To support such seamless transitions, our techniques need to be context- and target-aware. While a user's intention may be clear in one space, it may become ambiguous when transitioning to another. Considering the previous example, if a user is unable to trace a 3D point cloud on the 2D surface any technique needs to predict their selection intentions. This leads to another critical awareness we need to maintain: **situational awareness**—users should be able to predict the results of their actions in CR. Fig. 6(b) demonstrates this case—users use two pinch gestures to "grab" data from and to the surface. Their fingers naturally come into physical contact, promoting a continuous 3D pinch gesture to "grab" data into the 3D space. During this process, users have expected that the subsequent action would extract the data further. Another interaction technique, "lifting" (Fig. 7(a)), allows users to temporarily elevate the 3D volume for annotating a 2D slice, expecting that releasing the pinch gesture returns the data to its original position

on the surface once the annotation is complete. These interaction designs preserve spatial and situational awareness, thereby ensuring a deep engagement in task completion in the CR environment. Thus, they were highly favored by AR/VR experts in our evaluation.

Limitations. Similar to all visualization environments, SpatialTouch has its intrinsic limitations. Here, we focus on its limitations in hosting spatial visualizations. We discuss additional limitations caused by specific hardware settings in Appx. F. First, the stereoscopic rendering and the 2D monoscopic rendering may interfere with each other, potentially affecting the users' understanding of the spatial data. In our case, we use the HoloLens optical see-through HMD (OST)—when the 2D screen is too bright, it can interfere with the comprehension of the 3D rendering. Similarly, if a video see-through HMD (VST) is used, the stereoscopic rendering may occlude the 2D monoscopic rendering. In addition, the different visualization luminance between 2D displays and AR HMDs may also lead to a sense of disconnection. Therefore, blending the two renderings to obtain a coherent visualization is a meaningful research question for future development. Various factors should be considered to ensure an optimal color blending method, such as the specific AR device property (OST or VST), brightness, and other rendering-related settings. Second, 2D rendering is limited by the size of the screen, whereas AR theoretically provides unlimited space for 3D rendering. As a result, parts of the data visualization in the environment may not appear complete. While domain experts tend to focus on the local region of the data during exploration, we did not encounter any issues in the interviews. However, future researchers may need to take this into consideration when large-size visualizations are used in cross-reality interfaces that include fixed-size screens. Finally, collaborative data exploration is currently limited due to the design of FishTank VR monoscopic rendering on the 2D surface. Although users from different viewing angles can have their own view of the data in AR space, they share the same view rendered on the 2D surface. So only one user is able to see a cohesive 3D representation. A potential solution for the future would be to render several views on the screen at high frequency and use a technique similar to "shutter glasses," allowing each user to see their specific 2D rendering on the shared 2D surface.

8 CONCLUSION

Cross-reality is thus more than simply merging various levels of virtuality, where data is either positioned at a single space or moved to another place. Instead, it stands as an innovative and integrated environment for data presentation and exploration. Holding this vision, data can appear in any form and at any corner of the environment, tailored to the specific needs of each domain. Through the interviews with domain experts, we learned that transitioning their data between 2D and 3D spaces significantly motivates them to view and analyze their data from previously unexplored angles. In this light, we shared the design insights we gained through experimenting with various configurations for the CR environment, resulting in our recommendations to respect the users' mental models of the spaces in which they interact as well as of their data. Moreover, whether for understanding data or completing tasks, it is crucial to allow users to control data transformations, such as transforming data across spaces or change visual representations. Yet, the most critical aspect to consider is why CR is the appropriate choice for the task and data. This rationale shapes all other design aspects: the choices of environment, visualization, and interaction designs.

SUPPLEMENTAL MATERIAL POINTERS

We share our additional material at osf.io/avxr9. We also make our SpatialTouch simulator available at github.com/LixiangZhao98/Cross-Reality-Environment-SpatialTouch and the point cloud visualization and density estimation available at github.com/LixiangZhao98/PointCloud-Visualization-Tool.

REFERENCES

[1] R. Alharbi, O. Strnad, L. R. Luidolt, M. Waldner, D. Kouřil, C. Bohak, T. Klein, E. Gröller, and I. Viola. Nanotilus: Generator of immersive guided-tours in crowded 3D environments. *IEEE Trans Vis Comput Graph*, 29(3):1860–1875, 2023. doi: 10/gtnn26

[2] J. Auda, U. Gruenefeld, S. Faltaous, S. Mayer, and S. Schneegass. A scoping survey on cross-reality systems. *ACM Comput Surv*, 56(4), article no. 83, 38 pages, 2023. doi: 10/gtnn22

[3] B. Bach, R. Sicat, J. Beyer, M. Cordeil, and H. Pfister. The hologram in my hand: How effective is interactive exploration of 3D visualizations in immersive tangible augmented reality? *IEEE Trans Vis Comput Graph*, 24(1):457–467, 2018. doi: 10/gcp924

[4] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987. doi: 10/gf7bgr

[5] H. Benko and A. D. Wilson. Multi-point interactions with immersive omnidirectional visualizations in a dome. In *Proc. ITS*, pp. 19–28. ACM, New York, 2010. doi: 10/cxw789

[6] M. Benmahdjoub, A. Thabit, M.-L. C. van Veelen, W. J. Niessen, E. B. Wolvius, and T. v. Walsum. Evaluation of AR visualization approaches for catheter insertion into the ventricle cavity. *IEEE Trans Vis Comput Graph*, 29(5):2434–2445, 2023. doi: 10/mpq6

[7] L. Besançon, A. Ynnerman, D. F. Keefe, L. Yu, and T. Isenberg. The state of the art of spatial interfaces for 3D visualization. *Comput Graph Forum*, 40(1):293–326, 2021. doi: 10/gjbpxp

[8] S. Boorboor, Y. Kim, P. Hu, J. M. Moses, B. A. Colle, and A. E. Kaufman. Submerse: Visualizing storm surge flooding simulations in immersive display ecologies. *IEEE Trans Vis Comput Graph*, 13 pages, 2024. To appear. doi: 10/gt2smx

[9] A. Buja, J. A. McDonald, J. Michalak, and W. Stuetzle. Interactive data visualization using focusing and linking. In *Proc. Visualization*, pp. 156–163. IEEE Comp. Soc., Los Alamitos, 1991. doi: 10/c7wvqq

[10] T. Butkiewicz, A. H. Stevens, and C. Ware. Multi-touch 3D positioning with the Pantograph technique. In *Proc. I3D*, article no. 13, 9 pages. ACM, New York, 2019. doi: 10/gtnn2m

