GHOST in the Robots: Real-Time Exocentric Dual-Robot VR Teleoperation from Onboard Cameras

Yichen Wei*, Faisal Zaghloul*, Soujanya C Aryal, Aanya K. Agrawal, Chengfan Li, Jason Xinyu Liu‡, James Tompkin, and Stefanie Tellex

Fig. 1: **GHOST enables single-operator dual-robot control.** Our VR teleoperation system enables individual or simultaneous dual-robot control to perform complex mobile manipulation tasks, with a 3D virtual scene reconstructed and rendered in real time from onboard cameras.

***Abstract*—Teleoperating multiple robots simultaneously enables additional views and coordinated control. Yet, it poses fundamental challenges: the system must present sensor data cohesively and allow operators to manage multiple robot bases, arms, and cameras while maintaining low latency. Current multi-robot teleoperation systems require multiple operators, rely on autonomy, or restrict operators to high-level commands. We present GHOST: an open-source VR teleoperation system that enables single-operator control of two mobile manipulators via direct low-level commands using only onboard sensing. GHOST creates an exocentric 3D workspace by aligning real-time point clouds from the robots' RGB-D cameras, where scene coverage is improved through learning-based completion to aid operator spatial awareness. For control, the operator uses a mode-switching architecture to command either robot individually or both robots simultaneously. Experiments with 15 novice participants demonstrate 1.6–4× the success rate of an off-the-shelf tablet interface. For experts across nine challenging dual-robot tasks, our system enabled completion of two tasks that were infeasible with the tablet, and was 1.47× faster on average than the tablet. Website and code: https://h2r.github.io/GHOST/.**

***Index Terms*—Telerobotics and Teleoperation; Virtual Reality and Interfaces; Multi-Robot Systems.**

Manuscript received: March 20, 2026; Revised June 25, 2026; Accepted August 2, 2026.

This paper was recommended for publication by Editor Ki-Uk Kyung upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by NASA 80NSSC23M0075, NSF CNS-2038897, and Amazon.

*Equal contribution.

Yichen Wei, Faisal Zaghloul, Soujanya C Aryal, Aanya K. Agrawal, Chengfan Li, James Tompkin, and Stefanie Tellex are with the Department of Computer Science, Brown University, Providence, RI 02912, USA stefanie_tellex@brown.edu.

‡Jason Xinyu Liu is with CSAIL, Massachusetts Institute of Technology, Cambridge, MA 02139, USA. Work completed while at Brown University.

Digital Object Identifier (DOI): 10.1109/LRA.2026.3726328.

I. Introduction

TELEOPERATION systems aim to enable remote robot control in complex unstructured environments. This is often attempted via virtual reality (VR) because immersing operators in a virtual space with the robot can provide stereo depth perception, improved spatial awareness, and precise, intuitive control. Existing VR interfaces typically control only one robot. However, real-world tasks like transporting large or elongated objects require multiple support points, and this could be achieved with multiple robots. In addition, single-robot control often constrains the operator's field of view due to limited camera angles and arm occlusions that create blind spots. This forces operators to work with incomplete information or frequently reposition the robot to adjust their view.

Deploying multiple robots in VR teleoperation introduces new challenges to single-operator perception and control. The perception system must present precise and coherent spatial information to the operator. Additionally, operators must divide their limited attention across multiple robot bases and arms [1, 2, 3, 4]. Thus, the control system must efficiently map the operator's inputs to the high-dimensional robot control space. Lastly, the system overall must provide responsive control with multiple video streams, 3D scene data, and synchronized commands, requiring careful system design to maintain low latency over wireless connections. Existing multi-robot systems typically cannot meet these challenges. They may rely upon multiple operators [4], autonomy with only occasional human intervention [5], or constrain operators to only high-level commands rather than precise low-level control [6, 7].

To address these challenges, we present GHOST (GHOST Human Operator for Simultaneous Teleoperation), a system that enables a single operator to control two mobile manipulators, individually or simultaneously, from within an immersive

virtual scene, as if the operator were an invisible ghost standing alongside the robots. Our system reconstructs a 3D point cloud by processing RGB-D streams from the robots' onboard cameras. As each camera only provides sparse depth information, GHOST applies real-time depth completion to infer missing regions, then aligns the completed point clouds into a coherent virtual scene. This expands the operator's field of view and reduces the blind spot problem. The control scheme allows manipulation and navigation, enabling intuitive and real-time 6-DoF pose control of two mobile manipulators through two hand controllers. Simultaneous control of both robots through only two hand controllers is possible because their relative base poses can be 'locked', keeping cognitive load low. Our system design that combines real-time scene reconstruction, an immersive exocentric view, and low-level multi-robot control has not been previously shown.

We evaluate the system on nine teleoperation tasks that are difficult or impractical with a single robot but simpler with two. We compare to an off-the-shelf tablet interface. Across both novice and expert operators, our system improved task success rate and typically also completion time. For 15 novices, who attempted two easier tasks, our system improved the success rate by 1.6–4×. For three experts, our system enabled completion of two tasks that were infeasible with the tablet, and the remaining seven tasks were completed 1.47× faster on average. An ablation replacing the 3D scene with RGB camera feeds showed that, for experts, direct 6-DoF control provides much of the speed benefit, while the reconstructed 3D scene improves task completion reliability. Taken together, these results show that our new VR-based teleoperation system can provide improvements in efficiency and reliability and also enable new multi-robot capabilities on existing hardware.

We make the following contributions: (1) An exocentric 3D scene for multi-robot teleoperation; (2) A low-latency depth completion pipeline; (3) A mode-switching control design for dual-robot teleoperation; and (4) A set of challenging evaluation tasks for dual-robot teleoperation.

