A Shared Autonomy Framework for Construction Robot Teleoperation Using Mixed Reality

Yuezhen Gao[1], Ali Golabchi[2], and Qipei Mei[1]

[1]Department of Civil and Environmental Engineering, University of Alberta, Canada
[2]Department of Mechanical Engineering, University of Alberta, Canada
yuezhen@ualberta.ca, alireza1@ualberta.ca, qipei.mei@ualberta.ca

Abstract –

Construction environments are inherently unstructured and hazardous, making fully autonomous robotic operation difficult to achieve reliably. Teleoperation provides a practical alternative by allowing human operators to supervise and control robotic systems remotely. However, conventional teleoperation interfaces, such as joysticks and teach pendants, often require specialized training and provide limited situational awareness. To address these limitations, this paper presents a mixed reality (MR)-based shared autonomy framework for construction robot teleoperation. The proposed framework uses a mixed reality device to capture operator gestures and map them to control commands for a mobile robotic platform. A modular control architecture balances operator intent with system-enforced constraints: the human operator provides high-level motion guidance, while the robotic system executes low-level control and maintains predefined safety boundaries. This division of responsibility is well suited to construction tasks, where environmental variability, safety requirements, and task diversity limit the feasibility of fully autonomous solutions. The framework is implemented in a prototype system, and its applicability is demonstrated through representative construction scenarios, including remote tool delivery and material handling in constrained spaces. This work contributes a system architecture and design rationale for MR-based shared autonomy, providing a practical foundation for human-robot collaboration in construction teleoperation.

Keywords –

Construction robot; Shared autonomy; Robotic system; Teleoperation; Mixed reality

1 Introduction

The construction industry faces persistent challenges, including labor shortages, safety risks, and stagnating productivity [1]. These factors motivate the increasing adoption of mobile robots for construction tasks that are repetitive, physically demanding, or performed in hazardous conditions [2], including site inspection [3,4], material transportation [5,6], and assembly [7,8]. Among all types of mobile robots, quadruped robots offer superior mobility and terrain adaptability, making them suitable for navigating diverse construction sites [9, 10]. When equipped with robotic arms, these mobile robotic platforms can flexibly perform manipulation tasks such as object handling, tool delivery and material logistics [11]. However, despite advances in robotic hardware, the practical deployment of such systems in real construction settings remains limited.

The key barrier to deployment lies in how to control these robotic systems on-site [12,13]. While fully autonomous operation is theoretically feasible, it remains difficult to achieve due to the inherent diversity of construction environments, the presence of dynamic objects, and the complexity of tasks. Previous research on autonomous construction robots has demonstrated capabilities under structured or semi-structured environments [2,14] but extending these methods to unstructured real-world construction sites remains an open challenge. Teleoperation provides an alternative solution by keeping human operators in the control process, enabling them to directly supervise and guide robotic actions [15]. This method has been widely applied in tasks such as welding [16], delivery [17] and excavator operation [18]. However, conventional teleoperation devices typically include joysticks, game controllers, and teach pendants that require specialized training to operate, placing higher requirements on operators' cognitive and learning abilities. Such constraints hinder the broader adoption of teleoperated robots in the construction industry.

Shared autonomy has emerged as a promising paradigm for addressing the limitations of both fully autonomous and fully teleoperated systems. Instead of requiring operators to control every detail, shared autonomy distributes control responsibility between humans and robots: humans provide high-level intent, while robots handle low-level execution and enforce operational constraints [19]. This paradigm has been successfully applied in several domains. In assistive robotics, shared autonomy enables users with mobility impairments to control robotic arms with minimal input [20]. In surgical teleoperation, it allows surgeons to focus on high-level decisions while the system manages instrument stability and safety boundaries [21,22]. Shared autonomy has also been explored in space robotics [23] and rescue applications [24]. However, despite its demonstrated advantages in these domains, shared autonomy has received limited attention in the field of construction robotics, where environmental complexity and task diversity brought unique challenges. Moreover, while shared autonomy addresses the distribution of control responsibilities, the challenge of providing an intuitive operator interface for workers without specialized robotics training remains unresolved.

