AuxiScope: Handheld Augmented Reality Tablet as an Auxiliary Display for Large-Scale Display Systems

Matthew S. Castellana, Chahat Kalsi, Yoonsang Kim, Saeed Boorboor, Arie E. Kaufman, *Fellow, IEEE* *

Center for Visual Computing
Stony Brook University, Stony Brook, NY, USA

2025 IEEE International Symposium on Mixed and Augmented Reality (ISMAR) | 979-8-3315-8761-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ISMAR67309.2025.00169

Figure 1: Two collaborators explore urban data using AuxiScope, which overlays AR visualizations on handheld tablets aligned with data displayed on a large wall display, enabling personalized, multi-user analysis without visual interference.

Abstract

We present AuxiScope, a novel AR-based system designed to enhance personalized data exploration on large wall displays (LWDs) by integrating handheld tablets as auxiliary visualization interfaces. While LWDs offer expanded visual real estate and intuitive embodied interaction, they pose challenges related to effective interfaces for data exploration and analysis, specifically in multi-user settings. AuxiScope addresses these by overlaying supplementary visualizations onto corresponding LWD content, enabling individualized exploration without interference with the visual data displayed on the LWD. To achieve this, we have designed a geometric alignment pipeline that synchronizes the auxiliary visualizations atop the virtual scene. Specifically, by leveraging AR technology, AuxiScope resolves the tablet physical localization, viewpoint computation, and user interaction translation into the virtual space. Subsequently, based on a client-server architecture, it employs remote rendering and delegates computational tasks to the LWD compute nodes in order to minimize memory load on portable devices. We demonstrate the potential of AuxiScope through multiple AR-based interaction techniques across information and scientific visualization scenarios, for both 2D and 3D contexts.

Index Terms: Augmented Reality, Immersive Visualization, Large Wall Display, Handheld, Tablet, Multi-user

*Authors Castellana, Kalsi, Kim, Boorboor, and Kaufman emails: {matcastellan, ckalsi, yoonsakim, sboorboor, ari}@cs.stonybrook.edu

1 Introduction

Large wall displays (LWDs) have emerged as transformative tools for data exploration [2], offering expansive visual space, high-resolution detail, and intuitive embodied interaction [1, 21, 51, 57]. Unlike traditional desktop settings that rely on virtual navigation within a restricted field-of-view (FoV), LWDs enable users to naturally shift their perspectives through physical navigation, maintaining situational context while examining fine-grained details, and subsequently revealing new insights without disrupting cognitive flow [5, 51]. The scale of such displays not only enhances individual analytical processes by providing "spaces to think" [1, 38], but also fosters co-located collaboration, where multiple users can engage in collective sensemaking.

Realizing the full potential of LWDs for data exploration requires interaction techniques that consider the physical characteristics of the LWD environment, the user's spatial relationship to the data, and the specific context of the interactions themselves. Particularly in collaborative settings, data exploration encompasses a wide range of tasks that often require distinct interaction styles. For instance, consider a scenario where two experts, an engineer and an emergency manager, seek to analyze the impacts of a flooding simulation on an urban scene, displayed on an LWD, as illustrated in Fig. 1. The engineer may need to visualize critical infrastructure information, while the emergency manager simultaneously focuses on identifying evacuation sites based on flooding probabilities. Although displaying all relevant information on the LWD is advantageous, user interactions that modify the primary visualization can inadvertently disrupt the analytical workflow of other analysts sharing the space. Furthermore, for either analyst, toggling between the base visualization and their specific overlays can create context switching, which can hinder information retention [64].

This motivated us to develop AuxiScope, a system that lever-

2473-0726/25/$31.00 ©2025 IEEE
DOI 10.1109/ISMAR67309.2025.00169

ages augmented reality (AR) on handheld tablets to serve as auxiliary displays for interactively overlaying supplementary visualizations onto visual data presented on an LWD. Utilizing ubiquitous devices has been shown to enhance 3D interaction and exploration [4, 8, 26, 39] in immersive settings [10]. Moreover, advancements in AR technologies have enabled their integration with LWDs, where the physical wall display serves as the primary frame of reference, and the AR device adds personalized information, either on top of or around the LWD content, in the form of additional visualizations and highlights [28, 55, 63]. AuxiScope introduces a novel paradigm by processing user selection and generating AR-based visualizations within the virtual reality (VR) scene itself. By geometrically aligning the supplementary visualizations with the projected virtual scene and integrating them into the handheld device's video see-through view, it gives the perception that the new content is part of the LWD original rendering rather than a simple overlay on the physical LWD. This can be noticed in the right aux shown in Fig. 1, where buildings from the LWD view occlude parts of the flood-level visualization, giving a perception that the visualization is embedded within the virtual scene. Moreover, the AuxiScope system is modular, designed to support multiple visualization types that can be triggered in multi-user settings without visual interference, for any given LWD layout configuration.

Given that LWD environments typically operate within multi-node cluster architectures, AuxiScope is designed to function as an additional client within the LWD client-server framework, wherein the handheld tablet (hereafter referred to as *aux*) serves as an auxiliary display. To achieve geometric alignment of AR visualizations, the system first resolves the physical localization of the aux relative to the LWD. Subsequently, the server computes the appropriate viewpoint transformations and maps the user's selection from the aux to its corresponding position within the virtual space. To ensure computational efficiency and mitigate memory overhead on resource-constrained tablets, AuxiScope employs a remote rendering approach, delegating visualization processing tasks to the LWD cluster node while transmitting only the viewpoint-adjusted rendered images back to the aux. In leveraging this architecture, AuxiScope enables real-time augmentation of visual data without compromising the performance of the handheld device.

We demonstrate AuxiScope using augmentation techniques applied to use cases in information and scientific visualizations, encompassing diverse examples in 2D and 3D representations. Specifically, we explore interaction designs where AuxiScope can be used for (1) direct volume rendering visualization, where the aux locally updates transfer functions and acts as a virtual cutting plane, (2) enhanced navigation, where the aux acts as a *portal* that reveals hidden information by placing a secondary virtual camera in a 3D scene, and (3) augmentation of supplementary visualizations for visual analytics. We summarize our contributions as follows:

- We introduce AuxiScope, an auxiliary display paradigm that augments personalized 2D/3D supplementary visualizations augmented for virtual scene referents projected onto LWDs.
- We develop a custom post-rendered 3D warp technique for distributed, asynchronous video feeds in multi-node LWD visualization cluster setups, ensuring optimal registration, reducing latency, and increasing frame rate.
- We evaluate our system with a quantitative user study and qualitative feedback obtained through a series of information and scientific visualization-based demo applications.

2 Related Works

AuxiScope is informed by a rich body of research at the intersection of immersive analytics, hybrid user interfaces, and interaction techniques for LWDs.

2.1 Integrating AR and Displays

AR head-mounted displays (HMDs) and handheld tablets have enabled hybrid environments that augment traditional displays with virtual content, enhancing spatial interaction and providing immersive, context-aware experiences. Initial works have expanded LWD viewing areas through spatial augmentation. Nishimoto and Johnson [48] have projected views above or below the display environment, while James et al. [28] have added extra virtual walls.

Other systems integrate content directly onto or adjacent to physical displays to complement on-screen visuals. MARVIS [36] has presented a conceptual framework that integrates mobile devices with HMDs, enabling cross-device interactions such as linking and brushing to support flexible and coordinated exploration tasks. Specific to LWDs, Submerse [13] provides tablet-based interactivity for flooding simulations. Reipschläger et al. [55] have contributed a design space analysis of the intersection of LWDs, AR, and information visualization, specifically addressing key visualization challenges for aligning content with LWDs. AuxiScope not only overlays supplementary content but also supports 3D spatial interaction and visualization, tightly integrated with the scene rendered on the displays. It is also display-layout agnostic, operating across a wide range of physical configurations and orientations.

Typically, 2D screens and 3D AR content are treated as distinct modalities, motivating interest in seamless transitions between them. DesignAR [54] has introduced a workstation that combines stereoscopic AR with pen and touch 2D input, supporting operations to transform 2D sketches into 3D AR objects. Wu et al. [71] have developed a gestural interaction for AR that pulls 2D content into a 3D space. Rau et al. [53] have developed additional gesture- and input-based interaction techniques for transitioning 3D objects between the desktop and AR, and vice versa. SpatialTouch [72] has presented a system that merges a 2D interactive surface with stereoscopic AR, aligning perspectives across modalities.

Alternatively, AR can be used to generate an altered view of a display, a technique known as the magic lens. Originally developed for 2D [11], it has since been adapted for 3D contexts [69] and tangible displays [27]. In their survey, Tominski et al. [64] define lenses as lightweight tools that temporarily modify part of a visualization for local analysis, and present a taxonomy across interaction modality, display setting, and effect extent. We discuss works most relevant to AuxiScope that address tangible/touch input, multi-display setups, and separate views. Spindler et al. [61] have introduced passive paper-based lenses tracked above tabletop displays, allowing users to view altered representations such as semantic zoom or temporal filters. They later extended the system with a richer interaction vocabulary, mapping it to common information visualization tasks [62], and a further extension utilized high-resolution tablets [60]. Kim and Elmqvist [32] have developed a system for composable visual queries via tracked physical sheets placed over tabletop displays. In all these systems, the AR effect is spatially bound to the lens, acting as a dynamic filter.