II. Related Work

A. Visual Representations for Teleoperation

Robot teleoperation interfaces generally follow two representational paradigms: egocentric and exocentric [14, 15]. **Egocentric** interfaces, common in VR teleoperation [16, 17], present the environment from the robot's perspective and map commands to its frame. However, camera motion delays relative to human head and eye motion can induce motion sickness [18]. In multi-robot settings, operators need to mentally reconstruct spatial relationships by switching between cameras, increasing cognitive load. Simultaneous egocentric control is also impractical, as it requires separate live inputs, e.g., two tablet interfaces. GHOST instead uses an exocentric design that unifies multi-robot coordination in a shared 3D space.

Exocentric interfaces visualize robots and sensor data in a global frame, providing third-person views that naturally support multi-robot coordination [19]. Existing exocentric VR systems may require external capture cameras [7], limiting operational scope. Additionally, accurate 3D reconstruction remains challenging: stereo depth is sparse; Gaussian splatting and neural radiance fields [20] produce high-quality renderings but are too slow for real-time reconstruction. This motivates our use of depth completion models.

B. Depth Completion

One challenge that arises when generating point clouds is the sparsity of the input depth data. Depth completion techniques attempt to fill holes in depth camera data caused by low-reflectance objects, multi-path light effects, or distant scene regions, using an RGB image as a guide. Earlier methods used convolutional neural networks (CNNs) with different strategies to fuse depth features [21, 22, 23, 24], while more recent works use vision transformers (ViT) [25, 26, 27] and diffusion models [28] to improve depth completion quality [29, 30, 31].

For interactive robotics scenarios, we require a depth completion method that provides high-quality predictions, generalizes well to novel environments, and maintains an interactive frame rate. For our system, the ViT-based model PromptDA [29] best fits our needs, providing a balance between runtime and quality. PromptDA uses an adapter head that processes and fuses features from the transformer backbone at various layers with the input color and depth maps. While emerging works can produce point clouds from a batch of RGB(-D) images [32, 33], we found the resulting point clouds to be too low-quality for interactive real-time use.

C. Control Systems for Teleoperation

Beyond visual feedback, effective teleoperation requires control interfaces that balance ease of use and precise robot control. For navigation, direct physical guidance [34] provides intuitive control but is not remote teleoperation. Point-and-click interfaces on maps, images, or the physical scene [12, 35, 36] use high-level path planning but lack low-level control. Joysticks can offer both precise and remote control [37]. Learned controllers can generate base motion automatically while the operator teleoperates only the arm [38].

For manipulation, direct physical guidance of the robot [39] or a handheld gripper [40] again precludes remote teleoperation. Velocity-based devices, from familiar joysticks to 6-DoF masters and space mice [41, 42], are less intuitive for mapping to 6D pose [43]. Joint-copy and leader-follower systems enable direct joint control but require platform-specific leader hardware, such as robot-arm leaders [44] or low-cost printed replicas [45], that limits generalizability. Hand gesture tracking [16, 46, 47, 48, 49] can aid dexterous manipulation and imitation learning data collection if the tracking cameras have a clear view of the hands. VR systems with controller pose tracking [15, 50, 51, 52] offer promise as an intuitive manipulation control.

D. Multi-robot Teleoperation

Multi-robot coordination introduces challenges in information fusion, inter-robot communication, and shared operator attention, and prior systems span several design choices (Tab. I). **High-level autonomous systems** such as Spot-On [7] and

TABLE I: **GHOST is a new point in the multi-robot teleoperation design space.** Existing multi-robot teleoperation works did not produce real-time 3D scene depictions with low-level multi-robot control.

Work	Novel environments?	Single operator?	Low-level actions?	Multi-robot coordination	Real-time scene?	3D scene?	Onboard cameras?
Lewis et al. [8]	✓	✓	✓	Mirror	✓	✗	✓
Roldan et al. [6]	✗	✓	✗	High-level action only	✗	✓	✗
CollabTeleop [4]	✓	✗	✓	N/A	✓	✗	✗
Body Extension [9]	✓	✓	✓	Individual control	✓	✗	✓
Laghi et al. [10]	✗	✓	✓	Pivot	✗	✗	✗
Ozdamar et al. [11]	✗	✓	✓	Pivot	✗	✗	✗
Zick et al. [12]	✗	✓	✗	High-level action only	✓	✗	✗
Scaled Autonomy [5]	✓	✓	✓	Autonomous + occasional individual control	✗	✗	✗
Spot-On [7]	✗	✓	✗	High-level action only	Partial	✓	✗
Chen et al. [13]	✓	✓	✗	High-level action only	✓	✓	✓
Ours	✓	✓	✓	**Pivot + individual control**	✓	✓	✓

others [6, 12, 13] address these challenges through semi-autonomous behaviors and interfaces for scene exploration or navigation goals. However, they often rely on pre-built maps [6], static 3D overlays [7], or 2D abstract interfaces [12], assuming known environments and precluding low-level commands. Most similar in spirit, Chen et al. [13] co-localize multiple robots and an MR (Mixed Reality) headset using onboard sensors and overlay a live dense 3D reconstruction, but restrict the operator to high-level drag-and-drop goal poses executed by an autonomous path planner, targeting navigation rather than low-level mobile manipulation.

At the opposite end, **low-level multi-operator systems** such as CollabTeleop [4] provide direct control by assigning one operator per robot, but increase human-resource and coordination costs while limiting situational awareness to single onboard cameras. Between these extremes, **low-level single-operator** approaches use coordinated control strategies: shared pivot points [10, 11] control multiple robots as a group, while leader-follower architectures [8] make secondary robots mirror a primary robot. Attention-scheduling methods [5] instead learn operator preferences to decide which robot receives control, but assume otherwise autonomous robots needing only occasional intervention. These systems support simultaneous control or attention-switching, but offer limited independent execution [10] and typically provide only RGB streams [4, 8], which are insufficient for precise spatial reasoning.

Among these, Body Extension [9] is closest to ours, assigning one arm to each robot. However, it lacks both integrated 3D vision and coordinated base control, limiting scene understanding and precluding tasks such as cooperative carrying. No existing work provides a single operator with low-level independent and synchronized control of multiple mobile manipulators while maintaining real-time 3D awareness of a shared workspace.