[11] T. Butkiewicz and C. Ware. Multi-touch 3D exploratory analysis of ocean flow models. In *Proc. OCEANS*, pp. 746–755–10. IEEE, Piscataway, 2011. doi: 10/gjb2jq

[12] S. Butscher, S. Hubenschmid, J. Müller, J. Fuchs, and H. Reiterer. Clusters, trends, and outliers: How immersive technologies can facilitate the collaborative analysis of multidimensional data. In *Proc. CHI*, pp. 1–12. ACM, New York, 2018. doi: 10/ggfwqj

[13] M. Chavent, A. Vanel, A. Tek, B. Levy, S. Robert, B. Raffin, and M. Baaden. Gpu-accelerated atom and dynamic bond visualization using hyperballs: A unified algorithm for balls, sticks, and hyperboloids. *Journal of computational chemistry*, 32(13):2924–2935, 2011. doi: 10/c3tdp5

[14] Z. Chen, W. Zeng, Z. Yang, L. Yu, C.-W. Fu, and H. Qu. LassoNet: Deep lasso-selection of 3D point clouds. *IEEE Trans Vis Comput Graph*, 26(1):195–204, 2020. doi: 10/gmghvt

[15] 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, multitouch VR interface. *IEEE Trans Vis Comput Graph*, 18(10):1614–1626, 2012. doi: 10/fcds9g

[16] A. Çöltekin, I. Lochhead, M. Madden, S. Christophe, A. Devaux, C. Pettit, O. Lock, S. Shukla, L. Herman, Z. Stachoň, P. Kubíček, D. Snopková, S. Bernardes, and N. Hedley. Extended reality in spatial sciences: A review of research challenges and future directions. *ISPRS Int J Geo-Inf*, 9(7), article no. 439, 29 pages, 2020. doi: 10/ghk9xk

[17] B. R. De Araújo, G. Casiez, J. A. Jorge, and M. Hachet. Mockup Builder: 3D modeling on and above the surface. *Comput Graph*, 37(3):165–178, 2013. doi: 10/f4wv2s

[18] S. Doutreligne, T. Cragnolini, S. Pasquali, P. Derreumaux, and M. Baaden. Unitymol: interactive scientific visualization for integrative biology. In *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*, pp. 109–110. IEEE, 2014. doi: 10/m53v

[19] B. Ens, S. Goodwin, A. Prouzeau, F. Anderson, F. Y. Wang, S. Gratzl, Z. Lucarelli, B. Moyle, J. Smiley, and T. Dwyer. Uplift: A tangible and immersive tabletop system for casual collaborative visual analytics. *IEEE Trans Vis Comput Graph*, 27(2):1193–1203, 2021. doi: 10/ghgt5x

[20] B. Ferdosi, H. Buddelmeijer, S. Trager, M. Wilkinson, and J. Roerdink. Comparison of density estimation methods for astronomical. *Astron Astrophys*, 531, article no. A114, 16 pages, 2011. doi: 10/cj3s5m

[21] A. Franzluebbers, C. Li, A. Paterson, and K. Johnsen. Virtual reality point cloud annotation. In *Proc. SUI*, article no. 14, 11 pages. ACM, New York, 2022. doi: 10/gtn4jk

[22] B. Fröhler, C. Anthes, F. Pointecker, J. Friedl, D. Schwajda, A. Riegler, S. Tripathi, C. Holzmann, M. Brunner, H. Jodlbauer, H.-C. Jetter, and C. Heinzl. A survey on cross-virtuality analytics. *Comput Graph Forum*, 41(1):465–494, 2022. doi: 10/gtnn2w

[23] B. Fröhlich and J. Plate. The Cubic Mouse: A new device for three-dimensional input. In *Proc. CHI*, pp. 526–531. ACM, New York, 2000. doi: 10/dgg8k8

[24] L. A. Garrison, I. Kolesar, I. Viola, H. Hauser, and S. Bruckner. Trends & opportunities in visualization for physiology: A multiscale overview. *Comput Graph Forum*, 41(3):609–643, 2022. doi: 10/gtmk6z

[25] M. Hancock, T. ten Cate, S. Carpendale, and T. Isenberg. Supporting sandtray therapy on an interactive tabletop. In *Proc. CHI*, pp. 2133–2142. ACM, New York, 2010. doi: 10/dv4k73

[26] J. Hong, R. Hnatyshyn, E. A. Santos, R. Maciejewski, and T. Isenberg. A survey of designs for combined 2D+3D visual representations. *IEEE Trans Vis Comput Graph*, 30(6):2888–2902, 2024. doi: 10/gtnn2v

[27] B. Jackson, T. Y. Lau, D. Schroeder, K. C. Toussaint, and D. F. Keefe. A lightweight tangible 3D interface for interactive visualization of thin fiber structures. *IEEE Trans Vis Comput Graph*, 19(12):2802–2809, 2013. doi: 10/gh38r2

[28] S. Jadhav and A. E. Kaufman. MD-Cave: An immersive visualization workbench for radiologists. *IEEE Trans Vis Comput Graph*, 29(12):4832–4844, 2022. doi: 10/gt2smv

[29] R. Kijima and T. Ojika. Transition between virtual environment and workstation environment with projective head mounted display. In *Proc. VR*, pp. 130–137. IEEE Comp. Soc., Los Alamitos, 1997. doi: 10/fwtdxw

[30] M. Krüger, T. Gerrits, T. Römer, T. Kuhlen, and T. Weissker. IntenSelect+: Enhancing score-based selection in virtual reality. *IEEE Trans Vis Comput Graph*, 30(5):2829–2838, 10 pages, 2024. doi: 10/gtn4jm

[31] D. Kut'ák, P.-p. Vázquez, T. Isenberg, M. Krone, M. Baaden, J. Byška, B. Kozlíková, and H. Miao. State of the art of molecular visualization in immersive virtual environments. *Comput Graph Forum*, 42(6), article no. e14738, 29 pages, 2023. doi: 10/kt4n

[32] R. Langner, M. Satkowski, W. Büschel, and R. Dachselt. MARVIS: Combining mobile devices and augmented reality for visual data analysis. In *Proc. CHI*, article no. 468, 17 pages. ACM, New York, 2021. doi: 10/gksmhm

[33] B. Lee, M. Cordeil, A. Prouzeau, B. Jenny, and T. Dwyer. A design space for data visualisation transformations between 2D and 3D in mixed-reality environments. In *Proc. CHI*, article no. 25, 14 pages. ACM, New York, 2022. doi: 10/kvj4