AuxiScope introduces display-integrated AR that intercepts the rendering pipeline to apply user-specific 2D/3D modifications directly to the virtual scene. It streams personalized visualizations to each aux, making it appear as if the display itself rendered the augmentations, while the actual display content remains unchanged. It ensures that all users perceive temporally synchronized, perspective-correct overlays across tiled or multi-display walls, including multi-planar configurations. To our knowledge, no previous display-based magic lens approach achieves this combination of per-user personalization, per-plane integration, and support for shared, multi-screen environments.

2.2 Interaction Modalities for Large Interactive Displays

LWD and CAVE facilities offer expansive visual environments that support collaborative activities, where users often stand or

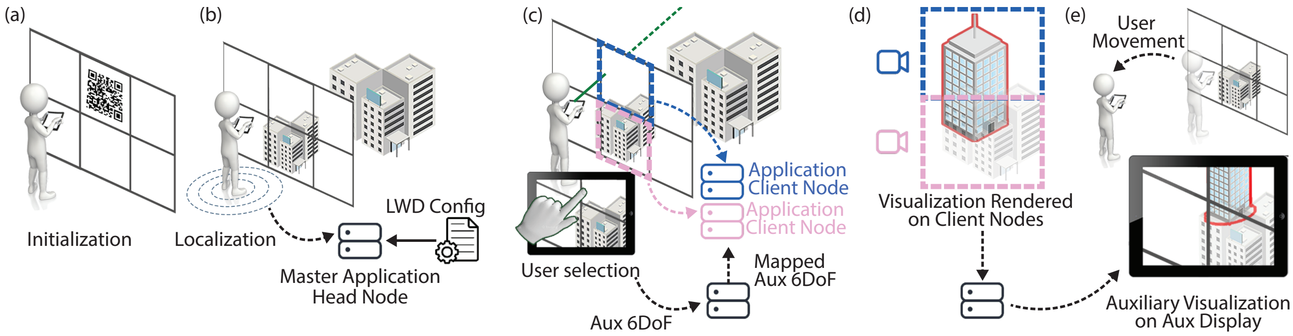

Figure 2: An illustration of the AuxiScope system. (a) Initialization begins by aligning the aux device with an image marker on the LWD. (b) The master instance maps the aux AR coordinate system to the LWD physical space using configuration details. The aux then transmits its 6DoF pose to the master. (c) On selection, the master forwards the aux 6DoF and selection ray to the target application on client nodes. (d) Target applications render updates in the background using a virtual camera to avoid rendering on the LWD. (e) Rendered updates are sent to the master and then to the aux. If delayed by processing or network latency, the visualization is warped to compensate for minor user motion.

walk during their interactions. This calls for interaction techniques beyond the traditional keyboard and mouse [5]. To address this, researchers have employed a range of embodied interaction techniques, including user movements [51], gestures [40, 42], speech [37], direct touch [52, 56], and speech+touch [20, 35].

As touchscreens have become ubiquitous, researchers have investigated integrating them with LWDs [16, 29, 40, 45]. Miguel et al. [46] have studied object selection and manipulation using touchscreens and physical buttons. Olwal and Feiner [50] compared touchscreen- and button-based zooming techniques on a mobile device serving as a viewport to a larger display. Bergé et al. [7] compared mid-air device and hand movements, and touchscreen interactions for input.

For a comprehensive review on specialized interfaces for 3D visualization, we refer readers to the survey by Besançon et al. [10], which categorizes interaction tasks into view and object manipulation, widget placement and control, and 3D data selection and annotation. We focus here on works that implement these tasks on hybrid touch/tangible tablets. López et al. [39] have introduced a system that decouples touch from stereoscopic display to enable precise 3D navigation and address reference frame mismatches. Besançon et al. [8] have designed a position-aware tablet that supports both tactile gestures and 6DoF tangible input for tasks such as slicing and seed placement. Subsequently, the authors developed Tangible Brush [9], a technique for precise, data-independent volume spatial selection. Sereno et al. [58] have adapted this approach by integrating a tablet with an HMD to study alternative spatial mappings to reduce cognitive load and improve control.

Specifically for LWDs, Bornik et al. [14] have developed a hybrid tablet user interface for volumetric medical data that enables control through more precise pen-based input or coarse physical device movement. Song et al. [59] have investigated multi-touch and tilt sensing to enable tangible manipulation of volume slicing planes through direct contact with the display as well as remotely. INSPECT [31] enables 6DoF object manipulation on LWDs using a rotation-tracked smartphone and indirect touch, allowing users to interact without diverting gaze from the screen.

Many existing systems built around hybrid touch/tangible interactions utilize the device display to show a customized AR view. However, these systems have not sufficiently explored maintaining personalization in a multi-user context [64], especially within complex virtual environments. When adapted to multi-user use, they often overlook the intricacies of maintaining shared scene integrity, supporting individualized perspectives, and preventing interference between simultaneous inputs. These challenges are amplified when the scene is computationally intensive and centrally rendered, as personalized views must coexist and remain synchronized within the same environment. AuxiScope enables co-located, parallel interaction with a shared scene through user-specific touch and tangible interfaces, grounded in a stable, shared reference display. This configuration supports mutual awareness and independent exploration without coupling inputs, facilitating multi-user interaction in a way not addressed by prior systems of this kind.

2.3 Post-Rendering 3D Warping

In XR, graphics for display are typically rendered from a camera that tracks the pose of a reference device. However, performance limitations can introduce issues; the user's head may have moved by the time the frame has displayed, resulting in a misalignment between the rendered image and current viewpoint. This discrepancy, known as *motion-to-photon delay* or *end-to-end latency*, can affect the user experience significantly. To address this, techniques have been developed to adjust the image pre-display for better alignment.

Early attempts at reducing motion-to-photon delay used simpler techniques, such as translating the image in 2D before display [15, 44], or a post-rendered 3D warp using the Z-buffer [41]. The term timewarp was later popularized by van Waveren [68], who introduced asynchronous timewarp for Oculus headsets. Recent Meta XR devices use positional timewarp, accounting for both head position and rotation. Timewarp techniques have also been incorporated into AR applications [33], using information from a faster RGB camera to warp and update images received from a slower IR camera. We present a two-stage timewarp mechanism to register AR graphics rendered on a device using a known LWD configuration and caching of timestamped camera view-projection matrices.

3 The AuxiScope System

We envision a scenario where one or more analysts navigate an LWD facility, each using a handheld tablet. Beaudouin-Lafon [6] has demonstrated that data consolidation and collaboration is more effective when LWDs are used in conjunction with personal devices, such as laptops and tablets, as these devices provide focused information that complements the shared LWD content. Based on this insight, we have developed a system that allows analysts to interact with and access supplementary visualizations for regions of interest (ROIs) displayed on the LWD using tablets. In this context, the tablet serves as an auxiliary display layer.

3.1 System Overview

AuxiScope is developed on a replicated execution model [22]. That is, for a multi-node, multi-GPU visualization cluster setup, instances of a target application are launched individually on the cluster client nodes (often referred to as rendering nodes). A master instance on the head node synchronizes interaction, tracking, and communications across all instances. Details outlining LWD system components, such as physical dimensions, display layout, cam-

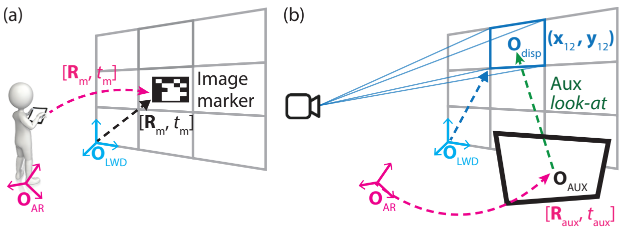

Figure 3: (a) Image marker used to align the aux AR coordinate system (O_{AR}) with the LWD physical space (O_{LWD}) at runtime. (b) This alignment allows AuxiScope to localize the aux relative to each LWD application instance.

era viewport for each target application instance, and cluster network details, are provided as input to the master instance.

An illustration of the AuxiScope system pipeline is shown in Fig. 2. Initialization begins with the user scanning an image-based marker displayed on the LWD using the aux (Fig. 2(a)). This allows the master instance to establish an alignment between the aux AR coordinate system and the LWD facility physical coordinate system (Sec. 3.2). Following initialization, the aux continuously transmits its position and orientation to the master instance, which localizes the device within the physical space (Fig. 2(b)).

In immersive display facilities, large-scale dataset rendering is typically distributed across multi-node clusters [12, 23, 43, 51]. Given the limited memory and computational capacity of tablets, AuxiScope leverages cluster resources to maintain a responsive, real-time user interface. When a user interacts with the LWD via the aux, the master instance transmits the localized aux six degree-of-freedom (6DoF) transformation and interaction metadata to the client nodes (Fig. 2(c)). Each relevant target instance then generates a corresponding visualization update (Sec. 3.3) and returns the rendered output to the master instance (Figs. 2(d)–(e)). The master instance composites the received outputs into a mosaiced frame and streams it back to the aux. The aux subsequently displays the visualization through its video see-through (VST) view, geometrically and spatially aligned with the underlying LWD content. To preserve accurate registration during user motion, AuxiScope applies lightweight post-rendering view adjustments (Sec. 2.3).