Recent advances in mixed reality (MR) technology offer a promising approach to address these interface limitations. Conventional teleoperation interfaces rely on 2D displays, which provide limited spatial awareness. This limitation persists regardless of operator location, as construction environments often involve visual occlusions, hazardous zones, and complex spatial relationships that are difficult to perceive through flat screens. MR devices overcome these challenges by providing immersive 3D visualization, overlaying operational information such as robot status and safety boundaries, and supporting natural input modalities including hand gestures, eye gaze, and head tracking [25, 26]. These capabilities reduce training requirements and lower cognitive load compared to traditional control devices. Several studies have explored MR-based teleoperation in laboratory and industrial settings, demonstrating head-mounted displays for robot control [27], gesture-based operation interfaces [28], and augmented visualization of robot motion and sensor data [29]. However, the application of MR to construction robot teleoperation remains limited. Existing MR teleoperation systems are primarily deployed in structured environments with minimal variability, and few studies have integrated MR with shared autonomy frameworks to explicitly allocate control responsibilities between humans and machines. This integration is crucial and highly valuable for the construction industry, as construction workers often lack specialized robotics expertise and must handle multiple complex tasks under safety constraints and time pressure.

While mobile robots provide significant potential for construction applications, their practical deployment remains hindered by control challenges. Fully autonomous solutions struggle with environmental variability, conventional teleoperation demands expertise and imposes cognitive load, and existing research on shared autonomy and MR teleoperation has not fully addressed construction-specific requirements. To fill the gap, this paper presents a shared autonomy framework for construction robot teleoperation using MR. The proposed framework applied a MR device for gesture-based control of a mobile robotic platform, supported by a modular control architecture that balances operator intent with safety constraints. Representative construction scenarios are discussed to illustrate the framework's applicability.

2 Methodology

2.1 System Overview

The proposed framework is designed to enable intuitive teleoperation of a mobile robotic platform in construction environments. Three primary design goals guide the system architecture. First, the framework prioritizes intuitive control. Minimizing the need for specialized training so that construction workers can operate the system effectively. Second, the framework incorporates safety constraints tailored to construction scenarios, where environmental hazards and task variability demand robust safeguards. Third, the architecture is designed to be modular, allowing adaptation to different hardware configurations and construction tasks.

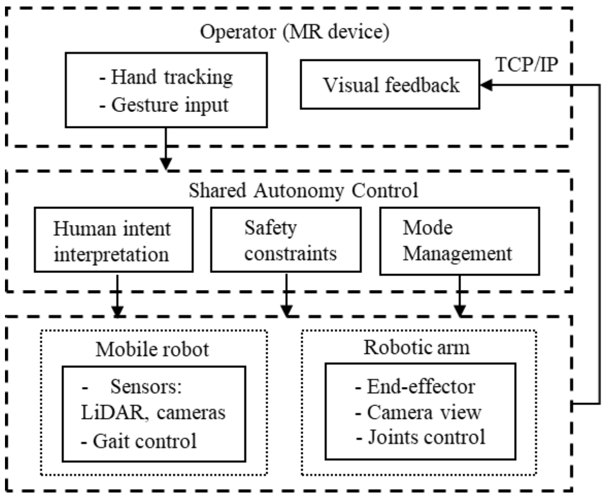

Figure 1. System design. All communications indicated by arrows are transmitted via the TCP/IP protocol.

The overall system architecture shows in Figure 1. The system consists of three main components: (1) a MR device for operator input and feedback, (2) a mobile manipulation platform for task execution, and (3) a control layer that coordinates between operator commands and robot actions. The operator wears a MR head-mounted display and controls the robot through tracked hand gestures. Sensor data collected by the robot is streamed back to the operator end, enabling remote monitoring of on-site conditions, safety constraints, and coordinates locomotion and manipulation functions. Communication between the MR device and the robotic platform is established via TCP/IP protocol over a shared wireless network. In the current prototype, both devices are connected to the same local network within a laboratory environment, with data transmitted through a server-client architecture. This configuration serves as a foundation for future deployment in larger-scale or remote construction settings.