[34] H.-N. Liang, H.-C. Jetter, F. Maurer, U. Gruenefeld, M. Billinghurst, and C. Anthes. *1st Joint Workshop on Cross Reality*. 2023. doi: 10/gtnn2r

[35] H.-N. Liang, L. Yu, and F. Liarokapis. Workshop: Mixing realities: Cross-reality visualization, interaction, and collaboration. In *VR and 3DUI Abstracts and Workshops*, pp. 298–300. IEEE Comp. Soc., Los Alamitos, 2023. doi: 10/gtnn2q

[36] D. López, L. Oehlberg, C. Doger, and T. Isenberg. Towards an understanding of mobile touch navigation in a stereoscopic viewing environment for 3D data exploration. *IEEE Trans Vis Comput Graph*, 22(5):1616–1629, 2016. doi: 10/f8ghxc

[37] P. Lubos, R. Beimler, M. Lammers, and F. Steinicke. Touching the cloud: Bimanual annotation of immersive point clouds. In *Proc. 3DUI*, pp. 191–192. IEEE Comp. Soc., Los Alamitos, 2014. doi: 10/gjb2n8

[38] J. F. Lucas. Design and evaluation of 3D multiple object selection techniques. Master's thesis, Virginia Polytechnic Institute and State University, USA, 2005. hdl: 10919/31769.

[39] C. Lundström, T. Rydell, C. Forsell, A. Persson, and A. Ynnerman. Multi-touch table system for medical visualization: Application to orthopedic surgery planning. *IEEE Trans Vis Comput Graph*, 17(12):1775–1784, 2011. doi: 10/bzwxj8

[40] A. Martinet, G. Casiez, and L. Grisoni. Integrality and separability of multitouch interaction techniques in 3D manipulation tasks. *IEEE Trans Vis Comput Graph*, 18(3):369–380, 2012. doi: 10/c6g7jt

[41] M. Meuschke, S. Voss, O. Beuing, B. Preim, and K. Lawonn. Com-

bined visualization of vessel deformation and hemodynamics in cerebral aneurysms. *IEEE Trans Vis Comput Graph*, 23(1):761–770, 2017. doi: 10/f92d8h

[42] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. In *Proc. Telemanipulator and Telepresence Technologies*, pp. 282–292. SPIE, Bellingham, 1995. doi: 10/dh8jnv

[43] H. Mohammed, A. K. Al-Awami, J. Beyer, C. Cali, P. Magistretti, H. Pfister, and M. Hadwiger. Abstractocyte: A visual tool for exploring nanoscale astroglial cells. *IEEE Trans Vis Comput Graph*, 24(1):853–861, 2018. doi: 10/gcqk6d

[44] R. A. Montano-Murillo, C. Nguyen, R. H. Kazi, S. Subramanian, S. DiVerdi, and D. Martinez-Plasencia. Slicing-Volume: Hybrid 3D/2D multi-target selection technique for dense virtual environments. In *Proc. VR*, pp. 53–62. IEEE Comp. Soc., Los Alamitos, 2020. doi: 10/gjb2j2

[45] N. Nguyen, O. Strnad, T. Klein, D. Luo, R. Alharbi, P. Wonka, M. Maritan, P. Mindek, L. Autin, D. S. Goodsell, and I. Viola. Modeling in the time of COVID-19: Statistical and rule-based mesoscale models. *IEEE Trans Vis Comput Graph*, 27(2):722–732, 2021. doi: 10/k8sh

[46] S. I. O'Donoghue, D. S. Goodsell, A. S. Frangakis, F. Jossinet, R. A. Laskowski, M. Nilges, H. R. Saibil, A. Schafferhans, R. C. Wade, E. Westhof, and A. J. Olson. Visualization of macromolecular structures. *Nat Methods*, 7(3):S42–S55, 2010. doi: 10/df36n3

[47] A. Olwal, S. Feiner, and S. Heyman. Rubbing and tapping for precise and rapid selection on touch-screen displays. In *Proc. CHI*, pp. 295–304. ACM, New York, 2008. doi: 10/dgz29t

[48] G. Perelman, E. Dubois, A. Probst, and M. Serrano. Visual transitions around tabletops in mixed reality: Study on a visual acquisition task between vertical virtual displays and horizontal tabletops. *Proc ACM Hum-Comput Interact*, 6(ISS):660–679, 2022. doi: 10/grd27g

[49] M. Pfeiffer, H. Kenngott, A. Preukschas, M. Huber, L. Bettscheider, B. Müller-Stich, and S. Speidel. Imhotep: virtual reality framework for surgical applications. *International journal of computer assisted radiology and surgery*, 13:741–748, 2018. doi: 10/gdj9p9

[50] P. Reipschläger and R. Dachselt. DesignAR: Immersive 3D-modeling combining augmented reality with interactive displays. In *Proc. ISS*, pp. 29–41. ACM, New York, 2019. doi: 10/gtdktf

[51] P. Reipschläger, T. Flemisch, and R. Dachselt. Personal augmented reality for information visualization on large interactive displays. *IEEE Trans Vis Comput Graph*, 27(2):1182–1192, 2021. doi: 10/ghgt5w

[52] A. Riegler, C. Anthes, H.-C. Jetter, C. Heinzl, C. Holzmann, H. Jodlbauer, M. Brunner, S. Auer, J. Friedl, B. Fröhler, C. Leitner, F. Pointecker, D. Schwajda, and S. Tripathi. Cross-virtuality visualization, interaction and collaboration. In *Proc. XR@ISS*, article no. 1, 4 pages, 2020. Online: ceur-ws.org/Vol-2779/paper1.pdf.