Our system design abstracts the underlying development of the visualization application. AuxiScope supports basic interaction via tap and scroll gestures. Developers may extend this interface with custom UI elements as needed. Applications need only handle interaction triggers and respond to camera pose updates from the master instance. In Sec. 4, we demonstrate use cases highlighting the utility and functionalities of AuxiScope.

3.2 Coordinate System Mapping

AR frameworks, such as ARKit [3], ARCore [25], and ARFoundation [66], use a device's cameras and motion sensors to track its position and orientation. Typically, these frameworks establish a local coordinate system at runtime, with the origin at the device's initial pose. Therefore, a robust mapping between the aux AR coordinate system and the LWD physical space is needed for spatial interaction and alignment of augmented visualizations.

To achieve this, an image marker of known dimensions, position, and rotation, as defined in the input config, is initially displayed on the LWD. On scanning and registering the marker using the aux VST, a coordinate system transformation, $T_{\mathscr{F}_{\mathrm{AR}} \to \mathscr{F}_{\mathrm{LWD}}}$, that maps the aux AR coordinate frame, $\mathscr{F}_{\mathrm{AR}}$, to the LWD physical space, $\mathscr{F}_{\mathrm{LWD}}$, is computed. Specifically, a transformation, $T_{\mathrm{AR}\to\mathrm{m}}$, is determined from the marker estimated pose relative to the AR origin, O_{AR}, and the marker pose with respect to the LWD origin, O_{LWD}, is specified in a configuration file, that defines $T_{\mathrm{LWD}\to\mathrm{m}}$. The mapping $T_{\mathscr{F}_{\mathrm{AR}} \to \mathscr{F}_{\mathrm{LWD}}}$ is then obtained as:

$$T_{\mathscr{F}_{\mathrm{AR}} \to \mathscr{F}_{\mathrm{LWD}}} = T_{\mathrm{m}\to\mathrm{LWD}} . T_{\mathrm{AR}\to\mathrm{m}} \tag{1}$$

Figure 4: (a) Image marker, highlighted by the red arrow, used for initialization. (b) Misaligned mapping of the AR and LWD coordinate spaces, visualized using a reference plane. (c) The mapping can be fine-tuned using widgets, highlighted by the arrows.

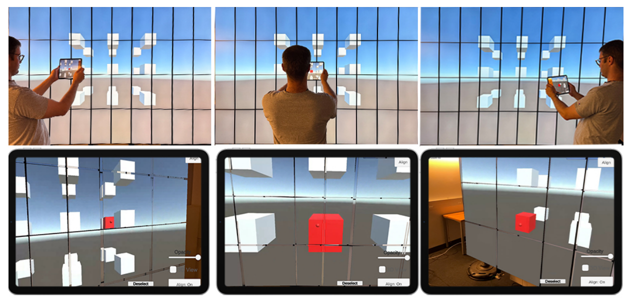

Figure 5: AuxiScope 3D interaction for selecting objects in the virtual scene from different angles.

This mapping is illustrated in Fig. 3. In our implementation, $T_{\mathrm{AR}\to\mathrm{m}}$ is calculated on the aux and transmitted to the master instance. The master instance calculates $T_{\mathrm{LWD}\to\mathrm{m}}$ and the mapping $T_{\mathscr{F}_{\mathrm{AR}} \to \mathscr{F}_{\mathrm{LWD}}}$, used for subsequent interactions and visualization updates between the aux and the target application.

AR frameworks detect image markers by matching visual patterns in the camera feed and estimating their 3D pose. However, accuracy may degrade under poor lighting, motion blur, or low resolution. To correct misalignment, users can interactively adjust the 3D rotation and translation via provided widgets, refining the mapping between the LWD and the aux view (Fig. 4).

3.3 Aligning and Rendering Visualizations

The referents of the augmented supplementary visualizations in AuxiScope are situated within a 3D virtual scene, projected onto planar LWD screens. As a result, overlaying and rendering AR elements are not dependent on the 3D position of the referent, but rather on the 2D projection of the virtual scene onto the LWD. This difference in viewpoint can be seen in Fig. 1, where two users view the LWD visual data from different positions and orientations.

Once the coordinate system is mapped, as the user navigates through the facility, the aux records its localization transformation, $T_{\mathrm{AR}\to\mathrm{aux}}$, at each frame, and transmits it to the master instance. The cached transformation enables alignment correction for latency compensation, discussed in Sec. 2.3. The master subsequently computes a display-to-aux transformation, $T_{\mathrm{disp}\to\mathrm{aux}}$, based on the spatial configuration of the corresponding target application within the LWD layout. It then distributes this transformation to the relevant target instances across all client nodes, illustrated in Fig. 3(b). For a given target application, the transformation is defined as:

$$T_{\mathrm{disp}\to\mathrm{aux}} = \left(T_{\mathrm{LWD}\to\mathrm{disp}}\right)^{-1} \cdot \left(T_{\mathscr{F}_{\mathrm{AR}} \to \mathscr{F}_{\mathrm{LWD}}}\right)^{-1} \cdot T_{\mathrm{AR}\to\mathrm{aux}} \tag{2}$$

where $T_{\mathrm{LWD}\to\mathrm{disp}}$ is calculated using the config information. This transformation enables both the accurate mapping of aux viewing rays into the LWD-projected scene for interaction, and the geometric alignment of remote visualizations rendered on the aux.

For selection and viewing, an aux ray, $\vec{\mathbf{v}}_{\mathrm{aux}}$, is defined as a ray from the center of the aux device, $\mathbf{o}_{\mathrm{aux}}$, in the direction of the camera *look-at*, illustrated in Fig. 3(b). The extrapolated ray into the

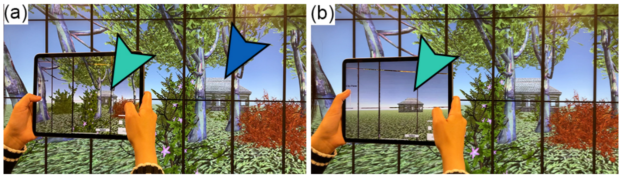

Figure 6: (a) A user selects a virtual object in the LWD scene from the aux, highlighted by the arrows. (b) The aux visualization update hides all other virtual objects in the scene.

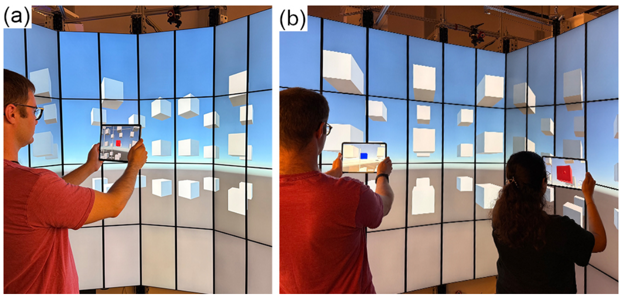

Figure 7: Using AuxiScope on different LWD configurations, such as (a) curved and (b) 2-wall layouts.

target instance scene is therefore a ray, $\vec{\mathbf{v}}_{\text{disp}}$:

$$\mathbf{o}_{\text{disp}} = T_{\text{disp}\to\text{aux}} \cdot \mathbf{o}_{\text{aux}} \quad , \quad \mathbf{l}_{\text{disp}} = T_{\text{disp}\to\text{aux}} \cdot \left[\mathbf{o}_{\text{aux}} + \vec{\mathbf{v}}_{\text{aux}}\right] \qquad (3)$$
$$\vec{\mathbf{v}}_{\text{disp}} = \mathbf{l}_{\text{disp}} - \mathbf{o}_{\text{disp}}$$

When a target instance receives an interaction trigger, such as a selection to overlay 2D information, updating the transfer function for volume rendering, or other examples demonstrated in Sec. 4, it renders visualization updates, as a background process, on a virtual texture reserved explicitly for aux visual elements. This avoids changing the visualizations displayed on the LWD. In a multi-user setting, the target application reserves a separate texture for each unique aux user, allowing independent visualizations per user. Once all relevant target instances transmit their reserved textures to the master instance, these textures are composited into a single texture. The master then applies the inverse transformation, $T_{\text{disp}\to\text{aux}}$, and transmits the result back to the aux.

Fig. 5 demonstrates the selection and projection process, where a cube in the LWD scene is selected by casting a ray from the aux into the 3D scene. The insets at the bottom show how changing the aux pose dynamically updates the 3D selections for the 2D projected cubes on the LWD. The selected cube (rendered in red) is aligned in the aux VST view, maintaining geometric and spatial consistency with its position in the LWD coordinate space. In more complex scenes containing numerous virtual objects, Fig. 6 demonstrates selective interaction, where for a scene shown in Fig. 6(a), a user taps on a specific building (indicated by arrows), prompting the system to *hide* all other virtual elements. The result in Fig. 6(b) shows a geometrically aligned, merged rendering of the selected object overlaid on the aux VST feed. Furthermore, Fig. 7 showcases AuxiScope's ability to support diverse LWD configurations, including curved and two-wall *L*-shaped layouts. The coordinate mapping and transformation framework ensures consistent spatial alignment across these different display topologies.