2.2 Hardware Platform

The mobile manipulation platform consists of a Unitree B2 quadruped robot equipped with a Unitree Z1 robotic arm, as shown in the Figure 2. B2 is an industrial level mobile robot capable of traversing uneven terrain of construction sites. It supports a continuous walking payload of 40kg and a maximum standing load of 120kg, with a top speed of 20 km/h. B2 is equipped with onboard sensors including a LiDAR unit on the top, two RealSense D435i depth cameras in the front and rear, providing environmental perception for obstacle detection, and image and video transmission for situational awareness. In addition, the platform has Simultaneous Localization and Mapping (SLAM) feature, which enable users to set waypoints on the map pre-scanned by the onboard LiDAR.

The Z1 robotic arm is mounted on the top of the B2 platform. It features six degree of freedom (6-DOF), a maximum reach of 740 mm, and a payload capacity of 2 kg. A depth camera is mounted at the wrist of the end-effector to help identify the object position when the arm is moving. This configuration enables the system to perform manipulation tasks such as grasping and transporting lightweight objects, including construction tools and small pieces of materials.

The operator interface is implemented using an Apple Vision Pro mixed reality headset. The device provides hand tracking capabilities that capture finger and wrist movements, which serve as input for robot control. Hand tracking data is obtained through Tracking Streamer, a visionOS app [30] developed with python library to stream hand/head tracking data from Vision Pro to python client, while viewing robot camera feeds via WebRTC protocol, which ensure real-time streaming requirement.

Figure 2. Mobile robotic platform: Z1 arm and B2.

2.3 Shared Autonomy Framework

The proposed framework adopts a shared autonomy paradigm in which control responsibilities are distributed between the human operator and the robotic system. Rather than requiring the operator to manage all system states, the framework assigns high-level decision-making to the human operator, while the robotic system handles low-level motion execution and enforces safety constraints. The control architecture is organized into two primary modes: locomotion control and manipulation control.

In locomotion mode, the operator controls the movement of the B2 platform while the robot executes gait control autonomously, maintaining stability and adapting to terrain variations. Real-time visual feedback streams from onboard cameras are transmitted to the operator's MR display, assisting the operator in making correct navigation and manipulation decisions. The robot possesses environmental awareness and executes actions within safety constraints detailed in section 2.5.

In manipulation mode, the operator controls the Z1 arm through tracked hand movements captured by the MR device. These movements are mapped to arm motion commands using end-effector control via inverse kinematics (IK) method, where the operator gives gripper position, and the system computes the required joint states and controls the arm to execute.

The operator can transition between locomotion and manipulation modes through one of two designated control inputs:

- Voice command: Say 'locomotion mode' or 'manipulation mode'.
- Hand gesture: The operator performs a synchronized pinch gesture with both hands simultaneously to switch modes.

2.4 Operator Interaction

Operator input is captured through the hand tracking feature of Apple Vision Pro. The system tracks pinch gestures, fingertip positions, and wrist orientation via the application described in section 2.2. The high sensitivity of Vision Pro to hand movements allows manipulation control with low latency and high flexibility. The 3D spatial awareness feature allows operators to keep interaction with the environment while using UI to control the robot system, ensuring smooth operation of the shared autonomy framework. This section details how operator controls are processed for robot manipulation and locomotion, then elaborates user interface for control integration.

2.4.1 Manipulation Control

For manipulation control, the operator's hands motions are mapped to the robotic arm end-effector position through a three-stage pipeline:

1. Raw tracking data including fingertip positions and wrist orientation are acquired from MR device.
2. Coordinate transformation and input scaling convert hand motion from the operator's reference frame into the robot's operational range.
3. Inverse kinematics computation determines the required joint states based on the target end-effector position.

