LaserDex: Improvising Spatial Tasks Using Deictic Gestures and Laser Pointing for Human–Robot Collaboration in Construction

Sungboo Yoon, S.M.ASCE[1]; Moonseo Park, M.ASCE[2]; and Changbum R. Ahn, A.M.ASCE[3]

Abstract: In the context of the unstructured and fast-changing construction environment, the adaptability of robots to human improvisation is crucial. However, existing construction robots are limited in understanding spatial goals communicated spontaneously in the workplace and require substantial human-user training. Incorporating deictic gestures into human–robot interfaces holds promise for enhancing the intuitive operation of construction robots and for on-site collaboration. However, applying deictic gestures in precision-demanding construction tasks presents challenges because in a large-scale work environment, humans have limited accuracy in pointing at objects at a distance. This study introduces LaserDex, a novel interface for distant spatial tasking based on a global-to-local identification approach, in which the human first guides the robot toward the general area of the task using a deictic gesture and then indicates the precise area by dynamically using a laser pointer. Our user study with 11 participants demonstrated that LaserDex can achieve an intersection over union (IoU) of 0.830 when outlining a rectangular drywall opening (compared with an IoU of 0.514 for the baseline, a handheld controller), and an 11.4-mm distance error of the center of the estimated rectangle compared with the center of the targeted rectangle. The findings of this study underscore the potential of LaserDex to seamlessly combine intuitive human–robot interaction with precise robot operations. **DOI: 10.1061/JCCEE5.CPENG-5715.**

Author keywords: Human–robot collaboration; Spatial tasking; Deictic gestures; Dynamic laser pointing.

Introduction

Construction jobsites present unique challenges for robots due to the highly complex and dynamic nature of the work environment (Zhang et al. 2023). During on-site construction, human workers inevitably make deviations from the original planned design (Jenny et al. 2020). These as-built deviations may lead to significant errors or failures during robot operation (Wang et al. 2021; Yin et al. 2022). Therefore, in the field, humans must make improvisations based on such unexpected situations. For example, when cutting drywall, human workers may decide on the most suitable positions and/or angles of the cut. The robot's role is to adapt to the new task plan and execute precise cuts accordingly. This form of human–robot interaction (HRI), in which a human instructs tasks based on specific spatial locations or areas in the environment, is referred to as spatial tasking (Yuan et al. 2019). However, current construction robots are limited in their ability to understand spatial instructions and goals on the worksite (in situ) (Yoon et al. 2023). Moreover, user interfaces, such as handheld controllers, require extensive training and operation time for users, which can be impractical for construction workers who typically are not experts in robotics (Liang et al. 2022).

Utilizing deictic gestures (e.g., pointing gestures) in the human–robot interface has the potential to provide a more effective and intuitive means of operating construction robots. Pointing is a natural mode of communication between construction workers and is preferred by novice users when interacting with robots, because pointing allows them to communicate with the robots without the need for understanding the robots' control mechanisms (Wang et al. 2023). Several approaches have been proposed in the literature to apply deictic gestures for spatial tasking. Specifically, deictic gestures have been employed to indicate a spatial location to which a mobile robot should navigate (Ikeda et al. 2023; Ürkmez and Bozma 2022); to navigate an unmanned aerial vehicle (UAV) (Gromov et al. 2019; Medeiros et al. 2021; Yuan et al. 2019); and to select objects for robotic manipulation (Čorňák et al. 2021; Hu et al. 2022; Strazdas et al. 2022). Some studies have attempted to recognize the user's intended target or action via their deictic gaze or head pose (Crocher et al. 2021; Yang et al. 2023). However, despite the advantages of deictic gestures, current gesture interfaces may not be suitable for construction tasks that require a high level of precision. Within large-scale environments, these interfaces cover only a restricted area of the entire workspace and have limited accuracy in estimating the spatial locations of distant targets (Yoon et al. 2023).

Therefore, this study presents a human–robot interface that leverages deictic gestures to spatially task a construction robot in situ with dynamic laser pointing. This study extends a previous study that used static laser pointing that recognized spatial goals as discrete points (Yoon et al. 2024); the present study proposes methods to accurately infer the distant spatial goals indicated by humans through the estimation of dynamic laser spots (e.g., continuous trajectories). The proposed interface, LaserDex was compared with a handheld controller as the baseline. Our findings show that

[1] Ph.D. Student, Dept. of Architecture and Architectural Engineering, Seoul National Univ., Seoul 08826, Republic of Korea. ORCID: https://orcid.org/0000-0003-4997-5792. Email: yoonsb24@snu.ac.kr

[2] Professor, Dept. of Architecture and Architectural Engineering, Seoul National Univ., Seoul 08826, Republic of Korea. Email: mspark@snu.ac.kr

[3] Associate Professor, Dept. of Architecture and Architectural Engineering, Institute of Construction and Environmental Engineering, Seoul National Univ., Seoul 08826, Republic of Korea (corresponding author). ORCID: https://orcid.org/0000-0002-6733-2216. Email: cbahn@snu.ac.kr

Note. This manuscript was submitted on September 12, 2023; approved on December 19, 2023; published online on March 9, 2024. Discussion period open until August 9, 2024; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801.

LaserDex leverages both intuitive human–robot interaction and the precise operation of construction robots.

Background

Human–Robot Interfaces in Construction

The human–robot interface is defined as the means for information and action exchanges between a human and a robot during HRI (ISO 2021). In collaborative systems, the successful completion of tasks relies on mutual interactions between humans and robots (Castro et al. 2021). Although these interactions share similarities with human–computer interactions (HCIs), the interfaces used in HRI differ in that they mediate an active relationship between a human and a physically situated agent, i.e., a robot (Murphy and Tadokoro 2019).

Achieving effective communication and interaction between workers and robots through human–robot interfaces is essential for seamless HRI in construction. Currently, there are two types of interfaces developed for construction robots: interfaces for programming, and interfaces for real-time control.

Programming involves writing a set of instructions or code that the robot can execute autonomously. The most widely used programming interface for industrial robots is a teach pendant, which is a handheld device that allows users to manipulate a robot's movements using a joystick while recording its trajectory (Ong et al. 2020). Teach pendants were used in the early development of construction robots, such as a robotic glazing system (Lee et al. 2013). However, a significant challenge with this interface is that the robot often possesses more degrees of freedom than the joystick can accommodate, leading to a more complex programming process that may require skilled operators (Ong et al. 2020).

Previous work has explored the use of emerging technologies in developing interfaces for programming robotic tasks in construction. These technologies include virtual reality (VR), augmented reality (AR), and mixed reality (MR), which have been investigated to provide interactive task visualization, planning, and execution (Amtsberg et al. 2021; Mitterberger et al. 2022; Wang et al. 2021). Furthermore, large language models (LLMs), such as the GPT series (Brown et al. 2020), LLaMA series (Touvron et al. 2023a, b), and T5 (Raffel et al. 2019) have demonstrated capabilities in processing natural language, making it possible for users to program construction robots using conversational language (Park et al. 2023; You et al. 2023). These studies suggest that integrating these technologies can offer the design of more-intuitive and user-friendly programming interfaces.

More recently, robot programming has been extended to the concept of learning from demonstration (LfD). LfD is a robot learning method that enables a robot to acquire new skills by imitating observed demonstrations from human experts, instead of relying on traditional robot programming (Argall et al. 2009; Liang et al. 2022). Expert demonstrations can be collected through virtual simulations (Huang et al. 2023; Liang et al. 2020, 2022) or by physically guiding the robot's movements (kinesthetic guidance) (Kramberger et al. 2021). However, despite the potential of LfD, its interfaces do not provide in situ tasking capabilities (Mitterberger et al. 2022). Presently, LfD applications in the construction domain are limited to replicating demonstrated trajectories. This is due to the scarcity and high costs of obtaining task-specific demonstration data and the difficulties in training the LfD models (Villani et al. 2018).

In contrast to programming interfaces, interfaces for real-time control, navigation, and teleoperation are designed for the operation of robots in the moment, whether in remote or colocated settings (Suzuki et al. 2022). Similar to programming interfaces, handheld controllers such as tablet PCs (Okishiba et al. 2019) and joystick controllers (Asadi et al. 2018; Koh et al. 2021) are used commonly in the field, as well as in research applications. Those controllers have been extended to virtual environments, for remote operation of demolition robots (Adami et al. 2022) or pipe skid maintenance robots (Zhou et al. 2023).

Additionally, haptic devices offer an effective interface by providing workers with kinesthetic and tactile haptic feedback, which is proving to be the most effective method for teleoperation among different types of sensory feedback in remote environments (Zhu et al. 2021). Although still relatively new in construction applications, haptic devices have shown promise in teleoperating robots for tasks, including welding (Ye et al. 2023), joint sealing (Brosque et al. 2021), and valve operation (Zhu et al. 2021). Some researchers have proposed a brain–computer interface (BCI) for leveraging electroencephalography (EEG) from a worker's brain activity to control an unmanned ground vehicle (UGV) robot's movements in the context of a masonry task (Liu et al. 2021).

However, for spatial tasking on the jobsite, current human–robot interfaces in construction have several limitations. First, complex interfaces for programming make flexible and immediate robot control difficult. Consequently, even minor alterations in the working conditions often require time-consuming reprogramming efforts (Huang et al. 2023). Given the nonstationary and ad hoc nature of construction projects, programming robotic construction tasks becomes impractical (Liang et al. 2022; Zhang et al. 2023). Second, the real-time control of robots with multiple degrees of freedom using interfaces such as handheld controllers is difficult (Wang et al. 2021). It requires extensive training for workers or even additional specialized operators, potentially increasing labor demands (Liang et al. 2022). Additionally, teleoperated robots often lack local accuracy due to the difficulties in obtaining situational awareness solely through images from robots (Lee and Brell-Cokcan 2021), leading to inaccuracies in conveying task instructions to construction robots.

Deictic Gesture–Based Spatial Tasking

For effective HRI in construction, it is important to enable intuitive control of robots by end users (operators) while also ensuring accurate robot perception of the human actions and commands (Al et al. 2020). Different strategies for user-friendly and intuitive HRI, including voice, gesture, and touch interfaces, have been studied (Al et al. 2020). Among these interaction modes, deictic reference, whether in the form of deictic gaze, deictic gesture, or deictic language, serves as a valuable method for establishing joint attention within a shared human–robot workspace for cooperative tasks, particularly in complex and noisy work environments (Stogsdill et al. 2021). Furthermore, deictic reference complements verbal descriptions in spatial guidance or when referring to specific objects (Jirak et al. 2021).

One natural form of deictic reference is deictic gaze, in which sustained eye contact with a target indicates interest. Previous research has focused on precise three-dimensional (3D) gaze estimation for object selection (Krupke et al. 2018; Shi et al. 2021; Wöhle and Gebhard 2021) and gaze-based intention recognition for manipulation tasks (Aronson et al. 2021; Yang et al. 2023). However, the use of gaze in spatial tasking is prone to errors due to inherent noise and outliers in eye-pointing data (Yang et al. 2023). Moreover, in far-range interactions, such as those encountered in construction environments, accurately detecting eye positions becomes challenging (Ürkmez and Bozma 2022). To address this limitation, head pose, which combines head position and face direction, has been used as a substitute for gaze-based indicators of a person's intention (Bovo et al. 2022).

However, it often is difficult to recognize the head pose of the user, especially compared with pointing (Rosen et al. 2020). Pointing-based spatial tasking presents several advantages, especially in industrial applications, because it does not require complex spatial reasoning, allows for free-hand interaction, requires minimal training, and relies on minimal infrastructure such as a red-green-blue-depth (RGB-D) camera to track arm orientation (Guzzi et al. 2022). Therefore, pointing gestures are an attractive interaction mechanism for humans colocated with robots. (Braeuer-Burchardt et al. 2020) developed a system using finger pointing for communication with a robot-guided measurement system in industrial quality checks. (Medeiros et al. 2021) proposed a human–drone interaction (HDI) method using pointing gestures estimated from a monocular camera for indicating a target in first-response emergency scenarios. (Mahmud et al. 2022) developed a system that uses pointing and dynamic commanding gestures that are robust to changes in lighting conditions and the environment to navigate mobile robots.

Compared with two-dimensional (2D) environments, selection in 3D environments is more challenging because of the higher number of degrees of freedom (DoFs) inherent in the task (Weller et al. 2021). Moreover, in cluttered environments or when dealing with moving targets, the lack of precision and fatigue can be an issue (Weller et al. 2021). Extensive research in robotics and computer science into how humans point at objects at a distance have revealed that humans have limited accuracy when pointing at remote objects using their hands or tools. Yoon et al. (2023) adopted a vision-based deictic gesture recognition method and evaluated the performance of deictic gestures in spatial communication tasks. Within these tasks, Yoon et al. manipulated factors such as target configuration, target distance, and relative positioning of humans and robots. To evaluate the robot's accuracy in classifying correct target panels and other panels, they used the $F1$-score which considers both true positives and misclassifications (false positives and false negatives). The results of the spatial communication tasks showed that in a large-scale environment with a complex target configuration, the robot's $F1$-score significantly decreased to a minimum of 0.404, whereas humans ranged from 0.730 to 0.956. Jirak et al. (2021) observed that computer-vision approaches exhibited a higher rate of incorrect object selection, reaching a 22.96% miss rate. This occurred as the level of ambiguity increased, with multiple objects placed on a table with overlapping arrangements. (Medeiros et al. 2021) achieved a 0.58 $F1$-score using a pointing gesture interface for drones at a maximum distance of 25 m between the drone and the target building. Notably, this problem is not limited to distant targets. In a study by Ürkmez and Bozma (2022), the estimation accuracy of pointing at close-range objects (0.9–1.5 m) fixed to a table reached only 77.3%.