3.4 Re-alignment of Aux Visualization due to Latency

Although remote rendering offers clear benefits, it introduces latency from round-trip transmissions. Updates in the aux pose between its transmission and the receipt of the remotely rendered frame can induce misalignments with respect to the VST view. These performance issues are influenced not only by the network throughput and connectivity of the devices and facility but also by the computational capacities of the involved hardware, which may be difficult to upgrade in existing LWD facilities. AuxiScope addresses this with a timewarp strategy.

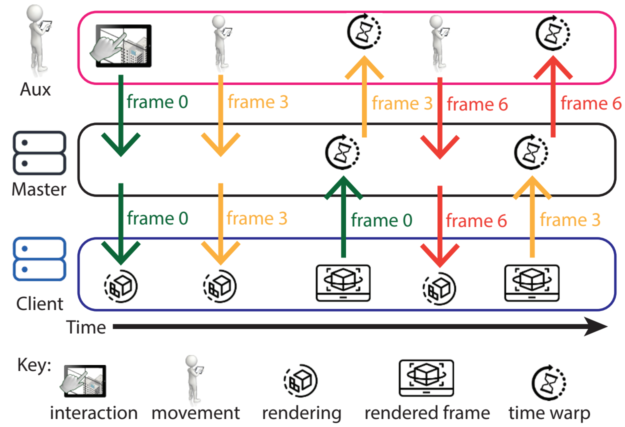

Figure 8: An illustration of the two-stage process of timewarping a remote rendered texture for projection on the aux device.

Timewarp techniques, discussed in Sec. 2.3, fundamentally rely on per-pixel 3D world positions to accurately compute corresponding adjusted locations. Our solution employs a two-stage timewarp mechanism, using the pre-existing knowledge of the LWD configuration and the additional systematic storage of a timestamped camera view-projection matrix (VP).

The first stage is performed on the master instance upon receiving rendered textures from the target instances. Due to variability in rendering times and network latencies, frames from different targets may arrive asynchronously. Therefore, we timestamp each rendered frame corresponding to the camera VP used during rendering. This enables per-target timewarp corrections, where each texture is individually reprojected based on its associated VP. As the master instance does not have immediate access to the aux current VP, it applies the most recently cached VP, VP_{last}, to compute the reprojection transformations and applies it to the composited texture. This composite texture is then re-encoded with a unique identifier linked to VP_{last} and transmitted back to the aux.

The second timewarp stage is executed on the aux upon receiving the composite texture. At this stage, a final reprojection is performed, using VP_{last} as the source camera view and the current aux camera view, VP_{aux}. The resulting image, now temporally synchronized, is then rendered to the aux VST view. The full communication and reprojection sequence is illustrated in Fig. 8.

Each timewarp operation begins with a source image I_{prev} and corresponding matrices VP_{prev}. The current camera VP initiates a custom render pass in which all scene objects are culled except for the rectangular proxy meshes representing the LWD. During rasterization, depth textures are used to construct an intermediate texture, WS, which stores world-space coordinates for each screen-space pixel (x,y). These 3D coordinates are encoded into the RGB channels of WS, while the alpha channel is used to denote validity.

Using WS, each pixel in screen space can be mapped to a corresponding 3D coordinate, $WS[(x,y)]$. To identify the location in the source image I_{prev} that corresponds to this 3D point, we project it into screen space using the previous VP matrix:

$$x_{prev}, y_{prev} = SS(VP_{prev} \cdot WS[x_{new}, y_{new}]) \qquad (4)$$

Here, SS indicates a transformation from clip space to screen space. The color value C is then sampled from I_{prev} at coordinates $(x_{\text{prev}}, y_{\text{prev}})$, using bilinear interpolation to account for subpixel positions. Finally, the destination image I is synthesized by assigning

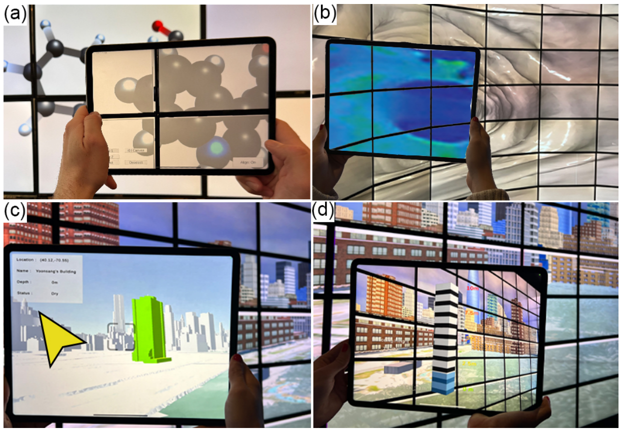

Figure 9: AuxiScope interaction examples. (a) Protein visualization: surface representation (aux) overlaid on ball-and-stick (LWD). (b) Immersive virtual colonoscopy: electronic biopsy (aux) overlaid on the colon (LWD). Urban visualization: (c) Occlusion removal of buildings (aux) for an urban scene (LWD). (d) 3D widget for measuring water-level (aux) for a coastal flooding scene (LWD).

pixel values according to:

$$I[(x,y)] = \begin{cases} C, & \text{if } WS[x_{new}, y_{new}] \text{ valid} \\ (0,0,0,0), & \text{otherwise} \end{cases} \tag{5}$$

4 Application Scenarios

We demonstrate AuxiScope for two interaction interfaces: (1) visualization overlays, with use cases in information and scientific visualization, and (2) 3D immersive navigation. We also conducted a user study to compare AuxiScope with a baseline shared-display condition, evaluating task completion time and accuracy. We refer the reader to our supplementary video for a demonstration of the presented interactions and applications.

We implemented AuxiScope using Unity 2022.3.30f1 with a modified UniCAVE plugin [65]. The system consists of two complementary projects: one designed for cluster deployment and another for the aux device. Development is done on the cluster project unless aux functionality also needs modification.Our prototype uses an iPad Pro with AR Foundation [66], and device communication is handled via Unity WebRTC [67]. We tested various LWD setups, each with three client nodes (2x Intel Xeon Ice Lake CPUs, 192GB RAM, and 8x RTX A5000 GPUs) and one master node (Dell Precision 5860, Xeon W-2495X, 128GB RAM, dual RTX 5000 GPUs).

4.1 AuxiScope Interaction Designs

One widely used AR interaction paradigm is overlaying supplementary visualizations to enhance understanding of primary content. Building on this, we enable spatially anchored 2D and 3D overlays on the LWD, delivering contextual information, annotations, or alternative views. Consider an example of visualizing a complex protein structure on an LWD, as shown in Fig. 9(a). Users can toggle between different representations of the scientific data on the aux, projected atop the reference visualization shown on the LWD. In Fig. 9(b), we demonstrate using the aux for visualizing electronic biopsy, a different medical visualization modality, for an immersive virtual colonoscopy [47] rendering on a curved LWD. In this example, the electronic biopsy view is a volume rendering performed for the visible colon surface, demonstrating that AuxiScope is capable of performing spatial alignment, rather than just overlaying views.

For analyzing large and complex datasets, focus+context visualization techniques systematically combine global overviews with detailed, localized views. In desktop settings, this is typically achieved through GUIs and frequent keyboard–mouse interactions. AuxiScope introduces a novel focus+context interaction paradigm in which the global dataset is visualized on the LWD, while spatially localized and context-sensitive information is presented on the aux. We demonstrate this capability using an urban visualization application, as illustrated in Figs. 9(c)-(d). In Fig. 9(c), the aux enables intuitive navigation within a dense 3D urban environment. Users can select individual buildings to reveal associated metadata, such as demographic attributes (indicated by the arrow), while automatically decluttering the view by removing occluding structures. Additionally, as shown in Fig. 9(d), interactive widgets such as a dipstick tool allow users to probe the scene and measure contextual variables, such as sea-level depth, by querying data embedded in the 3D scene grid.

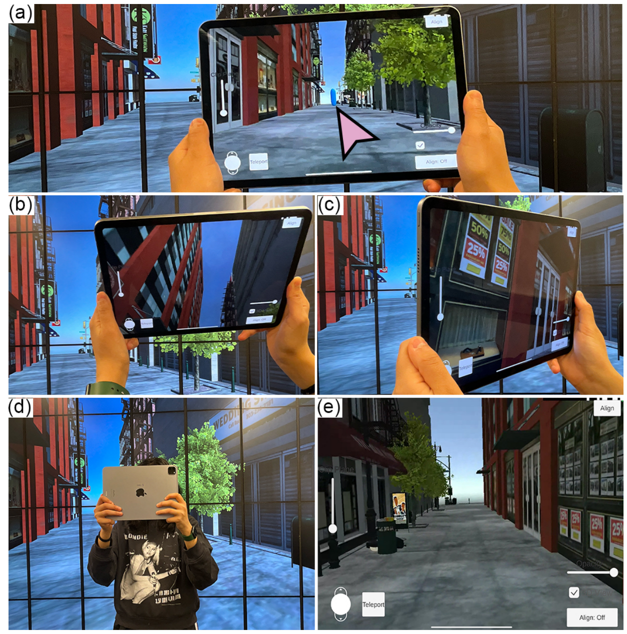

Figure 10: Immersive navigation using AuxiScope. (a) Using a 3D cursor, highlighted by the arrow, users can anchor a secondary virtual camera in the 3D scene on the LWD. (b) and (c) 3D scene views for corresponding aux orientations. (d) A user can also explore visual content due to missing display walls, e.g. a back wall. (e) The screenshot of the corresponding aux view, from (d).

