ASCE

# Interactive and Immersive Process-Level Digital Twin for Collaborative Human–Robot Construction Work

Xi Wang, S.M.ASCE[1]; Ci-Jyun Liang, S.M.ASCE[2]; Carol C. Menassa, M.ASCE[3]; and Vineet R. Kamat, M.ASCE[4]

**Abstract:** Human cognition plays a critical role in construction work, particularly in the context of high-level task planning and in-field improvisation. On the other hand, robots are adept at performing numerical computation and repetitive physical tasks with precise motion control. The unstructured and complex nature of construction environments and the inability to maintain tight tolerances in assembled workpieces pose several unique challenges to the wide application of robots in construction work. Thus, the robotization of field construction processes is best conceived as a collaborative human–robot endeavor that takes advantage of both human and robot intelligence as well as robots' physical operation capabilities to overcome uncertainties and successfully perform useful construction work onsite. This paper proposes an interactive and immersive process-level digital twin (I2PL-DT) system in virtual reality (VR) that integrates visualization and supervision, task planning and execution, and bidirectional communication to enable collaborative human–robot construction work. In this work paradigm, the human worker is responsible for high-level task planning and work process supervision. The robot undertakes workspace sensing and monitoring, detailed motion planning, and physical execution of the work. A drywall installation case study involving imperfect rough carpentry (wall framing) is presented using a KUKA mobile industrial robotic arm emulator. A human-in-the-loop study involving 20 subjects was conducted for system verification and to collect feedback for future improvements. The experimental results show that users can use the system to specify work sequences, select optimal task plans, and perform robot trajectory guidance after simple training and felt positive about the system functions and user experience. The system demonstrates the potential of transitioning the role of construction workers from physical task performers to robot supervisors. In addition, the system establishes a promising framework for construction workers to remotely collaborate with onsite construction robots. **DOI: 10.1061/(ASCE)CP.1943-5487.0000988.** 

**Author keywords:** Digital twin; Construction robots; Immersive virtual reality; Human–robot collaboration.



## Introduction

Due to the unstructured and dynamic nature of construction sites and the considerable physical demand construction work imposes on workers, the construction industry has been susceptible to higher fatalities and nonfatal injury rates among all the major industries (CPWR 2018; Liu et al. 2017). In addition to safety issues, the construction industry is vulnerable to natural and social restrictions because construction work generally cannot be performed remotely. As a result, the outbreak of the Covid-19 pandemic significantly affected construction projects across the US, causing 975,000 construction job losses in a single month (April 2020) that resulted in serious economic impacts (ENR 2020). Automation in construction enabled by human–robot collaboration (HRC) offers a promising approach to mitigate these issues by making optimum use of both human and robot capabilities and strengths (Liang et al. 2021; You et al. 2018).

### *Human and Robot Capabilities for Construction Work*

Robots are designed to manipulate objects with high precision and perform tasks repetitively and thus offer a promising alternative to relieve construction workers from physically demanding and repetitive tasks (Liang et al. 2019; Xu and Garcia de Soto 2020). Robots also allow some construction work to be conducted remotely and facilitate social distancing onsite so that construction projects are not significantly interrupted by unexpected circumstances such as the Covid-19 pandemic. Robots' reasoning intelligence in scene understanding, motion planning, and adaptability experienced rapid growth in recent years because of the progress in artificial intelligence and computational power (Brosque et al. 2021). However, construction robots face several challenges due to the unstructured and complex nature of construction environments and relatively loose tolerances of construction projects, which may lead to frequent robot failures while performing tasks onsite (Lundeen et al. 2019; Milberg 2006).

While robots are competent in the accurate and repetitive manipulation of heavy components, detection of minor deviations, and numerical computation, human beings are better at creative planning and sequencing based on domain knowledge, experience, and perceptual understanding (Seong et al. 2019; Sharif et al. 2016). Considering drywall installation as an example, when the wall frame deviates from the design, the human carpenter will tune the drywall panel or adjust nailing angles to ensure that the panel is

[1]Ph.D. Candidate, Civil and Environmental Engineering, Univ. of Michigan, Ann Arbor, MI 48109. ORCID: https://orcid.org/0000-0002-2583-0356. Email: wangix@umich.edu

[2]Ph.D. Candidate, Civil and Environmental Engineering, Univ. of Michigan, Ann Arbor, MI 48109. ORCID: https://orcid.org/0000-0002-0213-8471. Email: cjliang@umich.edu

[3]Associate Professor, Civil and Environmental Engineering, Univ. of Michigan, Ann Arbor, MI 48109 (corresponding author). ORCID: https://orcid.org/0000-0002-2453-0386. Email: menassa@umich.edu

[4]Professor, Civil and Environmental Engineering, Univ. of Michigan, Ann Arbor, MI 48109. ORCID: https://orcid.org/0000-0003-0788-5588. Email: vkamat@umich.edu

Note. This manuscript was submitted on December 31, 2020; approved on June 17, 2021; published online on August 26, 2021. Discussion period open until January 26, 2022; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801.

firmly connected to the wall frame, which is acceptable to a certain extent (NAHB 2015). Although human workers can quickly improvise a new plan in such situations, it is difficult for a robot to make decisions adaptively when it comes across unknown situations. Therefore, human expertise and ability to improvise (i.e., adjust the plan according to the current circumstance that differs from design) play a crucial role in overcoming these uncertainties and are indispensable during the construction process, making it unrealistic to completely replace construction workers with robots (Kyjanek et al. 2019).

Despite the significant promise of HRC, current techniques for humans to interact with large, industrial, construction robots are still inefficient (Kyjanek et al. 2019). One of the most common HRC methods for construction robots is teleoperation. However, it suffers from limited perception and accuracy reduction (Roldán et al. 2019). When it comes to mobile robotic arms with multiple degrees of freedom (DOFs) carrying large and heavy objects, a significant amount of training is required for operators since any errors or lapses can cause collisions and other safety issues (Hashimoto et al. 2011). Another typical HRC method in construction is to lead the robot by putting forward physical forces on the robot itself or the object carried by the robot, which is also fraught with safety issues since the worker needs to intimately share the workspace with the robot (Chung et al. 2010). Moreover, both of these techniques do not take advantage of robot intelligence in reasoning and still require human workers to continuously perform manual tasks during the whole work process.

### Human–Robot Collaborative Construction System

In order to overcome the limitations of existing HRC techniques and allow construction workers without robot programming expertise to seamlessly communicate with and intuitively operate robots for onsite construction work, this study proposes an interactive and immersive process-level digital twin (I2PL-DT) system for collaborative human–robot construction. The human worker is responsible for high-level task planning and supervision, and the robot undertakes detailed workspace sensing and monitoring, path planning, and physical execution of the work. During the work process, the human and the robot interact through a bidirectional seamless communication interface. The characteristics of the enabled collaborative workflow are as follows: (1) the workspace is continuously sensed and monitored by the robot and the information can be visualized by human workers through the VR digital twin, (2) human workers can perform high-level task planning and send task objectives and commands to the robot intuitively with the VR interface, (3) the robot can automatically develop collision-free motion plans and demonstrate the plans to human upon receiving requests from human, (4) human workers can preview the motion plans and approve a desirable plan for execution, and (5) the robot can physically execute the approved plan to perform the task while the human worker supervising the execution process in VR.

Allowing workers to interact with onsite robots from remote locations has the potential to reduce the number of onsite workers or facilitate their physical separation. In addition, with the help of immersive VR, people with disabilities (e.g., wheelchair users) can also perform construction work in collaboration with construction robots, offering potentially game-changing benefits toward making the construction workforce more inclusive. In order to evaluate the system and obtain feedback for future improvements, a drywall installation case study involving imperfect rough carpentry (wall framing) together with a human-in-the-loop experiment are conducted.

## Literature Review

### HRC Techniques for Robot Operation

HRC allows humans to operate robots for task execution. Human operators take over the decision making process and lead the robot to adaptively execute tasks. HRC techniques can thus significantly reduce the effort and technical challenges of preprogramming construction robots to be fully autonomous.

An intuitive method of human–robot collaborative operation is to lead the robot through physical contact, requiring human operators to apply physical forces directly to the robot or the object the robot is carrying to guide the robot to corresponding positions (Devadass et al. 2019; Lee and Moon 2014). The approach has several advantages. First, the robot is carrying the workpiece so that human workers are relieved from physical stress and can pay more attention to the task execution details. Second, the system retains the agility of human workers who can adapt to uncertainties such as workpiece deviations. This type of robot for physical interaction with humans needs to be specifically designed with safeguard functions. However, working alongside construction robots can still be dangerous since robot failures can cause serious accidents or fatalities in the presence of large and heavy construction workpieces (Sawacha et al. 1999; You et al. 2018).

Teleoperation allows the motion of human operators from remote locations to be replicated on the robot in real-time (David et al. 2014). It can protect humans from dangerous environments and thus is popular for construction robots. The joystick is often found convenient for navigating and thus has been widely used for unmanned ground vehicle (UGV) teleoperation (Khasawneh et al. 2019). In addition to navigation, joysticks serve as a device to operate robots with higher DOFs (Jung et al. 2013). During teleoperation, the robot working environment is captured by fixed cameras or robot onboard cameras. The human operator remotely observes the robot working environment captured by these cameras from computer screens. As a result, operators have a limited field of view, choices of perspective, and depth perception of the environment, leading to increased difficulty and reduced accuracy of manipulation (Chen et al. 2007; Roldán et al. 2019).

Haptic and force-reflecting devices can reflect contact force to the operator and provide the operator with tactile responses from the environment (Chotiprayanakul et al. 2012). It allows contact force control and makes teleoperation safer and smoother. Haptic and force-reflecting teleoperation have been used in a variety of robotized construction tasks, such as glass window panel fitting (Chung et al. 2010) and steel bridge grit blasting (Chotiprayanakul et al. 2012).

Robots can also be guided with human gestures from distanced or remote locations. Some studies have used vision-based systems to detect human gestures for robot guidance, such as hand position tracking (Du et al. 2012) and handheld light baton detection (Yu et al. 2014). Some studies used wearable sensors to track human body motions. Kim et al. (2009) used inclinometers, orientation sensors, and rotary encoders to detect human arm movement to operate an excavator. Seong et al. (2019) tracked dexterous human hand movement with gesture-controlling gloves and replicated the movement on a robotic hand.

In addition, researchers have also developed several other types of robot teleoperation interfaces. For example, the mobile phone has been used to teleoperate robots via text messages (Patra and Ray 2007) or voice commands (Kubik and Sugisaka 2001). David et al. (2014) remotely operate the cutting head of a tunnel boring machine with a master-slave system. With the proliferation of smartphones and tablets, multitouch interfaces with augmented reality (AR) techniques have been presented (Frank et al. 2016; Hashimoto et al. 2011). In recent years, several immersive teleoperation interfaces have also been developed (Whitney et al. 2018).

Although teleoperation can reduce programming workload and protect operators from danger, it has several limitations. First, the robot is moving at the same time while the human is operating. While real-time operation has some benefits, it poses additional security risks since operators have limited perception of the robot working environment; Second, the human operator needs to determine and lead the robot through the full path of manipulation and persistently engage in the whole work process. The human effort could be spared to some extent by making better use of robot intelligence; Lastly, teleoperating construction robots with multiple DOFs is difficult, and comprehensive training and expertise are required for human operators. Operational difficulties and safety risks increase significantly when it comes to the case of construction robots that manipulate heavy and large workpieces.

### Digital Twins in Robotics Applications

The concept of digital twins has become increasingly popular in recent years with the growth of sensing and computing capabilities and visualization technologies (Bilberg and Malik 2019). Digital twins include a virtual representation of the physical world that contains necessary and pertinent information from the physical world. Most importantly, digital twins also include data communication capabilities that connect and synchronize the virtual world with the physical world and exchange information between them (Deng et al. 2021). Such communication capabilities differentiate digital twins from 3D simulations and are inevitable elements of digital twins (Grieves 2014; Wang et al. 2017).

Digital twins have been used for several robotic applications in the manufacturing industry. For example, Kuts et al. (2019) proposed an industrial digital twin to program motions for an industrial robot arm to repeat in a real manufacturing process. Bilberg and Malik (2019) used a digital twin-based simulation for dynamic task sequence arrangement and allocation between a human and a robot in an assembly cell. Digital twins have also been used for HRC assembly system validation (Malik and Brem 2021) and safety protection while humans shared workspace with robots (Maragkos et al. 2019). Liang et al. (2020b) also developed a synchronization system to connect construction robots and digital twin simulations. However, the application of digital twins to construction robots is still limited.

### Immersive AR, VR, and Mixed Reality Technologies in HRC

With the emergence of low-cost commercial immersive devices, immersive technologies, including AR, VR, and MR, have been introduced to facilitate HRC from different aspects, including robot teleoperation (Sukumar et al. 2015; Whitney et al. 2018), joint angle control (Kuts et al. 2019), task objectives specification (Roldán et al. 2019; Wang et al. 2020b), trajectories planning (Kyjanek et al. 2019), and robots' intention indication (El Hafi et al. 2020; Walker et al. 2018).

Immersive technologies have also been utilized to study HRC in the construction industry. In a beam welding task, AR was used to show target welding positions so that the human operator can adjust the beam position for robotic welding (Tavares et al. 2019). Several studies have been conducted to study construction workers' reactions when they share the workspace with robots in order to develop safe HRC mechanisms (Kim et al. 2015; You et al. 2018). There are extensive studies in construction using immersive technologies for visualization, design, safety, and training purposes (Li et al. 2018; Liu et al. 2020). Zhou et al. (2020) used VR-based robot teleoperation for civil engineering applications. However, the application of immersive technologies for construction robot operation is limited in practice.