III. System Overview

Our system provides single-operator teleoperation of two Boston Dynamics Spot mobile manipulators within VR. Each Spot consists of a quadruped base, a 6-DoF arm with a gripper and gripper camera, a LiDAR unit on the back, and five body-mounted RGB-D cameras, of which we use the front two for scene reconstruction. The robots communicate via onboard 802.11n Wi-Fi with a ROS 2 [53] server and a VR client.

The system integrates off-the-shelf tools (PromptDA, ROS 2, Unity, Spot SDK) into three components (Fig. 2): (1) A **Unity-based VR client** that renders the visual feedback and provides the operator with mode-switching control of either robot individually or both simultaneously; (2) A **ROS 2 server** that manages low-bandwidth state streaming (i.e., small message data), command execution, multi-robot localization, and synchronized navigation; (3) **SpotObserver**, a custom GPU-accelerated library on the VR client that bypasses ROS 2 messaging to retrieve RGB-D data directly from each robot, performs real-time depth completion, and produces dense point clouds with minimal latency for Unity to render. The next two sections describe what these components provide to the operator: visual feedback (Sec. IV) and control (Sec. V).

IV. Visual Feedback

A. Scene Elements

The operator's visual feedback has four scene elements:

1) Robot models: Live URDF Spot models are rendered with joint states streamed from each robot via ROS 2. These models provide the operator with precise, always-visible kinematic state of each robot, including arm and gripper configurations.

2) 3D point clouds: Frames from each robot's front-facing RGB-D cameras are combined into a point cloud in real time (Sec. IV-B). Both robots' point clouds are rendered as splats in the virtual view given the robot pose estimates (Sec. IV-B2).

3) 2D gripper-camera feeds: Each gripper-camera feed is displayed as a floating panel near the operator's controller models. This close-up view compensates for cases where objects are occluded from the front-facing cameras, providing visual feedback for fine-grained grasping.

4) Virtual gripper targets: A virtual gripper is rendered at the commanded end-effector pose, distinct from the robot model's gripper. As network latency and robot dynamics delay the robot's physical response to the operator's input, this target gives immediate visual feedback of the operator's intended gripper pose; the robot model's gripper approaches the virtual gripper as the robot matches the target pose.

Of the four elements above, the robot models and virtual gripper targets are rendered directly from state data with negligible cost. The 2D gripper-camera feeds and 3D point clouds both originate from the vision processing pipeline.

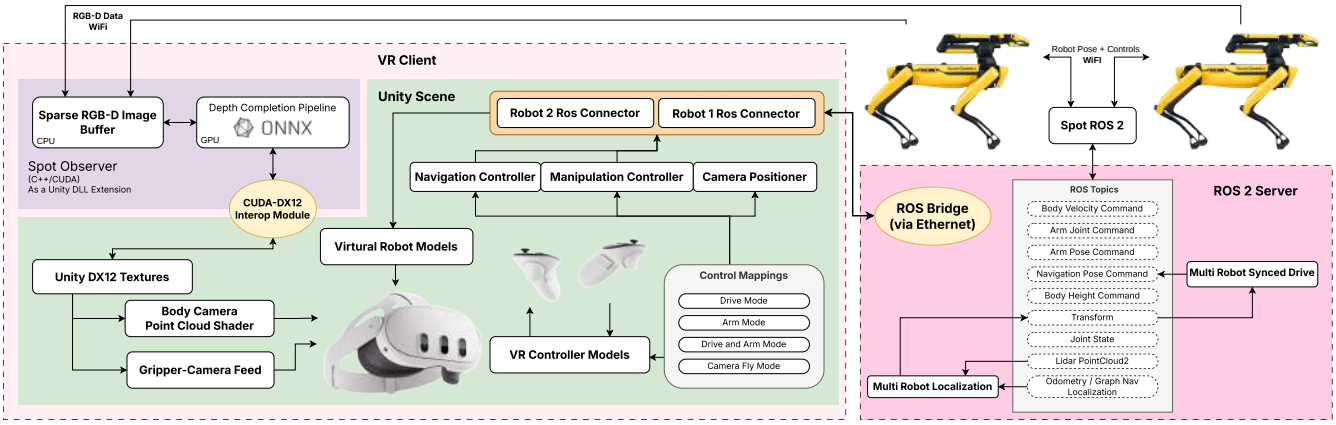

Fig. 2: **A diagram of the GHOST system.** The system integrates multi-robot ROS 2 communication, a GPU-accelerated RGB-D processing pipeline, and a Unity-based VR interface. Low-bandwidth state/command data are bridged through ROS 2 and drive the virtual robot models, while high-bandwidth RGB-D streams are processed on the GPU and rendered as interactive point clouds for low-latency teleoperation.

B. Vision Processing Pipeline

This pipeline must (1) have low latency, (2) register both robots' point clouds in a shared frame, and (3) complete sparse, noisy depth measurements into dense metric point clouds.

1) Latency reduction: High-bandwidth (i.e., large payload) RGB-D streams are processed by SpotObserver in threads separate from Unity's render loop: for each robot, dedicated threads fetch RGB-D data and run depth completion, while the main rendering thread transfers completed data to Unity. SpotObserver shares GPU resources with Unity through CUDA-DX12 interoperability, avoiding CPU-GPU copies when moving data into the rendering context.

For our experiments, the robots communicated with the VR client and ROS 2 server over a dedicated Wi-Fi network. The robots streamed their RGB-D data at 640×480 resolution. The system achieved 71 fps headset rendering with a 4.3 fps scene update rate, limited primarily by Wi-Fi bandwidth. Latency is dominated by sensor-frame delivery over the network rather than by depth completion or rendering (Tab. II).