This approach allows the operator to intuitively control the arm by moving their hand naturally, without needing to understand the detailed joint configuration. As mentioned in 2.2, the communication mechanism of the manipulation control is shown in Figure 3.

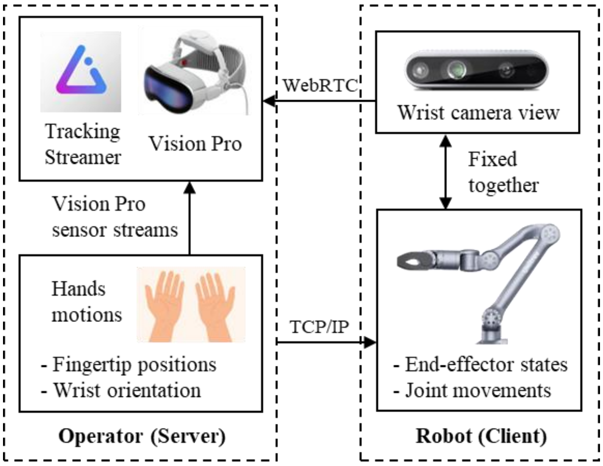

Figure 3. Communication between the operator end and the robotic arm end.

2.4.2 Locomotion Control

For locomotion control, the current prototype supports two ways:

- Using directional commands through software-based control scripts. The B2 platform responds to movement commands while managing gait execution and terrain adaptation.
- Using SLAM feature, the user setting waypoints on the pre-scanned map to customize a patrol route as shown in Figure 4, then the robot will patrol autonomously to the waypoints in sequence.

Effective teleoperation relies on real-time perception feedback from the robot's onboard sensors. The front and rear cameras stream live videos, and the LiDAR sensor provides distance data about the surrounding environment. The MR component integrates this data by overlaying ROS2 message feedback, such as LiDAR-detected obstacles and SLAM-generated reconstruction map, directly onto the operator's view. This deep integration enables a shared autonomy feedback loop where the operator provides high-level guidance based on the spatial perception, rather than just using MR as a basic input device.

Figure 4. Pre-scanned map using SLAM for waypoints and charging spot settings. Yellow dots and dash: user defined waypoints and patrol path. Cyan point: robot's current position and orientation.

2.4.3 User Interface

The proposed user interface is designed to centralize multi-modal feedback into an immersive workspace to enhance the operator's situational awareness. As shown in Figure 5, the interface is partitioned into two primary zones to support the framework's modular control architecture.

The left zone provides real-time visual perception through camera streams from the B2 robot's front and rear cameras. The lower-left window shows two ways locomotion control as described in 2.4.2. On the left side, the direction and speed of the robot are controlled by operator for general locomotion and map generation. On the right side, the operator can choose a SLAM-generated occupancy map and add waypoints on it for the robot to follow. The right zone manages manipulation control of the Z1 arm, where the operator can enable gesture-based tracking and monitor the state of the end-effector.

By overlaying system data including battery level, frame rate, and network latency onto the operator's field of view, this interface minimizes cognitive load and provides a continuous, shared autonomy feedback loop.

The integration of this UI into visionOS ensures that operator receives motion guidance while remaining aware of the robot's state and potential hazards in the environment.

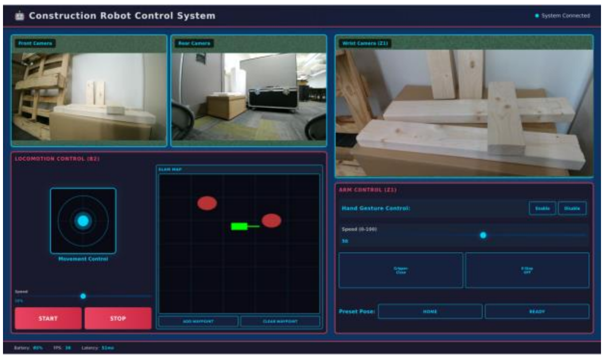

Figure 5. The user interface. Integrating camera streams, navigation, and locomotion controls.