### Comparison of I2PL-DT and Existing Studies

An efficient HRC system for construction must possess the following properties: first, human workers should be able to assist onsite construction robots to overcome loose tolerances and design deviations through effective guidance, instructions, and communication mechanisms; second, the system should relieve human physical and mental effort by transitioning the role of human workers from physical task performers to robot supervisors; last but not least, the system should ensure the safety of both human workers and construction site property with safeguards and collision avoidance mechanisms.

Based on the nature of the construction industry, we propose seven characteristics useful for HRC systems in construction, as shown in Table 1. The proposed characteristics include necessary information and functions that support users' remote interaction with onsite construction robots. Several closely related prior studies from a variety of applications are selected and the proposed characteristics they included are summarized in Table 2. The scale of the presented case study and the scale of objects manipulated in each study are summarized in the last column.

Although there are several existing studies utilizing immersive technologies or digital twins for robot operation, they cannot be directly applied to construction projects. On one hand, most of these systems are at a tabletop scale with a fixed robotic arm manipulating small objects. The same HRC approaches cannot be simply scaled up to construction tasks where both the robot workspace and the target objects are much larger than typical human workers. For instance, specifying the end-effector position as task goal (Characteristic 4) or previewing trajectory line for plan evaluation (Characteristic 5) is sufficient for manipulating small objects but is not adequate when objects are large (e.g., drywall panels). When the robot is manipulating a large object, it is critical to show how the object will move along with the robot for human workers during both the plan preview and execution supervision processes (Characteristic 5 and 6) to evaluate whether there are safety concerns. On the other hand, the execution process of construction robots involves significant uncertainties and less repeatability. As a result, HRC for construction robots needs a more intuitive approach that allows frequent human intervention rather than setting up a series of movements for the robot to repeat over an extended period of time, which should be considered for system design.

**Table 1.** Highlighted characteristics for HRC systems in construction

| Number | Characteristics |
|---|---|
| 1 | Human interaction from remote locations |
| 2 | Real-time visualization of the physical environment (if remote) |
| 3 | Augmented information useful for supervision purpose |
| 4 | Hierarchical task planning (high-level human task planning and improvisation and low-level robot automation for detailed motion plan) |
| 5 | Robot motion plan preview and evaluation |
| 6 | Real-time robot execution process and status supervision |
| 7 | Bidirectional communication between the human and the robot |

**Table 2.** Prior studies characteristics summary

| Study | Type | Characteristics 1 | 2 | 3 | 4 | 5 | 6 | 7 | Case (object) scale |
|---|---|---|---|---|---|---|---|---|---|
| Kyjanek et al. (2019) | AR | — | — | Work progress, detailed robot status | — | Trajectory line | Directly | Yes | Prefabrication (small) |
| Roldán et al. (2019) | VR | Yes | — | — | Drag end effector | — | Directly | Yes | Tabletop (no) |
| Kuts et al. (2019) | VR | Yes | — | Detailed robot status | Specify end effector | Robot movement | Robot synchronization | Yes | Tabletop (no) |
| Ong et al. (2020) | AR | — | — | Plan assistance | Select workpiece feature | — | Directly | Yes | Tabletop (small) |
| Frank et al. (2016) | MR | — | — | Estimated object pose | Drag block on tablet | — | Directly | — | Tabletop (small) |
| Sukumar et al. (2015) | AR | Yes | Stereo vision | Actual and target position | — | — | Directly through camera | Yes | Tabletop (small) |
| Zhou et al. (2020) | VR | Yes | Virtual screens | — | — | — | With teleoperation | Yes | Tabletop (small) |
| Cichon and Robmann (2018) | VR | Yes | Point clouds + models | Detailed robot status | — | — | — | Yes | Tabletop (small) |
| This study | VR | Yes | Point clouds + meshes + models | Workspace BIM, high-level robot status | Object target pose | Robot + object movement | Robot + object synchronization | Yes | Drywall Installation (large) |

It should be noted that these HRC systems are highly configurable and customizable. Characteristics are implemented differently in each study depending on their scale, applications, and focus. Take Characteristic 3 as an example, previous studies mainly show augmented information such as detailed robot status (e.g., joint angle), work progress (e.g., progress percentage), or object-related information. This study focuses on providing more intuitive high-level robot status (e.g., motion plan found) and environment-related information that facilitates human inspection and supervision. For Characteristic 2, this study combines 3D building information modeling (BIM) models, reconstructed 3D meshes, and point clouds to enable real-time visualization while reducing computational resources. We used markers for Characteristic 1 to show whether the system allows remote operation and Characteristic 7 since the content of bidirectional communication varies depending on the system needs.

In an effort to remedy the identified gaps in knowledge and current capabilities, the objective of the presented research is to develop an HRC system that is capable of conducting construction tasks with large involved objects and can offer interactive communication abilities to construction workers without robot programming expertise. Toward this end, an I2PL-DT HRC construction system that covers all seven characteristics is proposed.

## Technical Approach

### System Overview

The proposed I2PL-DT system integrates an immersive VR interface for human interaction, middleware for computation and communication, and a robot operational environment (ROE) for sensory data collection and construction task execution. The system framework is presented in Fig. 1. The immersive VR interface, developed on the Unity platform, allows users to interact with robots remotely with an augmented telepresence experience. The ROE is the construction environment in which the robot performs tasks. It consists of the robot, construction workspace, and sensors in the environment. The immersive VR interface is connected to the ROE via the robot operating system (ROS) as the middleware (Quigley et al. 2009). The middleware acts as the bridge between the human and the robot in ROE. It receives and processes data from both the immersive VR interface and the ROE, performs computation based on the information presented, and publishes processed data to corresponding clients.

The general system workflow is shown in Fig. 2, in which the roles of the human and the robot (i.e., middleware, sensors, and the real robot in ROE) are illustrated. *Workspace sensing and monitoring* are conducted as the system is initiated. The as-built workspace environment and robot states captured by the sensors in the ROE are processed by the middleware and sent to the VR interface continuously to be relayed in the human view. The human can perform *site inspection* by comparing this as-built workspace geometry with the as-designed BIM model in VR and based on this inspection perform *high-level task planning* to make decisions on work sequence, object to manipulate, installation position, and other tasks. High-level planning is achieved by interacting with objects and the information dashboard (billboard) in VR. At this stage, the human can test and compare different options without physical stress or risks from repetitively manipulating heavy construction materials. The user can confirm the task plan and send it to the middleware if satisfied.

The middleware processes the high-level task plan into specific goals for robot *motion planning*. The motion planner of the robot

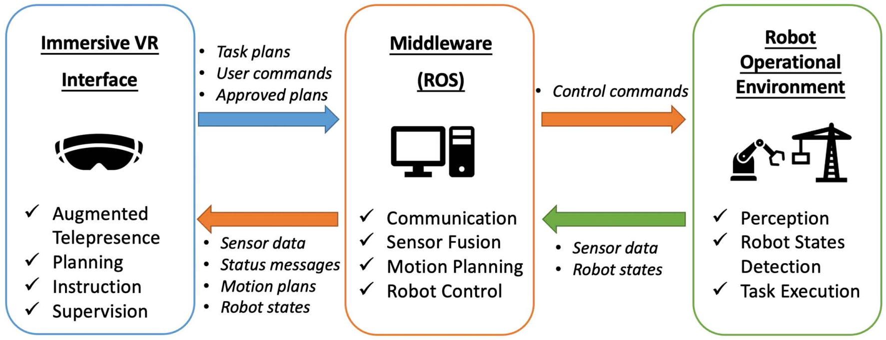


**Fig. 1.** HRC system framework.

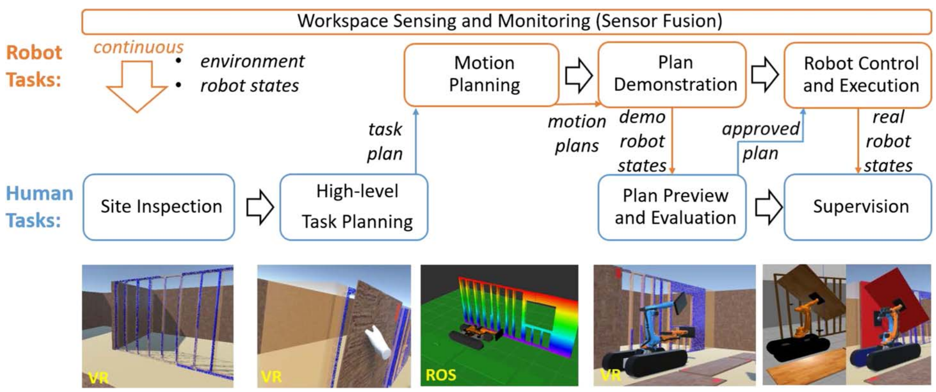


**Fig. 2.** System workflow and human–robot roles distribution.

then generates several collision-free motion plans to achieve the goals. In the meantime, the human can visualize the planning status messages of the robot from the billboard in VR. The motion plans are converted into robot states and published to VR for *demonstration* while the human *previews and evaluates* the plans on the virtual robot model. The motion plans are shown as full-scale realistic animations showing the robot and object movements that will happen at the execution stage. After viewing the motion plan, the human can approve the plan if satisfied or request a new plan demonstration.

At the same time, the middleware is notified of which motion plan has been approved by the user. It starts to *control* the real robot in the ROE to *execute* the approved motion plan. Joint states of the real robot are continuously captured by the encoders on the robot actuators and sent to the middleware, which are then relayed to the VR interface. Another virtual robot in VR synchronizes its joint states with the real robot in the ROE in real-time based on the state messages received. Therefore, the human can *supervise* the real robot states as it executes the task. In addition, the human can also obtain robot execution status messages from the billboard in VR.

In the remainder of this section, the technical approaches for developing the immersive VR interface and middleware are discussed in detail. The establishment of the ROE varies case-by-case and thus is discussed later in the case study.

## *Immersive VR Interface*

This study uses an immersive VR interface for several reasons. First, compared to AR and MR, immersive VR allows users to be present at remote locations away from the construction site, something that is particularly helpful to reduce construction site congestion and improve safety. For example, it can facilitate social-distancing requirements during periods such as the Covid-19 pandemic without compromising the progress of the work. Second, immersive VR provides realistic experiences to users. The human operator can navigate in the immersive VR environment and can observe objects from different perspectives just as they would do in the real world. This overcomes the limited field of view and depth perception of traditional teleoperation approaches and provides freedom for human operators to easily switch observation perspectives (Chen et al. 2007; Roldán et al. 2019). Furthermore, users can overcome some constraints of the real world within immersive VR. For example, users can defy gravity to "fly" near the roof or move construction materials around without being encumbered by their physical weight. They can also receive augmented information that cannot be directly obtained from the real world. Studies also show that the robot operator's situational awareness is improved while working in VR (Roldán et al. 2017; Ruiz et al. 2015).

### Immersive Virtual Environment Construction

The immersive virtual environment (IVE) is the digital twin of the ROE, where the users can perceive real-time construction workspace conditions, robot states, and augmented information such as the as-designed building geometry from remote locations. It consists of a virtual construction environment and two full-scale virtual robots (Fig. 3). One robot demonstrates the motion plan to the user, referenced as the "planning" robot in the rest of the paper [Fig. 3(a)]. The other robot, referred to as the "execution" robot [Fig. 3(b)], is synchronized with the actual robot [Fig. 3(c)] so that

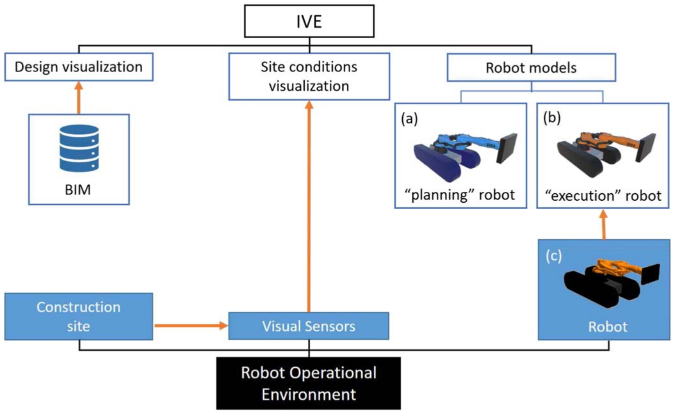


**Fig. 3.** IVE components scene graph.

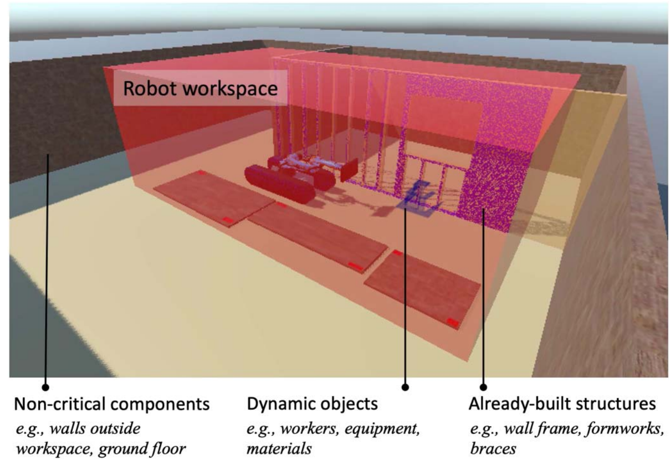


**Fig. 4.** Hybrid IVE construction.

the user can supervise the actual robot's execution process. The robot digital model, represented in URDF format, has the same size and configuration as the actual robot. It is first loaded in ROS and then transferred to VR via the ROS# library to be loaded as a game object. The VR robot models preserve the kinematic properties of the actual robot and can be controlled by subscribing messages from the middleware.