To address this problem, several studies have explored the use of laser pointing as a complementary or alternative means to gesture-based spatial tasking for robotic applications such as wheelchair-mounted robotic arms (Zhong et al. 2019) and mobile robots (Sprute et al. 2019). The authors suggest that using a laser pointer makes interaction easier because a laser indicates a position on the surface of the environment, without having to consider the robot's perspective. In addition, the laser spot on the surface provides visual feedback to the human user, enhancing their understanding of the interaction. Laser pointing is particularly suitable for spatial tasking in construction, because laser pointers are used commonly by workers to indicate hard-to-reach areas, such as high ceilings or tight corners. Furthermore, static laser pointing using line laser levels has wide application in construction, such as floor leveling, distance measurement, and plumbing alignment (Fig. 1). However, compared with static laser pointing, dynamic laser pointing presents unique challenges for spatial task interpretation, because hand tremors and unsteady movements can cause jitters and variability in the laser trajectory. Our study addresses these challenges by implementing smoothing techniques to reduce noise and a shape-fitting algorithm to improve the accuracy of trajectory estimation for spatial tasking in construction environments.

Methodology

This study developed a user-friendly interface that allows for intuitive HRI in the context of real-time, on-the-job (in situ) spatial tasking. Specifically, this study considers a scenario in which a human worker interacts with a robot to specify a target area on a drywall ceiling, tasking the robot to make an opening by cutting along the outlined area. Moreover, this interaction occurs as the task is designed spontaneously by a human worker rather than being predetermined in the 3D model [e.g., Building Information Modeling (BIM)], or because, in some cases, the 3D model might not be available. This process uses a multimodal interaction: a human worker points with their hand and with a laser pointer and speaks into a microphone. Along with deictic gestures and laser pointing, the proposed interface, LaserDex, incorporates speech as an auxiliary input mode, which functions as a cue for the robot to trigger movements. The proposed approach for target localization consists of two complementary phases: (1) global guidance, and (2) local refinement.

Global guidance aims to identify the human's region of interest (RoI) within the environment. During this phase, the human indicates their RoI by performing deictic pointing; then the system

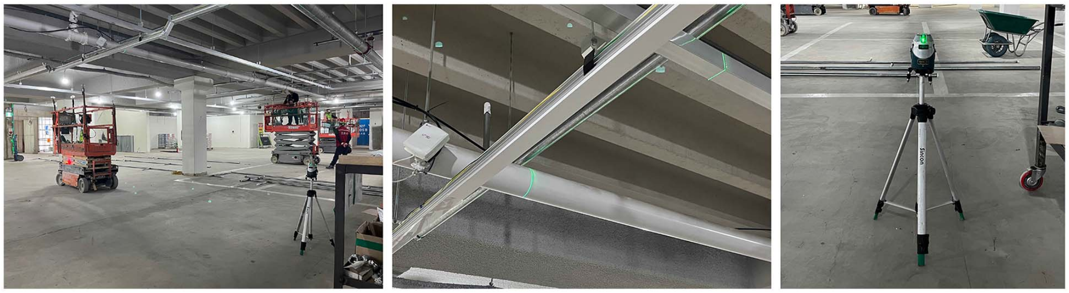

Fig. 1. Alignment of plumbing using a 360° line laser level in a basement construction site. Image courtesy of Hyundai E&C. (Images by Sungboo Yoon.)

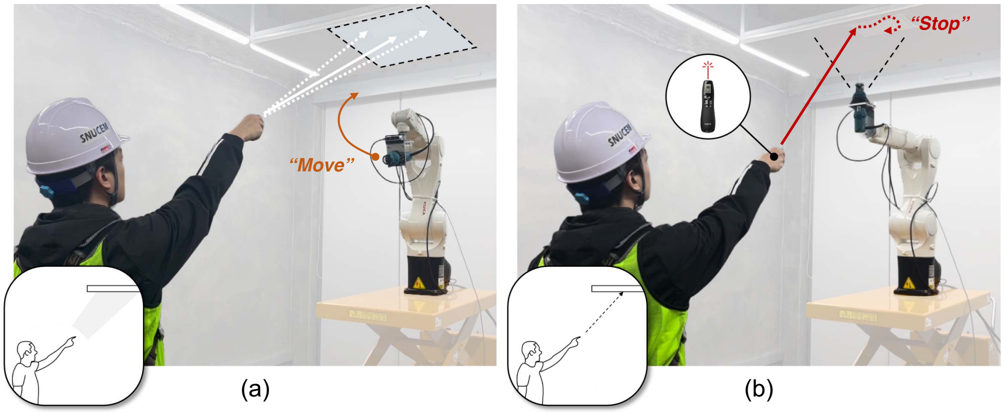

Fig. 2. Proposed global-to-local identification approach for spatial tasking in a drywall cutting scenario: (a) global guidance to indicate the general area; and (b) local refinement to precisely specify the cutout shape. (Images by Sungboo Yoon.)

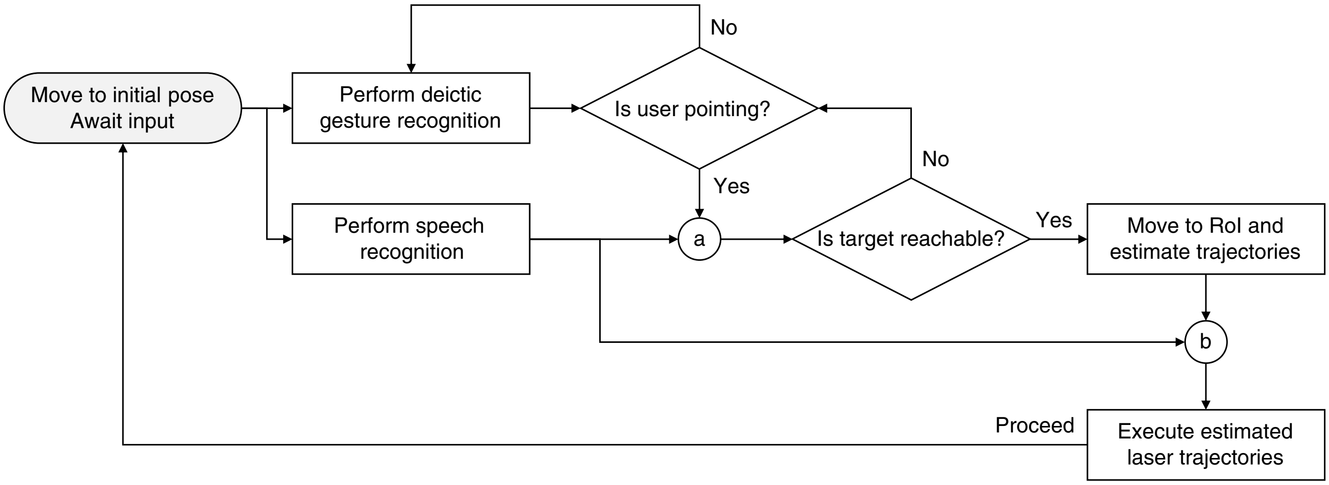
Move to initial pose
Await input
Perform deictic gesture recognition
Is user pointing?
No
Yes
Perform speech recognition
a
Is target reachable?
No
Yes
Move to RoI and estimate trajectories
b
Proceed
Execute estimated laser trajectories

Fig. 3. Control flow of LaserDex, with verbal commands "Move" (a) and "Stop" (b).

estimates the approximate location by recognizing the pointing direction through the robot vision [Fig. 2(a)]. The control flow in Fig. 3 illustrates how the proposed approach is implemented. The interaction starts when the robot identifies the deictic gesture by the human user. If the user is pointing, the approximate location is estimated, which is used for reachability analysis. To verify the target's reachability, a task limit sphere (Khatib et al. 2021) is implemented. This sphere represents a Cartesian boundary surrounding the robot using a virtual sphere (Khatib et al. 2021). The center of the sphere is positioned at the robot's second joint (the shoulder joint), with a radius equivalent to the total length from the second joint to the tip of the drywall cutting tool (Khatib et al. 2021). If the target falls within the task limit sphere, the estimated spatial location is referenced to guide the robot towards the human's RoI within the environment. Here, the user utters the command "move" to initiate the movement.

Local refinement aims to complement global guidance by collecting detailed spatial goals within the RoI. During this phase, the human explicitly gives the robot a spatial task by drawing trajectories using a laser pointer [Fig. 2(b)]. Local refinement allows the robot to estimate the laser trajectories precisely within the robot's field of view (FoV). By commanding "stop," the user directs the robot to stop laser tracking and to predict trajectories and execute the cutting of the drywall along the predicted path. Upon completing the task, the robot returns to its initial position and awaits further interaction input.

The control flow is executed based on the system architecture illustrated in Fig. 4. The system has five main components: environment mapping, deictic gesture recognition, laser pointing estimation, speech recognition, and robot motion planning. The process begins by simultaneously collecting the red-green-blue color (RGB) images I_{RGB} and the depth images I_{Depth} from a single robot-mounted RGB-D camera. Using the image input, depending on the interaction method, the deictic gesture recognizer and laser pointing estimator outputs the desired robot pose $\mathbf{p}_{\mathrm{desired}}$. During this process, the user's audio input is provided by a Bluetooth microphone. The speech recognizer then processes this audio input, producing a text-based command. Finally, the robot motion planner issues position commands to the robot server and receives real-time updates of the six robot joint states. The entire system is integrated with the Robot Operating System (ROS), with each component implemented as an individual ROS node. The detailed methods and algorithms utilized for developing each component are described in the following sections.

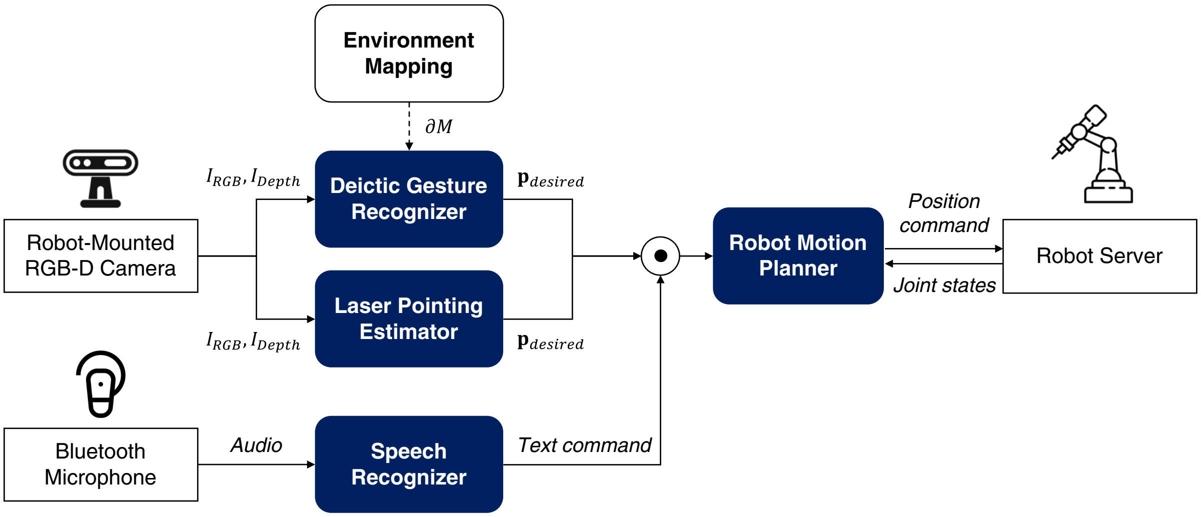

Fig. 4. System architecture for LaserDex.

Environment Mapping

In the process of environment mapping, a dense 3D global map is created to represent the surfaces of the robot's unknown surroundings. This map is built based on both RGB and depth images simultaneously captured from an RGB-D camera mounted on the robot end effector. First, a 3D point cloud of the environment is collected by following a predefined path. To collect the 3D point cloud, we selected real-time appearance-based mapping (RTAB-Map) (Labbé and Michaud 2019), a well-known visual simultaneous localization and mapping (SLAM) algorithm. In contrast to other visual SLAM methods [e.g., ORB-SLAM2 (Mur-Artal and Tardos 2017)], RTAB-Map constructs a dense 3D map of indoor environments, a feature that is particularly beneficial for our study's need for precise interaction within the environment (de Jesus et al. 2021). RTAB-Map is an RGB-D graph-based SLAM approach based on an incremental appearance-based loop closure detector (Labbé and Michaud 2019). The loop closure detector uses a bag-of-words approach to determinate the likelihood that a new image comes from a previous location or from a new location. When a loop closure hypothesis is accepted, a new constraint is added to the map's graph, and a graph optimizer then minimizes the errors in the map. A memory management approach is used to limit the number of locations used for loop closure detection and graph optimization, so that real-time constraints on large-scale environments are always respected (Labbé and Michaud 2019).

Next, the triangle meshes are generated from the dense point cloud using a Poisson surface reconstruction method (Kazhdan et al. 2006). RTAB-Map's ability to produce such detailed 3D maps enables the generation of high-quality triangle meshes that accurately represent the surfaces in the environment. These precollected meshes are saved as a global map, which can be used to estimate spatial locations on the surface of the environment, as indicated by the human operator.

