2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

# RADAR: Robotics Assembly by Demonstration via Augmented Reality

Wenhao Yang[1], Shi Bai[2], and Yunbo Zhang[3]

***Abstract*— With the widespread adoption of robots in high-mix, low-volume manufacturing, and the challenges posed by long-horizon assembly tasks, we introduce the RADAR system—an integrated human-robot collaboration system for Robotic Assembly by Demonstration via Augmented Reality. Existing frameworks lack a comprehensive, cross-task framework for effective assembly collaboration, limiting their applicability in complex tasks. We designed the RADAR system's conceptual model, detailing its workflow and components. The system integrates human input into robotic metal beam assembly through augmented reality interactions and interfaces. We also developed a task planner that dynamically adjusts human-robot assembly tasks at coarse-fine resolutions. Validating through practical scenarios, particularly the RAMP assembly benchmark, showed that human involvement significantly enhances assembly precision and success rates, proving RADAR's effectiveness and efficiency in human-robot collaborative assembly.**

## I. INTRODUCTION

The rise of Industry 4.0 and 5.0 is boosting the use of industrial robots in manufacturing, especially for Small and Medium-sized Enterprises (SMEs), enabling Intelligent Manufacturing and production processes. While robots are being introduced across various shop floors [1] for tasks such as assembly, material handling, welding, polishing, and painting [2], enhancing the flexibility and adaptability of robots remains a challenge for High-Mix Low-Volume (HMLV) manufacturing [3].

Human-robot collaboration (HRC) is considered a pivot for enhancing flexibility and adaptability from human involvement. Robots automate repetitive and physically demanding tasks and replace humans in hazardous or extreme working environments [4]. Given that the high-level cognitive and decision-making abilities of human workers still surpass those of robots [5], humans still play a crucial role in instructing robot manipulation, overseeing production, and handling complex tasks. In recent research, Augmented Reality (AR)-based HRC interfaces have been shown to be promising as they bridge the gap between humans and robots [6]–[8], provide bidirectional communication [9], [10] and enable robot learning from demonstration (LfD) [11]. With the progress that has been made, some recent AR-based HRC systems primarily concentrate on handle long-horizon tasks, such as [12]–[14]. In literature, the *long-horizon task* usually refers to a task that involves multiple subtasks/subgoals and the manipulation of several objects [15]. It requires a vast amount of possible action sequences and decision-making [16]. Moreover, the design of existing AR-based HRC systems is still task-specific and lacks systematic consideration. The paradigm of AR-based HRC and the associated system components across different tasks remain unclear. Therefore, the remaining research question is: *What should the AR-based HRC system for long-horizon assembly tasks look like?* To answer this research question, this paper presents a system design for an AR-based HRC interface, introduces a coarse-fine dual-level task planner for long-horizon tasks, and validates the approach with a robotic assembly task. The detailed contributions are listed as follows:

- We present the **R**obotics **A**ssembly by **D**emonstration via **A**ugmented **R**eality (RADAR) system—a new paradigm for AR-based HRC assembly systems (Fig.1(a)), consisting of both the conceptual design of a cross-task framework and key components for long-horizon tasks, along with its realization of a metal frame assembly task.
- We propose a task planner that divides tasks into coarse and fine levels, leveraging human high-level decision-making abilities at both levels to enable rapid programming and effective mitigation of failures in complex assembly operations.
- We validate the RADAR system with the RAMP assembly benchmark [17], demonstrating its superiority through higher success rates and enhanced efficiency.

## II. RELATED WORKS

### A. Robot-assisted Assembly

Existing fully automated robotic assembly is still challenging [18] as it suffers from the automated algorithms' holistic understanding of planning and the difficulties in executing certain steps. Human involvement, therefore, is introduced to teach robots new skills through learning from demonstration (LfD) methods. In literature, there are different types of LfD methods [19], including kinesthetic teaching [20], teleoperation [21], and passive observation [22]. These methods are limited to specific tasks or robots and lack a high-level understanding of long-horizon tasks. To tackle complex and long-horizon tasks, Lee et al. [23] introduced a virtual IKEA furniture assembly environment to test the performance of Reinforcement Learning (RL) and Imitation Learning (IL) based algorithms. Yet, evaluating virtual environment-developed algorithms in physical settings faces challenges due to the heterogeneity of hardware configurations and the limitations of standardized benchmarks.

[1]Wenhao Yang is with the Department of Industrial and Systems Engineering, Lamar University, Beaumont, TX 77705, USA wyang2@lamar.edu

[2]Shi Bai is with the iServe Robotics, Sunnyvale, CA 94002, USA

[3]Yunbo Zhang is with the Department of Industrial and Systems Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA ywzeie@rit.edu