[53] K. Satriadi, A. Cunningham, B. Thomas, A. Drogemuller, A. Odi, N. Patel, C. Aston, and R. Smith. Augmented scale models: Presenting multivariate data around physical scale models in augmented reality. In *Proc. ISMAR*, pp. 54–63. IEEE Comp. Soc., Los Alamitos, 2022. doi: 10/gtnn2p

[54] K. A. Satriadi, J. Smiley, B. Ens, M. Cordeil, T. Czauderna, B. Lee, Y. Yang, T. Dwyer, and B. Jenny. Tangible globes for data visualisation in augmented reality. In *Proc. CHI*, article no. 505, 16 pages. ACM, New York, 2022. doi: 10/grv5tm

[55] M. Schäfer, N. Brich, J. Byška, S. M. Marques, D. Bednář, P. Thiel, B. Kozlíková, and M. Krone. Invado: Interactive visual analysis of molecular docking data. *IEEE Trans Vis Comput Graph*, 30(4):1984–1997, 2024. doi: 10/gtnn2n

[56] D. Schwajda, J. Friedl, F. Pointecker, H.-C. Jetter, and C. Anthes. Transforming graph data visualisations from 2D displays into augmented reality 3D space: A quantitative study. *Front Virtual Reality*, 4, article no. 1155628, 19 pages, 2023. doi: 10/kvj9

[57] M. R. Seraji and W. Stuerzlinger. Hybridaxes: An immersive analytics tool with interoperability between 2D and immersive reality modes. In *Proc. ISMAR-Adjunct*, pp. 155–160. IEEE Comp. Soc., Los Alamitos, 2022. doi: 10/gtnn2x

[58] M. Sereno, L. Besançon, and T. Isenberg. Supporting volumetric data visualization and analysis by combining augmented reality visuals with multi-touch input. In *EuroVis Posters*, pp. 21–23, 2019. doi: 10/kt5f

[59] M. Sereno, S. Gosset, L. Besançon, and T. Isenberg. Hybrid touch/tangible spatial selection in augmented reality. *Comput Graph Forum*, 41(3):403–415, 2022. doi: 10/gqq53j

[60] M. Sereno, X. Wang, L. Besançon, M. J. Mcguffin, and T. Isenberg. Collaborative work in augmented reality: A survey. *IEEE Trans Vis Comput Graph*, 28(6):2530–2549, 2022. doi: 10/gjkq7w

[61] R. Shi, Y. Wei, X. Qin, P. Hui, and H.-N. Liang. Exploring gaze-assisted and hand-based region selection in augmented reality. *Proc ACM Hum-Comput Interact*, 7(ETRA), article no. 160, 19 pages, 2023. doi: 10/gt2sm2

[62] P. Song, X. Yan, W. B. Goh, A. Q. Chen, and C.-W. Fu. Hand-posture-augmented multitouch interactions for exploratory visualization. In *SIGGRAPH ASIA Technical Briefs*, article no. 27, 4 pages. ACM, New York, 2016. doi: 10/gtnn2k

[63] M. Sousa, D. Mendes, S. Paulo, N. Matela, J. Jorge, and D. S. o. Lopes. VRRRRoom: Virtual reality for radiologists in the reading room. In *Proc. CHI*, p. 4057–4062. ACM, New York, 2017. doi: 10/mn9z

[64] V. Springel, J. Wang, M. Vogelsberger, A. Ludlow, A. Jenkins, A. Helmi, J. F. Navarro, C. S. Frenk, and S. D. White. Aquarius project: The subhaloes of galactic haloes. *Mon Not R Astron Soc*, 391(4):1685–1711, 2008. doi: 10/fsjgzw

[65] V. Springel, S. D. M. White, A. Jenkins, C. S. Frenk, N. Yoshida, L. Gao, J. F. Navarro, R. J. Thacker, D. J. Croton, J. C. Helly, J. A. Peacock, S. Cole, P. A. Thomas, H. M. P. Couchman, A. E. Evrard, J. M. Colberg, and F. R. Pearce. Simulations of the formation, evolution and clustering of galaxies and quasars. *Nature*, 435:629–636, 2005. doi: 10/c3cmxr

[66] S. Strothoff, D. Valkov, and K. Hinrichs. Triangle cursor: Interactions with objects above the tabletop. In *Proc. ITS*, pp. 111–119. ACM, New York, 2011. doi: 10/b6jrqq

[67] M. van Der Zwan, W. Lueks, H. Bekker, and T. Isenberg. Illustrative molecular visualization with continuous abstraction. *Comput Graph Forum*, 30(3):683–690, 2011. doi: 10/c893rz

[68] I. Viola, M. Chen, and T. Isenberg. Visual abstraction. In *Foundations of Data Visualization*, chap. 2, pp. 15–37. Springer, Berlin, 2020. doi: 10/gk874c

[69] I. Viola and T. Isenberg. Pondering the concept of abstraction in (illustrative) visualization. *IEEE Trans Vis Comput Graph*, 24(9):2573–2588, 2018. doi: 10/gd3k7m

[70] N. Wang, D. Zielasko, and F. Maurer. User preferences for interactive 3D object transitions in cross reality – An elicitation study. In *Proc. AVI*, article no. 22, 9 pages. ACM, New York, 2024. doi: 10/gt2smz

[71] X. Wang, L. Besançon, D. Rousseau, M. Sereno, M. Ammi, and T. Isenberg. Towards an understanding of augmented reality extensions for existing 3D data analysis tools. In *Proc. CHI*, article no. 528, 13 pages. ACM, New York, 2020. doi: 10/gm4jj8

[72] X. Wang, L. Besançon, M. Ammi, and T. Isenberg. Augmenting tactile 3D data navigation with pressure sensing. *Comput Graph Forum*, 38(3):635–647, 2019. doi: 10/gjbgt9

[73] C. Ware, K. Arthur, and K. S. Booth. Fish tank virtual reality. In *Proc. CHI*, pp. 37–42. ACM, New York, 1993. doi: 10/cqjwdf

[74] A. Wiebel, F. M. Vos, D. Foerster, and H.-C. Hege. WYSIWYP: What you see is what you pick. *IEEE Trans Vis Comput Graph*, 18(12):2236–2244, 2012. doi: 10/f4ft8h

[75] G. Wills. Linked data views. In C.-h. Chen, W. Härdle, and A. Unwin, eds., *Handbook of Data Visualization*, chap. II.9, pp. 217–241. Springer, Berlin, 2008. doi: 10/dkvn57

[76] G. J. Wills. Selection: 524,288 ways to say "this is interesting". In *Proc. InfoVis*, pp. 54–60. IEEE Comp. Soc., Los Alamitos, 1996. doi: 10/bjq6dh

[77] S. Wu, D. Byrne, and M. W. Steenson. "megereality": Leveraging physical affordances for multi-device gestural interaction in augmented reality. In *CHI Extended Abstracts*, article no. INT008, 4 pages. ACM, New York, 2020. doi: 10/gmkzwj

[78] L. Yu, K. Efstathiou, P. Isenberg, and T. Isenberg. Efficient structure-aware selection techniques for 3D point cloud visualizations with 2DOF input. *IEEE Trans Vis Comput Graph*, 18(12):2245–2254, 2012. doi: 10/f4fv9z

[79] L. Yu, K. Efstathiou, P. Isenberg, and T. Isenberg. CAST: Effective and efficient user interaction for context-aware selection in 3D particle clouds. *IEEE Trans Vis Comput Graph*, 22(1):886–895, 2016. doi: 10/kt5n