Some studies used 3D computer-aided design (CAD) building models, such as BIM, as VR construction environments (Du et al. 2018). It can be loaded into IVE conveniently. However, this approach does not reflect the latest construction site conditions because the as-built structure may deviate from the design. It also cannot capture the moving workers and temporary equipment and structures onsite during the construction process, which should be considered for task planning. Point clouds of real-time construction site conditions can be captured using laser scanners or depth cameras, but it is expensive to render large point cloud data in VR because of its high refresh rate (Fang et al. 2016). Wang et al. (2019) generated BIM models from point clouds, which can be imported into the IVE. However, the dataset labeling and training processes consume significant resources.

In order to visualize actual construction site geometry in near real-time [i.e., with minimal delays caused by automatic data processing and electronic transmission (US DOD 2005)] while reducing the computational load and time delay, this study proposes a hybrid approach to create the IVE of the construction site. Components in the construction environment are first grouped into three categories, noncritical components, already-built structures, and dynamic objects, as shown in Fig. 4. Noncritical components indicate objects outside the robot workspace (e.g., walls outside the workspace) or components inside the robot workspace but with limited deviations from the design that do not affect the user's decision making or robot's operation processes (e.g., ground floor). For noncritical components, their as-designed BIM is directly used in the VR scene as a realistic working environment for the user.

Already-built structures are static building components or temporary structures inside the workspace (e.g., columns and formwork) that are closely related to the user's decision making or robot operation process. The as-built geometry of these structures is captured by depth cameras or laser scanners onsite as point clouds. The point clouds are reconstructed into 3D meshes in the middleware and sent to the immersive VR interface via ROS# to be loaded as scene objects in IVE (Bischoff 2020). The reasons for converting point clouds of already-built structures into 3D meshes are two-fold. First, it could significantly reduce the system computational load of refreshing large-size real-time point cloud data at every frame. Second, it allows colliders to be added onto the already-built structures for collision avoidance during the high-level task planning process. Their BIM is also loaded into the IVE. However, these models are set as semitransparent and are only used for visualization purposes to show users any discrepancies between the as-designed and the as-built structures.

The dynamic objects include human workers and moveable equipment that might intrude into the robot workspace and obstacles that temporarily stay in the robot workspace, which will affect human decision making and robot operation. It is critical to track these objects in near real-time because they may be present and move in the robot workspace at any time. Once dynamic objects appear in the robot workspace, their point clouds captured by depth cameras or laser scanners onsite are rendered in the VR scene so that the user can view construction environment conditions in real-time.

In the proposed approach, the categories of the components are decided by manually defining regions or selecting components. However, the proposed framework can be integrated with building components detection and recognition algorithms (Bassier et al. 2019; Sharif et al. 2017; Wang et al. 2020a) to automatically detect and classify components from point clouds into proposed categories. In addition, the greedy projection triangulation algorithm (Marton et al. 2009) has been used for point cloud 3D reconstruction. Nevertheless, other point cloud reconstruction approaches could be used based on the needs of different cases.

### VR Interface Development

The immersive VR interface acts as a visualization tool for the augmented telepresence experience, a planning tool for users to perform high-level task planning, and a supervision tool for robot motion plan evaluation and real-time status supervision. It contains several interactive elements for the user to perform task planning, guide the robot, and receive information. One of them is the message display media. It shows instructions to users and system messages. It may contain an internal user interface (e.g., buttons and sliders) inside the VR scene as a supplement for handheld controllers to provide the user with additional functions sending commands and interacting with the system.

The interface also includes some task-specific interactive elements as part of the VR scene. For example, for the pick-and-place related construction activities (e.g., assembly, installation),

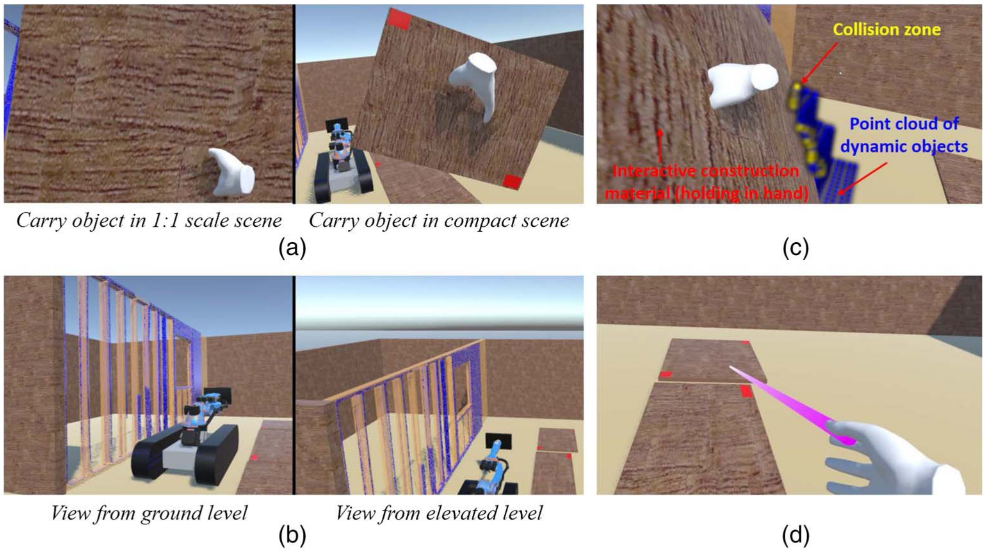


**Fig. 5.** Immersive VR interface interaction features: (a) scene scale adjustment; (b) elevation adjustment; (c) collision warning; and (d) material access sequence control.

the construction objects to be placed (e.g., bricks, panels) are included as interactive game objects in the VR interface. These interactive materials are of the same size and position as the actual construction materials onsite and can be grabbed, moved, and suspended in the air based on task needs. Users can use these elements to perform high-level task planning, indicate task goals, and guide the robot.

Several features have been developed in the interface to facilitate decision making and interaction processes (Fig. 5). The first feature is scene scale and viewpoint elevation adjustment. The users are given the ability to adjust the scene scale to be larger or smaller than the real world with handheld controllers during the interaction process [Fig. 5(a)]. The contracted scene can be used for general planning and supervision, while the enlarged scene can be used for detailed inspection and material pose fine-tuning. Furthermore, the user can adjust the elevation of their viewpoint to move around at any desired elevation to obtain an overview of the construction environment and inspect the geometry from the roof level [Fig. 5(b)].

The second feature is collision avoidance and checking. Colliders are added for the interactive construction materials, BIM models of noncritical components, and the 3D meshes of already-built structures. It provides collision protection at the user high-level planning stage because the user cannot place the construction materials in collision with the built structures. In addition, when users place the construction materials in collision with the dynamic objects, the part of the point cloud with collision will change its color as a warning. As shown in Fig. 5(c), the interactive construction material (wood panel) that the user is holding in hand collides with the point cloud of stacked boxes. As a result, the part of the point cloud that collides with the panel changes its color from blue to yellow.

The third included feature is material access sequence control. Following the practical convention, the system only allows the user to interact with materials stacked on the surface [Fig. 5(d)]. Once the material on the surface is removed, the material lying underneath is then set to be interactable. It should be noted that although this paper mainly discusses pick-and-place related cases, the system can be generalized to many different types of construction tasks (e.g., nailing, joint filling) after configuration.

### *Middleware*

ROS is used as the middleware for the proposed system, which is an integrative open-source robotic software framework (Quigley et al. 2009). It supports and can communicate with a variety of sensors, hardware, and robots. However, it is impractical for construction workers without robot programming expertise to operate robots directly through ROS. One of the reasons is that ROS is developed as a tool to facilitate robot programming. Although some software libraries in ROS provide operator interfaces, their availability and functionality are limited (Roldán et al. 2019). It is insufficient and is not intuitive to use when it comes to complex construction tasks that typically involve several procedures and objects. Therefore, in our framework, ROS is utilized as the middleware for communication between the human and the robot, robot motion planning, sensor fusion, and robot control. In this section, the techniques to establish the communication framework and conduct robot motion planning and robot control are introduced. Sensor fusion varies case-by-case and thus is discussed later in the case study.

#### Communication

The immersive VR interface and middleware communicates by exchanging ROS messages, including a variety of formats based on message types. The communication is established using ROS#, which is an open-source library developed for connecting ROS and Unity (Bischoff 2020). ROS can exchange messages with robots and their embedded sensors and environmental sensors with the MQTT communication protocol (Liang et al. 2020b). ROS can also communicate with robotics simulation software if ROE is in simulation. For example, an open-source meta-package, gazebo_ros_pkgs, can be used to exchange messages between ROS and the robot and sensor emulators in the Gazebo simulator (ROS Wiki 2020).

#### Robot Motion Planning

The motion planning method discussed in this study is based on the mobile industrial arm manipulator, which is a general case for construction robotics. Industrial robotic arms offer high DOFs and have high flexibility to be configured for a variety of complex construction tasks (Bock 2007; Liang et al. 2020a).



J. Comput. Civ. Eng., 2021, 35(6): 04021023

**Algorithm 1:** Pseudo Code for Motion Planning

**Input:** target $T$,
point cloud of dynamic objects $PCL_D$,
robot current pose $P$,
intermediate carrying pose (relative to robot base) $P_c$
**Output:** motion plan $MP$, success indicator *planSuccess*

**if** $\exists$ $PCL_D$ **then**
  Save $PCL_D$
**end**
**if** $\exists$ *Plan(P to T)* **then**
  $MP \leftarrow$ Plan($P$ to $T$)
  *planSuccess* $\leftarrow$ True //success (without moving base)
  **return**
**end**
$B_0 \leftarrow$ getBaseLocation($P$)
$S_{BT} \leftarrow$ getBaseLocations($T$)
find $B_T$ in $S_{BT}$, $\exists$ $MP_m \leftarrow$ Plan($B_0$ to $B_T$)
**if** no $B_T$ found **then**
  *planSuccess* $\leftarrow$ False // fail (no valid base movement path)
  **return**
**end**
$P_0 \leftarrow$ getPose($P_C$, $B_0$)
$P_T \leftarrow$ getPose($P_C$, $B_T$)
$MP_A \leftarrow$ Plan($P$ to $P_0$)
$MP_B \leftarrow$ Plan($P_T$ to $T$)
$MP \leftarrow$ concatenatePlan($MP_A$, $MP_m$, $MP_B$)
*planSuccess* $\leftarrow$ True //success (after moving base)

**Fig. 6.** Algorithm 1—Pseudocode for motion planning.

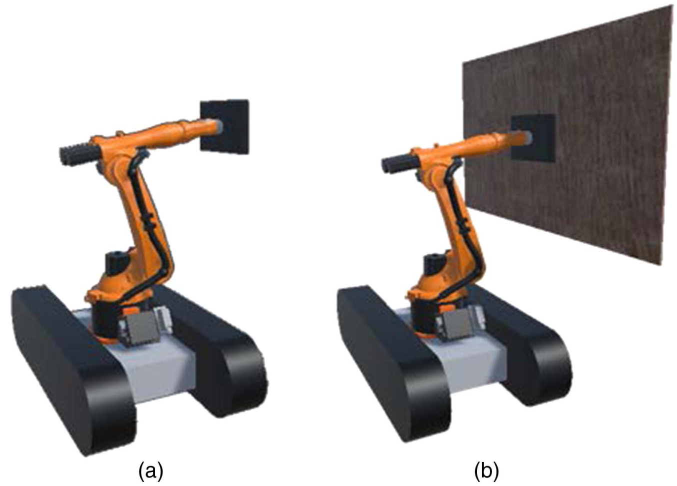


**Fig. 7.** Example of intermediate carrying pose: (a) without object; and (b) with object.

The motion plan is considered separately for the robot mobile base and the robotic arm. The robotic arm movement is given a higher priority than the mobile base movement. In other words, the robot will only move its base if its arm cannot directly find a motion plan to reach the target position from the target base location. This setting aims at reducing the localization error caused by frequent robot base movement. The robotic arm motion plan is generated by MoveIt (Chitta et al. 2012). The point cloud sensing data of the environment is processed into a 3D occupancy grid map with Octomap for collision avoidance (Hornung et al. 2013).

The task plan received from VR is first converted into the corresponding robot end-effector pose in ROS. Then, the Open Motion Planning Library (OMPL) (Sucan et al. 2012), which integrates several cutting-edge sampling-based motion planning algorithms, is used together with the Flexible Collision Library (Pan et al. 2012) to generate kinematics (i.e., position, velocity, and acceleration) of each joint to achieve the goal without collision. If the robot is carrying an object, the object is considered as part of the robot during the motion planning and collision checking process. As a result, both the robot and the object carried by the robot will not collide with the environment or with each other. The motion plan is only considered to be successful if it is collision-free.

The algorithm for motion planning is shown in Fig. 6. After receiving the target $\boldsymbol{T}$, the robot first checks whether there are any dynamic objects in its workspace by checking if there are point clouds other than the ones that represented the already-built building structure. If any point cloud of dynamic objects $\boldsymbol{PCL_D}$ is detected, the system will save it for future comparison at the execution stage. Then, the robotic arm attempts to find a motion plan $\boldsymbol{MP}$ to reach $\boldsymbol{T}$ from its original base location $\boldsymbol{B_0}$ (i.e., location of its base stays at without moving). If the robotic arm cannot find a plan after several attempts, it will try to move its base to a target base location $\boldsymbol{B_T}$ while holding the arm at an intermediate object carrying pose $\boldsymbol{P_c}$ (Fig. 7). The user can define the criteria to determine a set of $\boldsymbol{B_T}$ options $\boldsymbol{S_{BT}}$, which may contain one or multiple $\boldsymbol{B_T}$ near the target. For example, in our system, it is the nearest available location to the target on a specific path.