Deictic Gesture Recognition

The ability to accurately estimate pointing direction is crucial for spatial referencing, but there is currently no single, widely accepted method for estimating pointing direction (Ürkmez and Bozma 2022). Researchers have proposed several pointing ray casting techniques, classified based on the origin of the ray (Mayer et al. 2018; Strazdas et al. 2022). These techniques include (1) head ray cast (HRC), which defines the ray using the orientation of the head; (2) eye–finger ray cast (EFRC), which uses the dominant eye position as the root of the ray; (3) forearm ray cast (FRC), which uses the position of the elbow of the pointing arm as the root; and (4) shoulder–finger ray cast (SFRC), which uses the shoulder position as the root. In this study, to enhance computational efficiency for HRI in situ, we employ the position of the wrist as a proxy for the position of the index finger. Likewise, considering the practical constraints associated with using eye gaze in construction environments, as discussed in Section "Deictic Gesture–Based Spatial Tasking," we substitute the position of the head for the dominant eye position used in EFRC. We apply four modified ray casting techniques, namely shoulder–wrist ray cast (SWRC), head–wrist ray cast (HWRC), elbow–wrist ray cast (EWRC), and head ray cast (Fig. 5).

Yoon et al. (2021, 2023) demonstrated that vision-based deictic gesture recognition using SWRC can potentially be applied to HRI for spatial referencing in large-scale environments. Fig. 5 shows the normalized inclination angles and distance errors by four ray casting techniques during the interaction period. The three inclination angles represent the angles between the ray vector and the x-, y-, and z-axes. SWRC provides a relatively stabilized ray vector, resulting in consistent and precise pointing estimation (Fig. 5). These results are in line with those of previous studies that have shown that the pointing estimation using SWRC offers better accuracy than other methods (Jevtić et al. 2015; Strazdas et al. 2022).

The center of Fig. 6 shows the data flow diagram of the deictic gesture recognizer. The positions of the user's arm joints (i.e., shoulder, elbow, and wrist) are estimated in real-time through 3D human pose estimation (Zimmermann et al. 2018). This approach leverages robust human key-point detectors for RGB images and incorporates depth information for lifting into 3D (Zimmermann et al. 2018). The prediction of whether the user is pointing is made by applying a threshold to the angle of the elbow joint, θ_e, as described by Yoon et al. (2023). We set the threshold to 45°, i.e., $\theta_e < 45°$ indicates pointing. Assuming that it is known that the gesture ray intersects with a triangle from the mesh with index, i, the intersection point P can be estimated as follows:

$$P = P_s + \frac{\mathbf{n}_i \cdot (V_0 - P_s)}{\mathbf{n}_i \cdot \mathbf{v}_{sw}} \cdot \mathbf{v}_{sw} \qquad (1)$$

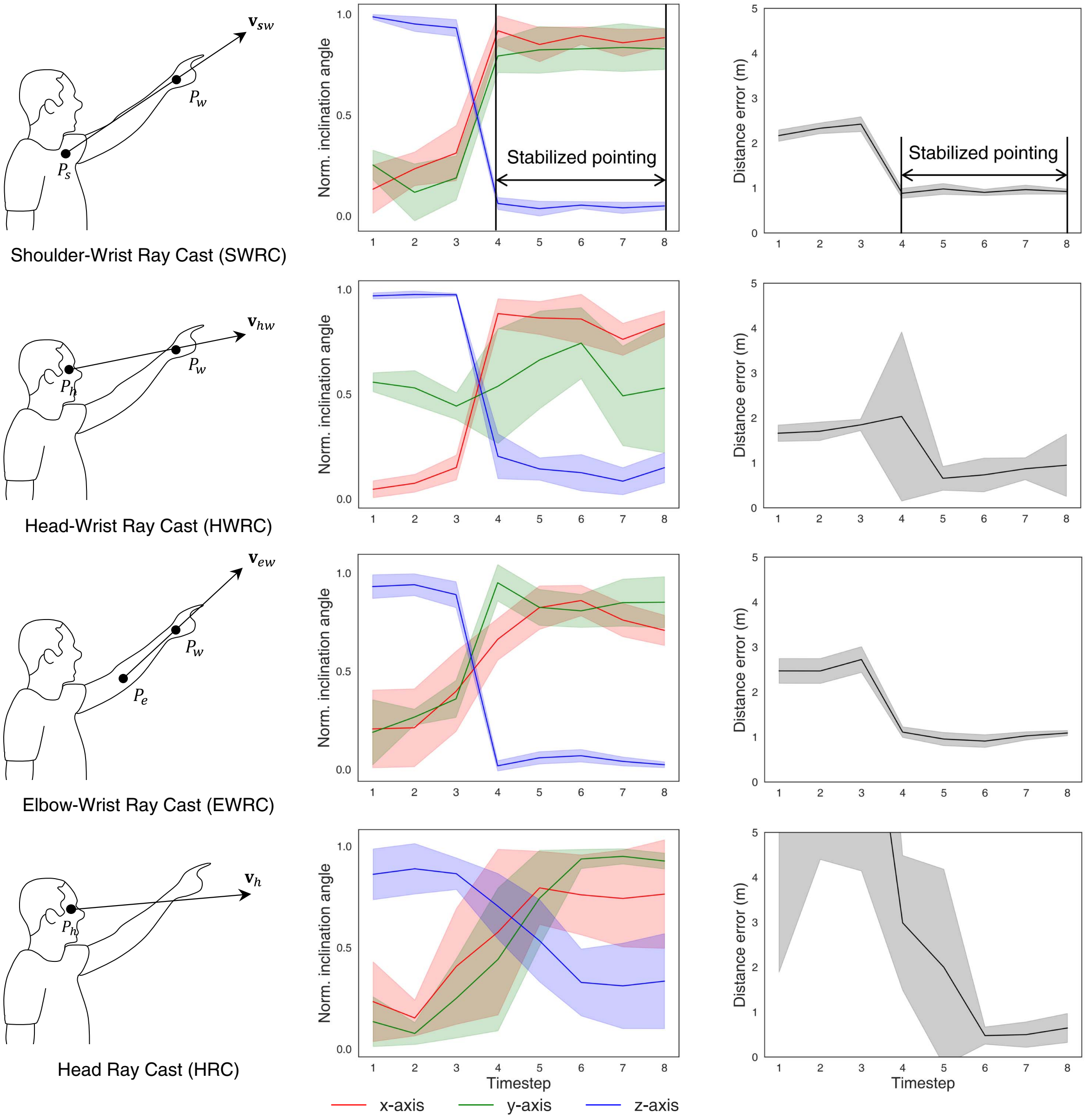

Fig. 5. Normalized inclination angles and distance errors of different ray casting techniques.

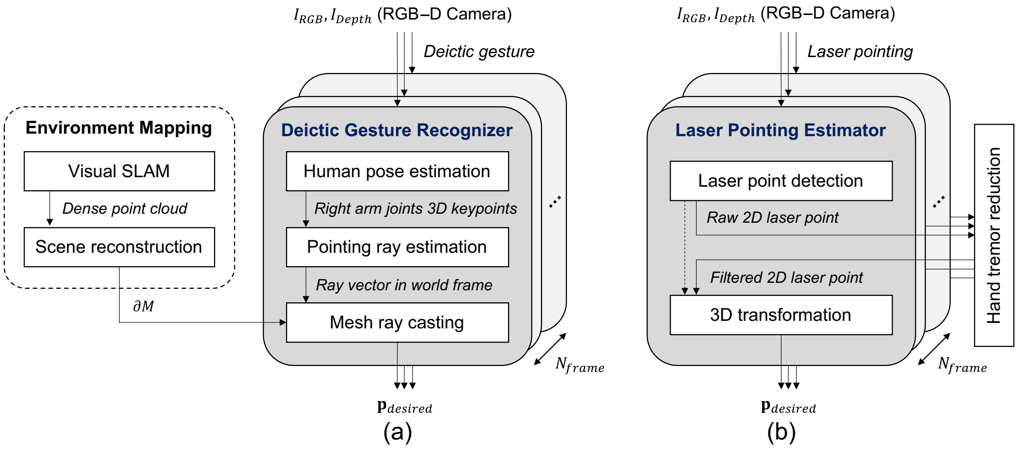

Fig. 6. (a) Data flow diagram for deictic gesture recognizer; and (b) laser pointing estimator.

where P_s = origin of the ray, which is the position of the user's shoulder in SWRC; $\mathbf{v}_{sw}$ = ray vector from the shoulder to the wrist joint; $\mathbf{n}_i$ = normal vector of the intersection triangle with index i; and V_0 = one of the vertices of the intersection triangle. The intersection point, P, represented in the world coordinate system, is regarded as the target location and is used to direct the robot toward the detection pose. To prevent collisions with the workpiece, we set an offset of 30 cm in the direction of $\mathbf{n}_i$.

Speech Recognition

Verbal cues are used to command the robot to move or stop. To identify these verbal cues from the user's spoken words, a speech recognition algorithm was implemented using the Google Speech-to-Text API. This API takes audio input and converts it into text transcriptions using deep learning algorithms.

Laser Pointing Estimation

When the robot transitions its pose to face the human's RoI, the human gives detailed spatial goals for completing the task by using dynamic laser pointing. Fig. 6(b) shows the data flow diagram of the laser pointing estimator. To ensure accurate estimation and interpretation of dynamic laser pointing, three procedures are proposed in the following subsections, namely laser point detection, tremor reduction, and a trajectory shape-fitting algorithm.

Laser Point Detection

The visual detection of the location of a laser point involves several steps. First, the input RGB image I_{RGB} is converted into a hue-saturation-value (HSV) image I_{HSV}. To segment the laser contour area within the HSV image, specific ranges are assigned to each layer. In this study, the color of the laser point is assumed to be red; thus, the hue layer is limited to a range [330°–358°], the saturation layer is limited to [10%–100%], and the value layer is limited to [78%–100%]. The laser point coordinates (x_l, y_l) are obtained by calculating the center of mass of the contour

$$M_{ij} = \sum_x \sum_y x^i y^j I_{\mathrm{HSV}}(x, y) \tag{2}$$

$$x_l = \left\lfloor \frac{M_{10}}{M_{00}} \right\rfloor, \qquad y_l = \left\lfloor \frac{M_{01}}{M_{00}} \right\rfloor \tag{3}$$

where $I_{\mathrm{HSV}}(x, y)$ = intensity value of the pixel at coordinates (x, y) within the contour; and M_{ij} = $(i + j)$th order image moment of $I_{\mathrm{HSV}}(x, y)$. The 2D coordinates in the image plane then are transformed into 3D space using the camera's intrinsic parameters based on the pinhole camera model (Sprute et al. 2019)

$${}^{C}L = \left(\frac{x_l - c_x}{f_x} d_l, \frac{y_l - c_y}{f_y} d_l, d_l \right)^T \tag{4}$$

where d_l = depth value of the corresponding pixel in the depth image; f_x and f_y = focal lengths of the camera; c_x and c_y = principle coordinates of the camera; and ${}^{C}L$ = 3D laser point coordinates in the camera frame. Finally, the 3D coordinates of the laser point from the camera frame ${}^{C}L$ are transformed to the world frame ${}^{W}L$ as follows:

$${}^{W}L = {}^{C}_{W}R^{T}\,{}^{C}L - {}^{C}_{W}R^{T}\,{}^{C}_{W}t \tag{5}$$

where ${}^{C}_{W}R$ and ${}^{C}_{W}t$ = rotation and translation from the world frame to the camera frame, respectively, obtained from the extrinsic camera parameters (Fig. 7). The intrinsic and extrinsic camera parameters are acquired through a calibration process.

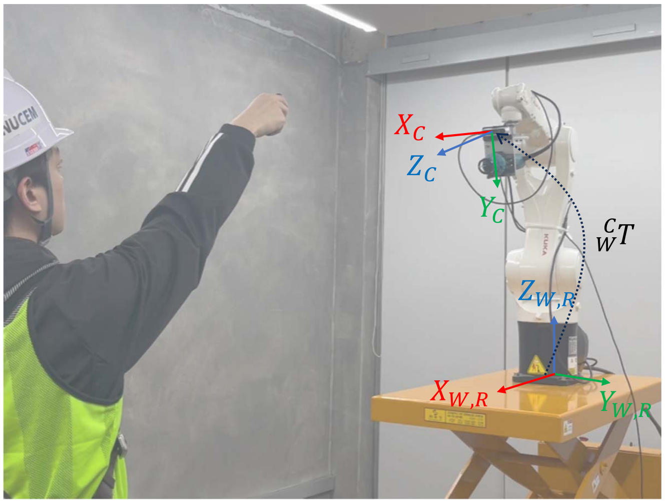

Fig. 7. Illustration of the camera frame (X_C, Y_C, Z_C), world frame (X_W, Y_W, Z_W), and robot frame (X_R, Y_R, Z_R). The world frame and the robot frame are aligned in this study. The transformation matrix from the world frame to the camera frame is represented as ${}^{C}_{W}T = [{}^{C}_{W}R|{}^{C}_{W}t]$. (Image by Sungboo Yoon.)