[80] L. Yu, J. Ouwerling, P. Svetachov, F. H. J. van Hoesel, P. M. A. van Ooijen, and J. Kosinka. VeLight: A 3D virtual reality tool for CT-based anatomy teaching and training. *J Vis*, 25(2):293–306, 2022. doi: 10/mpq7

[81] L. Yu, P. Svetachov, P. Isenberg, M. H. Everts, and T. Isenberg. FI3D: Direct-touch interaction for the exploration of 3D scientific visualization spaces. *IEEE Trans Vis Comput Graph*, 16(6):1613–1622, 2010. doi: 10/fc2df8

[82] L. Zhao, T. Isenberg, F. Xie, H.-N. Liang, and L. Yu. MeTACAST: Target- and context-aware spatial selection in VR. *IEEE Trans Vis Comput Graph*, 30(1):480–494, 2024. doi: 10/gtnn25

SpatialTouch: Exploring Spatial Data Visualizations in Cross-reality

Appendix

In this appendix we provide additional images and discussion beyond the material that we could include in the main paper due to space limitations or because it was not essential for explaining our approach.
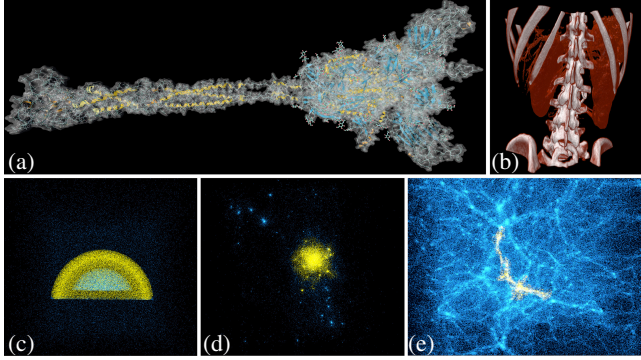

Fig. 8: Elicitation study datasets: (a) Protein data (data from [45]) , (b) MRI multi-slice anatomical data, (c)filament.

A DATASETS USED IN THE ELICITATION STUDY

Datasets. We used five datasets with a diverse set of data features, including:

Structured surface data: The molecular structure data (Fig. 8(a)) is a spike protein from the SARS-CoV-2 virus [45], reconstructed from the electron microscopy images. It is rendered as a ribbon diagram, which includes lines, sheets, and helix structures.

Volume visualization data: The MRI volume data (Fig. 8(b)) comprises multiple slices and facilitates interactive cutting plane exploration to focus on specific regions of the anatomical structures.

Unstructured point cloud data: We used three point cloud datasets: (1) a synthetic semi-spherical shell of particles (Fig. 8(c)) that partially encompasses a half-ball of interfering particles; (2) a cosmological N-body simulation [64] (Fig. 8(d)) with a vast, densely populated central cluster encircled by numerous smaller clusters; and (3) the Millennium-II data subset [65] (Fig. 8(e)), a complex network of filaments connecting high-density clusters.

B DESIGN SPACE FOR USER INTERACTION IN CR

To improve the readability of Fig. 4, we also include it in this appendix at full page width as Fig. 9.

C ALGORITHM FOR CAMERA SETTING

To ensure the data visualized on the surface and through the Hololens merge seamlessly into a unified 3D representation, we developed a rendering algorithm that aligns the visual content on the 2D surface with the AR visualization. To achieve this alignment we need to compute the surface position of data located below the surface. The positions of the surface camera and the AR HMD camera are denoted as $\mathbf{r}^{(c)}$ and $\mathbf{r}^{(h)}$, respectively. The configuration process involves the following key steps in each frame:

1. We translate **c** to the HMD's position, in which $\mathbf{r}^{(c)}{=}\mathbf{r}^{(h)}$.
2. We adjust the forward direction of the **c** (negative *z*-axis) to **s** perpendicularly, in which the *z*-axis of **c** aligns with the *z*-axis of **s**.
3. We modify the *x*-axis of **c** to align with the *x*-axis of **s**, maintaining consistency in horizontal orientation between **c** and **s**.
4. We compute the surface's center position $\mathbf{r}^{(s)}$, the bottom-left corner $\mathbf{r}^{(bl)}$, and the top-right corner $\mathbf{r}^{(tr)}$ in local coordinates w.r.t. $\mathbf{r}^{(c)}$.
5. The surface **s** serves as the near projection plane of **c**. The distance of **c**'s far projection plane is denoted by f. Then, we subsequently derive the projection matrix $\mathbf{m}^{(c)}$ as:

$$\mathbf{m}^{(c)} = \begin{pmatrix} \frac{2\mathbf{r}_z^{(s)}}{\mathbf{r}_x^{(tr)}-\mathbf{r}_x^{(bl)}} & 0 & \frac{\mathbf{r}_x^{(tr)}+\mathbf{r}_x^{(bl)}}{\mathbf{r}_x^{(tr)}-\mathbf{r}_x^{(bl)}} & 0 \\ 0 & \frac{2\mathbf{r}_z^{(s)}}{\mathbf{r}_y^{(tr)}-\mathbf{r}_y^{(bl)}} & \frac{\mathbf{r}_y^{(tr)}+\mathbf{r}_y^{(bl)}}{\mathbf{r}_y^{(tr)}-\mathbf{r}_y^{(bl)}} & 0 \\ 0 & 0 & \frac{-(f+\mathbf{r}_z^{(s)})}{f-\mathbf{r}_z^{(s)}} & \frac{-2f\mathbf{r}_z^{(s)}}{f-\mathbf{r}_z^{(s)}} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

D ADDITIONAL FIGURES FOR THE APPLICATION CASES

In this section, we present figures of three cases discussed in the paper to demonstrate how SpatialTouch is tailored to three distinct domains.

D.1 Molecular visualization

As shown in Fig. 10, we merge two distinct display spaces (AR HMD and Surface) to facilitate a concurrent visualization of different representations and abstractions. We developed a seamless interaction transition technique to enable users to move data across two spaces. It supports users to employ a familiar 2D pinch gesture on the touch surface to "pull" the visualization below the surface to the surface level and continue to use pinch gesture in 3D space to manipulate the visualization (shown in Fig. 11).

D.2 Medical anatomical visualization

SpatialTouch allows users to view 2D slices directly on the surface, while also observing stereoscopic renderings superimposed on the slice (Fig. 12).

Users can use the "lifting" feature to elevate the 3D volume visualization away from the 2D slice on the touch surface (Fig. 13).