For more complex navigation, we present an interaction design that allows users to utilize the aux to be temporarily *teleported* into the 3D scene, enabling enhanced and immersive scene exploration. This is shown in Fig. 10. By selecting a point in the 3D scene projected on the LWD, using a virtual cursor (Fig. 10(a)), the aux transitions into a 3D view of the scene, anchored to the selected point. Rotating the aux updates the camera orientation of the immersed view. Consider an urban visualization example; this interaction can help users navigate around intersections and buildings, as shown in Figs. 10(b)-(c), which may be difficult to view on the LWD or will require shifting the camera view. An interesting viewpoint is for single-wall LWDs, where users can use aux to view areas outside the bounds of the display (Figs. 10(d)-(e)).

4.2 Use Case I: Volume Rendering

Volume rendering is a widely used scientific visualization technique for exploring volumetric datasets. Volume exploration is inherently interactive, often involving user-driven adjustments such as transfer functions (TF), manipulation of cutting planes, and changes in view parameters to isolate and examine features of interest. These interactions are key for interpreting complex internal structures.

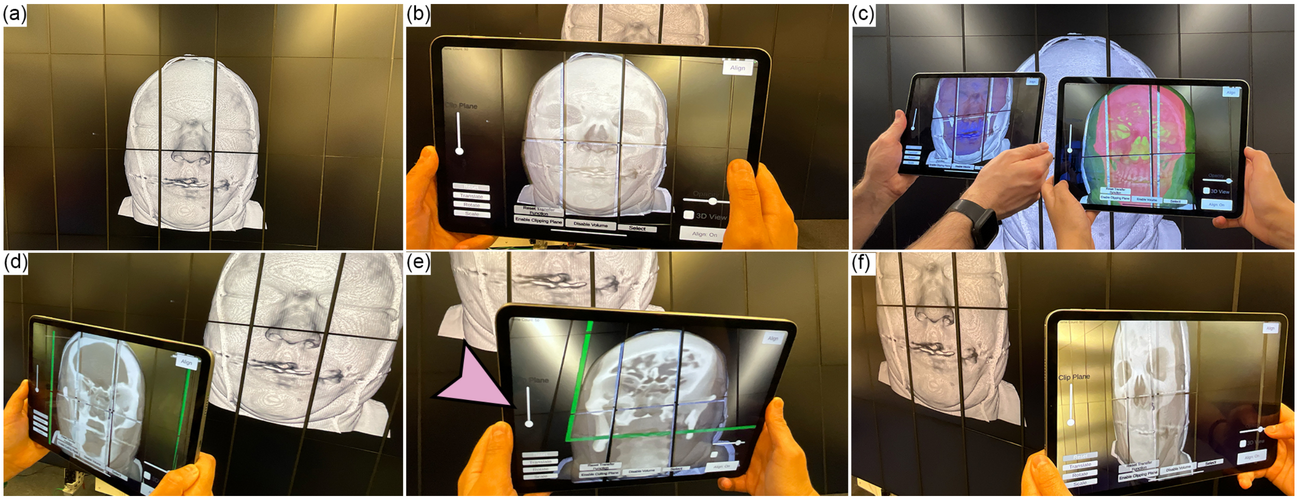

Figure 11: Interactive volume rendering using AuxiScope. (a) Head CT volume on an LWD. (b) Aux overlay with a different TF. (c) Two analysts viewing the volume on their aux, with different TFs. (d)–(f) AuxiScope interaction for volume cutting planes. Aux pose maps to cutting plane orientation. A slider, highlighted by the arrow, moves the plane along the viewing direction, *into* the LWD.

We demonstrate AuxiScope using a volume rendering of a CT head dataset. The volume is rendered on the LWD as a shared reference view (Fig. 11(a)). At the same time, individual users can adjust TFs via their aux interface and visualize the updated rendering spatially augmented on the shared view, analogous to overlaying a new visual layer that aligns with the original data (Fig. 11(b)). This allows users to perceive and compare personalized interpretations within the same spatial context, fostering both individual insight and collaborative analysis (Fig. 11(c)).

Additionally, we introduce an intuitive technique for dynamically inspecting the volume interior by mapping the aux pose to a cutting plane position and orientation. Users can physically translate and rotate their device to control the placement of the plane in real-time. Using a slider, users can move the cutting plane deeper *into* the volume, along the viewing direction of the aux. The resulting cross-section is displayed on the aux VST view, spatially coherent with the volume displayed on the LWD. Figs. 11(d)-(f) illustrate various cutting plane orientations based on the aux pose.

We invited a medical expert to provide feedback regarding AuxiScope's utility. Overall, the doctor expressed that this is a highly innovative tool with significant potential for clinical and educational applications. Regarding the cutting plane interaction, they highlighted that in conventional desktop-based systems, defining and adjusting cutting planes within volumetric scans, such as CT or MRI data, can be cumbersome, often involving challenging manipulation through sliders or input fields. In contrast, AuxiScope enables intuitive, hands-on control by allowing users to physically move and orient the aux. For instance, the doctor noted that inspecting specific ROIs, such as the paranasal sinuses or brainstem, becomes far more efficient and natural using spatial interaction.

The expert also emphasized AuxiScope's potential to support collaborative diagnostics. In multidisciplinary discussions (e.g., tumor board meetings or surgical planning), specialists often interpret the same volumetric data from their own clinical viewpoints. AuxiScope enables independent data exploration, allowing users to tailor visual parameters without disrupting the shared reference visualization on a large display. This capability, the doctor noted, could help bridge communication gaps, promote a deeper understanding across specialties, and accelerate consensus in complex cases.

4.3 Use Case II: Urban Flood Simulation Visualization

Recent research has shown that urban flood simulation visualization benefits from the scale and immersion of LWDs [13, 24, 18]. In this context, we present a multi-user use case for using AuxiScope in a flood visualization application. The application utilizes a simulated flooding scenario of Superstorm Sandy impacting New York City (NYC) [17], integrating elevation data, 3D building models, and public datasets from city sources [49].

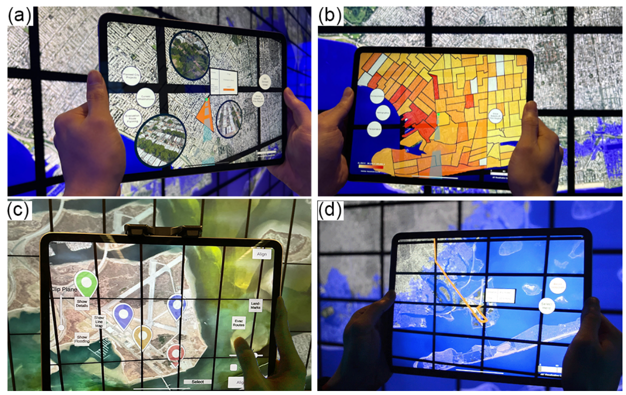

Figure 12: Using AuxiScope for urban flood visualization, users can (a) obtain detailed information by selecting ROIs on the map, (b) overlay visualization of demographic data, (c) highlight major landmarks, and (d) interactively visualize evacuation routes.

To evaluate this integrated environment, we invited participants from diverse backgrounds – government officials, community leaders, and emergency managers – to a focus group workshop. The participants were provided with two aux modes: *Analyst* and *Explorer*. Analyst mode (Figs. 12(a)–(b)) provided access to demographic data through dynamic heat maps, with census tracts color-coded by indicators such as ethnicity, education, or transportation access. Applying related filters updated the aux display accordingly. Explorer mode (Figs. 12(c)–(d)) visualized flood levels, landmarks, and evacuation routes across census tracts. These modes were developed in collaboration with social scientists. Participants used the aux devices to examine four locations, collaboratively assessing flooding impact on various demographics and communities.

Prior to the workshop, participants received tutorials on the aux and its modes. During the workshop, interactions were recorded and later reviewed using a custom framework for analyzing XR interactions [34]. Written and verbal feedback was collected, and participant behavior was observed and analyzed throughout.

Participants found the aux intuitive, supporting a convenient "*point-and-see*" metaphor for accessing supplementary information. Some with less experience using ubiquitous devices experienced minor difficulties in handling the aux and showed signs of the gorilla arm effect. Overall, participants valued the ability to tangibly overlay contextual visualizations onto the map and flooding data on the LWD reference display. One emergency manager noted the advantage over traditional command centers, where attention must shift across multiple displays, each with a different display of the data. In contrast, AuxiScope allowed them to maintain a single viewing direction and "*conveniently pull up*" the required data, "*without being distracted due to drastically shifting the view from one screen on one end of the room to another.*" Participants also appreciated maintaining personalized interactive views without disrupting collaborators. Collaborative use was common, with participants naturally communicating to share information.