Laser Tremor Reduction

One of the challenges in tracking laser movements is the user's natural hand tremor. This hand tremor causes jitters in the laser spot, making it difficult to keep the laser focused on a particular surface of the environment or screen (Chung and Kim 2013). Although it is not possible to completely eliminate hand tremors without using an additional mechanical device (Matveyev et al. 2003), it is possible to systematically reduce jitters in the raw data collected during certain periods using various smoothing methods. Fig. 8 shows sample raw data which include these tremor-induced jitters, and postprocessed trajectories using the moving average filter (MAF), single exponential smoothing (SES), double exponential smoothing (DES), and one Euro filter (OEF). These algorithms produce smoothed trajectories that are less affected by jitters. Each of these algorithms was implemented in LaserDex and tested in the user study to quantitatively gauge their performance in mitigating hand tremors.

Laser Trajectory Shape-Fitting

The shape-fitting procedure converts smoothed laser trajectories into a rectangular shape, which allows the robot to execute the corresponding task. The rectangle-fitting (RF) algorithm takes as an input a set of 2D points in $\mathbb{R}^2$ and outputs the rectangle's central coordinates $P_c = (x_c, y_c)$, rotation angle θ of the rectangle around P_c, width w, and height h (Fig. 9). Given a set of m data points, $\boldsymbol{S} = \{(x_i, y_i) | i = 1, 2, \ldots, m\}$, observed coordinate $P_i = (x_i, y_i)$, and the parameters $\boldsymbol{\beta} = (x_c, y_c, \theta, w, h)$, the process of fitting a rectangle can be described as a least-squares problem, which can be solved using optimization algorithms, such as the Levenberg–Marquardt algorithm (Gavin 2019)

$$\min_{\boldsymbol{\beta}} \sum_{i \in \boldsymbol{S}}^{m} r_i^2 \tag{6}$$

where the residuals r_i are given by

$$r_i = d(P_i, Q_i) = \sqrt{(x_i - x_{i_{cp}})^2 + (y_i - y_{i_{cp}})^2} \tag{7}$$

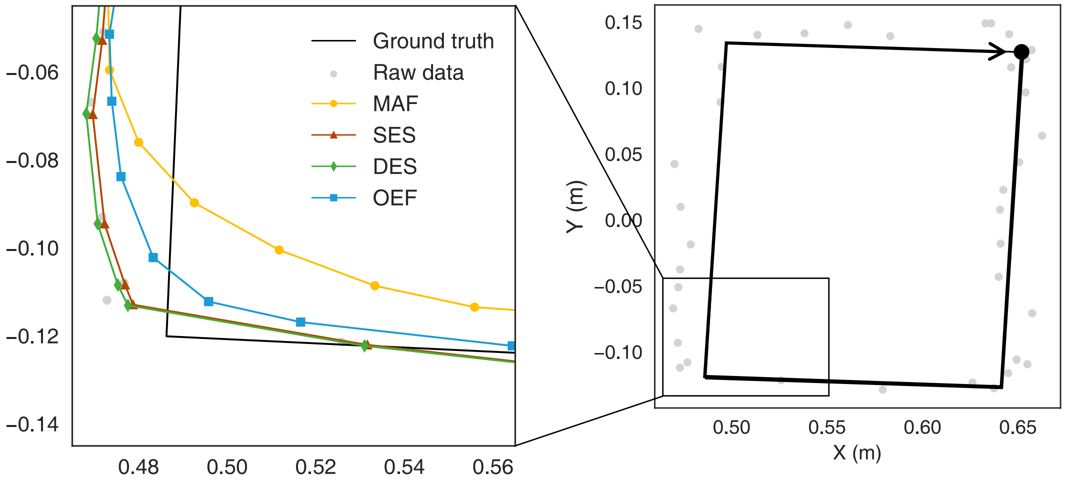

Fig. 8. Sample results of raw laser point data and smoothed trajectories using four smoothing algorithms. Ground truth refers to the actual position of the targeted shape. MAF = moving average filter, SES = single exponential smoothing, DES = double exponential smoothing, and OEF = one Euro filter.

$$Q_i = (x_{i_{cp}}, y_{i_{cp}}) = g(P_i;\boldsymbol{\beta}) \tag{8}$$

where $g(P_i;\boldsymbol{\beta})$ is a function that calculates the coordinates of the closest point on the rectangle based on the observed coordinate P_i and the parameters $\boldsymbol{\beta}$.

Robot Motion Planning

Sections "Gesture Recognition" and "Laser Pointing Estimation" explored the process of obtaining the desired positions and orientations of the robot tool center point (TCP) based on the user's spatial instructions. When the TCP is determined, motion planning is performed. For motion planning, we employ the stochastic trajectory optimization for motion planning (STOMP) (Choi et al. 2022; Kalakrishnan et al. 2011) algorithm, which is known for its ability to generate smooth trajectories in real time (Mainprice and Berenson 2013). However, during the execution phase, we assume that the workpiece is positioned on the ceiling, parallel to the ground. As a result, the tool-point orientation is implemented as a Cartesian path constraint, ensuring that the tool-point stays upright throughout the task execution.

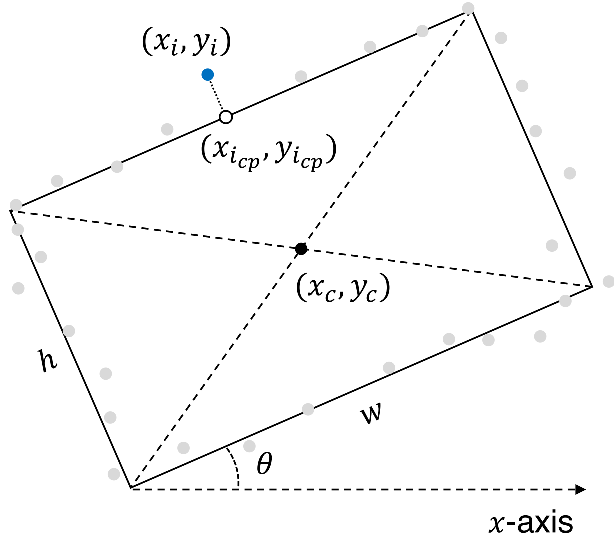

Fig. 9. Geometric definitions for the rectangle fitting and residual function.

Experiments

To assess the advantages of using LaserDex, we conducted a comparative user study. During this study, participants interacted with a colocated robot (within approximately 2.5 m) and tasked it with cutting a rectangle in a drywall. To enhance user immersion and provide contextual cues of a real-world jobsite, the interfaces were evaluated in a realistic construction site setting with unfinished concrete walls and drywalls.

Participants

After approval by the Institutional Review Board (IRB) at Seoul National University (IRB 2306/001–004), we recruited a total of 11 participants, six male and five female, with an age range from 18 to 25 years [23.1 ± 1.97 (mean ± standard deviation)]. Six of the participants were undergraduates and five were graduate students. Selection criteria included participants with knowledge of construction to better ground our results in real-world construction and to ensure that they understood the context of drywall finishing. None of the participants had prior experience in (tele)operating robotic arms. In terms of experience with joystick-operated video games, four participants reported no prior experience, whereas seven participants reported playing such games only on a yearly basis. Participants were compensated with approximately $15.

Hardware Setup

Fig. 10 shows the robotic setup used in the experiments. The hardware configuration consisted of a KUKA KR 6 R900 6-DoF (KUKA AG, Augsburg, Germany) manipulator mounted on a mobile table lift [Fig. 10(a)]. The robot's end effector was equipped with a Makita 3706 drywall cutting tool and an Intel RealSense D435 RGB-D camera [Fig. 10(b)]. The Makita 3706 (Makita Corporation, Anjo, Japan) drywall cutting tool is capable of operating at a maximum speed of 32,000 rpm for cutting purposes. The Intel RealSense D435 RGB-D camera (Intel Corporation, Santa Clara,

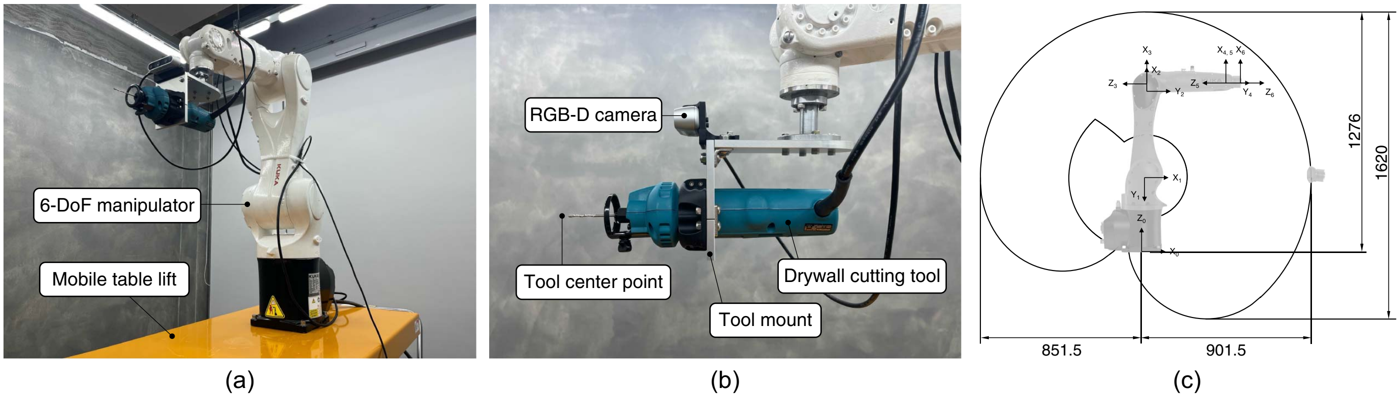

Fig. 10. Hardware setup used in this study: (a) hardware configuration; (b) end-effector detail; and (c) dimensions and coordinate systems of KUKA KR 6 R900. (Images by Sungboo Yoon.)

California) provides 640 × 480-pixel RGB and depth images at a frame rate of 30 frames/s (FPS). Fig. 10(c) illustrates the dimensions of the manipulator and coordinate system of each joint. Based on this information, we set the radius of task limit sphere to 850 mm.

Baseline: Handheld Controller Interface

For comparison, we chose a handheld controller interface as the baseline. Handheld controllers, such as joysticks, have gained widespread use in interfaces for construction machines and robots (Okishiba et al. 2019) This conventional acceptance of handheld controllers in construction robotics ensures an effective evaluation of LaserDex's performance in real-world applications. Fig. 11 shows the handheld controller used in this study; the three translational and three rotational dimensions of the robot's TCP were mapped to two analog joysticks and a button of an Xbox 360 (Microsoft Corporation, Redmond, Washington) controller.

Task and Procedure

We designed a within-subjects experiment in which each participant completed two spatial tasks cutting a rectangle in a 2 × 10-mm-thick drywall, using LaserDex and the handheld controller interface. Based on an a priori power analysis [$\alpha = 0.05, (1 - \beta) = 0.80$], the minimum required sample size was determined to be 10 pairs (10 samples for each condition).

Before starting the experiment, the participants were assigned randomly on a first-come, first-served basis to one of two groups. When one group reached its limit of five participants, any additional participants were assigned to the other group. In the experiment, one group used LaserDex first and then the handheld controller interface, and the other group conducted the two tasks in the reverse order to mitigate any potential learning effect on the measured performance.

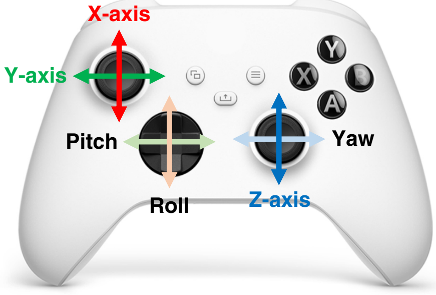

Fig. 11. Handheld controller (Xbox 360 controller) used in this study and its six-DoF mappings.

In each experimental condition (LaserDex and handheld controller), participants were asked to specify the trajectory of the rectangular cut on the drywall in order to spatially task the robot. The participants were allowed to determine the direction of the path themselves. To ensure consistency in performance across participants and interfaces, other factors—such as the initial positions of the robot and human (approximately 2.5 m apart), the initial robot configuration, and the speed of robot motion—were kept constant. The drywall was affixed to a metal stud frame suspended from the ceiling. The dimensions of the rectangular opening were 10 in. (254 mm) wide × 6 in. (152.4 mm) high, based on the specifications of the steel air grill intended for installation in the opening. This rectangular shape was presented on the drywall by printed targets to give a visual guidance only for the participants.

In the LaserDex condition, participants used a laser pointer and a Bluetooth microphone. The experimenter provided necessary training on using deictic gestures, verbal commands, and the laser pointer to spatially task the robot. The participants were instructed to indicate the central point of the rectangular target with the deictic gesture to guide the robot for a proper FoV for detecting the laser points. However, the experimental instructions did not include specific guidance on how to perform the deictic gestures, except the instruction to use only their right arm.

In the handheld controller interface condition, participants used an Xbox 360 joystick controller. Participants received training on the mapping between joystick axes and the robot, as described in the "Baseline: Handheld Controller Interface" section. Each participant performed a single test for each interface condition. They were allowed to practice the interaction methods until they felt confident about using them.

Upon completing both conditions, participants were asked to fill out a questionnaire regarding their experiences.