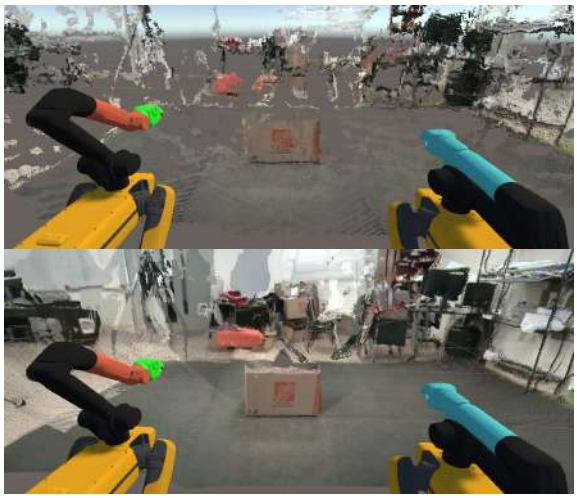

Fig. 3: **Shared 3D scene before (above) and after (below) real-time depth completion.** Completion improves scene coverage and makes the point clouds from both robots' cameras more coherent.

TABLE II: **Average SpotObserver latency per stage.**

Sensor-frame delivery	Depth completion	Render
230 ms	131 ms	14 ms

2) Relative 3D transform: Each robot's built-in localization estimates its base pose in a world frame, giving an initial inter-robot transform. Because these estimates are insufficient for point-cloud alignment, we refine the transform with an iterative closest point (ICP) correction computed from both robots' LiDAR data. Robot localization updates continuously, while the ICP correction is recomputed every 4 seconds.

3) Metric depth completion: SpotObserver uses PromptDA for depth completion to clean up noisy and sparse depth-sensor data (Fig. 3). To improve completion quality, we apply nearest-neighbor interpolation to prefill sparse depth measurements before feeding them to the model. Both prefilling and model inference run on the GPU via CUDA and ONNX Runtime.

V. Control

A. Design

Controlling two mobile manipulators simultaneously is a challenge in human attention allocation and interface design. The operator must manage each robot's base and arm controls, and the exocentric camera, resulting in five simultaneous control channels. The Meta Quest 3 provides two hand controllers, each equipped with four buttons, a thumbstick, and 6-DoF tracking. To effectively map the available controls to the required control channels, we designed a mode-switching architecture that presents operators with task-relevant control combinations.

GHOST separates *robot selection* from *control mode*. In single-robot operation, each hand controller can be assigned to one robot and placed in one of four task-level modes: *Fly*, which repositions the exocentric camera; *Drive*, which controls the robot base; *Arm*, which maps controller pose to end-effector pose while providing gripper control; and *Arm-Drive*, which combines base and arm control on one controller. In addition, GHOST provides a distinct *Dual-Robot* mode for coordinated

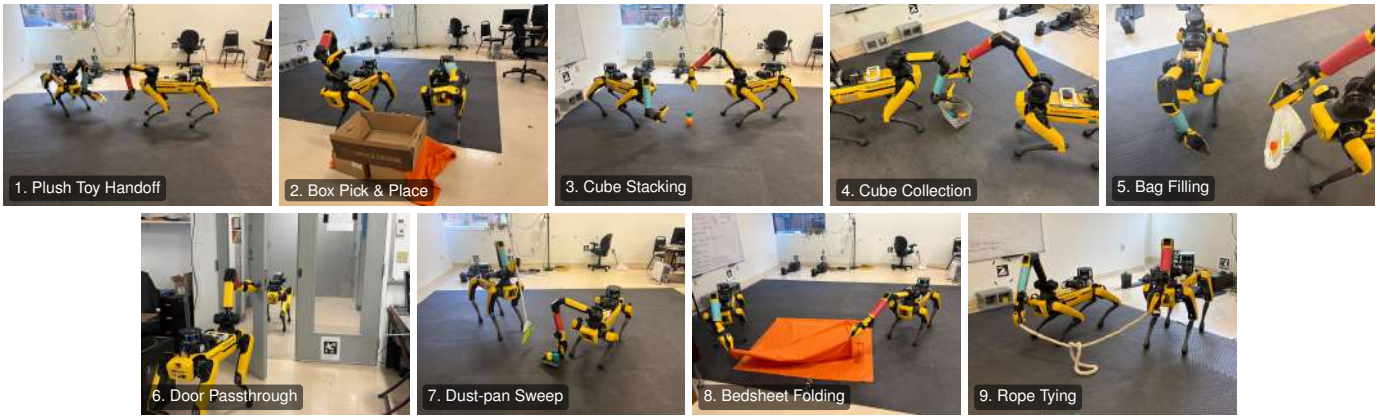

Fig. 4: **Nine evaluation tasks for dual-robot teleoperation.** Tasks require bimanual manipulation (1, 2, 5, 9), address perceptual blind spots (3, 4), or demand spatial context awareness (6, 7, 8). All require coordinated control of two mobile manipulators.

operation. In this mode, base commands are applied to a shared formation pivot so both robots move synchronously and maintain their relative configuration, while arm commands remain independently controlled by the operator's two hands. This separation lets the operator switch between individual single-robot control and synchronized dual-robot motion using the limited inputs of two hand controllers.

B. Mappings

1) Single-robot control: For *arm control*, GHOST maps the hand controller's relative 6-DoF motion to the robot end-effector. When the operator holds the controller trigger, the system registers the current controller and end-effector poses; subsequent controller motion is applied as a relative displacement from that initial pose. This latch mechanism lets operators release and resume control, covering the robot workspace without excessive physical motion. The virtual gripper target immediately shows the commanded pose in VR, while the physical arm follows at approximately 10 Hz through the Boston Dynamics API and IK solver.

For *base control*, thumbstick displacement is sent as translational and rotational velocity commands to the selected robot. For *camera control*, the operator's head pose is tracked relative to a movable virtual base frame. Thumbstick input in *Fly* mode translates or rotates this base frame, allowing the operator to navigate the exocentric scene without needing to physically walk through the workspace.