2.5 Safety Constraints and Management

Detailed safety design is essential for robot teleoperation in construction environments, where operators may have limited experience with robotic systems and environmental hazards are common. Unlike general teleoperation, this framework addresses construction-specific challenges, such as navigating uneven terrain through autonomous gait control, mitigating visual occlusions through multi-angle camera streams, and enforcing strict distance thresholds using LiDAR to protect personnel in dynamic, unstructured work zones. These operational challenges are integrated into the feedback loop through the MR device, providing the operator with an intuitive control reference. Moreover, the system logs safety-related interventions within the MR environment, generating valuable data for refining and optimizing control rules.

The safety design follows three principles: 1) Minimizing risk to personnel, equipment, and surroundings. 2) Keeping the operator informed of safety states of the system; 3) Ensuring the system stops when emergency happens.

2.5.1 Motion Constraints

Motion constraints regulate the physical movement of robotic system to prevent collisions and mechanical damage.

- **Speed limits.** Both the locomotion of B2 robot and the motion of Z1 end-effector are restricted to maximum speeds. During teleoperation, the speed of robot is set below its maximum capability to ensure operators have sufficient reaction time in cluttered environments.
- **Workspace boundaries.** The reachable zone of Z1 arm is restricted to prevent self-collision with the B2 body and overextension beyond mechanical limits. Commands moving the end-effector outside the defined workspace are rejected.
- **Joint limits.** Angular boundaries are enforced for each Z1 joint to prevent mechanical damage, particularly relevant in confined scenarios where the arm operates near walls or obstacles.

2.5.2 Input Constraints

Input constraints determine the mapping relationship between operator commands and robotic movements.

- **Input scaling.** The range of human hand movements is scaled proportionally to the robot's operational range, preventing small gestures from triggering excessive robotic motion. The scaling factor balances responsiveness with safety.
- **Rate limits.** Sudden or abnormal movements of operator are filtered before transmission to prevent abrupt acceleration that could make the platform unbalanced or damage the manipulated object.

2.5.3 Environmental Constraints

These constraints are implemented to address external factors that impact safe operation.

- **Distance threshold.** The system monitors distance to detected obstacles using onboard LiDAR and cameras. When an obstacle is detected within a designated distance, the robot stops motion and alerts the operator. Movement resumes after the operator acknowledges the alert.
- **Communication timeout.** If network connect is lost beyond the preset duration, the robot stops all actions and holds its states to prevent uncontrolled operation.

2.5.4 Constraint Violation Response

The interaction between operator inputs and autonomous correction is managed through a hierarchical priority system. Operator gestures captured by the MR device are first processed through input scaling and rate limits to eliminate jerky or extreme movements that could destabilize the platform. The system then cross-references these commands with real-time sensors data from LiDAR and onboard cameras to identify potential conflicts.

Conflict resolution is executed based on a response strategy:

Adjustment (Soft Constraints): When a gesture command approaches a non-critical boundary, such as the limit of the robotic arm's reachable workspace, the system autonomously re-calculates and adjusts the command to the nearest safe coordinate. This allows the operator to maintain task flow without a total system halt, while the MR interface provides visual feedback of the correction.

Rejection and Override (Hard Constraints): When there is a critical conflict, such as a gesture that would cause a collision or violate a distance threshold near personnel, the system prioritizes autonomous safety over operator intent. When the command is rejected, the robot stops all motion immediately and enters a locked mode.

To resolve this locked state, the operator must manually acknowledge a safety alert within the MR headset before the system will accept new inputs, ensuring the human is fully aware of the environmental hazard before proceeding.