First, the robot checks whether there is a valid pathway to move its base from $\boldsymbol{B_0}$ to $\boldsymbol{B_T}$. For robots that move forward and backward along a straight line, this process can be achieved by setting a bounding box along the robot pathway and checking the occupancy within that bounding box using point clouds captured by onsite depth cameras. For robots that can move around the workspace freely, more advanced path planning algorithms like Dijkstra's Algorithm are needed to find a valid path (Dijkstra 1959). If multiple $\boldsymbol{B_T}$ options are determined, the robot will attempt all the potential options in $\boldsymbol{S_{BT}}$ until a valid pathway is found.

Once a valid pathway from $\boldsymbol{B_0}$ to $\boldsymbol{B_T}$ is found, the robotic arm will generate its motion plan $\boldsymbol{MP_A}$ to move to the intermediate carrying pose $\boldsymbol{P_0}$ (i.e., $\boldsymbol{P_c}$ with the base at $\boldsymbol{B_0}$) in preparation for the base movement. It will then generate another motion plan $\boldsymbol{MP_B}$ from the intermediate carrying pose $\boldsymbol{P_T}$ (i.e., $\boldsymbol{P_c}$ with the base at $\boldsymbol{B_T}$) to $\boldsymbol{T}$. These two robotic arm motion plans, together with the robot base movement plan $\boldsymbol{MP_m}$, are combined into the robot final motion plan $\boldsymbol{MP}$.

The system can generate and save several motion plans. The user can specify the cost functions (e.g., time duration) to sort motion plans so that the plans can be demonstrated within a certain order (e.g., time duration from short to long) in the VR interface based on users' preferences. To view the arm plan in VR, we extract discrete joint states from the generated motion plan and publish it to VR at its timepoint specified in the motion plan to move the "planning" robot in VR. For the mobile base movement plan visualization, we simulate the base movement plan in the middleware by selecting discrete location points along the base movement path, publish it to VR at a given frequency, and have the "planning" robot move to certain points while maintaining its arm pose as the intermediate carrying pose.

### Robot Control and Execution

The approved trajectory plan is converted into robot control commands with the *ros_control* package, which generates output to real robot actuators with PID controllers according to the motion plan (Chitta et al. 2017). When the robot is executing the task, joint states and location data from the encoders of the robot actuators are obtained and sent to VR. As the "execution" robot in VR receives the data, it adjusts its joint states and location to synchronize itself with the real robot.

The construction environment is relatively open and dynamic. Even though the robot is not designed to share the workspace with workers and other moveable equipment, they may still accidentally intrude into the robot workspace. Therefore, instead of blindly following the trajectory approved by the user in VR, safeguard functions are needed during the execution process to prevent accidents.

As mentioned, at the start of the planning stage, the point cloud of the dynamic objects is saved for future use. These objects are considered for collision avoidance during motion planning. Therefore, it is acceptable if the objects stay at the same place during the execution process. However, if point clouds other than the previously detected ones are detected in the workspace, it means that either the workspace is intruded upon or the previously detected objects are moved after planning. As soon as an intrusion is detected, the robot will stop emergently. The user can inspect the site condition and request the robot to replan its motion based on the latest environment. If the robot workspace is very large, it can be separated into different areas. The robot will only stop if a certain area is intruded upon. In addition to the system safeguard functions, the user has the privilege to stop the robot with the handheld controller at any time.

## Case Study and Experiments

A drywall installation case study involving imperfect rough carpentry (wall framing) is used to demonstrate the proposed I2PL-DT HRC system. For some complex systems, there are several distinct and interdependent technologies and subsystems that need to come together before the system can be computationally analyzed or applied in real world settings. For example, a variety of technological advancements (e.g., perception, localization, hardware design) are needed for a construction robot to successfully perform construction activities onsite (Lundeen 2019). Instead of attempting to address all these challenges at once, we focus on verifying that the proposed I2PL-DT system framework and its associated modules allow human workers to interact with and collaboratively perform construction tasks with the robot; as well as receiving feedback to further improve our system in the future.

The use of virtual simulators such as ROS Gazebo is the first step of evaluating the feasibility of this new method as indicated by several existing studies such as Lin and Berenson (2016) and Murali et al. (2020). Gazebo is a robotics simulator with a robust physical engine that allows rapid prototyping of robotic tasks and direct subsequent transfer of the methods to the corresponding real robotic platforms (Koenig and Howard 2004). When connected with ROS, Gazebo is capable of communicating with real physical robots with high accuracy. It has been demonstrated that a real KUKA KR120 robotic arm can be synchronized with its emulator in Gazebo with average errors of each joint angle less than $2.4 \times 10^{-5}$ in radians (Liang et al. 2020b). In addition, Gazebo allows emulation of unstructured and dynamic construction site conditions such as generating dynamic objects, which would be especially useful for offline system testing before physically deploying the system on actual construction sites. Therefore, the case study utilizes a 6DOF KUKA industrial robotic arm emulator mounted on a tracked mobile robotic platform, which is capable of construction work. The ROE, including the construction site, sensors, and the robot, is emulated in Gazebo.

With the focus on demonstrating the interaction framework between the human worker and the robot, three assumptions are made and considered reasonable because they have already been extensively studied in the literature. First, the case study assumes accurate registration of the IVE and ROE (Feng et al. 2015). Second, it is assumed that the construction materials are firmly placed (i.e., no sliding) and their placement locations are known (Son et al. 2010). Third, we assume the robot can accurately localize itself onsite (Lundeen et al. 2017; Xu et al. 2020).

Verification is defined as the "process of evaluating a system or components to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase" (IEEE 2017). It involves special tests to model a subsystem (e.g., developing scenarios as proof-of-concept implementation) or using repeating tests to ensure the system meets initial design requirements. For an interfacing system like the one proposed in this study, proof-of-concept implementation is used as verification to confirm that all the modules of the proposed system can work well with each other to reach the goal (Ge and Kuester 2015; Kim et al. 2012, 2021; Kurien et al. 2018). Some studies also conducted user tests as the preliminary usability study (Akanmu et al. 2020; Chen et al. 2016; Mantha et al. 2020; Quintero et al. 2015). In this study, we presented three scenarios from the drywall installation case study as proof-of-concept implementation. The drywall installation system setup and the technical details of the three scenarios are discussed in depth. A human-in-the-loop study with 20 subjects is also conducted as the preliminary usability study of the system, in which the subject guides the robot to pick up different types of drywall panels stacked on the ground and places the panels on a wall frame which is built with deviations from design. Feedback and suggestions from human subjects are used for system evaluation and to propose future improvements.

### *Digital Twin Environment Setup*

We emulated the ROE in Gazebo [Fig. 8(a)]. An imperfect wall frame with a window opening has already been constructed. A few pieces of drywall panels in three different sizes are stacked near the wall frame. A robotics arm emulator on a tracked mobile robotic base is ready for conducting the work. The environment also contains a few Microsoft Kinect camera emulators, which are fixed at certain locations, facing the wall frame and the robot.

The VR digital twin of the ROE is created in Unity [Fig. 8(b)]. Some stacked drywall panels of the same size and position as the ones in the ROE are created as interactive construction materials, which will be used for high-level task planning and robot guidance. Only the pieces sitting on the top of each stack are activated to be interactable. As the top one is removed, the interactivity of the piece below is activated.

An interactive billboard is developed as an integration of the display media and the internal user interface. The interactive billboard can be separated into three functional zones, as shown in Fig. 9. The upper zone is used to display augmented robot status messages (e.g., robot planning) and instructions to users (e.g., robot needs to pick first). The middle zone is the function panel. It provides some functions to facilitate user's interaction with the VR scene. The function provided by each button in this panel is summarized in Table 3. The bottom zone is the command panel for the user to send instructions to the robot. It consists of four buttons, "Pick," "Place," "Hold," and "Release." The detailed usage of these buttons is introduced along with the scenarios in the following subsections.

### *Sensor Fusion*

The type and number of sensors to use and their placement should be decided according to the environment and the type of work the robot will conduct. The sensors should be able to provide sufficient

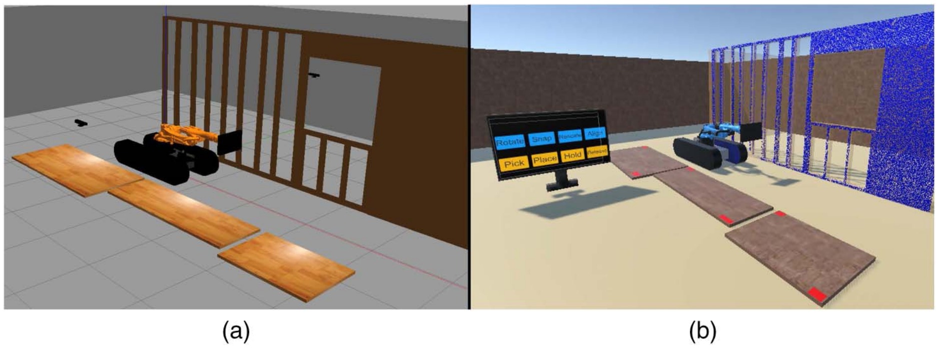


**Fig. 8.** Digital twin environment settings: (a) robot operational environment; and (b) immersive VR environment.

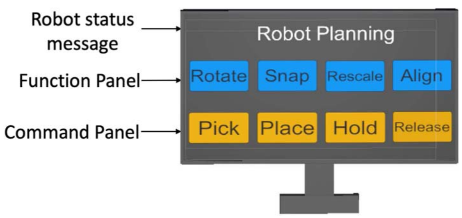


**Fig. 9.** Interactive billboard functional zones.

**Table 3.** Buttons on the functional panel

| Button | Function |
|---|---|
| Rotate | Open a new page with options to snap the target panel orientation along the X-, Y-, and Z-axis to certain angles quickly and accurately. |
| Snap | Snap the target panel to be side by side and at the same orientation with a designated previously installed panel. |
| Align | Show highlighted vertical lines from each corner of the drywall to the ground panel for users to check panel alignment with the wall frame. |
| Rescale | Quickly rescale the VR scene back to a 1: 1 scale. |

information to support human high-level planning, robot trajectory planning and collision avoidance, and any customized functions to achieve the specific goal of the system. In our case study, four Microsoft Kinect depth cameras are used to visually capture the drywall installation workspace in Gazebo. We chose Kinect since the work is done indoors and Kinect offers acceptable performance under such conditions. They are fixed in the construction environment instead of being installed onboard the robot because the views of cameras mounted on the robot can be easily occluded when the robot is carrying large construction objects. In addition, fixed cameras have lower noise compared to cameras mounted on robots. While depth cameras such as Kinect have relatively lower costs, their performance is limited in outdoor environments because they use infrared sensors to capture depth data and have limited measurement ranges (Liu et al. 2019). Therefore, for outdoor construction tasks or large robot workspaces, 3D laser scanners or stereo cameras that have larger measurement ranges and better outdoor performance should be considered for point cloud capture (Wang et al. 2020a). The process of sensor data processing is shown in Fig. 10.

First, the RGB and depth images captured by each Kinect camera are converted into point clouds and concatenated into a single point cloud. The point cloud is downsampled with the voxel grid filter [Fig. 11(a)]. The downsampled point cloud is then sent to MoveIt and goes through the self-filtering process. Self-filtering removes the points that represent the robot itself [Fig. 11(b)]. The ground plane, stacked drywall panels, and Kinect cameras installed onsite are added as collision objects in MoveIt. As a result, they are considered for collision checking, but their point clouds are removed by self-filtering.

The point cloud after self-filtering is converted into a 3D occupancy grid map using Octomap for collision avoidance during motion planning (Hornung et al. 2013). After that, a PassThrough filter is used to separate the point cloud of the already-built structure [Fig. 11(c)], which is then converted into a 3D mesh with the greedy projection triangulation algorithm (Marton et al. 2009) and sent to Unity, and dynamic objects [Fig. 11(d)], which is sent to Unity for visualization after further downsampling and updates in real-time. The point cloud for dynamic objects is also used to detect workspace clearance.

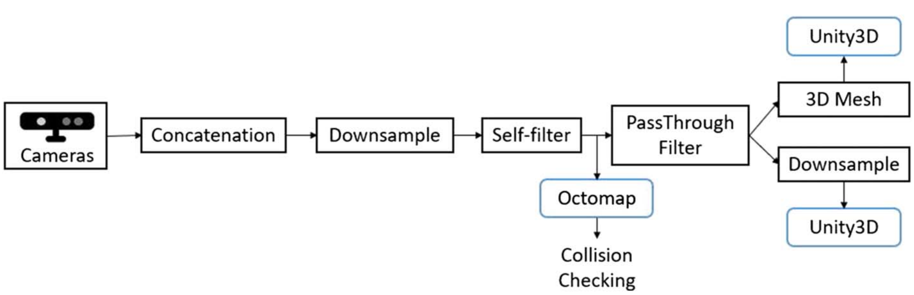


**Fig. 10.** Sensor data processing.



J. Comput. Civ. Eng., 2021, 35(6): 04021023

## *Experimental Verification Scenarios*

### Working Sequence Guidance

Most construction tasks involve the installation of materials in a certain order (e.g., top to bottom, large to small). According to the wall frame condition, the user can control the work sequence, including determining the order of conducting tasks and selecting the specific workpiece to manipulate.