2) Dual-robot synchronous control: To support coordinated navigation, such as jointly carrying a shared object, we implement a synchronous navigation mode. Upon activation, the initial base poses are recorded as $T_{w\to i}(0) \in SE(2)$ for $i = 1, \ldots, n$ robots. The formation pivot position is the centroid of all robot positions $\mathbf{p}_{\text{pivot}}(0) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{p}_i(0)$ with yaw defined as the circular mean of robot yaw $\theta_i(0)$:

$$\bar{\theta}_{\text{yaw}}(0) = \text{atan2}\left(\frac{1}{n}\sum_{i=1}^{n}\sin(\theta_i(0)), \frac{1}{n}\sum_{i=1}^{n}\cos(\theta_i(0))\right). \quad (1)$$

Each robot's offset from the pivot is recorded as $T_{\text{piv}\to i} = T_{\text{piv}\to w}(0) \cdot T_{w\to i}(0)$. When the operator issues a navigation command $\Delta T \in SE(2)$, the pivot updates to $T_{w\to\text{piv}}(t) = T_{w\to\text{piv}}(t-1) \cdot \Delta T$, and each robot receives its target waypoint $T^{\text{target}}_{w\to i}(t) = T_{w\to\text{piv}}(t) \cdot T_{\text{piv}\to i}$ via the Boston Dynamics API at 10 Hz. Empirically, this maintains sufficient formation accuracy for cooperative carrying over Wi-Fi. Arm controls remain independent across robots in this mode, since this independence aligns with humans' natural bimanual coordination.

VI. Evaluation

A. Evaluation Design

We evaluate GHOST on nine dual-robot tasks (Fig. 4) spanning three categories:

- **Bimanual manipulation** (Tasks 1, 2, 5, 9): Tasks requiring coordinated, simultaneous two-handed manipulation.
- **Perceptual blind spots** (Tasks 3, 4): Tasks that benefit from multi-viewpoint observations to reduce occlusion and improve scene coverage.
- **Spatial context awareness** (Tasks 6, 7, 8): Tasks that require reasoning about the relative spatial configuration of multiple robots and the environment.

Together, these tasks cover a range of coordination scenarios where multi-robot systems are required or provide practical advantages, while still capturing core single-robot capabilities.

We compare to the official Boston Dynamics tablet interface: an egocentric, velocity-control baseline that requires two tablets (Fig. 5) for simultaneous dual-robot teleoperation.

B. Novice User Study

To evaluate system usability with novice operators, we conducted a within-subjects study comparing the interfaces.

Fig. 5: **Tablet interface.** Teleoperating two robots requires using two tablets, typically by switching between them.

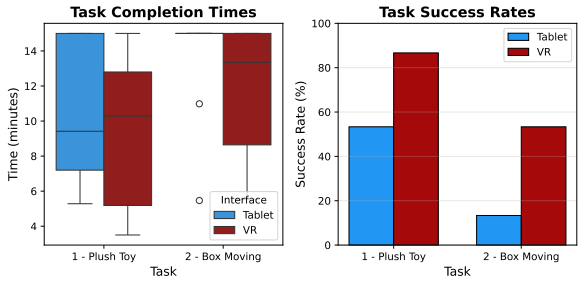

Fig. 6: **Novice task completion times and success rates.**

TABLE III: **Expert task feasibility.**

	GHOST (Ours)	GHOST (RGB-only)	Tablet
1 - Plush Toy Handoff	✓	✓	✓
2 - Box Pick & Place	✓	2/3	✓
3 - Cube Stacking	✓	✓	✓
4 - Cube Collection	✓	✓	✓
5 - Bag Filling	✓	✓	✓
6 - Door Passthrough	✓	2/3	2/3
7 - Dust-pan Sweep	✓	1/3	✗
8 - Bedsheet Folding	2/3	2/3	✗
9 - Rope Tying	✓	✓	✓

Participants: We recruited 15 novice participants (mean age 21.0 ± 2.4 years). Participants had minimal to no prior experience interacting with VR systems (mean rating 1.53 ± 0.74 on a 5-point scale from 1=never used to 5=use daily) or robotic systems (1.87 ± 0.83). Participants reported slightly more exposure to joystick-based games (2.17 ± 1.03).

Procedure: Novice participants attempted two of the nine evaluation tasks: plush toy handoff and box pick-and-place. We selected these tasks because they are simple enough for first-time users while still requiring dual-robot coordination. Before each interface condition, participants received 10–15 minutes of familiarization and practiced picking up a cube from the ground. They then attempted both tasks with both GHOST and the tablet, while we recorded task success and completion time. Interface order was counterbalanced: 8 participants used GHOST first, and 7 used the tablet first.

Metrics: We measured task success rate and completion time. To assess perceived usability, participants also completed the System Usability Scale (SUS) after using each interface.

Results: Our system achieved 1.6× the tablet's success rate on plush toy handoff and 4× on box pick-and-place, with comparable or faster completion times among successful trials (Fig. 6). On the SUS, our interface scored higher than the tablet on average (64.8 ± 19.2 vs. 52.8 ± 22.4 out of 100). This difference was not statistically significant (Wilcoxon signed-rank test, $p = 0.16$, $d_z = 0.34$); we note the low n in this preliminary study. In feedback, participants cited point-cloud reconstruction quality and motion-induced discomfort as the main drawbacks of GHOST, while praising its intuitive control.

C. Expert User Study

We conducted an expert study to evaluate the capability of GHOST on the full set of dual-robot teleoperation tasks.

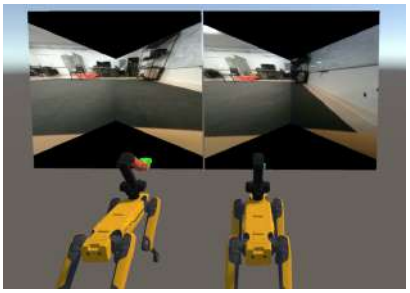

Fig. 7: **Ablation study:** Operator view in RGB-only GHOST.