Measures

In this study, we used both objective and subjective measures to compare the task performance and the user experience between the two interfaces. The three objective measures were intersection over union (IoU) between the ground truth target B_{gt} and the estimated box trajectory of the robot's TCP B_{eb} (Fig. 12), the total task completion time, and the trajectory length. In this study, the robot

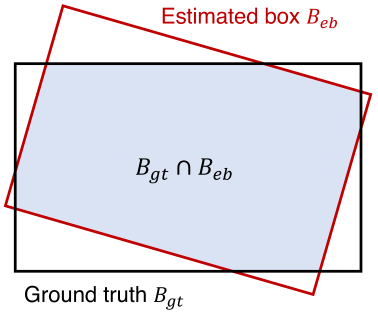

Fig. 12. Illustration of the intersection over union of two boxes. The shaded area is the intersection of the two boxes; the area of the outer boundaries is the union of the two boxes; and the IoU is the ratio of these two areas.

performed dry runs of the given spatial tasks. Therefore, the objective measures were derived from the robot's trajectories, rather than physical cutting outcomes.

The IoU ratio quantified the degree of overlap (0%–100%) between the two rectangles—the ground truth target, and the estimated box; it is a proxy metric for assessing the accuracy of the robotic task, and is calculated as follows:

$$\mathrm{IoU} = \frac{\mathrm{area}(B_{gt} \cap B_{eb})}{\mathrm{area}(B_{gt} \cup B_{eb})} \tag{9}$$

During the experiments, the entire trials were recorded to measure total task completion time offline. The starting time was defined as the time at which the participants started the interaction with the robot, and the finishing time was defined as the time at which the participants validated that they finished their task. The task completion time was a proxy metric for assessing the efficiency of the robotic task.

Additionally, for subjective measures, we adapted two questionnaires: the NASA Task Load Index (NASA-TLX) questionnaire, and the System Usability Scale (SUS). The NASA-TLX is a multidimensional rating procedure that derives an overall workload score, ranging from 0 to 100, based on a weighted average of ratings on six subscales: mental demand, physical demand, temporal demand, performance, effort, and frustration (Adamides et al. 2017). The higher the score, the higher is the perceived workload; therefore, a lower score is preferred. Whereas the other subscales used a rating scale ranging from very low to very high, the performance question used a scale from perfect to failure. Therefore, a lower score on the performance subscale also is preferable to a higher score. The SUS consists of a 10-item questionnaire with five response options for respondents, from strongly agree to strongly disagree. The SUS score ranges from 0 to 100, and the higher the score, the better is the perceived usability of the system (Adamides et al. 2017).

Results

Objective Results

Fig. 13 shows the results of two objective metrics—IoU and task completion time—as well as the trajectory length of the robot's TCP. We used a paired t-test to compare the performance of LaserDex with the baseline handheld controller. In terms of task accuracy and efficiency, LaserDex outperformed the baseline handheld controller. Participants using LaserDex achieved higher accuracy [$t(10) = -3.35$, $p < 0.01$] and completed the task in less time [$t(10) = 9.94$, $p < 0.0001$] than those using the baseline handheld controller. Notably, LaserDex not only had better average accuracy but also had lower variance. The handheld controller resulted in an IoU of 0.514 ± 0.270 (mean ± standard deviation), whereas LaserDex achieved 0.830 ± 0.074. Moreover, the trajectory length of the robot's TCP was significantly shorter [$t(10) = 13.3$, $p < 0.0001$] when using LaserDex, a trend which is depicted in the trajectories [position; orientation is omitted for clarity (Cui et al. 2023)] of all participants in Fig. 14(b). In contrast, the trajectories of the handheld controller [Fig. 14(a)] were less smooth and often had movements that deviated from the intended target. This indicates that novice users faced difficulties in understanding and aligning with the control mechanisms of the manipulator, even though

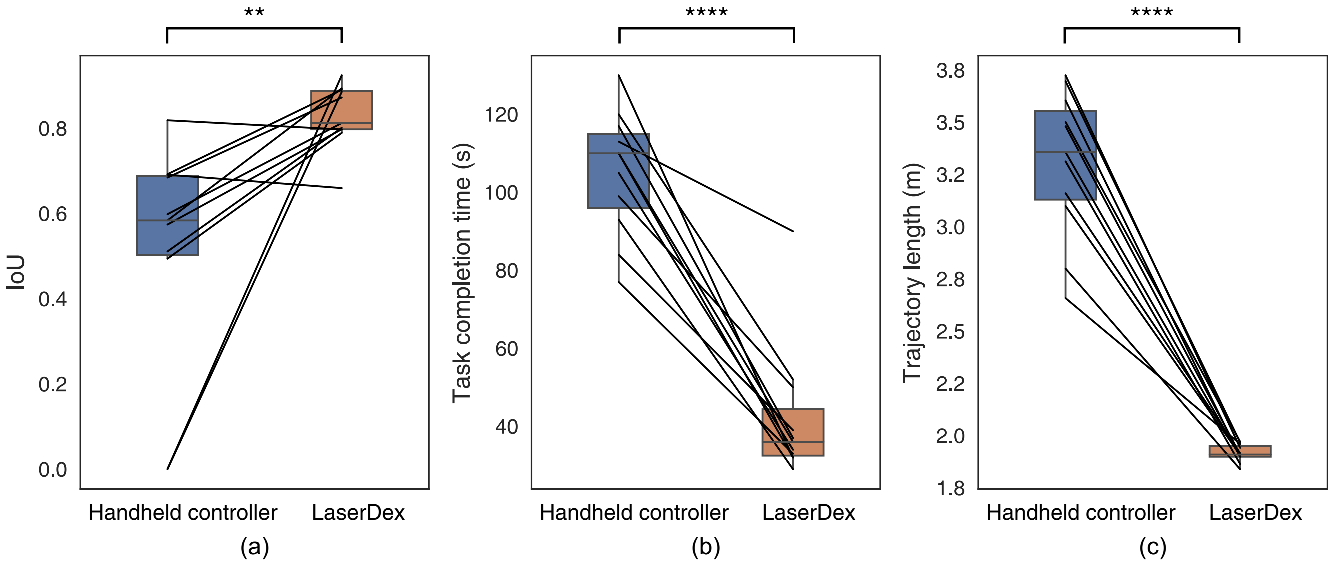

Fig. 13. Results of our objective metrics: (a) IoU; (b) task completion time; and (c) trajectory length of the robot's tool center point (TCP). Lines connect paired data points from the same participant. **$p < 0.01$; ****$p < 0.0001$.

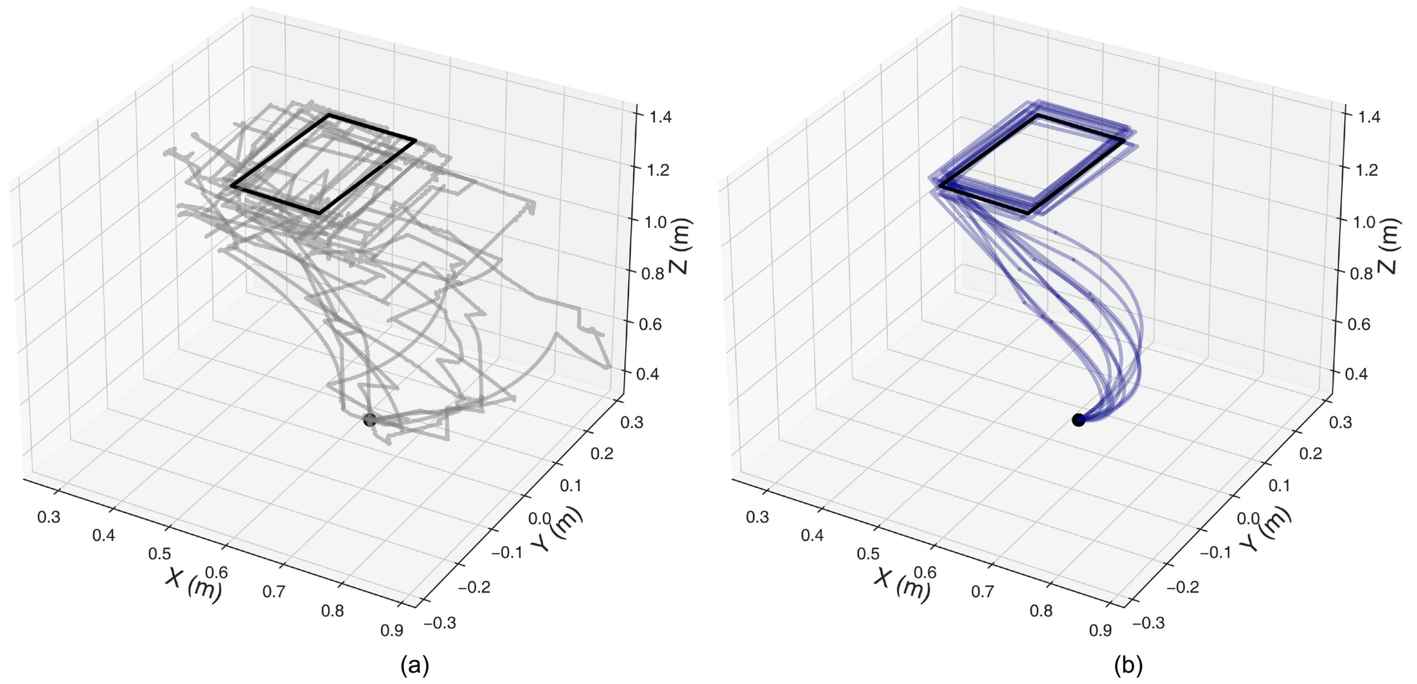

Fig. 14. Robot TCP trajectories: (a) handheld controller; and (b) LaserDex. The solid circle represents the initial position of the robot TCP, and the hollow rectangle represents the ground truth target.

they received sufficient training on task space control with the handheld controller (Losey et al. 2022). LaserDex had smoother and more-accurate trajectories [Fig. 14(b)], allowing participants intuitive control without the need to consider control strategies.

For a clearer understanding of LaserDex's workflow, an example task sequence of the trial performed by Subject 4 is shown in Fig. 15. The trial took a total of 37 s, with 26 s spent on interaction and an additional 11 s for robot execution. The interaction using deictic gestures took approximately 2 s, 4 s were required for the system to recognize the verbal cues, and the interaction using a laser pointer took 9 s.

Fig. 16 shows the sample results of the four smoothing algorithms. Each of the four sets of original data points was selected randomly from 11 participants. In this study, from the set of 2D data points, the minimum bounding box was calculated by finding the extreme coordinates along the x- and y-axes. The results demonstrate that the minimum bounding box algorithms tended to be less accurate than the rectangle-fitting algorithms. Additionally, the results show that the OEF with rectangle-fitting method produced a rectangle that closely matched the ground truth.

Furthermore, to investigate quantitatively the effects of the proposed methods for laser instruction estimation, the IoUs of the four smoothing algorithms and the two rectangular shape estimation algorithms for 11 participants are shown in Fig. 17(a). The parameters for each of the four smoothing algorithms were optimized through empirical search, and the resulting values are reported in Table 1. Among the smoothing methods, OEF, a first-order low-pass filter, outperformed the others for both the minimum bounding box and rectangle-fitting algorithms. Moreover, OEF is fast, simple to tune, and offers a good trade-off between precision and latency (Baloup et al. 2019; Casiez et al. 2012). In terms of rectangular shape estimation, the rectangle fitting algorithm, described in the "Laser Trajectory Shape-Fitting" section, significantly increased the IoUs of smoothed points compared with the minimum bounding box algorithm for the SES [$t(10) = -5.66$, $p < 0.001$], DES [$t(10) = -5.92$, $p < 0.001$], and OEF [$t(10) = -3.00$, $p < 0.05$].

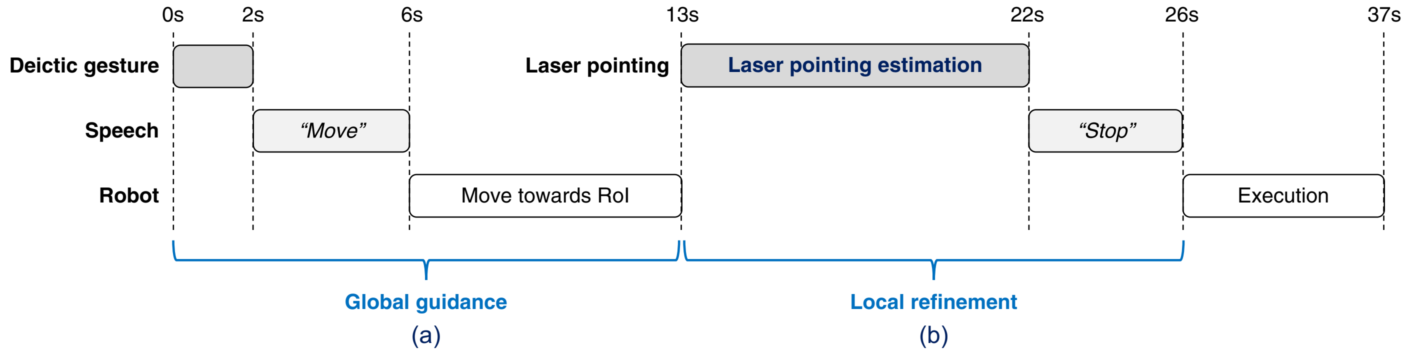

Fig. 15. Example task sequence: (a) deictic gesture recognition; and (b) laser pointing estimation.