The user can aim the controller at the panel they want to install next and grab it. Then, the user can place the panel onto the wall frame at their preferred position and orientation. The pose of the panel can be fine-tuned several times at different scene scales until the user is satisfied. As the user confirms the task plan, the position ($\boldsymbol{P}'$) and orientation ($\boldsymbol{Q}'$) of the target panel are sent to the middleware. After receiving the task goal, the middleware processes the target panel pose into the target end-effector pose. The target panel pose is first converted from the VR world coordinate ($\boldsymbol{P}', \boldsymbol{Q}'$) to the ROS world coordinate ($\boldsymbol{P}, \boldsymbol{Q}$) with Eq. (1)

$$\begin{aligned} [P_x, P_y, P_z] &= [P'_z, -Px', P'_y] \\ [Q_x, Q_y, Q_z, Q_w] &= [-Q'_z, Qx', -Q'_y, Q'_w] \end{aligned} \tag{1}$$

It is then subsequently converted into the end-effector pose ($\boldsymbol{P_E}, \boldsymbol{Q_E}$) with Eq. (2)

$$\Delta \boldsymbol{P} = \begin{bmatrix} 1-2Q_yQ_y-2Q_zQ_z & 2Q_xQ_y-2Q_zQ_w & 2Q_xQ_z+2Q_yQ_w \\ 2Q_xQ_y+2Q_zQ_w & 1-2Q_xQ_x-2Q_zQ_z & 2Q_yQ_z-2Q_xQ_w \\ 2Q_xQ_z-2Q_yQ_w & 2Q_yQ_z+2Q_xQ_w & 1-2Q_xQ_x-2Q_yQ_y \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dfrac{T_a+T_b}{2} \end{bmatrix} \tag{2}$$

$$\boldsymbol{P_E} = \boldsymbol{P} + \Delta \boldsymbol{P}$$

$$\boldsymbol{Q_E} = \boldsymbol{Q}$$

where $T_a$ = thickness of the drywall panel; and $T_b$ = thickness of the gripper.

There are four types of instructions the user can send through the command panel on the interactive billboard, "Pick," "Place," "Hold," and "Release." Because the robot uses a vacuum gripper, it does not directly reach the target end-effector pose while picking or placing objects with the "Pick" and "Place" commands. Instead, the gripper needs to first pause at an intermediate prepick or preplace pose, which is 10 cm before it reaches the target. The intermediate end-effector pose can also be calculated with Eq. (2) by replacing $(T_a + T_b)/2$ with 10 cm. Then, the robot end-effector follows the cartesian path to move from the intermediate pose to the target. The cartesian motion is divided into several small steps at a resolution of 1 cm. If a collision is detected before the target position is reached, the robot will stop.

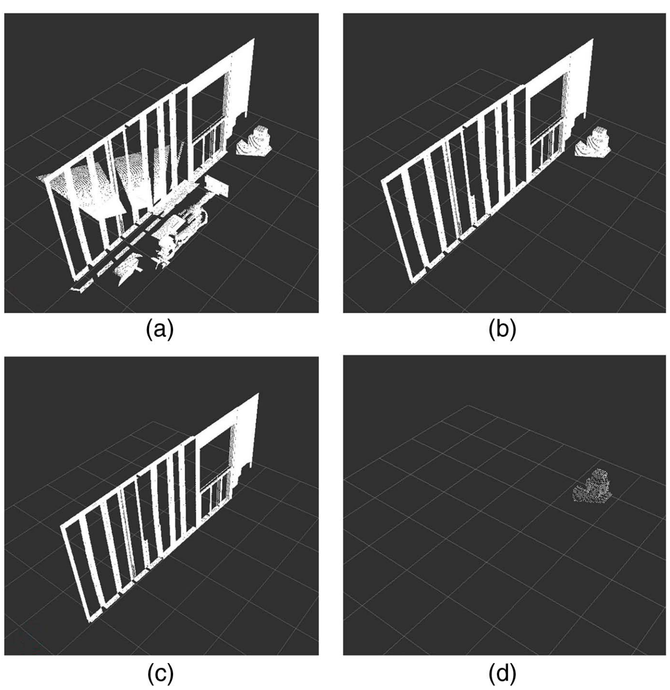


**Fig. 11.** Point cloud processing procedures: (a) concatenated point cloud; (b) self-filtered point cloud; (c) already-built structures; and (d) dynamic objects.

For the "Pick" command, the robot picks up the drywall panel as soon as it reaches the target or the collision point. For the "Place" command, the robot will wait until the user presses the "Release" button to release the drywall panel from the end-effector, indicating that the user confirmed the drywall was secured (e.g., screwed or nailed) and is safe to release. For the "Hold" command, the robot directly moves to the target (without pausing at the intermediate pose) and waits for another command before taking any action.

By repeatedly specifying target panels, installation positions, and guiding the robot through the pick-place or pick-hold-place installation process, the user can collaboratively work with the robot to complete a series of construction activities in a specific work sequence. The snapshot graphs in Fig. 12 show the work sequence guidance process of installing four drywall panels onto the wall frame. The yellow arrows point to the human-specified targets, which provide information in terms of types of panels and the target installation positions at this step. The four figures in each step show the real robot in ROE and the virtual VR robot picking up the corresponding panel and placing it in the target position.

### Optimal Motion Plan Selection

Although OMPL can plan trajectories for the robot to reach the target, it does not guarantee that the trajectory is optimal. Therefore, the proposed system requires the middleware to generate

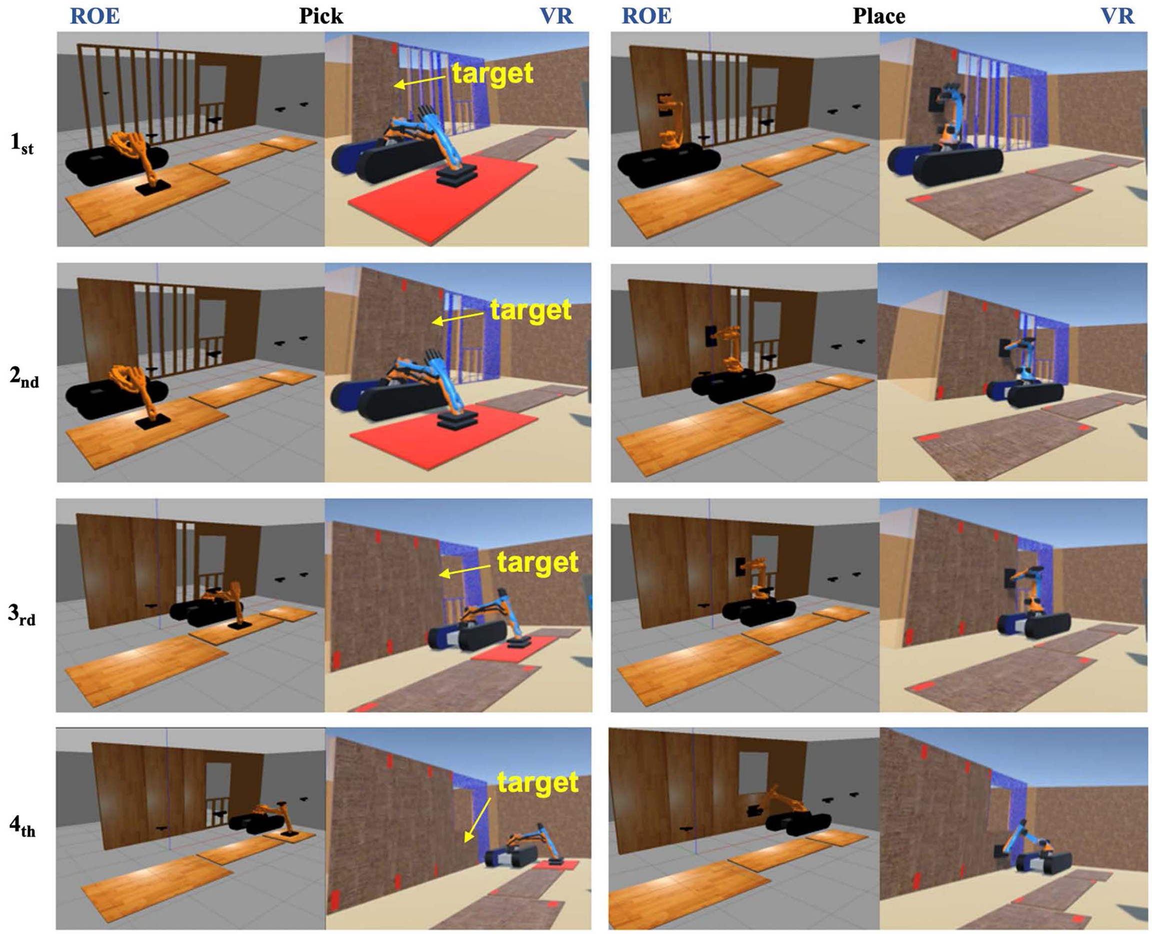


**Fig. 12.** Work sequence guidance process demonstration.

multiple motion plans and allows the user to select the most desirable plan after viewing generated plans in VR.

The robot motion consists of several planning stages. Preprocessing is needed so that entire planned motions can be viewed by the user before the actual robot can take any action. This depends on the type of command the user gives and whether the robot base movement is needed, as shown in Table 4.

Since the processes for *MoveBase* and *CartesianMotion* are relatively monotonous in this case study, multiple motion plans are developed for *ArmToCarryPose* and *ArmToPrePick/Place/Hold* stages only. For each of these stages, five stage-level motion plans are generated. After concatenating stage-level plans into entire motion plans that cover all needed stages, four entire motion plans with the shortest time durations are saved. The entire motion plan with the shortest time duration is demonstrated to the user first. If the user is satisfied with the plan, they approve it by pressing a controller button and the robot will execute the plan. Otherwise, the next plan, the one with the second shortest time duration is demonstrated, and so on. Fig. 13 shows the snapshots of robot execution supervision processes for the different stages of picking operation with the temporal order from left to right. The description of each stage is given at the bottom. The execution robot the user sees in VR is synchronized with the real robot in ROE.

## Trajectory Guidance with Intermediate Object Poses

The construction environment presents more challenges to the robot motion planning process than ordinary robot working environments because of its complexity. Therefore, the motion planner might fail to develop a motion plan or the robot might be stuck at some locations even if a valid trajectory exists. Even though in some situations the robot can find valid motion plans, the user may have preferences for the robot to perform the task in a specific way.

**Table 4.** Motion plan preprocessing for visualization in different situations

| Base | Command | Planning stages |
|---|---|---|
| Move base | Pick | *ArmToCarryPose* + *MoveBase* + *ArmToPrePick* + *CartesianMotion* |
| | Place | *ArmToCarryPose* + *MoveBase* + *ArmToPrePlace* + *CartesianMotion* |
| | Hold | *ArmToCarryPose* + *MoveBase* + *ArmToHold* |
| Not move base | Pick | *ArmToPrePick* + *CartesianMotion* |
| | Place | *ArmToPrePlace* + *CartesianMotion* |
| | Hold | *ArmToHold* |

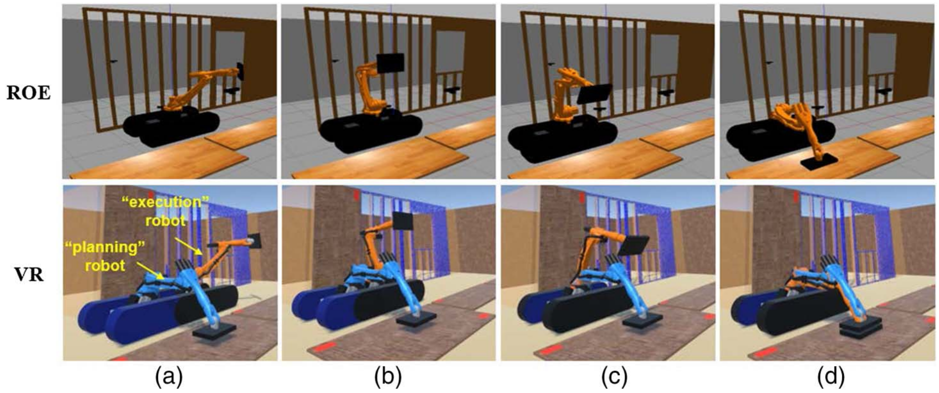


**Fig. 13.** Robot execution processes demonstration: (a) arm to carry pose; (b) move base; (c) arm to prepick; and (d) cartesian motion.

Some existing studies allow users to guide the robot by specifying end-effector paths or waypoints (Fang et al. 2012; Ong et al. 2020). However, it is very challenging for users to specify collision-free paths or waypoints when the manipulated object is large and the workspace is complex. In addition, paths and waypoints only possess the end-effector position information. When the object is large, its orientation on the trajectory is also important and can make a notable difference. Therefore, in the proposed system, the user guides the robot by specifying the intermediate object poses on the trajectory. The user sets the poses by placing the interactive drywall panels at desired positions and pressing the "Hold" button. The robot carries the panel to the intermediate poses and holds the panel to wait for another command. Multiple intermediate poses can be specified to guide the robot trajectory step by step. Fig. 14 demonstrates the process that the user guides the robot trajectory in four steps from top to bottom. In each step, the user specifies an intermediate object pose (Target 1–3) by placing the virtual panel and finally guides the robot to the final installation target.