One notable insight related to usability came from a participant who initially planned evacuation routes based solely on flooding data shown on the LWD. After overlaying supplementary data—such as economic status, demographics, and evacuation paths—their perception of social vulnerability shifted. They noted that while this is possible on a desktop, "*context switching between different windows or tabs... severely affects information retention.*"

5 User Study Design

To assess the effectiveness of AuxiScope, we posed the following research question: *How does AuxiScope's personalized rendering affect user performance?* Prior work by Cordeil et al. [19] demonstrated faster task completion with similar accuracy using individualized HMD views compared to shared CAVE displays. Inspired by these findings, we investigated whether similar benefits extend to aux in LWDs. We conducted an exploratory study comparing AuxiScope against a baseline shared-display condition, examining task completion time and accuracy.

5.1 Task Design

We designed a two-user collaborative task simulating scientific and information visualization scenarios, while remaining accessible to non-experts. In each trial, one to four city images appeared on the shared LWD, with city count systematically varied to test all task parameter combinations. Each participant received an exploratory prompt on their aux, requiring comparison of attributes across cities. To answer, users interacted with the cities using a 3D cursor and accessed associated data views. Each city contained one or more interactive visualizations relevant to the prompt. Once both participants submitted their responses, the next question appeared on the LWD. Task performance was measured by individual completion time and response accuracy (correct vs. incorrect).

To reduce confounds related to height-based accessibility or spatial positioning, cities appeared only within a central 50-inch horizontal region of the LWD. While limiting the spatial footprint of the task may constrain ecological validity, the study's primary focus is on isolating the impact of personalized rendering. Implications for full-screen or spatially distributed designs are discussed in Sec. 6.

5.2 Task Dimensions

To evaluate AuxiScope across varying conditions, we systematically manipulated four task dimensions: interaction modality, task breadth, task depth, and information overlap, resulting in 40 unique conditions per participant pair. Sessions were designed to last approximately 30 minutes, based on prior feedback on fatigue in tablet-based studies.

Modality – AuxiScope vs. Shared Display: The primary independent variable is rendering modality. In the *AuxiScope* condition, participants select and view data on their aux devices. In the *Shared Display* condition, the aux is used as input, but all data views are rendered on the LWD. To reflect realistic collaboration, both users can manipulate the shared display, with temporary locking to avoid conflicts. We hypothesize that personalized aux will reduce completion time and improve accuracy by minimizing visual resource contention.

Task Breadth – Areas of Interest (AoI): This dimension varies the number of cities displayed simultaneously, representing visual density and the potential for spatial overlap in user focus. We hypothesize that performance improves with additional cities due to reduced interaction conflicts, and that the relative benefit of aux diminishes under lower competition.

Task Depth – Number of Views Needed: This dimension varies the number of data views per city, representing analysis complexity. Each city includes one or two views showing different data facets. We hypothesize that greater depth increases cognitive load, lengthening completion times and reducing accuracy, while aux mitigates these effects through uninterrupted exploration.

Information Overlap – Number of Shared Views: This dimension captures the extent of visual information overlap between participants, ranging from zero to two shared views. Total views per task vary accordingly, from one (fully shared) to four (no overlap). We hypothesize that higher overlap reduces redundancy and improves collaborative efficiency by minimizing unique visual elements, and that aux will offer greater benefits in high-overlap conditions by enabling parallel, individualized access to shared content.

5.3 Participants and Procedure

We recruited 24 participants (16 male, 8 female, mean = 27.88 years, SD = 3.83) who gave informed consent under IRB approval. Prior AR experience was assessed via a pre-study survey: 5 participants reported *Never*, 11 *Rarely*, 5 *Sometimes*, 2 *Often*, and 1 *Daily*. Participants received a task overview and training on the aux interface before beginning the study. They were organized into 12 pairs, with each pair completing 40 trials across two blocks (AuxiScope and Shared Display). Block order was counterbalanced to mitigate order effects, and task conditions within each block were randomized. Breaks were provided between blocks or upon request.

5.4 Results

Since user performance in Shared Display tasks is affected by partner behavior, we report pair-level task time (the average completion time of both users) and pair-level accuracy (the combined number of correct responses for each question).

Completion time and accuracy comparisons between Shared Display and AuxiScope revealed exploratory trends. AuxiScope users completed tasks faster (26.53s vs. 28.65s) and slightly more accurately (1.41 vs. 1.35 on a 0–2 scale). Completion time increased with increasing AoI (from 14.88s for 1 to 39.48s for 4), while accuracy decreased (from 1.78 to 1.16). A similar time increase was observed with Views Needed (20.43s for one view vs. 32.37s for two) and Overlap (from 26.23s with no overlap to 33.91s with two). Interestingly, while accuracy declined with more AoIs, it slightly improved with more Views Needed and Overlap. High standard deviations (up to 20.86s for time, 0.65 for accuracy) highlight the need for further statistical testing.

To evaluate the effects of each dimension on the dependent variables, we fit two mixed-effects models with a shared fixed and random effect structure:

$$Time \sim \text{Modality} + AoI + \text{Views Needed} + \text{Overlap} + (1|P_{ID}) \quad (6)$$

where P_{ID} is a unique identifier given to each pair to account for between-pair dependencies.

Completion time was analyzed using a linear mixed-effects model. Accuracy, defined as the number of correct responses per pair (0–2), was modeled as a binomial outcome using a generalized linear mixed-effects model (GLMM) with a logit link. Model results are summarized in Tables 1 and 2, with estimated marginal means (EMMs) shown in Fig. 13.

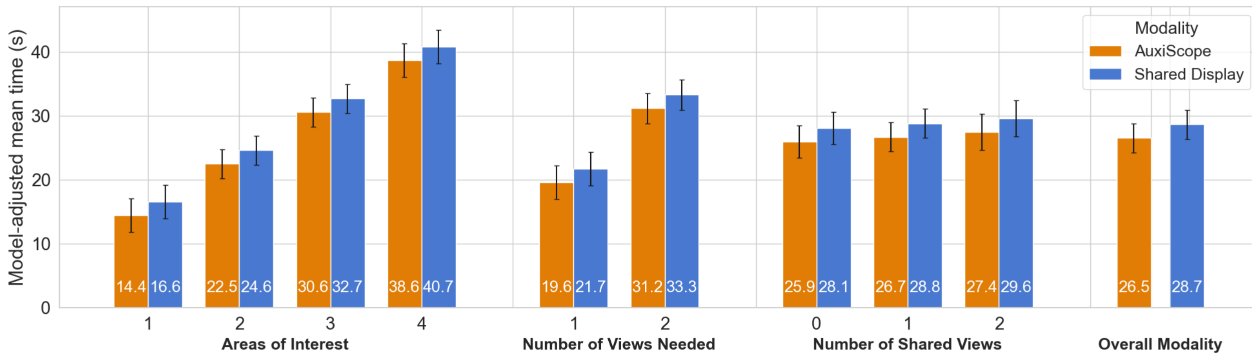

Figure 13: Model-adjusted task-completion times by predictor and display modality. Bars show mixed-model predicted means for each condition; error bars are their 95% confidence intervals

Table 1: Linear mixed-effect for pairwise completion time

Predictor	**Coef.**	**Std.Err.**	**z**	**P> |z|**	**95% CI**	
					0.025	**0.975**
Intercept	-12.73	2.34	-5.43	<001	-17.31	-8.13
Modality	2.12	1.03	2.05	0.04	0.09	4.14
Areas of Interest	11.56	1.12	10.36	<.001	9.37	13.75
Views Needed	8.06	0.46	17.45	<.001	7.16	8.97
Overlap	0.75	0.73	1.03	0.30	-0.68	2.18

Table 2: Binomial GLMM estimates for pairwise accuracy

Predictor	**Coef.**	**Std.Err.**	**z**	**P> |z|**	**95% CI**	
					0.025	**0.975**
Intercept	1.86	0.31	5.95	<.001	1.25	2.48
Modality	0.15	0.14	-1.01	0.31	-0.43	0.14
Areas of Interest	-0.51	0.07	-7.59	<.001	-0.64	-0.38
Views Needed	0.16	0.15	1.03	0.30	-0.14	0.46
Overlap	0.14	0.10	1.39	0.16	-0.06	0.35

Model results indicate a modest but statistically significant time advantage for the aux condition, with Shared Display tasks averaging 2.1 seconds longer. Task complexity also impacted time: each additional AoI and view increased completion time by 11.5 and 8 seconds, respectively. Contrary to our hypothesis, the number of shared views showed no significant effect on time or accuracy. For accuracy (Tab. 2), only *AoI* was statistically significant. Although the *Modality* coefficient suggests a slight accuracy benefit for aux, the effect was not conclusive. Overall, aux modestly improves task speed and may enhance accuracy, warranting further investigation to confirm these trends and uncover underlying mechanisms.