Interfaces: We tested three interfaces: full GHOST, GHOST with RGB-only camera feeds (Fig. 7), and the official tablet interface (Fig. 5). RGB-only GHOST keeps the VR interface but removes the point-cloud 3D scene, separating the effect of 6-DoF control from 3D reconstruction.

Participants and Procedure: Three authors experienced with all interfaces served as expert participants. Each participant attempted all nine tasks with each interface, performing three primary trials with up to two retries after failures.

Metrics: We measured task success, which defines feasibility, and completion time. If an expert failed more than three of five attempts, the task was deemed infeasible for that expert and excluded from timing comparisons.

Results: Full GHOST achieved the highest task completion reliability, with 26/27 feasible participant-task conditions (Tab. III): 8 of 9 tasks were feasible for all three experts, and bedsheet folding for 2 of 3. The tablet baseline achieved 20/27 conditions; dust-pan sweeping and bedsheet folding were infeasible for all experts.

On the seven tasks where the tablet succeeded, GHOST achieved a 1.47× average per-task speedup over the tablet (Fig. 8). The largest gains were on box pick-and-place (2:48 vs. 7:09), plush toy handoff (1:39 vs. 3:00), cube stacking (4:01 vs. 5:50), and cube collection (2:39 vs. 3:49). GHOST was also slightly faster on bag filling and rope tying. Although GHOST was slower than the tablet on successful weighted-door passthrough trials (Task 6), this task was more often feasible with GHOST (3/3 vs. 2/3).

The RGB-only GHOST ablation achieved 22/27 feasible participant-task conditions. Its task completion reliability and completion times mostly lie between the tablet and full GHOST. RGB-only GHOST failed more on tasks requiring coordinated spatial reasoning, including box pick-and-place, weighted-door passthrough, dust-pan sweeping, and bedsheet folding. This suggests that direct 6-DoF control provides much of the timing benefit, while the reconstructed 3D scene improves reliability and spatial awareness when the point cloud is accurate.

D. Discussion

While preliminary, the results show clear advantages of our approach over RGB-based, joystick-controlled dual-robot teleoperation. We examine the key factors behind these differences: coordinated dual-arm control, direct pose control, predictive visual feedback, and 3D scene representation.

1) Coordinated control: VR maps each robot arm to the operator's corresponding hand, enabling simultaneous dual-arm

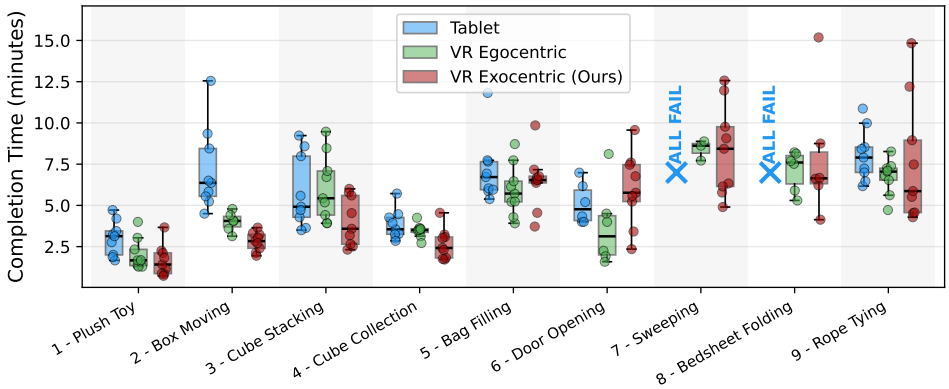

Fig. 8: **Expert task completion times across interfaces.** Full GHOST is faster and more reliable than the tablet baseline on most tasks, while the RGB-only GHOST ablation isolates the effect of direct 6-DoF control without the point-cloud-based exocentric scene. Two tasks were infeasible with the tablet. RGB-only GHOST generally falls between the tablet and full GHOST; when point-cloud quality is poor, it can match or exceed full GHOST.

control without mode switching. This bimanual mapping was important for tightly coordinated tasks: synchronized grasping (Task 2), and bedsheet folding (Task 8). In practice, tablet operators alternated between arms as simultaneous input across two tablets proved too demanding; this was a primary factor in the tablet's failure on Task 8. However, simultaneous control helped little on sequential tasks like weighted-door passthrough (Task 6) or cube collection (Task 4).

2) Intuitive high-precision and rotation control: Direct hand-pose mapping provides an advantage over joystick-based velocity control, particularly for precise alignment (Task 3) and rotation-intensive tasks (Tasks 5, 7). Because VR maps hand pose directly to end-effector pose, fine spatial adjustments are immediate and natural, whereas velocity commands make precise placement and orientation cumbersome.

3) Virtual gripper target reduces perceived latency: The virtual gripper target provides a preview of the commanded pose, reducing perceived latency and improving confidence. However, during tool use (Task 7), it only previews gripper motion, not the tool tip, creating a perceptual mismatch that partially negates this benefit. Despite this, VR still outperforms the tablet for tool use, as velocity-based rotation makes compounded rotations at a tool tip far from the gripper difficult.

4) 3D representation tradeoffs: A 3D point cloud representation enables precise spatial alignment (Task 3) and is a key driver of our system's performance: while the dual-robot setup partially mitigates the viewing-angle limitations of RGB-only systems, operators must still switch between camera views to achieve full alignment, whereas the reconstructed 3D scene provides a unified spatial reference for reasoning about object pose, inter-robot alignment, and spatial constraints. This advantage is more evident in task completion reliability. When comparing RGB-only GHOST versus full GHOST, the 3D scene yielded only marginal gains in completion time but raised completions from 22/27 to 26/27, suggesting it helps operators with coordinated spatial reasoning that RGB views alone make difficult. However, the benefit depends on reconstruction quality: accurate depth completion provided useful spatial cues, while inaccurate reconstruction resulted in reduced efficiency for tasks that involve deformable or cluttered objects (Tasks 5 and 9). This likely drives the high variability: in some cases RGB-only GHOST matched or exceeded full GHOST, indicating that an inaccurate 3D scene can be worse than none at all.