### Human-in-the-Loop User Study

We conducted a human-in-the-loop user study as a preliminary usability test of the I2PL-DT system on collaborative human–robot construction work and to receive feedback and suggestions for future improvements. There were 20 subjects recruited to perform the drywall installation task with the proposed system. The main objective of the user study is to verify that subjects who were not involved in the system development process and unfamiliar with

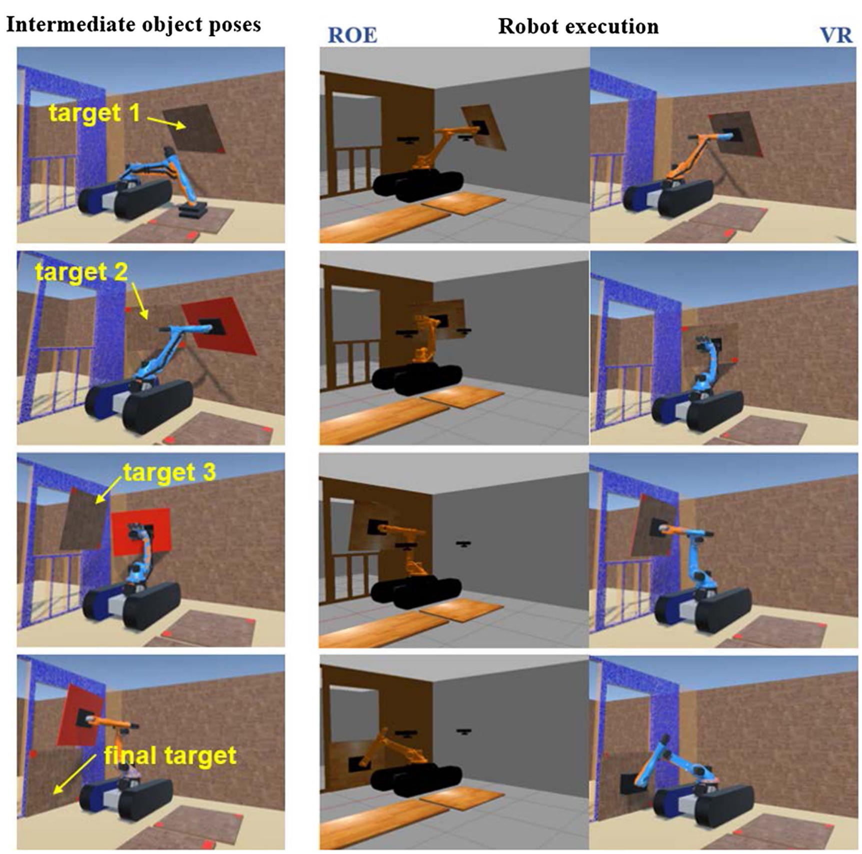


**Fig. 14.** Trajectory guidance with intermediate object poses.

the system's technical background can successfully use the system to collaborate with the robot and achieve task objectives. Following the drywall installation task, a survey is carried out to collect user feedback for improving the system functions and interaction design. The survey supports three main functions: (1) evaluate the general usefulness, effectiveness, and understandability of the system, (2) assess system functions and user experience with the VR interface, and (3) elicit user feedback and suggestions. In this section, the experimental protocol is introduced and the quantitative ratings from users are analyzed according to their feedback and suggestions.

### Experimental Protocol

The experiments were conducted one subject at a time in a university research laboratory following all health-informed safety guidelines in place at the time on account of the Covid-19 pandemic. The experimental protocol was approved by the Institutional Review Board at the University of Michigan. There were 20 subjects, eleven female and nine male, who were recruited and completed the experiment. Several prior studies have invited college students to test a system at its prototype stage to assist with system verification and design (Akanmu et al. 2020; Chen et al. 2016; Mantha et al. 2020; Quintero et al. 2015). Since the VR interface of the proposed system is fundamentally different from the traditional drywall installation approaches, we invited graduate students, who have basic knowledge of visualization and are more familiar with gaming and computer technologies, as the users for our study at this stage. Most subjects have civil engineering, construction, and/or robotics backgrounds, and they were introduced to the basic drywall installation knowledge at the start of the experiment. This allows for minimal training time before they can perform the requested task.

The timeline of the experiment can be found in Fig. 15. As the experiment started, the researcher spent 15 min introducing the experiment to the subject and answer their questions. Next, the subject put on the headset and completed a trial session to get familiar with the system.

After trial, the system was reset and the subject was given 30 min to perform the main task. The environment settings of the main task are the same as shown in Fig. 8. The subject was requested to install four drywall panels vertically onto the wall frame. Installations of the first three panels were implemented with the pick-place approach. As the subjects got more familiar with the system while installing the first three, they were requested to use the pick-hold-place approach for the last panel by indicating intermediate object poses on the trajectory. The subject made their own decisions on the working sequence and the type of drywall to install for each operation.

The subject was asked to take a survey after the main task. The survey contained five different sections. The first section asked the subjects' basic information and their task completeness. In the second to the fourth sections, subjects evaluated different aspects of the system with a 7-point scale, where 7 represents the most positive evaluation and 1 represents the most negative evaluation. In the last section, subjects provided written comments on the system and made suggestions for improvements.

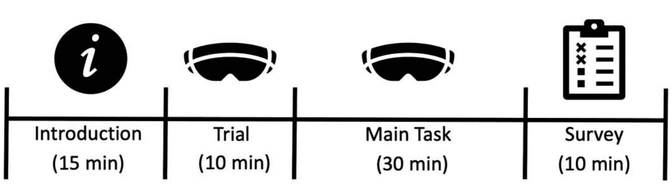


**Fig. 15.** Experiment timeline.

### Results

All subjects were able to use the proposed I2PL-DT system and take advantage of the provided system functions to collaborate with the robot. Out of 20 subjects, 16 completed the installation of all four panels during the 30-min main task period. In addition, 19 out of 20 subjects successfully noticed and avoided the deviation on the wall frame. An approximate productivity comparison between the subjects using the I2PL-DT system and the standardized data of an experienced carpenter (assuming they will complete the installation by themselves) obtained from RSMeans data (Gordian/RSMeans Data 2019) has been performed. Fig. 16 shows the cumulative time taken to install one to all four panels with the orange line indicating the average time taken for all subjects and the box plot illustrating the time distribution among subjects. The standardized RSMeans data is shown in the dark red line. The RSMeans database uses the area of panels installed as the output to quantify productivity. Therefore, the time required to install a larger panel is proportionally longer than a smaller one. In this comparison, it is assumed that the panels are installed in the order from larger size to smaller size when calculating with RSMeans. While a robot performs the task, the time taken for installing a larger and smaller panel is almost the same, and the panels are installed with an individual subject's preferred order. Although it is an approximate comparison, it can clearly show that the proposed system takes significantly less time than traditional methods in addition to reducing the physical stress and increasing the safety of construction workers.

In the second section of the survey, subjects gave a general assessment of the system's usefulness, effectiveness, and understandability. The mean and standard deviation of ratings along with the box plots are shown in Fig. 17. Subjects generally thought that the system is very useful and understandable. However, the ratings of system effectiveness are relatively lower. One of the comments we received is that the plan preview process almost doubles the time needed because the subjects first previewed the motion plan animation and then supervised the robot to execute the same plan. Although safety is ensured, the subjects reflected that the process reduces the overall equipment effectiveness and thus the job progress is not optimal. Nevertheless, even with the plan preview process, the approximate productivity comparison between the subjects using the I2PL-DT system ($9.20\ \mathrm{m}^2/30$ min) and RSMeans data [$6.64\ \mathrm{m}^2$ (71.43 S.F.)/h] shows that subjects' productivity

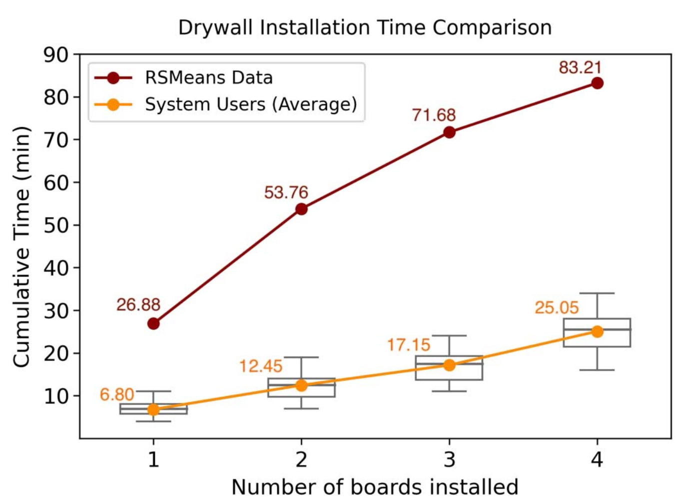


**Fig. 16.** Drywall installation time comparison.

| Indicator | Mean | SD |
|---|---|---|
| *Usefulness* | 6.200 | 0.616 |
| *Effectivenss* | 5.500 | 0.827 |
| *Undertandability* | 6.250 | 0.910 |

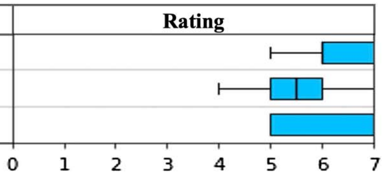


**Fig. 17.** General assessment results.

with the proposed system is highly comparable to that of experienced construction workers (Gordian/RSMeans Data 2019).

In the third section, subjects evaluated different system functions (Fig. 18). The functions correspond with the seven useful characteristics for HRC construction systems (Table 1). Subjects thought the motion plan evaluation process is clear and that they could easily and clearly supervise real robot states with the VR robot models. They can understand the differences between as-designed and as-built geometry. However, they have relatively lower satisfaction with the automatic motion plan generated by the system since there are some unnecessary rotations on the panel that reduce the system efficiency. Some subjects also expected the robot to move faster.

Some subjects experienced difficulties communicating with the robot because they were not familiar with the usage of the handheld VR controller, especially when their eyes were covered by the VR headset. For the information from the robot, they suggested adding haptic and sound feedback in addition to visual messages. One subject suggested showing messages in front of the users' view instead of showing them on the TV screen. For the high-level task planning, a subject suggested planning the installation poses of all four panels before the robot starts to develop the motion plan and perform the task instead of planning and installing the panels one-by-one. In addition, one subject suggested adding a function to indicate whether the intermediate pose the user selected for the robot to temporarily hold the object is valid (i.e., within the robot's reaching range).

In the fourth section of the survey, the subjects were asked to evaluate their VR user experience from eight aspects. The assessment results are shown in Fig. 19. The questions in this survey are adapted from the presence questionnaire developed by UQO Cyberpsychology Lab (UQO Cyberpsychology Lab 2004). Subjects were generally satisfied with their VR experience. They indicated that the interaction with the VR environment is natural and they could well-anticipate system responses to their actions. They could quickly adapt to the VR experience and visually search the environment for information they need. However, some subjects reported they experienced some difficulties manipulating the panel. Even though they can scale down the scene, the panel still blocked their vision to some extent because it was very close to their body when they hold it in their hand. Haptic feedback would also be helpful for accurate manipulation of the drywall panel. In addition, several subjects reflected that they experienced motion sickness after working for a while in VR and the handheld controllers were

| Indicator | Characteristics | Mean | SD |
|---|---|---|---|
| *Construction site condition visualization* | 2 | 5.600 | 0.598 |
| *As-design VS. as-built* | 3 | 5.750 | 1.164 |
| *High-level task planning* | 4 | 5.750 | 0.550 |
| *Robot automatic motion plan* | 4 | 4.900 | 1.518 |
| *Motion plan evaluation* | 5 | 6.150 | 0.933 |
| *Real robot states supervision* | 6 | 5.850 | 0.933 |
| *Communicate to robot* | 7 | 5.550 | 0.686 |
| *Information from robot* | 7 | 5.400 | 1.046 |

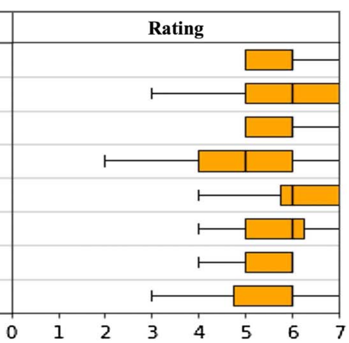


**Fig. 18.** System functions assessment results.

| Indicator | Mean | SD |
|---|---|---|
| *Involvement* | 5.900 | 0.912 |
| *Events control* | 5.500 | 0.888 |
| *Visual search* | 6.000 | 0.858 |
| *Movement and Manipulation* | 5.450 | 1.050 |
| *Interaction (Natural)* | 6.050 | 0.826 |
| *Interaction (Overall)* | 5.650 | 0.933 |
| *Concentration* | 5.750 | 0.910 |
| *Adaption* | 6.000 | 1.076 |

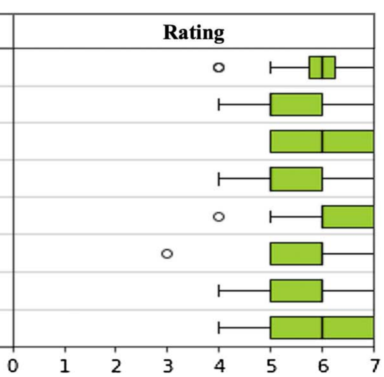


**Fig. 19.** VR user experience assessments results.

not sensitive enough to show the laser pointer in some situations, affecting their concentration and overall interaction experience. Some subjects reported delays in graphics rendering near the end.

## Discussion

Overall, subjects felt positive about the proposed I2PL-DT system for HRC construction. In addition, the valuable feedback from subjects has provided us with remarkable insights for improving the proposed system. To reduce the work progress delay caused by motion plan preview while ensuring safety, the motion plan demonstration speed can be adjusted to be faster than the real robot's movement speed to save plan preview time. Another possible solution to this problem is to offer the users the option to skip previewing the motion plan with a disclaimer that it is better to preview to ensure safety and prevent unexpected accidents. However, the real robot's movement speed could not be made faster because of the hardware limitation of the robot. Several changes could be made to improve the system experience, including adding haptic and audible feedback to facilitate communication and improve object manipulation, showing messages in front of the user's view, making the panel semitransparent when being grabbed to preserve users' vision while manipulating large-size objects, allowing several steps of high-level task planning to be developed at once, validating the intermediate poses for robot "Hold" operation, and optimizing the system for fast rendering. Interface design techniques should be used to reduce motion sickness. More advanced motion planning algorithms should be developed to reduce unnecessary robot movement. Moreover, systematic training is needed to get users familiar with the system before applying it to real-world construction projects.