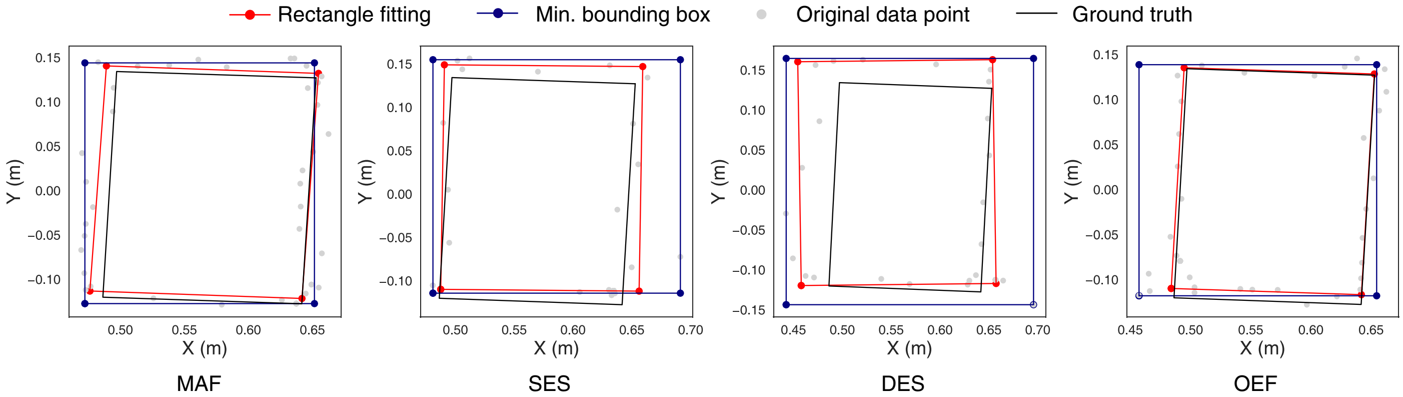

Fig. 16. Sample results of the estimated rectangle by smoothing algorithms from four different participants.

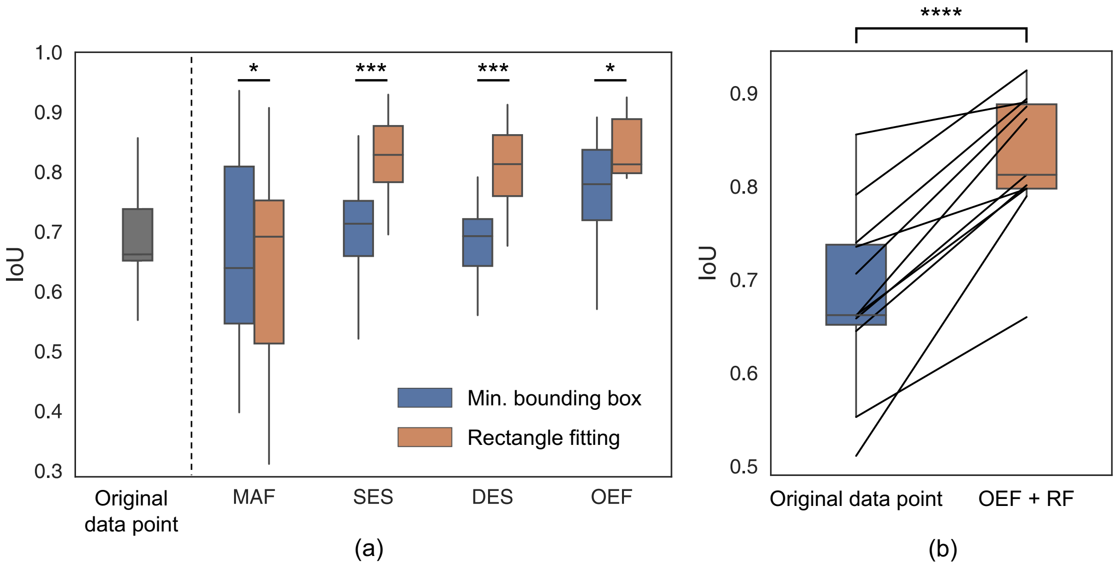

Fig. 17. IoU ratios: (a) four smoothing algorithms and two rectangular shape estimation algorithms; and (b) proposed method (OEF + RF) versus original data point. Lines connect paired data points from the same participant. $^{*}p < 0.05$; $^{***}p < 0.001$; $^{****}p < 0.0001$.

In the case of the MAF, however, the minimum bounding box algorithm was shown to be statistically significant for increasing the IoUs [$t(10) = 2.27$, $p < 0.05$)]. These results suggest that the OEF with rectangle-fitting is the most robust and accurate method for estimating dynamic laser trajectories. Fig. 17(b) illustrates the results of the OEF + RF method and the original data points. Statistical analysis using the paired t-tests showed that the IoUs of estimated rectangles significantly increased using the OEF + RF method [$t(10) = -7.28$, $p < 0.0001$].

Table 1. Parameters of different smoothing methods and their values

Smoothing method	Parameter	Value
Moving average filter (MAF)	Window size	8
Single exponential smoothing (SES)	α	0.9
Double exponential smoothing (DES)	α β	0.9 0.1
One Euro filter (OEF)	Min cutoff β	0.173 0.01

Subjective Results

Fig. 18 shows the results of two subjective questionnaires. Fig. 18(a) shows the perceived workload (NASA-TLX index). Averaging all six subscales with weighted scores shows that the handheld controller scored 54.4 ± 19.6, whereas LaserDex achieved a better score, 46.8 ± 23.2. However, no statistically significant differences were observed in the average scores. The subscales also showed no statistically significant differences, except for the mental demand [$t(10) = 2.68$, $p < 0.05$]. Fig. 18(b) shows the results of the SUS. Participants perceived LaserDex as more usable [$t(10) = -5.11$, $p < 0.001$] than the handheld controller (83.4 versus 62.5, out of 100).

Discussion

Construction robots require novel human–robot interfaces to leverage the benefits of both intuitiveness and precise robotic operation. However, these advantages act as contrasting objectives for designing and modeling human–robot interfaces based on deictic gestures (Carfi and Mastrogiovanni 2021). To address this challenge, this

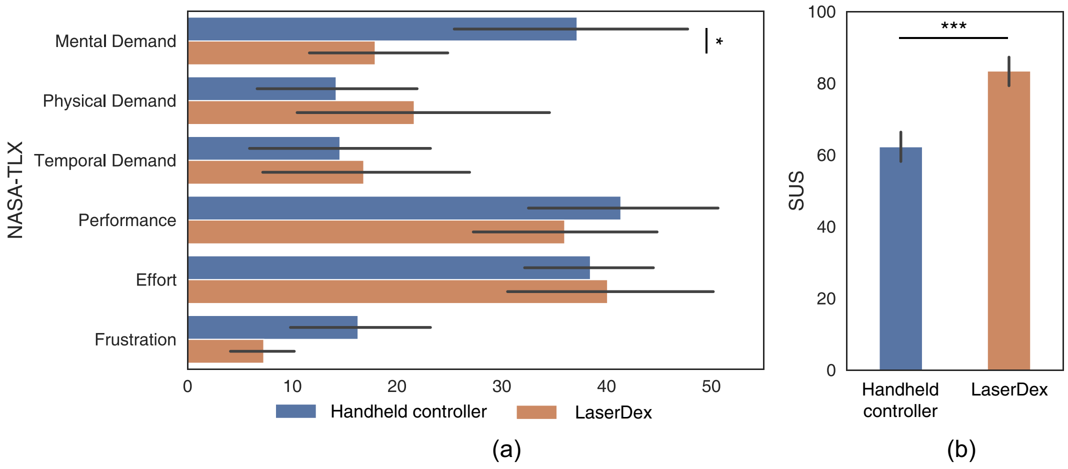

Fig. 18. Results from two subjective questionnaires: (a) NASA Task Load Index (NASA-TLX), in which a lower score is preferable; and (b) System Usability Scale (SUS), in which a higher score is preferable. *$p < 0.05$; ***$p < 0.001$.

study proposes LaserDex, a human–robot interface that integrates deictic gestures and dynamic laser pointing and presented a computational method to accurately and efficiently interpret spatial tasks.

An intuitive human–robot interface enables construction workers, regardless of their prior experience with robots, to quickly learn how to interact with the robot and exert control. LaserDex facilitates seamless and user-centered HRI by incorporating multiple input streams, specifically deictic gesture, speech, and laser pointing. The results of the user study involving nonexpert participants demonstrated a significant enhancement in perceived usability compared with the baseline, particularly in the context of construction tasks involving far-to-reach targets that require significant changes in the robot's configuration for execution.

Moreover, by applying dynamic laser pointing with smoothing using the OEF algorithm and rectangular shape estimation using the RF method, LaserDex showed noteworthy accuracy in estimating spatial tasks, particularly in addressing the challenges associated with deictic gestures, such as limited coverage and accuracy within 3D workspaces. Fig. 19 shows the distance errors with and without laser pointing. For the deictic gesture only condition, the distance error was calculated as the Euclidean distance between the central point of the ground truth target and the estimated target point at the time the user gave verbal cues. For the deictic gesture plus laser pointing condition, the distance error was calculated as the Euclidean distance between the central point of the ground truth target and the central point of the estimated rectangle. Without laser pointing, relying solely on deictic gestures, the average distance error for LaserDex was 228 ± 141 mm, whereas for deictic gestures with laser pointing, the average distance error was 11.4 ± 5.15 mm. The results show that implementing a laser pointing strategy in LaserDex not only significantly enhances the accuracy of spatial location estimation [$t(10) = 5.05$, $p < 0.001$], but also improves robustness and repeatability, as evidenced by the low variance among participants.

Furthermore, these results were found to be competitive with those of related research conducted in other domains, including robotics. Table 2 presents distance errors from previous studies. Interfaces from other domains were tested mostly on close-proximity tabletops, whereas this study focused on scenarios with more-distant targets. This underscores the relevance and applicability of LaserDex in real-world scenarios in which construction robots operate within expansive 3D workspaces.

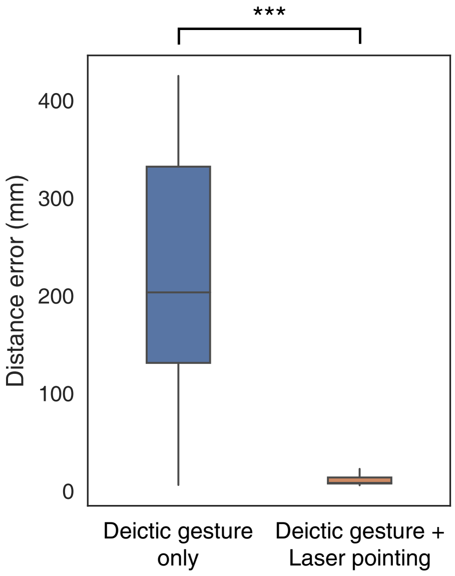

Fig. 19. Distance error. For deictic gesture only, the error is the Euclidean distance between the central point of the ground truth target and the estimated target point at the time the user gave verbal cues. For deictic gesture plus laser pointing, the error is the Euclidean distance between the central point of the ground truth target and the central point of the estimated rectangle. ***$p < 0.001$.

LaserDex has the potential to empower construction robots—such as drywall finishing robots (Canvas 2022), wall plastering robots (Okibo 2022), and concrete drilling robots (Hilti 2020)—to interact effectively with human workers in improvising spatial locations (e.g., positions, orientations, and areas). This aligns with the results of Kim et al. (2022), who outlined the preferences of architecture finishing groups in improvising their tasks with human skills. By enabling human workers to collaborate with robots in situ, LaserDex potentially can help to address the concerns of workers who fear losing control over their work and can maintain their professional autonomy.

Three limitations of this study need to be addressed in future research. First, the robot pose was fixed at a predefined distance

Table 2. Distance errors of LaserDex and deictic gesture–based interfaces in other domains

Interaction input	Estimation technique	Experimental setup	Distance error (mean ± standard deviation)	Reference
Deictic pointing + laser pointing + speech	Vision-based pose estimation	Target selection task using robot arm. Rectangular target area on ceiling. Human–target distance = 2.5 m.	11.4 ± 5.15 mm	Present study
Deictic pointing	Marker-based six-DoF motion capture system	Target selection task; 80 virtual targets (16 × 5 grid, 360° rotation). Human–target distance = 4 m.	400 mm	Mayer et al. (2020)
Deictic pointing	IMU-based pose estimation and motion capture system	Quadrotor landing task. Four landing targets at corners of flying arena with edges of 3.6 m. Subjects located in middle of arena.	70.0 mm	Gromov et al. (2020)
Deictic pointing	Vision-based pose estimation	Object grasping task using robot arm; 27 objects on floor. Human–robot distance = 2.0 m.	52.9 ± 59.4 mm	Hu et al. (2022)
Deictic pointing	Motion sensor–based hand recognition	Target selection task. Virtual target markers on interactive LED monitors. Distance to target plane = 0.4 m.	50 mm	Čorňák et al. (2021)
Deictic pointing	Vision-based pose estimation	AMR navigation task. Human–target distance = 4.0 m.	50 mm	Ikeda et al. (2023)
Deictic gaze or head pose	Eye tracking glasses and motion capture system	Waypoint selection task using robot arm. Five waypoints on tabletop. Human–target distance = 0.3–1.1 m.	DG: 27.4 ± 21.8 mm; HP: 19.0 ± 15.7 mm	Wöhle and Gebhard (2021)
Deictic pointing + head pose	Vision-based hand and face detection	Mobile robot navigation task; 25 targets on floor. Human–target distance = 1.5–5.5 m.	161 ± 19 mm (at 1.5 m); 484 ± 123 mm (at 5.5 m)	Azari et al. (2019)
Deictic pointing + head pose + speech	HMD-based head pose estimation	Object selection task for pick-and-place using robot arm. Five cylinder targets on table. Human–target distance = 0.3–0.5 m.	10.0 ± 3.0 mm	Krupke et al. (2018)

Note: IMU = inertial measurement unit; AMR = autonomous mobile robot; and HMD = head-mounted display.

from the workpiece for continuous observation of the dynamic laser pointing using the RGB-D camera. However, in situations in which the workspace is constrained and securing an adequate offset from the workpiece is not possible, or when the content of the spatial task is considerably large, the robot may lose track of the laser pointing, resulting in data loss in the laser trajectories. To ensure seamless HRI in construction, we suggest that future work should investigate the integration of a FoV adjustment mechanism to dynamically reposition the robot pose before losing track of the laser pointing.