2.6 System Implementation

This section describes the software implementation and communication of the proposed framework. The software system adopts a modular architecture consists of three components: the operator interface end running on the MR device, the control module processing command and safety validation, and the robot end managing hardware control and executing tasks. These independent modules facilitate system maintenance and adaptation to different hardware configurations.

The communication between the MR device and robotic platform is established via TCP/IP protocol over a shared wireless network. A server-client architecture implemented in Python manages data exchange between system components. The system manages two data streams simultaneously: a command stream transmits the tracking data of the operator to the control module, and a feedback stream returns sensor data from the robot to the operator end. The robot execution uses ROS2 as middleware, managing message passing for sensor data acquisition, motion planning and navigation, and joint control. The modular nature of ROS2 facilitates integration with the B2 and Z1 APIs.

The current prototype operates within a laboratory environment, with all devices connected to a shared local network. This implementation serves as a foundation for future deployment in larger-scale construction settings, where alternative communication methods such as 5G may be required.

3 Discussion

3.1 Application Scenarios

The proposed framework is designed to support various construction-related teleoperation tasks. This section discusses representative scenarios to illustrate the framework's intended applicability.

3.1.1 Remote tool delivery

On construction sites, workers often need tools or small components delivered to hard-to-reach locations. With the proposed framework, an operator can remotely guide the mobile platform to a storage area, switch to manipulation mode to grasp the required item, and then navigate to the delivery point. This shared-autonomy approach enables the operator to focus on high-level task decisions, while the robot handles navigation stability and safety constraints.

3.1.2 Material Handling in Restricted Areas

Construction sites often contain confined spaces that pose risks to human workers, such as areas near heavy machinery or structural hazards. The teleoperation framework enables operators to manipulate materials in these zones from a safe distance. Real-time visual feedback from the robot's cameras provides environmental perception data, while safety constraints prevent the robot from entering excessively hazardous areas.

3.1.3 Site Inspection and Monitoring

The mobile platform's ability to traverse uneven terrain makes it suitable for inspection tasks across construction sites. Operators can remotely guide the robot to designated locations to assess site conditions. Onboard cameras and other sensors enable remote inspections without requiring on-site presence.

3.1.4 Routine Inspection in Stable Environments

The framework supports waypoint based autonomous navigation. In scenarios where the construction layout remains static, such as completed building interiors, or equipment storage areas, the robot operate with reduced human intervention. Using the onboard SLAM feature, the operator can set inspection waypoints on the pre-scanned spatial map, and the robot will move to the waypoints in sequence. This mode is particularly suitable for repetitive monitoring tasks, such as daily safety inspections or equipment management. Additionally, the operator can designate a charging station on the map, enabling the robot to autonomously return for charging when battery level falls below a preset threshold. This feature extends operational duration and reduces the need for continuous human supervision during routine tasks.

3.2 Limitations

Several limitations of the current prototype are acknowledged. First, the system is under development and testing in a laboratory environment simulating construction scenarios, and its performance in actual construction sites with dust, vibration, and communication interference remains to be validated. Second, current locomotion relies on user-interface-based control rather than gesture input, requiring integration with motions captured by MR device. Besides, although safety constraints have been established, comprehensive testing under diverse conditions has not been conducted. Finally, communication latency may also impact response times for precision operation tasks, further reduction in latency is necessary.

3.3 Future Work

Future development will focus on addressing the limitations. Key enhancements include:

- Gesture-based mobile control will be integrated in the system to ensure easy and smooth teleoperation for the remote operator.
- The user interface will be integrated into visionOS to support operator using gestures for robot mode selection and situational awareness. Alert sounds will also be added to notify the operator for emergency stop.
- Field experiments in real construction settings will be conducted to evaluate the system performance under different practical conditions. Quantitative metrics including task completion time, material handling success rate, and error frequency, will be compared between the MR-based shared autonomy framework and traditional joystick-based teleoperation to demonstrate the advantages of proposed framework.
- User studies with construction workers will assess usability of the framework, user cognitive workload, and training requirement.