In addition to user feedback, the authors identify several limitations to be addressed in future work. First, the robotic arm has a fixed intermediate pose to carry the object while moving its mobile base, which will not change during the moving process. In the future, algorithms will be developed for the robotic arm to dynamically adjust its pose according to the environment geometry while moving the robot base. This will provide more flexibility in collision avoidance for a mobile robotic arm carrying large-size objects.

Second, the robot will follow the exact motion plan once the plan is approved. If a new object appears in the robot's workspace during execution, the robot will terminate its execution and wait for instructions from human workers for safety reasons. In future work, the robot's autonomy can be enabled out of human supervision to dynamically adapt its path during execution along with the investigation of how to reduce the impact of certain autonomy on system safety.

Third, this study assumes the material stacked position is known to the robot. In fact, the materials stacked onsite might be moved from time to time and the position recorded might not be accurate. Computer vision and deep learning-based approaches can be used for the robot to automatically detect materials.

Fourth, there will be time delays caused by point cloud processing and rendering and data exchange, especially when the working environment is large and complex. In the future, more advanced optimization algorithms and computational power could be used for the system to stay close to real-time for large-scale projects.

Lastly, since the main focus of this study is the system framework design and its verification instead of field tests, the experiments are conducted in simulation primarily to ensure user and workspace safety in this first set of experiments. The authors are exploring gripper hardware design that is capable of large construction object manipulation (e.g., drywall panels) and improving the bidirectional communication for state synchronization between the Gazebo simulation and the physical robot to experiment on real KUKA KR120 robotic arms (Liang et al. 2020b). We will also conduct human factor studies and invite construction workers to use the system to improve system design and investigate the long-term effects of the system on workers' physical workload, mental stress, and job satisfaction.

## Conclusions

This paper proposed an I2PL-DT system for construction workers without robot programming expertise to remotely collaborate with construction robots to perform construction work. The proposed system has several contributions. First, it uses immersive VR and proposes a hybrid approach to create an augmented telepresence experience for the human workers while preventing them from being exposed to potential hazards on construction sites and allows people with physical disabilities to participate in performing construction activities. Not only can workers access pertinent information they will obtain by actually present on construction sites, but they can also obtain augmented information that they cannot directly perceive onsite, such as the as-designed building model and robot status information.

Second, the system provides an intuitive interface to assist human workers to perform high-level task planning. The user can specify task objectives by interacting with virtual objects in VR and try different options without exerting substantial physical effort or using up actual resources.

Third, human effort can be notably reduced since the robot is responsible for planning collision-free trajectories after receiving the task objectives. The user can also guide the robot to perform the task by specifying intermediate object poses on the trajectory.

Fourth, the system enables seamless bidirectional communication between the human worker and the robot and allows real-time robot status supervision. The human worker can easily send task objectives or commands to the robot by clicking buttons on handheld controllers. It also allows supervision of robots' intentions, actual states, and implicit robot status information in VR.

Overall, the proposed system offers a promising approach for construction workers to collaborate with onsite construction robots from remote locations and demonstrates the potential of transitioning the role of construction workers from physical task performers to robot supervisors, laying the groundwork for future construction work at the collaborative human–robot frontier.

## Data Availability Statement

Some data, models, or codes that support the findings of this study are available from the corresponding author upon reasonable request. The models and codes that can be requested from the corresponding author are working environment 3D models and code for robot planning and control. All anonymized human subject data are also available upon request.

## Acknowledgments

The authors would like to acknowledge the financial support for this research received from the US National Science Foundation (NSF—FW Award No. 2025805). Any opinions and findings in this paper are those of the authors and do not necessarily represent those of the NSF.

## Supplemental Materials

Video S1 is available online in the ASCE Library (www.ascelibrary.org).

## References


Akanmu, A. A., J. Olayiwola, O. Ogunseiju, and D. McFeeters. 2020. "Cyber-physical postural training system for construction workers." *Autom. Constr.* 117 (Sep): 103272. https://doi.org/10.1016/j.autcon.2020.103272.

Bassier, M., B. Van Genechten, and M. Vergauwen. 2019. "Classification of sensor independent point cloud data of building objects using random forests." *J. Build. Eng.* 21 (Jan): 468–477. https://doi.org/10.1016/j.jobe.2018.04.027.

Bilberg, A., and A. A. Malik. 2019. "Digital twin driven human–robot collaborative assembly." *CIRP Ann.* 68 (1): 499–502. https://doi.org/10.1016/j.cirp.2019.04.011.

Bischoff, M. 2020. "ROS#." Accessed May 4, 2020. https://github.com/siemens/ros-sharp.

Bock, T. 2007. "Construction robotics." *Auton. Robots* 22 (3): 201–209. https://doi.org/10.1007/s10514-006-9008-5.

Brosque, C., G. Skeie, and M. Fischer. 2021. "Comparative analysis of manual and robotic concrete drilling for installation hangers." *J. Constr. Eng. Manage.* 147 (3): 05021001. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002002.

Chen, J. Y. C., E. C. Haas, and M. J. Barnes. 2007. "Human performance issues and user interface design for teleoperated robots." *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* 37 (6): 1231–1245. https://doi.org/10.1109/TSMCC.2007.905819.

Chen, Y.-C., H.-L. Chi, S.-C. Kang, and S.-H. Hsieh. 2016. "Attention-based user interface design for a tele-operated crane." *J. Comput. Civ. Eng.* 30 (3): 04015030. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000489.

Chitta, S., et al. 2017. "ros_control: A generic and simple control framework for ROS." *J. Open Source Software* 2 (20): 456. https://doi.org/10.21105/joss.00456.

Chitta, S., I. Sucan, and S. Cousins. 2012. "MoveIt! [ROS Topics]" *IEEE Rob. Autom. Mag.* 19 (1): 18–19. https://doi.org/10.1109/MRA.2011.2181749.

Chotiprayanakul, P., D. K. Liu, and G. Dissanayake. 2012. "Human-robot-environment interaction interface for robotic grit-blasting of complex steel bridges." *Autom. Constr.* 27 (Nov): 11–23. https://doi.org/10.1016/j.autcon.2012.04.014.

Chung, J., S. H. Lee, B. J. Yi, and W. K. Kim. 2010. "Implementation of a foldable 3-DOF master device to a glass window panel fitting task." *Autom. Constr.* 19 (7): 855–866. https://doi.org/10.1016/j.autcon.2010.05.004.

Cichon, T., and J. Robmann. 2018. "Digital twins: Assisting and supporting cooperation in human-robot teams." In Vol. 2 of *Proc., 2018 15th Int. Conf. on Control, Automation, Robotics and Vision, ICARCV 2018*, 486–491. New York: IEEE.

CPWR (Center for Construction Research and Training). 2018. *The construction chart book: The US construction industry and its workers*. 6th ed. Silver Spring, MD: CPWR.

David, O., F. X. Russotto, M. Da Silva Simoes, and Y. Measson. 2014. "Collision avoidance, virtual guides and advanced supervisory control teleoperation techniques for high-tech construction: Framework design." *Autom. Constr.* 44 (Aug): 63–72. https://doi.org/10.1016/j.autcon.2014.03.020.

Deng, M., C. C. Menassa, and V. R. Kamat. 2021. "From BIM to digital twins: A systematic review of the evolution of intelligent building representations in the AEC-FM industry." *J. Inf. Technol. Construct.* 26 (5): 58–83. https://doi.org/10.36680/j.itcon.2021.005.

Devadass, P., S. Stumm, and S. Brell-Cokcan. 2019. "Adaptive haptically informed assembly with mobile robots in unstructured environments." In *Proc., 36th Int. Symp. on Automation and Robotics in Construction, ISARC 2019*, 469–476. Oulu, Finland: International Association for Automation and Robotics in Construction.

Dijkstra, E. W. 1959. "A note on two problems in connexion with graphs." *Numer. Math.* 1 (1): 269–271. https://doi.org/10.1007/BF01386390.

Du, G., P. Zhang, J. Mai, and Z. Li. 2012. "Markerless Kinect-based hand tracking for robot teleoperation." *Int. J. Adv. Rob. Syst.* 9 (2): 1–10. https://doi.org/10.5772/50093.

Du, J., Z. Zou, Y. Shi, and D. Zhao. 2018. "Zero latency: Real-time synchronization of BIM data in virtual reality for collaborative decision-making." *Autom. Constr.* 85 (Jan): 51–64. https://doi.org/10.1016/j.autcon.2017.10.009.

El Hafi, L., et al. 2020. "System for augmented human–robot interaction through mixed reality and robot training by non-experts in customer service environments." *Adv. Rob.* 34 (3–4): 157–172. https://doi.org/10.1080/01691864.2019.1694068.

ENR (Engineering News-Record). 2020. "Construction loses 975,000 jobs in April, due to COVID-19 impacts." Accessed August 8, 2020. https://www.enr.com/articles/49333-construction-loses-975000-jobs-in-april-due-to-covid-19-impacts.

Fang, H. C., S. K. Ong, and A. Y. C. Nee. 2012. "Interactive robot trajectory planning and simulation using augmented reality." *Rob. Comput. Integr. Manuf.* 28 (2): 227–237. https://doi.org/10.1016/j.rcim.2011.09.003.

Fang, Y., Y. K. Cho, and J. Chen. 2016. "A framework for real-time pro-active safety assistance for mobile crane lifting operations." *Autom. Constr.* 72 (Dec): 367–379. https://doi.org/10.1016/j.autcon.2016.08.025.

Feng, C., Y. Xiao, A. Willette, W. McGee, and V. R. Kamat. 2015. "Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites." *Autom. Constr.* 59 (Nov): 128–138. https://doi.org/10.1016/j.autcon.2015.06.002.

Frank, J. A., A. Brill, and V. Kapila. 2016. "Interactive mobile interface with augmented reality for learning digital control concepts." In *Proc., 2016 Indian Control Conf., ICC 2016*, 85–92. New York: IEEE.

Ge, L., and F. Kuester. 2015. "Integrative simulation environment for conceptual structural analysis." *J. Comput. Civ. Eng.* 29 (4): B4014004. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000405.

Gordian/RSMeans Data. 2019. *Building construction costs with RSMeans data, 2020*. Melbourne, FL: Florida Institute of Technology.

Grieves, M. 2014. *Digital twin: Manufacturing excellence through virtual factory replication.* Milford, MA: Digital Twin Consortium.

Hashimoto, S., A. Ishida, M. Inami, and T. Igarash. 2011. "TouchMe: An augmented reality based remote robot manipulation." In *Proc., 21st Int. Conf. on Artificial Reality and Telexistence (ICAT)*, 1–6. Tokyo: Virtual Reality Society of Japan.

Hornung, A., K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard. 2013. "OctoMap: An efficient probabilistic 3D mapping framework based on octrees." *Auton. Robots* 34 (3): 189–206. https://doi.org/10.1007/s10514-012-9321-0.

IEEE. 2017. *ISO/IEC/IEEE International Standard—Systems and software engineering—Vocabulary.* ISO/IEC/IEEE 24765:2017(E). New York: IEEE. https://doi.org/10.1109/IEEESTD.2017.8016712.

Jung, K., B. Chu, S. Park, and D. Hong. 2013. "An implementation of a teleoperation system for robotic beam assembly in construction." *Int. J. Precis. Eng. Manuf.* 14 (3): 351–358. https://doi.org/10.1007/s12541-013-0049-3.

Khasawneh, A., H. Rogers, J. Bertrand, K. C. Madathil, and A. Gramopadhye. 2019. "Human adaptation to latency in teleoperated multi-robot human-agent search and rescue teams." *Autom. Constr.* 99 (Mar): 265–277. https://doi.org/10.1016/j.autcon.2018.12.012.

Kim, B., C. Kim, and H. Kim. 2012. "Interactive modeler for construction equipment operation using augmented reality." *J. Comput. Civ. Eng.* 26 (3): 331–341. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000137.

Kim, D., J. Kim, K. Lee, C. Park, J. Song, and D. Kang. 2009. "Excavator tele-operation system using a human arm." *Autom. Constr.* 18 (2): 173–182. https://doi.org/10.1016/j.autcon.2008.07.002.

Kim, J., S. You, S. Lee, V. R. Kamat, and L. P. Robert. 2015. "Evaluation of human robot collaboration in masonry work using immersive virtual environments need for IVE for advancing HRC in construction." Accessed May 4, 2020. http://www.convr2015.com/wp-content/uploads/2015/10/Web-Proceedings-CONVR2015.pdf.

Kim, S., M. Peavy, P. C. Huang, and K. Kim. 2021. “Development of BIM-integrated construction robot task planning and simulation system.” *Autom. Constr.* 127 (Jul): 103720. https://doi.org/10.1016/j.autcon.2021.103720.

Koenig, N., and A. Howard. 2004. “Design and use paradigms for Gazebo, an open-source multi-robot simulator.” In *Proc., 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2149–2154. New York: IEEE.

Kubik, T., and M. Sugisaka. 2001. “Use of a cellular phone in mobile robot voice control.” In *Proc., SICE Annual Conf.*, 106–111. New York: IEEE.