In this way, users can obtain a clear view of the 2D slices and perform accurate interaction on the surface, such as precise annotation (Fig. 14 (a)) and distance measurements (Fig. 14 (b)). SpatialTouch supports users in marking features on the 3D visualization directly on the 2D slice (on the surface). This method is particularly useful when features appear in depth but are rendered on the surface.

D.3 Astronomical point cloud visualization

As shown in Fig. 15, users can view 3D point cloud visualizations in AR space to gain a comprehensive understanding of data density distribution and context, while performing precise data analysis, such as selection or annotation, on a 2D surface.

We developed two seamless spatial selection techniques for point cloud visualization. With BrushWYP (Fig. 16), users can brush over the string-like shape of 3D point cloud data in the 3D space and continue to brush the rest along the structure on the 2D surface.

With BrushLasso (Fig. 17), users can brush target points in mid-air and encircle points on the surface through a single, seamless input.

In addition, users are able to brush target points only in mid-air with MeTABrush [82] (Fig. 18), or draw a lasso around them on the surface with CloudLasso [78] (Fig. 19).

E THE CR VISUALIZATION LANDSCAPE

With SpatialTouch we developed and studied a new CR environment that merges a monoscopic 2D surface with a stereoscopic 3D space for data visualization and exploration. As we demonstrated, SpatialTouch facilitates comprehensive 3D spatial data analysis across various visual representations, scales, and data abstractions. This section extends beyond the design of SpatialTouch to discuss the broader landscape of CR solutions for visualization. We believe this discussion will be useful for future CR designers in creating CR environments for spatial data visualizations and more.
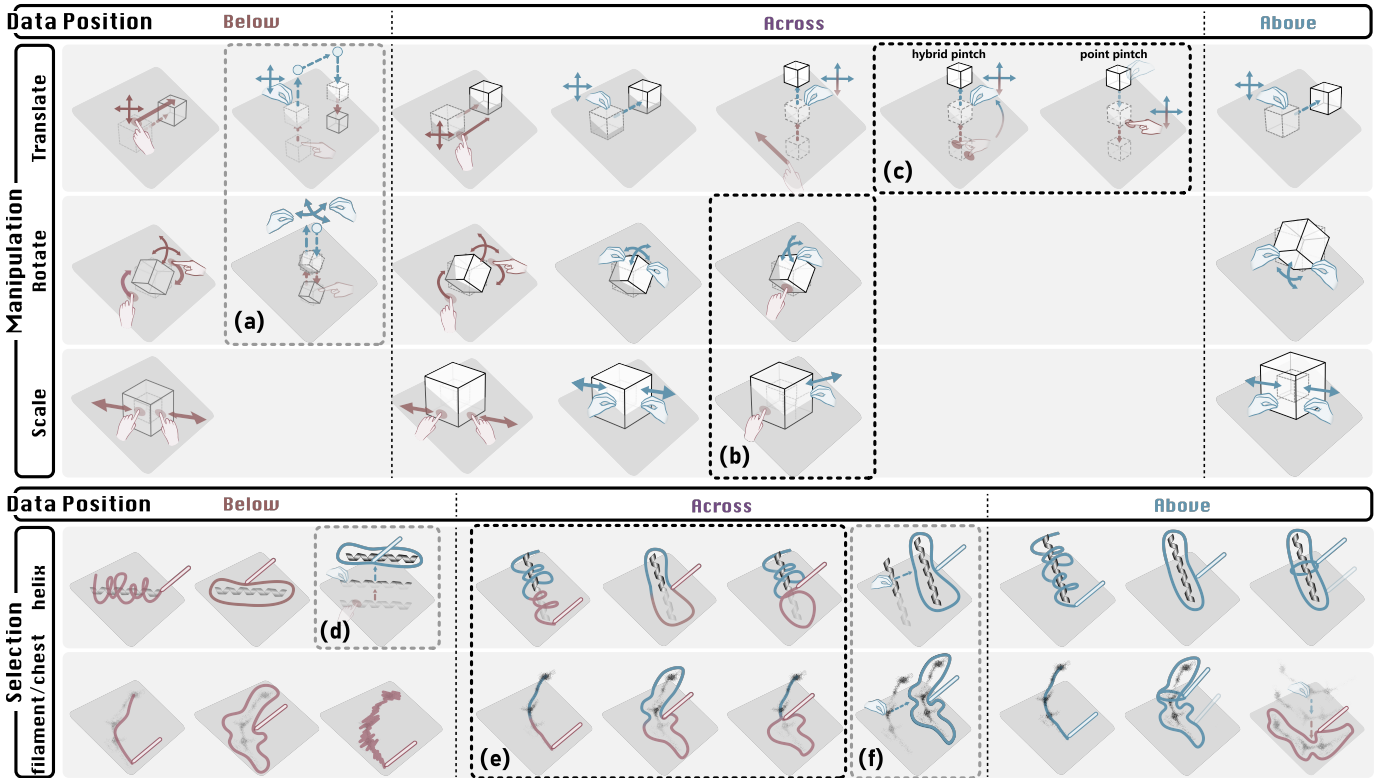
Fig. 9: Design space for interaction techniques for two visualization tasks: data manipulation and selection. Red: interactions on 2D surface; Blue: interactions in 3D space. **Below**, **Across**, and **Above**: positions of the target data/location. (a), (d), (f): move data above the surface for interaction. (b) (c) and (e): interaction transitions across both spaces.
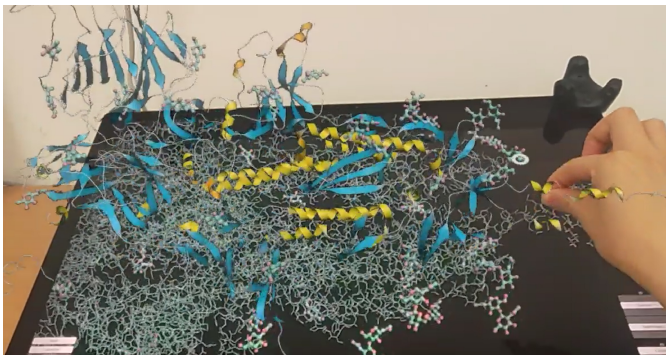

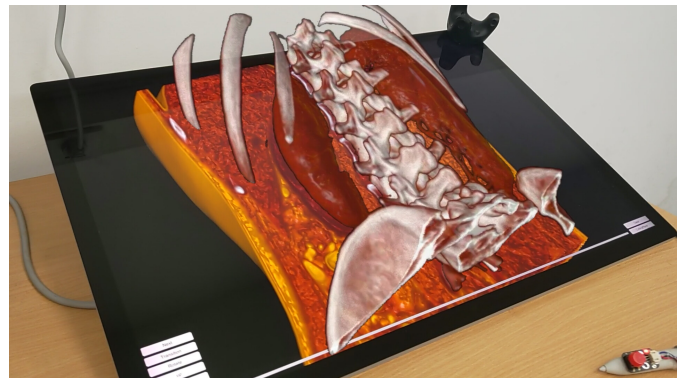Fig. 10: Protein data visualization [45] with the ribbon (shown in AR) and the licorice (shown on the surface) visual representations.