4 Conclusion

This paper presented a MR-based shared-autonomy framework for construction robot teleoperation. The prototype system integrates a mixed reality head-mounted display with a quadruped robot equipped with a robotic arm, enabling intuitive remote control through the operator’s gestures. The framework distributes control responsibilities between the human operator and the robotic system: the operator provides high-level guidance through natural hand movements, while the robot performs low-level execution and enforces safety constraints.

The framework adopts a modular control architecture that supports two operating modes: locomotion of the robot base and manipulation of the robotic arm, with intuitive mode switching through voice commands or gestures. To address construction-specific requirements, the system incorporates multiple safety mechanisms, including motion constraints, input scaling, environmental monitoring, and violation response strategies. Communication between system modules is implemented using the TCP/IP protocol and ROS 2 middleware, providing a foundation for future deployment and extension. Representative construction scenarios, including remote tool delivery, material handling, and site inspection, were discussed to demonstrate the framework’s potential applicability. Although the current prototype has been tested in a laboratory environment simulating construction conditions, its modular design facilitates adaptation to different hardware platforms and task settings.

This work contributes a system architecture and design foundation for shared autonomy in construction teleoperation. Future work will focus on experimental validation, user studies, and improvements in autonomous capabilities and system reliability to support practical deployment in real construction environments.

References

[1] K. S. Saidi, J. T. O'Brien, and A. M. Lytle. Robotics in construction. In Springer Handbook of Robotics, pages 1493–1520. Springer, Cham, 2016.

[2] K. Asadi, H. Ramshankar, H. Pullagurla, A. Bhandare, S. Shanbhag, P. Mehta, S. Kundu, K. Han, E. Lobaton, and T. Wu. Vision-based integrated mobile robotic system for real-time applications in construction. Automation in Construction, 96:470–482, 2018.

[3] J. Li, X. Zhou, C. Gui, M. Yang, F. Xu, and X. Wang. Adaptive climbing and automatic inspection robot for variable curvature walls of industrial storage tank facilities. Automation in Construction, 172:106049, 2025.

[4] J. Tuomisto, Y. Zheng, I. Chauhan, O. Seppänen, and J. Pajarinen. Automating on-site object inspection with a quadruped robot and BIM. Automation in Construction, 181(Part A):106571, 2026.

[5] G. Yang, S. Wang, H. Okamura, B. Shen, Y. Ueda, T. Yasui, T. Yamada, Y. Miyazawa, S. Yoshida, Y. Inada, S. Ino, K. Okuhata, and Y. Mizobuchi. Hallway exploration-inspired guidance: applications in autonomous material transportation in construction sites. Automation in Construction, 128:103758, 2021.

[6] G. Yang, S. Wang, H. Okamura, T. Yasui, S. Ino, K.