Second, we utilized an Xbox controller as a proxy for the handheld controllers in construction robotics. Although this choice was made because of the general acceptance of gaming controllers as teleoperation interface for robotic arms, we acknowledge that this is a simplifying assumption. Future work should investigate the use of authentic control devices that fully represent the control mechanisms and interface design of handheld controllers utilized in the field.

Third, our participant demographic comprised students with a background in construction, but not professional construction workers. Therefore, although our results reflect the interaction between nonexpert humans in robotics and robots, they may overlook the hands-on experience and knowledge that professional workers bring to the drywall cutting task when operating a robotic arm. Future work should investigate how construction expertise impacts the task performance and user experience of LaserDex.

Conclusion

This study addresses the challenges of human–robot collaboration in construction jobsites by proposing a human–robot interface that enables in situ spatial tasking. Our resultant interface, LaserDex, aims to provide an effective and intuitive means of operating construction robots, enabling construction workers to communicate spatial goals without the need for extensive training in robotics.

The results of our study showed that deictic gestures can be implemented successfully in distant spatial tasking, with the assistance of laser pointing. LaserDex achieved a distance error of only 11.4 mm, outperforming similar deictic gesture–based interfaces in other domains. Moreover, the proposed method—which integrates laser spot detection, laser trajectory smoothing using the OEF algorithm, and rectangular shape estimation using the RF algorithm—was found to be precise and robust in interpreting spatial tasking for target areas. The proposed method achieved an IoU of 0.830, with less variation among participants compared with the baseline handheld controller. Last, subjective results also suggest that this intuitive and natural interaction method enhances perceived usability; LaserDex received higher scores in the SUS than did the baseline.

The integration of deictic gestures and laser pointing in a human–robot interface contributes to both precise robotic operation and intuitive HRI. These findings highlight the potential of our interface as a viable alternative to conventional handheld controllers in the context of in situ construction task improvisations.

The authors' ongoing research explores different forms of spatial tasks. The authors envision several variations of spatial tasks, beyond the rectangular-shaped opening examined in this study, which can expand the applicability of LaserDex to a broader spectrum of construction tasks.

Data Availability Statement

Data will be made available on request.

Acknowledgments

This study was supported by the New Faculty Startup Fund and the Institute of Construction and Environmental Engineering (ICEE) at Seoul National University (SNU). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of ICEE or SNU.

References

Adami, P., P. B. Rodrigues, P. J. Woods, B. Becerik-Gerber, L. Soibelman, Y. Copur-Gencturk, and G. Lucas. 2022. "Impact of VR-based training on human–robot interaction for remote operating construction robots." *J. Comput. Civ. Eng.* 36 (3): 04022006. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001016.

Adamides, G., C. Katsanos, Y. Parmet, G. Christou, M. Xenos, T. Hadzilacos, and Y. Edan. 2017. "HRI usability evaluation of interaction modes for a teleoperated agricultural robotic sprayer." *Appl. Ergon.* 62 (Jul): 237–246. https://doi.org/10.1016/j.apergo.2017.03.008.

Al, G. A., P. Estrela, and U. Martinez-Hernandez. 2020. "Towards an intuitive human-robot interaction based on hand gesture recognition and proximity sensors." In *Proc., 2020 IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 330–335. New York: IEEE.

Amtsberg, F., X. Yang, L. Skoury, and A. Menges. 2021. "iHRC: An AR-based interface for intuitive, interactive and coordinated task sharing between humans and robots in building construction." In *Proc., 38th ISARC*. Cambridge, UK: International Association for Automation and Robotics in Construction.

Argall, B. D., S. Chernova, M. Veloso, and B. Browning. 2009. "A survey of robot learning from demonstration." *Rob. Auton. Syst.* 57 (5): 469–483. https://doi.org/10.1016/j.robot.2008.10.024.

Aronson, R. M., N. Almutlak, and H. Admoni. 2021. "Inferring goals with gaze during teleoperated manipulation." In *Proc., 2021 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 7307–7314. New York: IEEE.

Asadi, E., B. Li, and I.-M. Chen. 2018. "Pictobot: A cooperative painting robot for interior finishing of industrial developments." *IEEE Rob. Autom. Mag.* 25 (2): 82–94. https://doi.org/10.1109/MRA.2018.2816972.

Azari, B., A. Lim, and R. Vaughan. 2019. "Commodifying pointing in HRI: Simple and fast pointing gesture detection from RGB-D images." In *Proc., 2019 16th Conf. on Computer and Robot Vision (CRV)*, 174–180. New York: IEEE.

Baloup, M., T. Pietrzak, and G. Casiez. 2019. "RayCursor: A 3D pointing facilitation technique based on raycasting." In *Proc., 2019 CHI Conf. on Human Factors in Computing Systems, CHI '19*, 1–12. New York: Association for Computing Machinery.

Bovo, R., D. Giunchi, L. Sidenmark, H. Gellersen, E. Costanza, and T. Heinis. 2022. "Real-time head-based deep-learning model for gaze probability regions in collaborative VR." In *Proc., 2022 Symposium on Eye Tracking Research and Applications, ETRA '22*, 1–8. New York: Association for Computing Machinery.

Braeuer-Burchardt, C., F. Siegmund, D. Hoehne, P. Kuehmstedt, and G. Notni. 2020. "Finger pointer based human machine interaction for selected quality checks of industrial work pieces." In *Proc., ISR 2020; 52th Int. Symp. on Robotics*, 1–6. New York: IEEE.

Brosque, C., E. G. Herrero, Y. Chen, and M. A. Fischer. 2021. "Collaborative welding and joint sealing robots with haptic feedback." In *Proc., 38th ISARC*. Cambridge, UK: International Association for Automation and Robotics in Construction.

Brown, T. B., et al. 2020. "Language models are few-shot learners." Preprint, submitted July 24, 2020. https://arxiv.org/abs/2005.14165.

Canvas. 2022. "Canvas.Build." Accessed April 7, 2023. https://www.canvas.build/.

Carfi, A., and F. Mastrogiovanni. 2021. "Gesture-based human–machine interaction: Taxonomy, problem definition, and analysis." *IEEE Trans. Cybern.* 53 (1): 497–513. https://doi.org/10.1109/TCYB.2021.3129119.

Casiez, G., N. Roussel, and D. Vogel. 2012. "1€ filter: A simple speed-based low-pass filter for noisy input in interactive systems." In *Proc., SIGCHI Conf. on Human Factors in Computing Systems, CHI '12*, 2527–2530. New York: Association for Computing Machinery.

Castro, A., F. Silva, and V. Santos. 2021. "Trends of human-robot collaboration in industry contexts: Handover, learning, and metrics." *Sensors* 21 (12): 4113. https://doi.org/10.3390/s21124113.

Choi, A., M. K. Jawed, and J. Joo. 2022. "Preemptive motion planning for human-to-robot indirect placement handovers." In *Proc., 2022 Int. Conf. on Robotics and Automation (ICRA)*, 4743–4749. New York: IEEE.

Chung, M. G., and S.-K. Kim. 2013. "Efficient jitter compensation using double exponential smoothing." *Inf. Sci.* 227 (Apr): 83–89. https://doi.org/10.1016/j.ins.2012.12.008.

Čorňák, M., M. Tölgyessy, and P. Hubinský. 2021. "Innovative collaborative method for interaction between a human operator and robotic manipulator using pointing gestures." *NATO Adv. Sci. Inst. Ser. E Appl. Sci.* 12 (1): 258. https://doi.org/10.3390/app12010258.

Crocher, V., R. Singh, J. Newn, and D. Oetomo. 2021. "Towards a gaze-informed movement intention model for robot-assisted upper-limb rehabilitation." In *Proc., 43rd Annual Int. Conf. IEEE Engineering in Medicine & Biology Society (EMBC)*, 6155–6158. New York: IEEE.

Cui, Y., S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh. 2023. "No, to the right: Online language corrections for robotic manipulation via shared autonomy." In *Proc., 2023 ACM/IEEE Int. Conf. on Human-Robot Interaction, HRI '23*, 93–101. New York: Association for Computing Machinery.

de Jesus, K. J., H. J. Kobs, A. R. Cukla, M. A. de Souza Leite Cuadros, and D. F. T. Gamarra. 2021. "Comparison of visual SLAM algorithms ORB-SLAM2, RTAB-map and SPTAM in internal and external environments with ROS." In *Proc., 2021 Latin American Robotics Symp. (LARS), 2021 Brazilian Symp. on Robotics (SBR), and 2021 Workshop on Robotics in Education (WRE)*. New York: IEEE.

Gavin, H. P. 2019. *The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems*. Durham, NC: Duke Univ.

Gromov, B., G. Abbate, L. M. Gambardella, and A. Giusti. 2019. "Proximity human-robot interaction using pointing gestures and a wrist-mounted IMU." In *Proc., 2019 Int. Conf. on Robotics and Automation (ICRA)*, 8084–8091. New York: IEEE.

Gromov, B., L. Gambardella, and A. Giusti. 2020. "Guiding quadrotor landing with pointing gestures." In *Human-friendly robotics 2019*, 1–14. Cham, Switzerland: Springer.

Guzzi, J., G. Abbate, A. Paolillo, and A. Giusti. 2022. "Interacting with a conveyor belt in virtual reality using pointing gestures." In *Proc., 2022 ACM/IEEE Int. Conf. on Human-Robot Interaction, HRI '22*, 1194–1195. New York: IEEE Press.

Hilti. 2020. "Hilti Jaibot." Accessed April 7, 2023. https://www.hilti.com/content/hilti/W1/US/en/business/business/trends/jaibot.html.

Hu, Z., Y. Xu, W. Lin, Z. Wang, and Z. Sun. 2022. "Augmented pointing gesture estimation for human-robot interaction." In *Proc., 2022 Int. Conf. on Robotics and Automation (ICRA)*, 6416–6422. New York: IEEE.

Huang, L., Z. Zhu, and Z. Zou. 2023. "To imitate or not to imitate: Boosting reinforcement learning-based construction robotic control for long-horizon tasks using virtual demonstrations." *Autom. Constr.* 146 (Feb): 104691. https://doi.org/10.1016/j.autcon.2022.104691.

Ikeda, T., N. Noda, S. Ueki, and H. Yamada. 2023. "Gesture interface and transfer method for AMR by using recognition of pointing direction and object recognition." *J. Rob. Mechatron.* 35 (2): 288–297. https://doi.org/10.20965/jrm.2023.p0288.

ISO. 2021. *Robotics—Vocabulary.* ISO 8373. Geneva: ISO.

Jenny, S. E., H. Blum, A. Gawel, R. Siegwart, F. Gramazio, and M. Kohler. 2020. "Online synchronization of building model for on-site mobile robotic construction." In *Proc., Int. Symp. on Automation and Robotics in Construction*, 1508–1514. Cambridge, UK: International Association for Automation and Robotics in Construction.

Jevtić, A., G. Doisy, Y. Parmet, and Y. Edan. 2015. "Comparison of interaction modalities for mobile indoor robot guidance: Direct physical interaction, person following, and pointing control." *IEEE Trans. Hum.-Mach. Syst.* 45 (6): 653–663. https://doi.org/10.1109/THMS.2015.2461683.

Jirak, D., D. Biertimpel, M. Kerzel, and S. Wermter. 2021. "Solving visual object ambiguities when pointing: An unsupervised learning approach." *Neural Comput. Appl.* 33 (7): 2297–2319. https://doi.org/10.1007/s00521-020-05109-w.

Kalakrishnan, M., S. Chitta, E. Theodorou, P. Pastor, and S. Schaal. 2011. "STOMP: Stochastic trajectory optimization for motion planning." In *Proc., 2011 IEEE Int. Conf. on Robotics and Automation*. New York: IEEE.

Kazhdan, M., M. Bolitho, and H. Hoppe. 2006. "Poisson surface reconstruction." In *Proc., Fourth Eurographics Symp. on Geometry Processing*. Goslar, Germany: Eurographics Association.

Khatib, M., K. Al Khudir, and A. De Luca. 2021. "Human-robot contactless collaboration with mixed reality interface." *Rob. Comput. Integr. Manuf.* 67 (Feb): 102030. https://doi.org/10.1016/j.rcim.2020.102030.

Kim, Y., H. Kim, R. Murphy, S. Lee, and C. R. Ahn. 2022. "Delegation or collaboration: Understanding different construction stakeholders' perceptions of robotization." *J. Manage. Eng.* 38 (1): 04021084. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000994.

Koh, K. H., M. Farhan, K. P. C. Yeung, D. C. H. Tang, M. P. Y. Lau, P. K. Cheung, and K. W. C. Lai. 2021. "Teleoperated service robotic system for on-site surface rust removal and protection of high-rise exterior gas pipes." *Autom. Constr.* 125 (May): 103609. https://doi.org/10.1016/j.autcon.2021.103609.