6 Conclusion and Future Work

We presented AuxiScope, a novel interaction system integrating handheld AR devices with LWDs to support personalized, co-located data exploration. Through a combination of spatially aligned AR overlays, AuxiScope enables users to independently interact with supplementary visualizations without disrupting shared content on the LWD. To this end, we address components such as AR localization, remote rendering, and timewarp-based latency correction for an effective system design. To demonstrate AuxiScope's utility, we presented a wide range of visualization scenarios. We validated them through both technical demonstrations and a user study comparing task performance across modalities. Our findings underscore the potential of auxiliary AR displays to enrich collaborative data analysis in immersive environments.

While AuxiScope introduces novel opportunities for co-located collaboration, several enhancements could augment its capabilities. These include collaborative features like aux-to-aux sharing or aux-to-display screencasting. Additionally, a remote AuxiScope mode could allow some users to experience a virtualized version of the scene through AR, tailored to lower-end mobile devices. Bridging the experiential gap between in-person and remote participants presents an exciting challenge for hybrid collaborative analysis.

Future iterations could optimize the graphics pipeline for a broader range of hardware configurations. Currenly, content is streamed from a head node to reduce client-side demands, but several strategies could improve adaptability. Lightweight visual elements such as cursors or glyphs could render locally to reduce latency. The system could adjust resolution or bandwidth usage based on device and network conditions. A profiling step at startup could guide dynamic load balancing, shifting rendering tasks to capable tablets or reducing fidelity on constrained hardware. These strategies would improve responsiveness while supporting diverse deployment scenarios.

AuxiScope current interaction model utilizes a pointing technique that employs a straight ray cast from the device center to the LWD. While simple and effective, this approach can lead to fatigue during extended use – commonly known as the *gorilla arm effect* – particularly when users must hold the device at chest or shoulder height to point. Based on existing research on mitigating fatigue for gesture interaction [70, 30], we hypothesize that the effect could also be mitigated on AuxiScope and similar tablet AR devices through novel interaction techniques that enhance pointing efficiency and accuracy while minimizing physical strain. We propose several ideas for future work: (1) curved selection rays that better align with natural hand and wrist movements; (2) predictive stabilization to filter out jitter from small device motions, and (3) context-aware snapping that guides the ray toward likely targets based on scene semantics. Additionally, incorporating AR mappings based on relative motion and clutched interaction, as described by Sereno et al. [58], may allow users to interact from more ergonomic holding positions, helping to further reduce fatigue. For future work, we plan to conduct studies with users belonging to different demographics to gather accuracy and fatigue data for varying distances from the user to the LWD, and the ROI on the LWD.

Lastly, while our preliminary results are promising, further validation with a larger sample, broader age range, varied tasks, and diverse collaborative scenarios will clarify the benefits of using aux. We are also interested in longitudinal deployments and qualitative assessments to examine their effect on group dynamics, strategy formation, and decision-making over time.

Acknowledgment

This work was supported in part by awards NSF IIS2107224, ONR N000142312124 and N000142412287, and IBM/SUNY 2404.

References

[1] C. Andrews, A. Endert, and C. North. Space to think: large high-resolution displays for sensemaking. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, p. 55–64, 10 pages, 2010. 1

[2] C. Andrews, A. Endert, B. Yost, and C. North. Information visualization on large, high-resolution displays: Issues, challenges, and opportunities. *Information Visualization*, 10(4):341–355, 2011. 1

[3] Apple. ARKit6 - Augmented Reality. https://developer.apple.com/augmented-reality/arkit/. Accessed: March 15, 2025. 4

[4] T. Babic, H. Reiterer, and M. Haller. Understanding and creating spatial interactions with distant displays enabled by unmodified off-the-shelf smartphones. *Multimodal Technologies and Interaction*, 6(10):94, 2022. 2

[5] R. Ball, C. North, and D. A. Bowman. Move to improve: promoting physical navigation to increase user performance with large displays. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, p. 191–200, 10 pages, 2007. 1, 3

[6] M. Beaudouin-Lafon. Lessons learned from the wild room, a multisurface interactive environment. *Proceedings of the Conference on L'interaction Homme-machine*, pp. 1–8, 2011. 3

[7] L.-P. Bergé, M. Serrano, G. Perelman, and E. Dubois. Exploring smartphone-based interaction with overview+detail interfaces on 3d public displays. *Proceedings of the ACM International Conference on Human-Computer Interaction with Mobile Devices & Services*, 2014. 3

[8] L. Besançon, P. Issartel, M. Ammi, and T. Isenberg. Hybrid tactile/tangible interaction for 3D data exploration. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):881–890, 2017. 2, 3

[9] L. Besançon, M. Sereno, L. Yu, M. Ammi, and T. Isenberg. Hybrid touch/tangible spatial 3D data selection. *Computer Graphics Forum*, 38(3):553–567, 2019. 3

[10] L. Besançon, A. Ynnerman, D. F. Keefe, L. Yu, and T. Isenberg. The state of the art of spatial interfaces for 3D visualization. *Computer Graphics Forum*, 40(1):293–326, 2021. 2, 3

[11] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. Toolglass and magic lenses: the see-through interface. *Proceedings of the ACM Conference on Computer Graphics and Interactive Techniques*, p. 73–80, 8 pages, 1993. 2

[12] S. Boorboor, D. Gutiérrez-Rosales, A. Shoaib, C. Kalsi, Y. Wang, Y. Cao, X. Gu, and A. E. Kaufman. Silo: Half-gigapixel cylindrical stereoscopic immersive display. *IEEE Conference Virtual Reality and 3D User Interfaces*, pp. 483–493, 2025. 4

[13] 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 Transactions on Visualization and Computer Graphics*, 30(9):6365–6377, 2024. 2, 7

[14] A. Bornik, R. Beichel, E. Kruijff, B. Reitinger, and D. Schmalstieg. A hybrid user interface for manipulation of volumetric medical data. *IEEE 3D User Interfaces*, pp. 29–36, 2006. 3

[15] D. Breglia, M. Spooner, and D. Lobb. Helmet mounted laser projector. *Proceedings of the Image Generation and Display Conference*, pp. 241–258, 1981. 3

[16] O. Chapuis, A. Bezerianos, and S. Frantzeskakis. Smarties: an input system for wall display development. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, p. 2763–2772, 10 pages, 2014. 3

[17] B. A. Colle, M. J. Bowman, K. J. Roberts, M. H. Bowman, C. N. Flagg, J. Kuang, Y. Weng, E. B. Munsell, and F. Zhang. Exploring water level sensitivity for metropolitan New York during Sandy (2012) using ensemble storm surge simulations. *Journal of Marine Science and Engineering*, 3(2):428–443, 2015. 7

[18] B. A. Colle, J. R. Hathaway, E. J. Bojsza, J. M. Moses, S. J. Sanders, K. E. Rowan, A. L. Hils, E. C. Duesterhoeft, S. Boorboor, A. E. Kaufman, et al. Risk perception and preparation for storm surge flooding: A virtual workshop with visualization and stakeholder interaction. *Bulletin of the American Meteorological Society*, 104(7):E1232–E1240, 2023. 7

[19] M. Cordeil, T. Dwyer, K. Klein, B. Laha, K. Marriott, and B. H. Thomas. Immersive collaborative analysis of network connectivity: CAVE-style or head-mounted display? *IEEE Transactions on Visualization and Computer Graphics*, 23(1):441–450, 2017. 8

[20] E. Courtoux, C. Appert, and O. Chapuis. SurfAirs: Surface + mid-air input for large vertical displays. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, 2023. 3

[21] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. V. Kenyon, and J. C. Hart. The CAVE: audio visual experience automatic virtual environment. *Communications of ACM*, 35(6):64–72, 9 pages, 1992. 1

[22] S. Eilemann, M. Makhinya, and R. Pajarola. Equalizer: A scalable parallel rendering framework. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):436–452, 2009. 3

[23] A. Febretti, A. Nishimoto, T. Thigpen, J. Talandis, L. Long, J. D. Pirtle, T. Peterka, A. Verlo, M. Brown, D. Plepys, D. Sandin, L. Renambot, A. Johnson, and J. Leigh. CAVE2: a hybrid reality environment for immersive simulation and information analysis. *The Engineering Reality of Virtual Reality*, 8649:864903, 2013. 4

[24] L. Ferreira, G. Moreira, M. Hosseini, M. Lage, N. Ferreira, and F. Miranda. Assessing the landscape of toolkits, frameworks, and authoring tools for urban visual analytics systems. *Computers & Graphics*, 123(C), 2024. 7

[25] Google. Google for Developers - ARCore. https://developers.google.com/ar. Accessed: March 15, 2025. 4

[26] T. Horak, S. K. Badam, N. Elmqvist, and R. Dachselt. When David meets Goliath: Combining smartwatches with a large vertical display for visual data exploration. *ACM CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2018. 2

[27] H. Hua and L. D. Brown. Magic Lenses for Augmented Virtual Environments . *IEEE Computer Graphics and Applications*, 26(04):64–73, 2006. 2

[28] R. James, A. Bezerianos, and O. Chapuis. Evaluating the extension of wall displays with AR for collaborative work. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, article no. 99, 17 pages, 2023. 2

[29] Y. Jansen, P. Dragicevic, and J.-D. Fekete. Tangible remote controllers for wall-size displays. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, p. 2865–2874, 10 pages, 2012. 3