VII. Conclusions

We present GHOST, an exocentric VR system that enables a single operator to perform low-level dual-robot mobile manipulation using only onboard sensing. Across novice and expert studies, GHOST improved success rate and typically also completion time relative to an egocentric tablet baseline, and enabled tasks that were infeasible with the baseline.

Our user study has limitations. First, the expert participants are the developers of the system. Although they are familiar with both interfaces, bias is inevitable. Second, there were few novice participants, and each attempted only two tasks to remain manageable for new users. Thus, this study should be understood as a preliminary assessment of the system's capability. Future larger-scale studies with independent expert operators and a broader pool of novice participants could systematically characterize trends across teleoperation modalities.

Other promising directions include training behavior cloning models from the collected data, scaling to larger multi-robot teams, integrating autonomous assistance for shared control, and improving the 3D reconstruction pipeline to be more precise and consistent for better scene understanding.

Acknowledgments

We thank Calvin Bauer, Are Oelsner, and Janeth Meraz for their work on an earlier version of GHOST [19]. We also thank Ziyan Liu, Shiyu Yan, Arin Idhant, Simon Juknelis, Kamya Raman, Arib Syed, and Automne Petitjean for working on the project, and thank Ivy He, Mingxi Jia, and Gary Lvov for their valuable suggestions on the paper.

References

[1] C. Wickens, A. Vieanne, B. Clegg, A. Sebok, and J. Janes, "Visual attention allocation between robotic arm and environmental process control: Validating the STOM task switching model," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. SAGE Publications Sage CA: Los Angeles, CA, 2015.

[2] N. Cowan, "The magical mystery four: How is working memory capacity limited, and why?" *Current Directions in Psychological Science*, 2010.

[3] C. D. Wickens, "Multiple resources and mental workload," *Human Factors*, 2008.

[4] A. Tung *et al.*, "Learning multi-arm manipulation through collaborative teleoperation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[5] G. Swamy, S. Reddy, S. Levine, and A. D. Dragan, "Scaled autonomy: Enabling human operators to control robot fleets," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020.

[6] J. J. Roldán, E. Peña-Tapia, A. Martín-Barrio, M. A. Olivares-Méndez, J. Del Cerro, and A. Barrientos, "Multi-robot interfaces and operator situational awareness: Study of the impact of immersion and prediction," *Sensors*, 2017.

[7] T. Engelbracht *et al.*, "Spot-On: A mixed reality interface for multi-robot cooperation," *arXiv preprint arXiv:2505.22539*, 2025.

[8] B. Lewis and G. Sukthankar, "Two hands are better than one: Assisting users with multi-robot manipulation tasks," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2011.

[9] Y. Hirao, W. Wan, D. Kanoulas, and K. Harada, "Body extension by using two mobile manipulators," *Cyborg and Bionic Systems*, 2023.

[10] M. Laghi *et al.*, "Shared-autonomy control for intuitive bimanual telemanipulation," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018.

[11] I. Ozdamar, M. Laghi, G. Grioli, A. Ajoudani, M. G. Catalano, and A. Bicchi, "A shared autonomy reconfigurable control framework for telemanipulation of multi-arm systems," *IEEE Robotics and Automation Letters*, 2022.

[12] L. A. Zick, D. Martinelli, A. Schneider de Oliveira, and V. Cremer Kalempa, "Teleoperation system for multiple robots with intuitive hand recognition interface," *Scientific Reports*, 2024.

[13] J. Chen, B. Sun, M. Pollefeys, and H. Blum, "A 3D mixed reality interface for human-robot teaming," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024.

[14] J. I. Lipton, A. J. Fay, and D. Rus, "Baxter's homunculus: Virtual reality spaces for teleoperation in manufacturing," *IEEE Robotics and Automation Letters*, 2017.

[15] D. Whitney, E. Rosen, D. Ullman, E. Phillips, and S. Tellex, "ROS reality: A virtual reality framework using consumer-grade hardware for ROS-enabled robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018.

[16] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-TeleVision: Teleoperation with immersive active visual feedback," in *Conference on Robot Learning*. PMLR, 2025.

[17] I. Chuang, A. Lee, D. Gao, M.-M. Naddaf-Sh, and I. Soltani, "Active vision might be all you need: Exploring active vision in bimanual robotic manipulation," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025.

[18] H. Xiong, X. Xu, J. Wu, Y. Hou, J. Bohg, and S. Song, "Vision in action: Learning active perception from human demonstrations," in *Conference on Robot Learning (CoRL)*. PMLR, 2025.

[19] C. Bauer, J. Meraz, A. Oelsner, J. Tompkin, and S. Tellex, "GHOST in the robot: Virtual reality teleoperation for mobile manipulation," 2024, MSc Project.

[20] M. Wilder-Smith, V. Patil, and M. Hutter, "Radiance fields for robotic teleoperation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024.

[21] X. Cheng, P. Wang, and R. Yang, "Learning depth with convolutional spatial propagation network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[22] A. Eldesokey, M. Felsberg, and F. S. Khan, "Confidence propagation through CNNs for guided sparse depth regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[23] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018.

[24] J. Tang, F.-P. Tian, B. An, J. Li, and P. Tan, "Bilateral propagation network for depth completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[25] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2021.

[26] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec *et al.*, "DINOv2: Learning robust visual features without supervision," *Transactions on Machine Learning Research*, 2024.

[27] L. Yang *et al.*, "Depth Anything V2," in *Advances in Neural Information Processing Systems*, 2024.