Kramberger, A., A. Kunic, I. Iturrate, C. Sloth, R. Naboni, and C. Schlette. 2021. "Robotic assembly of timber structures in a human-robot collaboration setup." *Front. Rob. AI* 8 (Jan): 768038. https://doi.org/10.3389/frobt.2021.768038.

Krupke, D., F. Steinicke, P. Lubos, Y. Jonetzko, M. Görner, and J. Zhang. 2018. "Comparison of multimodal heading and pointing gestures for co-located mixed reality human-robot interaction." In *Proc., 2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 1–9. New York: IEEE.

Labbé, M., and F. Michaud. 2019. "RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation." *J. Field Rob.* 36 (2): 416–446. https://doi.org/10.1002/rob.21831.

Lee, H. J., and S. Brell-Cokcan. 2021. "Cartesian coordinate control for teleoperated construction machines." *Constr. Rob.* 5 (1): 1–11. https://doi.org/10.1007/s41693-021-00055-y.

Lee, S., S. Eom, and J. Moon. 2013. "Design of teach pendant for robot glazing system." In *Proc., 30th Int. Symp. on Automation and Robotics in Construction and Mining (ISARC 2013): Building the Future in Automation and Robotics*. Cambridge, UK: International Association for Automation and Robotics in Construction.

Liang, C.-J., V. R. Kamat, and C. C. Menassa. 2020. "Teaching robots to perform quasi-repetitive construction tasks through human demonstration." *Autom. Constr.* 120 (Dec): 103370. https://doi.org/10.1016/j.autcon.2020.103370.

Liang, C.-J., V. R. Kamat, C. C. Menassa, and W. McGee. 2022. "Trajectory-based skill learning for overhead construction robots using generalized cylinders with orientation." *J. Comput. Civ. Eng.* 36 (2): 04021036. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001004.

Liu, Y., M. Habibnezhad, and H. Jebelli. 2021. "Brain-computer interface for hands-free teleoperation of construction robots." *Autom. Constr.* 123 (Mar): 103523. https://doi.org/10.1016/j.autcon.2020.103523.

Losey, D. P., H. J. Jeon, M. Li, K. Srinivasan, A. Mandlekar, A. Garg, J. Bohg, and D. Sadigh. 2022. "Learning latent actions to control assistive robots." *Auton. Rob.* 46 (1): 115–147. https://doi.org/10.1007/s10514-021-10005-w.

Mahmud, J. A., B. C. Das, J. Shin, K. M. Hasib, R. Sadik, and M. F. Mridha. 2022. "3D gesture recognition and adaptation for human–robot interaction." *IEEE Access* 10 (Nov): 116485–116513. https://doi.org/10.1109/ACCESS.2022.3218679.

Mainprice, J., and D. Berenson. 2013. "Human-robot collaborative manipulation planning using early prediction of human motion." In *Proc., 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. New York: IEEE.

Matveyev, S., M. Göbel, and P. Frolov. 2003. *Laser pointer interaction with hand tremor elimination*. 376–740. Berlin: Springer.

Mayer, S., J. Reinhardt, R. Schweigert, B. Jelke, V. Schwind, K. Wolf, and N. Henze. 2020. "Improving humans' ability to interpret deictic gestures in virtual reality." In *Proc., 2020 CHI Conf. on Human Factors in Computing Systems, CHI '20*, 1–14. New York: Association for Computing Machinery.

Mayer, S., V. Schwind, R. Schweigert, and N. Henze. 2018. "The effect of offset correction and cursor on mid-air pointing in real and virtual environments." In *Proc., 2018 CHI Conf. on Human Factors in Computing Systems, CHI '18*, 1–13. New York: Association for Computing Machinery.

Medeiros, A. C. S., P. Ratsamee, J. Orlosky, Y. Uranishi, M. Higashida, and H. Takemura. 2021. "3D pointing gestures as target selection tools: guiding monocular UAVs during window selection in an outdoor environment." *ROBOMECH J.* 8 (1): 1–19. https://doi.org/10.1186/s40648-021-00200-w.

Mitterberger, D., S. Ercan Jenny, L. Vasey, E. Lloret-Fritschi, P. Aejmelaeus-Lindström, F. Gramazio, and M. Kohler. 2022. "Interactive robotic plastering: Augmented interactive design and fabrication for on-site robotic plastering." In *Proc., 2022 CHI Conf. on Human Factors in Computing Systems, CHI '22*, 1–18. New York: Association for Computing Machinery.

Mur-Artal, R., and J. D. Tardos. 2017. "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras." *IEEE Trans. Rob.* 33 (5): 1255–1262. https://doi.org/10.1109/TRO.2017.2705103.

Murphy, R. R., and S. Tadokoro. 2019. "User interfaces for human-robot interaction in field robotics." In *Disaster robotics: Results from the ImPACT tough robotics challenge*, edited by S. Tadokoro, 507–528. Cham, Switzerland: Springer.

Okibo. 2022. "Our robot." Accessed April 7, 2023. https://okibo.com/our-robot/.

Okishiba, S., R. Fukui, M. Takagi, H. Azumi, S. Warisawa, R. Togashi, H. Kitaoka, and T. Ooi. 2019. "Tablet interface for direct vision teleoperation of an excavator for urban construction work." *Autom. Constr.* 102 (Jun): 17–26. https://doi.org/10.1016/j.autcon.2019.02.003.

Ong, S. K., A. W. W. Yew, N. K. Thanigaivel, and A. Y. C. Nee. 2020. "Augmented reality-assisted robot programming system for industrial applications." *Rob. Comput. Integr. Manuf.* 61 (Feb): 101820. https://doi.org/10.1016/j.rcim.2019.101820.

Park, S., X. Wang, C. C. Menassa, V. R. Kamat, and J. Y. Chai. 2023. "Natural language instructions for intuitive human interaction with robotic assistants in field construction work." Preprint, submitted July 9, 2023. https://arxiv.org/abs/2307.04195.

Raffel, C., N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. 2019. "Exploring the limits of transfer learning with a unified text-to-text transformer." Preprint, submitted October 23, 2019. https://arxiv.org/abs/1910.10683.

Rosen, E., D. Whitney, M. Fishman, D. Ullman, and S. Tellex. 2020. "Mixed reality as a bidirectional communication interface for human-robot interaction." In *Proc., 2020 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 11431–11438. New York: IEEE.

Shi, L., C. Copot, and S. Vanlanduit. 2021. "GazeEMD: Detecting visual intention in gaze-based human-robot interaction." *Robotics* 10 (2): 68. https://doi.org/10.3390/robotics10020068.

Sprute, D., K. Tönnies, and M. König. 2019. "This far, no further: Introducing virtual borders to mobile robots using a laser pointer." In *Proc., 2019 Third IEEE Int. Conf. on Robotic Computing (IRC)*, 403–408. New York: IEEE.

Stogsdill, A., G. Clark, A. Ranucci, T. Phung, and T. Williams. 2021. "Is it pointless? Modeling and evaluation of category transitions of spatial gestures." In *Proc., Companion of the 2021 ACM/IEEE Int. Conf. on Human-Robot Interaction, HRI '21 Companion*, 392–396. New York: Association for Computing Machinery.

Strazdas, D., J. Hintz, A. Khalifa, A. A. Abdelrahman, T. Hempel, and A. Al-Hamadi. 2022. "Robot system assistant (RoSA): Towards intuitive multi-modal and multi-device human-robot interaction." *Sensors* 22 (3): 923. https://doi.org/10.3390/s22030923.

Suzuki, R., A. Karim, T. Xia, H. Hedayati, and N. Marquardt. 2022. "Augmented reality and robotics: A survey and taxonomy for AR-enhanced human-robot interaction and robotic interfaces." In *Proc., 2022 CHI Conf. on Human Factors in Computing Systems, CHI '22*, 1–33. New York: Association for Computing Machinery.

Touvron, H., et al. 2023a. "LLaMA: Open and efficient foundation language models." Preprint, submitted February 27, 2023. https://arxiv.org/abs/2302.13971.

Touvron, H., et al. 2023b. "Llama 2: Open foundation and fine-tuned chat models." Preprint, submitted July 18, 2023. https://arxiv.org/abs/2307.09288.

Ürkmez, M., and H. I. Bozma. 2022. "Detecting 3D hand pointing direction from RGB-D data in wide-ranging HRI scenarios." In *Proc., 2022 ACM/IEEE Int. Conf. on Human-Robot Interaction, HRI '22*, 441–450. New York: IEEE Press.

Villani, V., F. Pini, F. Leali, and C. Secchi. 2018. "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications" *Mechatronics* 55 (Nov): 248–266. https://doi.org/10.1016/j.mechatronics.2018.02.009.

Wang, X., C.-J. Liang, C. C. Menassa, and V. R. Kamat. 2021. "Interactive and immersive process-level digital twin for collaborative human–robot construction work." *J. Comput. Civ. Eng.* 35 (6): 04021023. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000988.

Wang, X., H. Shen, H. Yu, J. Guo, and X. Wei. 2023. "Hand and arm gesture-based human-robot interaction: A review." In *Proc., 6th Int. Conf. on Algorithms, Computing and Systems, ICACS '22*, 1–7. New York: Association for Computing Machinery.

Weller, R., W. Wegele, C. Schröder, and G. Zachmann. 2021. "LenSelect: Object selection in virtual environments by dynamic object scaling." *Front. Virtual Reality* 2 (Jun): 684677. https://doi.org/10.3389/frvir.2021.684677.

Wöhle, L., and M. Gebhard. 2021. "Towards robust robot control in cartesian space using an infrastructureless head- and eye-gaze interface." *Sensors* 21 (5): 1798. https://doi.org/10.3390/s21051798.

Yang, B., J. Huang, X. Chen, X. Li, and Y. Hasegawa. 2023. "Natural grasp intention recognition based on gaze in human–robot interaction." *IEEE J. Biomed. Health Inf.* 27 (4): 2059–2070. https://doi.org/10.1109/JBHI.2023.3238406.

Ye, Y., T. Zhou, and J. Du. 2023. "Robot-assisted immersive kinematic experience transfer for welding training." *J. Comput. Civ. Eng.* 37 (2): 04023002. https://doi.org/10.1061/JCCEE5.CPENG-5138.

Yin, H., J. M. Liew, W. L. Lee, M. H. Ang, and K. W. Yeoh. 2022. "Towards BIM-based robot localization: A real-world case study." In *Proc., Int. Symp. on Automation and Robotics in Construction (IAARC)*. International Association for Automation and Robotics in Construction. Cambridge, UK: International Association for Automation and Robotics in Construction.

Yoon, S., Y. Kim, C. R. Ahn, and M. Park. 2021. "Challenges in deictic gesture-based spatial referencing for human-robot interaction in construction." In *Proc., 38th Int. Symp. on Automation and Robotics in Construction (ISARC)*, 491–497. Cambridge, UK: International Association for Automation and Robotics in Construction.

Yoon, S., Y. Kim, M. Park, and C. R. Ahn. 2023. "Effects of spatial characteristics on the human–robot communication using deictic gesture in construction." *J. Constr. Eng. Manage.* 149 (7): 04023049. https://doi.org/10.1061/JCEMD4.COENG-12997.

Yoon, S., J. Park, M. Park, and C. R. Ahn. 2024. "A deictic gesture-based human-robot interface for in situ task specification in construction." *Comput. Civil Eng.* 445–452.

You, H., Y. Ye, T. Zhou, Q. Zhu, and J. Du. 2023. "Robot-enabled construction assembly with automated sequence planning based on ChatGPT: RoboGPT." *Buildings* 13 (7): 1772. https://doi.org/10.3390/buildings13071772.

Yuan, L., C. Reardon, G. Warnell, and G. Loianno. 2019. "Human gaze-driven spatial tasking of an autonomous MAV." *IEEE Rob. Autom. Lett.* 4 (2): 1343–1350. https://doi.org/10.1109/LRA.2019.2895419.

Zhang, M., R. Xu, H. Wu, J. Pan, and X. Luo. 2023. "Human–robot collaboration for on-site construction." *Autom. Constr.* 150 (Jun): 104812. https://doi.org/10.1016/j.autcon.2023.104812.

Zhong, M., Y. Zhang, X. Yang, Y. Yao, J. Guo, Y. Wang, and Y. Liu. 2019. "Assistive grasping based on laser-point detection with application to wheelchair-mounted robotic arms." *Sensors* 19 (2): 303. https://doi.org/10.3390/s19020303.

Zhou, T., Q. Zhu, Y. Ye, and J. Du. 2023. "Humanlike inverse kinematics for improved spatial awareness in construction robot teleoperation: Design and experiment." *J. Constr. Eng. Manage.* 149 (7): 04023044. https://doi.org/10.1061/JCEMD4.COENG-13350.

Zhu, Q., J. Du, Y. Shi, and P. Wei. 2021. "Neurobehavioral assessment of force feedback simulation in industrial robotic teleoperation." *Autom. Constr.* 126 (Jun): 103674. https://doi.org/10.1016/j.autcon.2021.103674.

Zimmermann, C., T. Welschehold, C. Dornhege, W. Burgard, and T. Brox. 2018. "3D human pose estimation in RGBD images for robotic task learning." In *Proc., 2018 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1986–1992. New York: IEEE Press.