[30] P. Kartick, A. Uribe-Quevedo, and D. Rojas. Piecewise: A non-isomorphic 3D manipulation technique that factors upper-limb ergonomics. *Virtual Worlds*, 2(2):144–161, 2023. 9

[31] N. Katzakis, R. J. Teather, K. Kiyokawa, and H. Takemura. Inspect: extending plane-casting for 6-dof control. *Human-centric Computing and Information Sciences*, 5:1–22, 2015. 3

[32] K. Kim and N. Elmqvist. Embodied lenses for collaborative visual queries on tabletop displays. *Information Visualization*, 11(4):319–338, 20 pages, 2012. 2

[33] P. Kim, J. Orlosky, K. Kiyokawa, P. Ratsamee, and T. Mashita. Dot-Warp: Dynamic object timewarping for video see-through augmented reality. *IEEE International Symposium on Mixed and Augmented Reality*, pp. 184–185, 2017. 3

[34] Y. Kim, Z. Aamir, M. Singh, S. Boorboor, K. Mueller, and A. E. Kaufman. Explainable XR: Understanding user behaviors of XR environments using LLM-assisted analytics framework. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–11, 2025. 7

[35] R. Langner, U. Kister, and R. Dachselt. Multiple coordinated views at large displays for multiple users: Empirical findings on user behavior, movements, and distances. *IEEE Transactions on Visualization and Computer Graphics*, 25:608–618, 2019. 3

[36] R. Langner, M. Satkowski, W. Büschel, and R. Dachselt. MARVIS: Combining mobile devices and augmented reality for visual data analysis. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, article no. 468, 17 pages, 2021. 2

[37] G. M. León, A. Bezerianos, O. Gladin, and P. Isenberg. Talk to the wall: The role of speech interaction in collaborative visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 2024. 3

[38] L. Lisle, K. Davidson, L. Pavanatto, I. A. Tahmid, C. North, and D. A. Bowman. Spaces to think: A comparison of small, large, and immersive displays for the sensemaking process. *IEEE International Symposium on Mixed and Augmented Reality*, pp. 1084–1093, 2023.

1
[39] 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 Transactions on Visualization and Computer Graphics*, 22(5):1616–1629, 2016. 2, 3
[40] S. Malik, A. Ranjan, and R. Balakrishnan. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. *Proceedings of the ACM Symposium on User Interface Software and Technology*, p. 43–52, 10 pages, 2005. 3
[41] W. R. Mark, L. McMillan, and G. Bishop. Post-rendering 3D warping. *Proceedings of the ACM Symposium on Interactive 3D Graphics*, p. 7–ff., 1997. 3
[42] F. Matulic and D. Vogel. Multiray: Multi-finger raycasting for large displays. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, p. 1–13, 13 pages, 2018. 3
[43] E. Mayer, T. Odaker, D. Kolb, S. Müller, and D. Kranzlmüller. LED CAVE - new dimensions for large-scale immersive installations. *IEEE Conference on Virtual Reality and 3D User Interfaces*, pp. 515–519, 2024. 4
[44] T. Mazuryk and M. Gervautz. Two-step prediction and image deflection for exact head tracking in virtual environments. *Computer Graphics Forum*, 14(3):29–41, 1995. 3
[45] D. C. McCallum and P. Irani. ARC-Pad: absolute+relative cursor positioning for large displays with a mobile touchscreen. *Proceedings of the ACM Symposium on User Interface Software and Technology*, p. 153–156, 4 pages, 2009. 3
[46] M. M. Miguel, T. Ogawa, K. Kiyokawa, and H. Takemura. A pda-based see-through interface within an immersive environment. *International Conference on Artificial Reality and Telexistence*, pp. 113–118, 2007. 3
[47] S. Mirhosseini, I. Gutenko, S. Ojal, J. Marino, and A. Kaufman. Immersive virtual colonoscopy. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):2011–2021, 2019. 6
[48] A. Nishimoto and A. E. Johnson. Extending virtual reality display wall environments using augmented reality. *ACM Symposium on Spatial User Interaction*, article no. 7, 5 pages, 2019. 2
[49] NYC Planning. BYTES of the BIG APPLE. www.nyc.gov/site/planning/data-maps/open-data.page. Online; accessed March 11, 2025. 7
[50] A. Olwal and S. Feiner. Spatially aware handhelds for high-precision tangible interaction with large displays. *Proceedings of the ACM International Conference on Tangible and Embedded Interaction*, 2009. 3
[51] C. Papadopoulos, K. Petkov, A. E. Kaufman, and K. Mueller. The Reality Deck–an immersive gigapixel display. *IEEE Computer Graphics and Applications*, 35(1):33–45, 2015. 1, 3, 4
[52] A. Prouzeau, A. Bezerianos, and O. Chapuis. Evaluating multi-user selection for exploring graph topology on wall-displays. *IEEE Transactions on Visualization and Computer Graphics*, 23(8):1936–1951, 2017. 3
[53] T. Rau, T. Isenberg, A. Koehn, M. Sedlmair, and B. Lee. Traversing dual realities: Investigating techniques for transitioning 3D objects between desktop and augmented reality environments. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, article no. 1236, 16 pages, 2025. 2
[54] P. Reipschläger and R. Dachselt. Designar: Immersive 3d-modeling combining augmented reality with interactive displays. *Proceedings of the ACM International Conference on Interactive Surfaces and Spaces*, 2019. 2
[55] P. Reipschlager, T. Flemisch, and R. Dachselt. Personal augmented reality for information visualization on large interactive displays. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1182–1192, 2021. 2
[56] A. Saktheeswaran, A. Srinivasan, and J. Stasko. Touch? speech? or touch and speech? investigating multimodal interaction for visual network exploration and analysis. *IEEE Transactions on Visualization and Computer Graphics*, 26(6):2168–2179, 2020. 3
[57] D. Schikore, R. Fischer, R. Frank, R. Gaunt, J. Hobson, and B. Whitlock. High-resolution multiprojector display walls. *IEEE Computer Graphics and Applications*, 20(4):38–44, 2000. 1
[58] M. Sereno, S. Gosset, L. Besançon, and T. Isenberg. Hybrid touch/tangible spatial selection in augmented reality. *Computer Graphics Forum*, 41(3):403–415, 2022. 3, 9
[59] P. Song, W. B. Goh, C.-W. Fu, Q. Meng, and P.-A. Heng. Wysiwyf: exploring and annotating volume data with a tangible handheld device. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, 2011. 3
[60] M. Spindler, W. Büschel, C. Winkler, and R. Dachselt. Tangible displays for the masses: spatial interaction with handheld displays by using consumer depth cameras. *Personal and Ubiquitous Computing*, 18:1213–1225, 2014. 2
[61] M. Spindler, S. Stellmach, and R. Dachselt. PaperLens: advanced magic lens interaction above the tabletop. *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, p. 69–76, 8 pages, 2009. 2
[62] M. Spindler, C. Tominski, H. Schumann, and R. Dachselt. Tangible views for information visualization. *ACM International Conference on Interactive Tabletops and Surfaces*, p. 157–166, 10 pages, 2010. 2
[63] T. Sun, Y. Ye, I. Fujishiro, and K.-L. Ma. Collaborative visual analysis with multi-level information sharing using a wall-size display and see-through hmds. *IEEE Pacific Visualization*, pp. 11–20, 2019. 2
[64] C. Tominski, S. Gladisch, U. Kister, R. Dachselt, and H. Schumann. Interactive lenses for visualization: An extended survey. *Computer Graphics Forum*, 36(6):173–200, 2017. 1, 2, 3
[65] R. Tredinnick, B. Boettcher, S. Smith, S. Solovy, and K. Ponto. Unicave: A unity3d plugin for non-head mounted vr display systems. *Proceedings of the IEEE Virtual Reality*, p. In Print, 2017. 6
[66] Unity3D. AR Foundation. https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.1/manual/index.html. Accessed: March 15, 2025. 4, 6
[67] Unity3D. WebRTC. https://docs.unity3d.com/Packages/com.unity.webrtc@2.4/manual/index.html. Accessed: March 15, 2025. 6
[68] J. M. P. van Waveren. The asynchronous time warp for virtual reality on consumer hardware. *Proceedings of the ACM Conference on Virtual Reality Software and Technology*, p. 37–46, 10 pages, 2016. 3
[69] J. Viega, M. J. Conway, G. Williams, and R. Pausch. 3d magic lenses. *Proceedings of the ACM Symposium on User Interface Software and Technology*, p. 51–58, 8 pages, 1996. 2
[70] J. Wentzel, G. d'Eon, and D. Vogel. Improving virtual reality ergonomics through reach-bounded non-linear input amplification. *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, p. 1–12, 12 pages, 2020. 9
[71] S. Wu, D. Byrne, and M. W. Steenson. Megereality: Leveraging physical affordances for multi-device gestural interaction in augmented reality. *ACM CHI Conference on Human Factors in Computing Systems*, p. 1–4, 4 pages, 2020. 2
[72] L. Zhao, T. Isenberg, F. Xie, H.-N. Liang, and L. Yu. SpatialTouch: Exploring Spatial Data Visualizations in Cross-Reality . *IEEE Transactions on Visualization & Computer Graphics*, 31(01):897–907, 2025. 2