Okuhata, and Y. Mizobuchi. Autonomous cart docking for transportation robots in complex and dynamic construction environments. Automation in Construction, 171:105968, 2025.
[7] C. Feng, Y. Xiao, A. Willette, W. McGee, and V. R. Kamat. Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites. Automation in Construction, 59:128–138, 2015.
[8] Z. Chen and A. Adel. Advancing robotic assembly in construction: Innovations, challenges, and opportunities. Automation in Construction, 178:106370, 2025.
[9] S. Halder and K. Afsari. Robots in inspection and monitoring of buildings and infrastructure: A systematic review. Applied Sciences, 13(4):2304, 2023.
[10] J. Tuomisto, Y. Zheng, I. Chauhan, O. Seppänen, and J. Pajarinen. Automating on-site object inspection with a quadruped robot and BIM. Automation in Construction, 181:106571, 2026.
[11] B. Xiao, C. Chen, and X. Yin. Recent advancements of robotics in construction. Automation in Construction, 144:104591, 2022.
[12] Y. Fu, J. Chen, and W. Lu. Human-robot collaboration for modular construction manufacturing: Review of academic research. Automation in Construction, 158:105196, 2024.
[13] Z. Li, Y. Yu, N. Zeng, F. Tian, S. Zhang, H. Sun, and Q. Li. Human-centric human-robot collaboration in on-site construction (2014–2024): Advances, barriers, and future directions. Automation in Construction, 180:106566, 2025.
[14] C. Zhao, J.-Y. Chen, T. Sun, W. Fan, X.-Y. Sun, Y. Shao, G.-Q. Guo, and H.-L. Wang. Robotic motion planning for autonomous in-situ construction of building structures. Automation in Construction, 171:105993, 2025.
[15] J. S. Lee, Y. Ham, H. Park, and J. Kim. Challenges, tasks, and opportunities in teleoperation of excavator toward human-in-the-loop construction automation. Automation in Construction, 135:104119, 2022.
[16] S. Yoon, M. Park, and C. R. Ahn. Comparing dynamic viewpoint control techniques for teleoperated robotic welding in construction. Automation in Construction, 172:106053, 2025.
[17] A. Boker, E. Grimberg, F. Tener, and J. Lanir. Navigating the last mile: A stakeholder analysis of delivery robot teleoperation. Sustainability, 17(13):5925, 2025.
[18] D. Liu, J. Kim, and Y. Ham. Multi-user immersive environment for excavator teleoperation in construction. Automation in Construction, 156:105143, 2023.
[19] D. A. Abbink, M. Mulder, and E. R. Boer. A topology of shared control systems—Finding common ground in diversity. IEEE Transactions on Human-Machine Systems, 48(5):509–525, 2018.
[20] D. Gopinath, S. Jain, and B. D. Argall. Human-in-the-loop optimization of shared autonomy in assistive robotics. IEEE Robotics and Automation Letters, 2(1):247–254, 2017.
[21] A. Attanasio, B. Scaglioni, E. De Momi, P. Fiorini, and P. Valdastri. Autonomy in surgical robotics. Annual Review of Control, Robotics, and Autonomous Systems, 4:651–679, 2021.
[22] M. Selvaggio, G. A. Fontanelli, F. Ficuciello, L. Villani, and B. Siciliano. Passive virtual fixtures adaptation in minimally invasive robotic surgery. IEEE Robotics and Automation Letters, 3(4):3129–3136, 2018.
[23] J. Artigas, R. Balachandran, C. Riecke, M. Stelzer, B. Weber, J. H. Ryu, and A. Albu-Schaeffer. Kontur-2: Force-feedback teleoperation from the International Space Station. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 1166–1173, Stockholm, Sweden, 2016.
[24] A. Birk and S. Carpin. Rescue robotics—A crucial milestone on the road to autonomous systems. Advanced Robotics, 20(5):595–605, 2016.
[25] X. Wang, M. J. Kim, P. E. Love, and S. C. Kang. Augmented reality in built environment: Classification and implications for future research. Automation in Construction, 32:1–13, 2016.
[26] A. H. Behzadan, S. Dong, and V. R. Kamat. Augmented reality visualization: A review of civil infrastructure system applications. Advanced Engineering Informatics, 29(2):252–267, 2016.
[27] D. Whitney, E. Rosen, D. Ullman, E. Phillips, and S. Tellex. ROS reality: A virtual reality framework using consumer-grade hardware for ROS-enabled robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5018–5025, Madrid, Spain, 2018.
[28] C. J. Liang, V. R. Kamat, and C. C. Menassa. Teaching robots to perform quasi-repetitive construction tasks through human demonstration. Automation in Construction, 120:103370, 2020.
[29] M. Walker, H. Hedayati, J. Lee, and D. Szafir. Communicating robot motion intent with augmented reality. In Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction, pages 316–324, Chicago, IL, 2018.
[30] Y. Park. Teleoperation System using Apple Vision Pro(Version0.1.0).Online: https://github.com/Improbable-AI/VisionProTeleop, Accessed: 10/01/2026.