Fig. 12: 3D anatomic visualization on the surface and AR device with different transfer function settings.
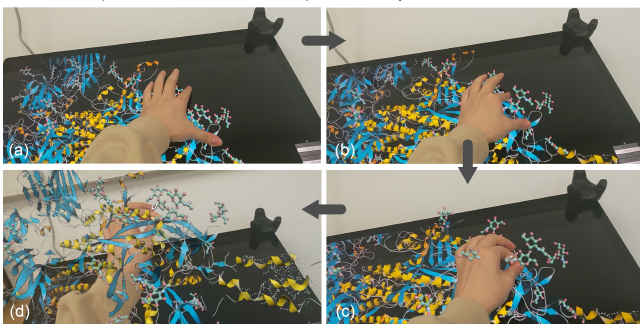
Fig. 11: The interaction transition technique to move the data visualization from 2D to 3D spaces. (a) start pinching on the surface to "pull" the visualization from 2D to 3D. (b) during the pinching, the visualization "moves towards 3D space" with the distance between the thumb and index finger decreasing. (c) the two fingers merge and the visualization is pulled up to near the level of the surface. (d) continue to move the visualization by pinching in the air.

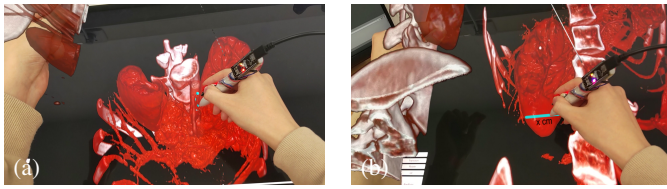Fig. 13: Lifting the stereo rendering with left pinch.

Fig. 14: Lifting the data and (a) annotating the features in depth, (b) measuring the width of the lung with pen on 2D surface.

Fig. 15: Cosmological N-body simulation visualization [64] across 2D (dark blue) and 3D environments (light blue).
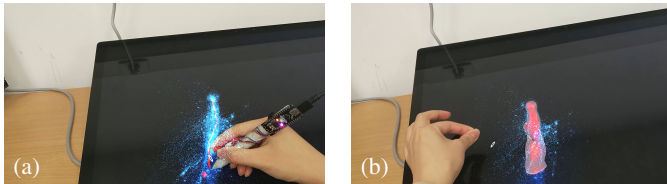

Fig. 16: BrushWYP: (1) direct brushing on the target points across two spaces, and (b) the selection result.
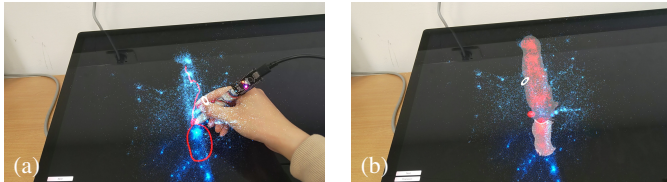

Fig. 17: BrushLasso: (1) brushing the target points in mid-air and drawing a lasso on the surface, (b) the selection results.
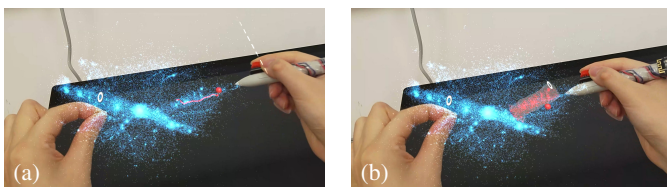

Fig. 18: MeTACAST: (1) brushing the target points directly in mid-air, (b) the selection results.

E.1 Motivation (why?)

When considering a CR environment for visualization, the first factor to consider is the underlying motivation as it shapes the entire design. A key advantage of CR is its ability to **augment visual representations or show additional information** for enhanced comprehension, data analysis support, or sense-making. The different CR designs we illustrated in Fig. 20 all exemplify this capability. Additional information may include extra data dimensions, contextual detail, or 3D represen-
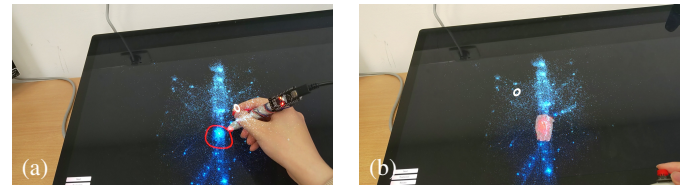

Fig. 19: CloudLasso: drawing a lasso around the target points on the surface, (b) the selection results.

tations. Examples are an AR plus traditional PC-based data analysis tool for 3D data [71] (Fig. 20(a)), techniques for navigating volume data through overview and detail views [15] (Fig. 20(d)), embedded AR visualizations for presenting multivariate data [51] (Fig. 20(f)), and SpatialTouch for presenting 3D visualization and multiple selected 2D slices (Fig. 7(b)). Furthermore, understanding the context of data and regions of interest allows users to focus on their interaction and may lead to more precise input, as they gain a clear understanding of the data's relative position and their interaction. In our elicitation study we observed that our participants' actions typically centered on regions of interest and crucial data points. CR environments enhance user engagement by offering **diverse interaction spaces** with additional information. Users can, e. g., accurately position a viewing window inside a medical structure on a touch table, while they observe both the overview and detail [15] (Fig. 20(d)). In SpatialTouch, users can precisely circle point data on a 2D surface, after assessing its spatial distribution in AR space (Fig. 20(h)). Another key motivation for employing CR is **collaborative data analysis**. CR empowers multiple collaborators to simultaneously visualize and discuss shared visual content, especially for spatial 3D data. Fig. 20(b) illustrates how users in a shared AR space can interact with a 3D visualization, each manipulating the data independently via a private tablet [59]. Conversely, Fig. 20(g, f) show how display content is shared among collaborators, while AR HMDs provide private views for individual analysis [19, 51]. These private views can be merged with the public one as required for sharing insights. A particularly innovative design is the use of a shared AR visualization, while participants have personal tablets directly inside the AR space with exact spatial alignment for individual analysis (Fig. 20(e)). Furthermore, while not extensively covered in previous work, we demonstrated in SpatialTouch the use of **varied visual abstractions** both Below and Above the surface to deepen users' comprehension of data features. The surface + AR designs in Fig. 20(e, h) are particularly effective for this goal because their unified depiction presents spatial data alongside embedded representations, facilitating an intuitive understanding of complex data.