Kurien, M., M. K. Kim, M. Kopsida, and I. Brilakis. 2018. “Real-time simulation of construction workers using combined human body and hand tracking for robotic construction worker system.” *Autom. Constr.* 86 (Feb): 125–137. https://doi.org/10.1016/j.autcon.2017.11.005.

Kuts, V., T. Otto, T. Tahemaa, Y. Bondarenko, and E. Equilibrium. 2019. “Digital twin based synchronised control and simulation of the industrial robotic cell using virtual reality.” *J. Mach. Eng.* 19 (1): 128–144. https://doi.org/10.5604/01.3001.0013.0464.

Kyjanek, O., B. Al Bahar, L. Vasey, B. Wannemacher, and A. Menges. 2019. “Implementation of an augmented reality AR workflow for human robot collaboration in timber prefabrication.” In *Proc., 36th Int. Symp. on Automation and Robotics in Construction, ISARC 2019*, 1223–1230. Oulu, Finland: International Association for Automation and Robotics in Construction.

Lee, S., and J. I. Moon. 2014. “Introduction of human-robot cooperation technology at construction sites.” In *Proc., 31st Int. Symp. on Automation and Robotics in Construction and Mining, ISARC 2014*, 978–983. Oulu, Finland: International Association for Automation and Robotics in Construction.

Li, X., W. Yi, H. L. Chi, X. Wang, and A. P. C. Chan. 2018. “A critical review of virtual and augmented reality (VR/AR) applications in construction safety.” *Autom. Constr.* 86 (Feb): 150–162. https://doi.org/10.1016/j.autcon.2017.11.003.

Liang, C. J., V. R. Kamat, and C. C. Menassa. 2019. “Teaching robots to perform construction tasks via learning from demonstration.” In *Proc., 36th Int. Symp. on Automation and Robotics in Construction, ISARC 2019*, 1305–1311. Oulu, Finland: International Association for Automation and Robotics in Construction.

Liang, C. J., V. R. Kamat, and C. C. Menassa. 2020a. “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., W. McGee, C. Menassa, and V. Kamat. 2020b. “Bi-directional communication bridge for state synchronization between digital twin simulations and physical construction robots.” In *Proc., 37th Int. Symp. on Automation and Robotics in Construction, ISARC 2020*, 1480–1487. Oulu, Finland: International Association for Automation and Robotics in Construction.

Liang, C. J., X. Wang, V. R. Kamat, and C. C. Menassa. 2021. “Human-robot collaboration in construction: Classification and research trends.” *J. Constr. Eng. Manage.* https://doi.org/10.1061(ASCE)CO.1943-7862.0002154.

Lin, Y. C., and D. Berenson. 2016. “Using previous experience for humanoid navigation planning.” In *Proc., IEEE-RAS Int. Conf. on Humanoid Robots*, 794–801. New York: IEEE.

Liu, C., S. Shirowzhan, S. M. E. Sepasgozar, and A. Kaboli. 2019. “Evaluation of classical operators and fuzzy logic algorithms for edge detection of panels at exterior cladding of buildings.” *Buildings* 9 (2): 40. https://doi.org/10.3390/buildings9020040.

Liu, M., S. Han, and S. Lee. 2017. “Potential of convolutional neural network-based 2D human pose estimation for on-site activity analysis of construction workers.” In *Proc., Computing in Civil Engineering 2017*, 141–149. Reston, VA: ASCE.

Liu, Y., F. Castronovo, J. Messner, and R. Leicht. 2020. “Evaluating the impact of virtual reality on design review meetings.” *J. Comput. Civ. Eng.* 34 (1): 04019045. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000856.

Lundeen, K. M. 2019. “Autonomous scene understanding, motion planning, and task execution for geometrically adaptive robotized construction work.” Ph.D. dissertation, Dept. of Civil and Environmental Engineering, Univ. of Michigan.

Lundeen, K. M., V. R. Kamat, C. C. Menassa, and W. McGee. 2017. “Scene understanding for adaptive manipulation in robotized construction work.” *Autom. Constr.* 82 (Oct): 16–30. https://doi.org/10.1016/j.autcon.2017.06.022.

Lundeen, K. M., V. R. Kamat, C. C. Menassa, and W. McGee. 2019. “Autonomous motion planning and task execution in geometrically adaptive robotized construction work.” *Autom. Constr.* 100 (Apr): 24–45. https://doi.org/10.1016/j.autcon.2018.12.020.

Malik, A. A., and A. Brem. 2021. “Digital twins for collaborative robots: A case study in human-robot interaction.” *Rob. Comput. Integr. Manuf.* 68 (Apr): 102092. https://doi.org/10.1016/j.rcim.2020.102092.

Mantha, B. R. K., C. C. Menassa, V. R. Kamat, and C. R. D’Souza. 2020. “Evaluation of preference- and constraint-sensitive path planning for assisted navigation in indoor building environments.” *J. Comput. Civ. Eng.* 34 (1): 04019050. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000865.

Maragkos, C., G. C. Vosniakos, and E. Matsas. 2019. “Virtual reality assisted robot programming for human collaboration.” *Procedia Manuf.* 38 (Jan): 1697–1704. https://doi.org/10.1016/j.promfg.2020.01.109.

Marton, Z. C., R. B. Rusu, and M. Beetz. 2009. “On fast surface reconstruction methods for large and noisy point clouds.” In *Proc., 2009 IEEE Int. Conf. on Robotics and Automation*, 3218–3223. New York: IEEE.

Milberg, C. T. 2006. “Application of tolerance management to civil systems.” Ph.D. dissertation, Dept. of Civil and Environmental Engineering, Univ. of California.

Murali, P. K., K. Darvish, and F. Mastrogiovanni. 2020. “Deployment and evaluation of a flexible human–robot collaboration model based on AND/OR graphs in a manufacturing environment.” *Intell. Serv. Rob.* 13 (4): 439–457. https://doi.org/10.1007/s11370-020-00332-9.

NAHB (National Association of Home Builders). 2015. *Residential construction performance guidelines: For professional builders & remodelers*. Washington, DC: NAHB.

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.

Pan, J., S. Chitta, and D. Manocha. 2012. “FCL: A general purpose library for collision and proximity queries.” In *Proc., IEEE Int. Conf. on Robotics and Automation*, 3859–3866. New York: IEEE.

Patra, A. K., and S. J. Ray. 2007. “Guiding robots using mobile phone.” In *Proc., 24th Int. Symp. on Automation and Robotics in Construction, ISARC 2007*, 339–344. Oulu, Finland: International Association for Automation and Robotics in Construction.

Quigley, M., B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. 2009. “ROS: An open-source robot operating system.” In *Proc., ICRA Workshop on Open Source Software*. New York: IEEE.

Quintero, C. P., R. Tatsambon, M. Gridseth, and M. Jagersand. 2015. “Visual pointing gestures for bi-directional human robot interaction in a pick-and-place task.” In *Proc., IEEE International Workshop on Robot and Human Interactive Communication*, 349–354. New York: IEEE.

Roldán, J. J., E. Peña-Tapia, D. Garzón-Ramos, J. de León, M. Garzón, J. del Cerro, and A. Barrientos. 2019. “Multi-robot systems, virtual reality and ROS: Developing a new generation of operator interfaces.” In *Robot operating system (ROS)*, 29–64. Cham, Switzerland: Springer.

Roldán, J. J., E. Peña-Tapia, A. Martín-Barrio, M. A. Olivares-Méndez, J. Del Cerro, and A. Barrientos. 2017. “Multi-robot interfaces and operator situational awareness: Study of the impact of immersion and prediction.” *Sensors* 17 (8): 1720. https://doi.org/10.3390/s17081720.

ROS Wiki. 2020. “gazebo_ros_pkgs.” Accessed August 8, 2020. http://wiki.ros.org/gazebo_ros_pkgs.

Ruiz, J. J., A. Viguria, J. R. Martinez-De-Dios, and A. Ollero. 2015. “Immersive displays for building spatial knowledge in multi-UAV operations.” In *Proc., 2015 Int. Conf. on Unmanned Aircraft Systems*, 1043–1048. New York: IEEE.

Sawacha, E., S. Naoum, and D. Fong. 1999. "Factors affecting safety performance on construction sites." *Int. J. Project Manage.* 17 (5): 309–315. https://doi.org/10.1016/S0263-7863(98)00042-8.

Seong, S., C. Lee, and J. Kim. 2019. "Intelligent multi-fingered dexterous hand using virtual reality (VR) and robot operating system (ROS)." In Vol. 751 of *Advances in intelligent systems and computing*, 35–41. Cham, Switzerland: Springer.

Sharif, M.-M., M. Nahangi, C. Haas, and J. West. 2017. "Automated model-based finding of 3D objects in cluttered construction point cloud models." *Comput.-Aided Civ. Infrastruct. Eng.* 32 (11): 893–908. https://doi.org/10.1111/mice.12306.

Sharif, S., T. R. Gentry, and L. M. Sweet. 2016. "Human-robot collaboration for creative and integrated design and fabrication processes." In *Proc., 33th Int. Symp. on Automation and Robotics in Construction, ISARC 2016*, 596–604. Oulu, Finland: International Association for Automation and Robotics in Construction.

Son, H., C. Kim, and K. Choi. 2010. "Rapid 3D object detection and modeling using range data from 3D range imaging camera for heavy equipment operation." *Autom. Constr.* 19 (7): 898–906. https://doi.org/10.1016/j.autcon.2010.06.003.

Sucan, I. A., M. Moll, and L. E. Kavraki. 2012. "The open motion planning library." *IEEE Rob. Autom. Mag.* 19 (4): 72–82. https://doi.org/10.1109/MRA.2012.2205651.

Sukumar, D. K., S. Lee, C. Georgoulas, and T. Bock. 2015. "Augmented reality-based tele-robotic system architecture for on-site construction." In *Proc., 32th Int. Symp. on Automation and Robotics in Construction, ISARC 2015*, 1–8. Oulu, Finland: International Association for Automation and Robotics in Construction.

Tavares, P., C. M. Costa, L. Rocha, P. Malaca, P. Costa, A. P. Moreira, A. Sousa, and G. Veiga. 2019. "Collaborative welding system using BIM for robotic reprogramming and spatial augmented reality." *Autom. Constr.* 106 (Oct): 102825. https://doi.org/10.1016/j.autcon.2019.04.020.

UQO Cyberpsychology Lab. 2004. "Revised presence questionnaire 3.0." Accessed August 8, 2020. http://w3.uqo.ca/cyberpsy/docs/qaires/pres/PQ_va.pdf.

US DOD (Department of Defense). 2005. *Dictionary of military and associated terms*. Washington, DC: US DOD.

Walker, M., H. Hedayati, J. Lee, and D. Szafir. 2018. "Communicating robot motion intent with augmented reality." In *Proc., ACM/IEEE Int. Conf. on Human-Robot Interaction*, 316–324. New York: IEEE.

Wang, Q., J. Guo, and M. K. Kim. 2019. "An application oriented scan-to-BIM framework." *Remote Sensing* 11 (3): 365. https://doi.org/10.3390/rs11030365.

Wang, Q., Y. Tan, and Z. Mei. 2020a. "Computational methods of acquisition and processing of 3D point cloud data for construction applications." *Arch. Comput. Methods Eng.* 27 (2): 479–499. https://doi.org/10.1007/s11831-019-09320-4.

Wang, X., C. J. Liang, C. Menassa, and V. Kamat. 2020b. "Real-time process-level digital twin for collaborative human-robot construction work." In *Proc., 37th Int. Symp. on Automation and Robotics in Construction, ISARC 2020*, 1528–1535. Oulu, Finland: International Association for Automation and Robotics in Construction.

Wang, X. V., Z. Kemény, J. Váncza, and L. Wang. 2017. "Human–robot collaborative assembly in cyber-physical production: Classification framework and implementation." *CIRP Ann.* 66 (1): 5–8. https://doi.org/10.1016/j.cirp.2017.04.101.

Whitney, D., E. Rosen, D. Ullman, E. Phillips, and S. Tellex. 2018. "ROS reality: A virtual reality framework using consumer-grade hardware for ROS-enabled robots." In *Proc., IEEE Int. Conf. on Intelligent Robots and Systems*, 5018–5025. New York: IEEE.

Xu, L., C. Feng, V. R. Kamat, and C. C. Menassa. 2020. "A scene-adaptive descriptor for visual SLAM-based locating applications in built environments." *Autom. Constr.* 112 (Apr): 103067. https://doi.org/10.1016/j.autcon.2019.103067.

Xu, X., and B. Garcia de Soto. 2020. "On-site autonomous construction robots: A review of research areas, technologies, and suggestions for advancement." In *Proc., 37th Int. Symp. on Automation and Robotics in Construction*, 385–392. Oulu, Finland: International Association for Automation and Robotics in Construction.

You, S., J. H. Kim, S. H. Lee, V. Kamat, and L. P. Robert. 2018. "Enhancing perceived safety in human–robot collaborative construction using immersive virtual environments." *Autom. Constr.* 96 (Dec): 161–170. https://doi.org/10.1016/j.autcon.2018.09.008.

Yu, Y. H., C. H. Yeh, T. T. Lee, P. Y. Chen, and Y. H. Shiau. 2014. "Chip-based real-time gesture tracking for construction robot's guidance." In *Proc., 31th Int. Symp. on Automation and Robotics in Construction, ISARC 2014*, 821–828. Oulu, Finland: International Association for Automation and Robotics in Construction.

Zhou, T., Q. Zhu, and J. Du. 2020. "Intuitive robot teleoperation for civil engineering operations with virtual reality and deep learning scene reconstruction." *Adv. Eng. Inf.* 46 (Oct): 101170. https://doi.org/10.1016/j.aei.2020.101170.