[28] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[29] H. Lin *et al.*, "Prompting Depth Anything for 4K resolution accurate metric depth estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[30] Y. Zhang, X. Guo, M. Poggi, Z. Zhu, G. Huang, and S. Mattoccia, "CompletionFormer: Depth completion with convolutions and vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[31] M. Viola *et al.*, "Marigold-DC: Zero-shot monocular depth completion with guided diffusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[32] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "VGGT: Visual geometry grounded transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[33] N. Keetha *et al.*, "MapAnything: Universal feed-forward metric 3D reconstruction," in *International Conference on 3D Vision (3DV)*. IEEE, 2026.

[34] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile ALOHA: Learning bimanual mobile manipulation using low-cost whole-body teleoperation," in *Conference on Robot Learning*. PMLR, 2025.

[35] C. C. Kemp, C. D. Anderson, H. Nguyen, A. J. Trevor, and Z. Xu, "A point-and-click interface for the real world: laser designation of objects for mobile manipulation," in *Proceedings of the 3rd ACM/IEEE international conference on human robot interaction*, 2008.

[36] M. Gu, E. Croft, and A. Cosgun, "AR point & click: An interface for setting robot navigation goals," in *International Conference on Social Robotics*. Springer, 2022.

[37] S. Dass *et al.*, "TeleMoMa: A modular and versatile teleoperation system for mobile manipulation," *arXiv preprint arXiv:2403.07869*, 2024.

[38] D. Honerkamp, H. Mahesheka, J. O. von Hartz, T. Welschehold, and A. Valada, "Whole-body teleoperation for mobile manipulation at zero added cost," *IEEE Robotics and Automation Letters*, 2025.

[39] S. Macciò, M. Shaaban, A. Carfì, and F. Mastrogiovanni, "Kinesthetic teaching in robotics: a mixed reality approach," in *2024 33rd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2024.

[40] T. Yang *et al.*, "MoMa-Force: Visual-force imitation for real-world mobile manipulation," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023.

[41] D. Ryu, J.-B. Song, C. Cho, S. Kang, and M. Kim, "Development of a six DOF haptic master for teleoperation of a mobile manipulator," *Mechatronics*, 2010.

[42] V. Dhat, N. Walker, and M. Cakmak, "Using 3D mice to control robot manipulators," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024.

[43] S. K. Long and J. P. Bliss, "The effect of control device on performance in a robotic arm task," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. SAGE Publications Sage CA: Los Angeles, CA, 2016.

[44] T. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," in *Robotics: Science and Systems (RSS)*, 2023.

[45] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "GELLO: A general, low-cost, and intuitive teleoperation framework for robot manipulators," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[46] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: Learning a robotic hand imitator by watching humans on YouTube," in *Robotics: Science and Systems (RSS)*, 2022.

[47] Y. Qin *et al.*, "AnyTeleop: A general vision-based dexterous robot arm-hand teleoperation system," in *Robotics: Science and Systems (RSS)*, 2023.

[48] R.-Z. Qiu *et al.*, "Humanoid policy ~ human policy," *arXiv preprint arXiv:2503.13441*, 2025.

[49] A. Posadas-Nava, A. Carrasco, and R. Linares, "BEAVR: Bimanual, multi-embodiment, accessible, virtual reality teleoperation system for robots," *arXiv preprint arXiv:2508.09606*, 2025.

[50] A. Mandlekar *et al.*, "RoboTurk: A crowdsourcing platform for robotic skill learning through imitation," in *Conference on Robot Learning*. PMLR, 2018.

[51] T. Lin *et al.*, "Learning visuotactile skills with two multifingered hands," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025.

[52] A. Iyer *et al.*, "OPEN TEACH: A versatile teleoperation system for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2025.

[53] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot Operating System 2: Design, architecture, and uses in the wild," *Science Robotics*, 2022.

APPENDIX

A. Task Descriptions

#	Name	Description	Initial Scene	Final Scene
1	Plush Toy Handoff	Initially, a white plush toy is placed in front of the robots. The operator needs to pick up the toy with one hand and hand it over to the other robot.		
2	Large Box Picking and Placing	Initially, a large box is placed in front of the two robots. The box is too large that it has to be carried by two robots. The operators need to pick up the box, carry it near the platform covered with orange cloth, and place it on top of the platform.		
3	3-Cube Stacking	Initially, three cubes are placed in a line in front of the robot. The operator needs to stack the three cubes in one stack in whatever order they deem best.		
4	Collect 3 Scattered Cubes into the Basket	Initially, a basket and three cubes are scattered around in the scene in front of the robot. The operator is required to navigate and find all three cubes and place them into the basket in the center.		
5	Opening and Holding Bags While Filling Them with 2 Cubes	Initially, one white plastic bag and two cubes are placed in front of the robots. The plastic bag is almost flattened, so it requires one robot to hold up the bag in order to have the opening big enough to fit the gripper and the cube. Thus, the operator will have to use one robot to hold up the bag, while the other robot looks around and searches for the cubes, and then places them into the held-up bag.		
6	Weighted Door Opening and Pass Through	The robots need to open a weighted door and make one robot go through the door. The door is too heavy for the Boston Dynamics Spot's automatic door-opening function, which will not properly open the door and let the robot through.		

TABLE IV: Task Descriptions for each task (Part 1)

#	Name	Description	Initial Scene	Final Scene
7	Dust Pan Sweep	Initially, two cubes, one broom, and one dust pan are placed in front of the robot. The operator needs to control the robot to use the broom to sweep the two cubes into the dust pan.		
8	Bedsheet Folding	Initially, a very large, long sheet is placed in front of the robots. The operator needs to fold it in half. Note that since the sheet is so large, the robot will have to move the base in order to properly fold the sheet.		
9	Rope Tying 1 Knot	Initially, a thick rope is placed in front of the two robots. The operator needs to control the robot and tie a knot on the rope.		

TABLE V: Task Descriptions for each task (Part 2)