E.2 Environment (where?) and dataset (what?)

As illustrated in Fig. 2, CR environments exploit various levels of virtuality to enhance user comprehension and facilitate task completion. Typically, they rely on monoscopic displays—mobile devices, surface interfaces, or desktop screens—, augmented with one or more stereoscopic AR/VR views (Fig. 20). The 2D displays offer multiple perspectives for data visualization, while 3D spaces are often dedicated to spatial data, resulting in a comprehensive data presentation.

For these configurations, it is crucial to create effective connections between contextual information and the visualized data attributes across multiple views as well as within spatial data. Various environments (Fig. 20(a–d)) employ approaches such as visual cues (e. g., color, highlighting), linked views, and animated transitions to achieve this coherence. Notably, designs Fig. 20(e–h) emphasize a strong connection and precise **spatial alignment** between AR and display content. Reipschläger and Dachselt [50] previously defined three levels of spatial proximity: AR content positioned in front of or behind the display, AR content arranged close to or at the edge of the display, and AR content rendered with no spatial relation to the display. Analyzing these setups, Fig. 20(f, g) use the display as a frame of reference for AR content placement. Designs in Fig. 20(e, h)—including SpatialTouch—leverage the spatial nature of 3D data that is rendered in AR space and
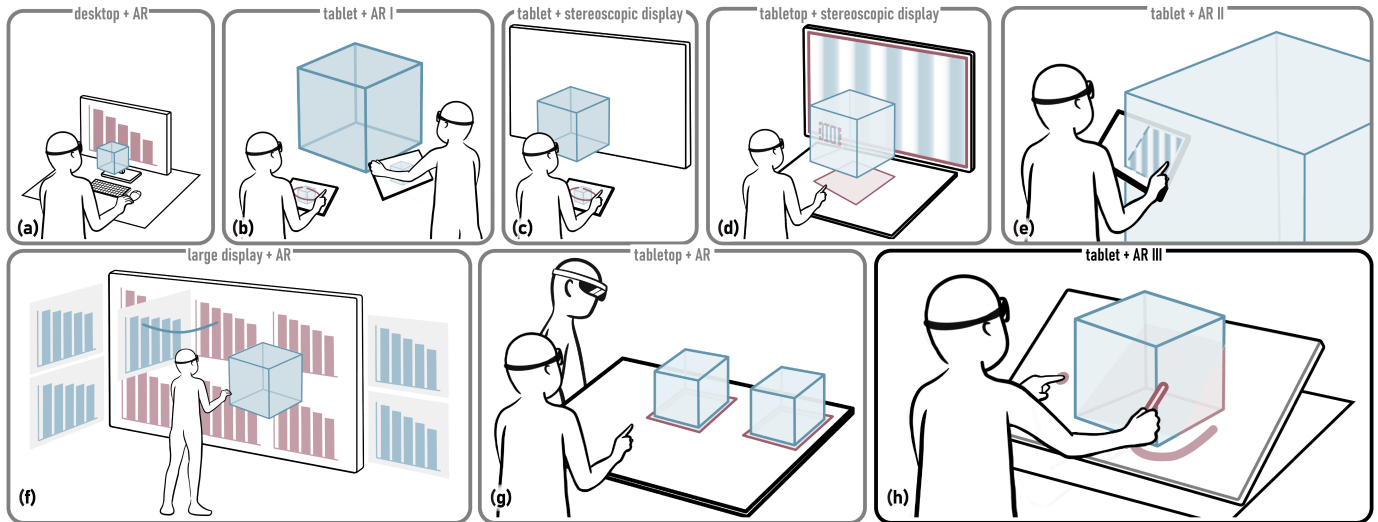
Fig. 20: Existing CR environments for visualization: (a) monitor + AR [29, 57, 71], (b) tablet + AR [58, 59], (c) tablet + Stereoscopic display [36], (d) tablet + stereoscopic display [15], (e) tablet + AR [3, 48, 59] , (f) large display + AR [51], (g) tablet + AR [19], (h) tablet + AR [50], SpatialTouch.

position the 2D display inside the AR visualization. This approach is particularly effective for demonstrating the relationship between contents in two different spaces, as demonstrated in spatial selection tasks [59] and the visualization of medical volumes using cutting planes in SpatialTouch.

F   LIMITATIONS OF THE SPECIFIC HARDWARE SETTINGS

In this section, we discuss the limitations caused by the specific hardware settings (Surface Studio and Hololens HMD) of SpatialTouch. First, SpatialTouch requires high precision in the registration between the 2D surface and the 3D space to ensure accurate alignment between the two views. In our prototype, the HoloLens and Surface Studio run the program separately as two clients, with visualization and interaction states synchronized via the server. Similar to past work [50], after starting the application on Hololens, we need to manually align the two coordinate systems. Thus, although the view synchronization algorithm presented in Sec. 3 works accurately, as shown in the supplemental demo, it was time-consuming and difficult to achieve a high level of precision in the coordinate system registration. For future development, we recommend the researchers leverage automatic registration techniques with high precision or develop targeted CR display ecology with a unique coordinate system. Second, the limited field of View (FOV) and the restricted gesture recognition area of the HoloLens significantly hinder the user's immersive experience. The narrow FOV confines the visualization to a small window in the user's vision, which breaks the sense of immersion as users are constantly reminded of the boundaries of the display. The constrained gesture recognition area means that users need to perform interactions within a small space in front of the body. It increases the cognitive load of users as they need to be conscious of keeping their hands within the detectable zone. We recommend that researchers leverage HMDs or glasses with large FOVs and extensive gesture recognition areas. For the FOV, current mainstream video see-through HMDs, such as the Oculus Quest 3, can sample the real world and map the results onto display devices with a large FOV. However, the video quality of VST HMDs is not yet sufficient for users to clearly observe the data visualizations on 2D surfaces in the real world. We believe that in the future, the image quality of VST HMDs will continue to improve, and issues such as video stream latency and image distortion will be resolved. For gesture recognition, external sensors such as Leap Motion can be introduced to facilitate gesture recognition and tracking.

FIGURES LICENSE/COPYRIGHT