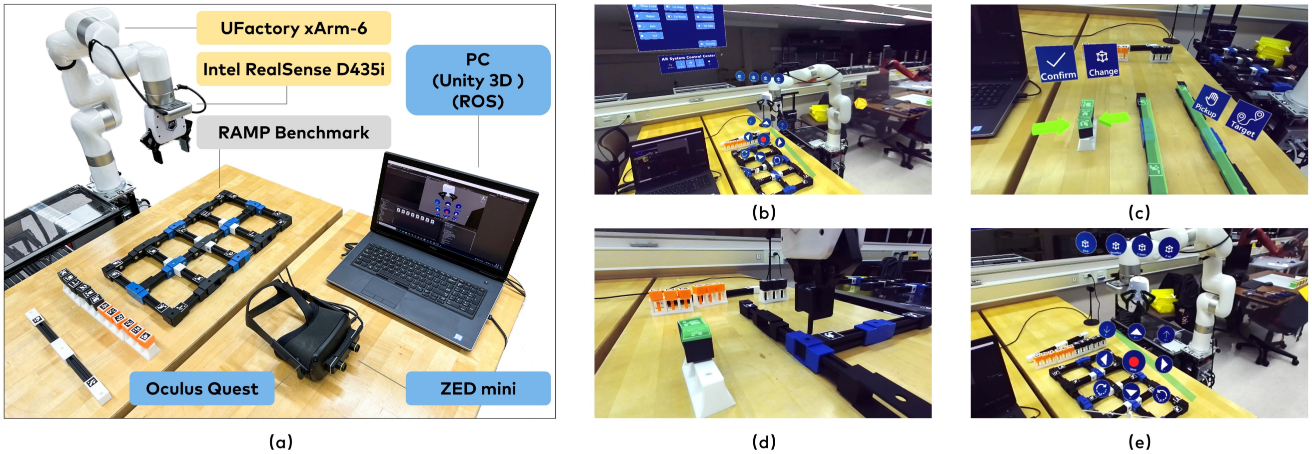


Fig. 1: Overview of the proposed RADAR (Robotics Assembly by Demonstration via Augmented Reality) system, highlighting its key components: (a) hardware configuration, (b) user's Augmented Reality interface, (c) demonstrate the robot with constraints; (d) human involvement in peg insertion, and (e) button-based control for fine-grained robotics operations.

There is a critical need for optimization algorithms to devise efficient assembly sequences. Assembly predicates, representing constraints, significantly influence sequence generation [24]. Optimization methods include library-based solutions, learning from human demonstration, and learning by exploration. For example, Zhang et al. [25] enhanced robot task planners with explicability and predictability. Stenmark and Topp introduced a programming-by-demonstration approach for creating and refining robot program components [26]. Mollard et al. proposed a method integrating learning from demonstration to detect constraints, decompose subtasks, and learn hierarchical assembly tasks [27]. It is necessary to develop a comprehensive framework for robot assembly, which should integrate library-based task planners, facilitate learning from human demonstration, and autonomously address complex problems through exploration.

Robot assembly involves manipulating components in discrete subtasks like peg-in-hole insertion, slide-in-the-groove alignment, bolt screwing, pick-and-place maneuvers, and pipe connection [28]. However, intricate tasks such as screwing and flipping require specialized end effectors. Knepper et al. [29] developed modular tools for such tasks, advocating for collaborative robot use, whereas Dhanaraj et al. [30] proposed human intervention integration. Precision-centric tasks like gear assembly and peg insertion require closed-loop control mechanisms drawing from force/torque measurements [31], computer vision [32], or CAD models [33]. With the reviewed remarks, our objective is to optimize human involvement to the greatest extent feasible, diminishing reliance on specific tasks, thus augmenting the efficiency of general robot assembly.

### B. AR-enhanced Human-robot Interaction

Integrating human input in autonomous robotic assembly shows promise for handling complex tasks efficiently. HRC systems leverage individual strengths [34], with humans excelling at decision-making and error correction. AR has been proven to effectively facilitate interaction and information exchange between robots and humans, thereby enhancing the efficiency of HRI [35]. AR benefits HRI by improving safety in shared workspaces, facilitating communication of robot intentions, and providing intuitive interactions [36]–[40]. Additionally, AR simplifies robot programming tasks, supports real-time communication, and aids in teleoperation, thus improving robot adoption in industries [3], [41]–[45].

Recent advancements in AR have garnered interest in its application within Human-Robot Interaction (HRI) and collaborative assembly processes. Notable innovations include improvements in LfD, such as hand-tracking for robot trajectories [46], task-level constraints [47], and visualizations of robot-learned skills [11]. However, *a cross-task AR-based framework for complex tasks remains an unanswered question*. Meanwhile, several studies have explored integrating AR into collaborative assembly. Makris et al. [48] utilized AR to enhance safety mechanisms and provide production-related information. Luxenburger et al. [49] investigated AR as a communication tool for hybrid teams in aircraft assembly. Initial demonstrations of AR's application in the collaborative assembly were conducted by Danielsson et al. [50], and Kousi et al. [51] developed software to support operators using mobile robots. Still, these approaches have limitations: *They frequently concentrate on application-specific tasks rather than addressing the broader systemic integration of human and robotic components, and they tend to emphasize assisting human operators over fully exploiting the potential capabilities of robots*. Moreover, the attainment of precision in control and the execution of intricate operations present challenges, influenced by hardware visualization and tracking [52], accurate visual representation [53], control interface precision [54], and the adequacy of feedback mechanisms [55]. These combined factors create a gap in AR-based systems for real-world manufacturing applications.

## III. Proposed System

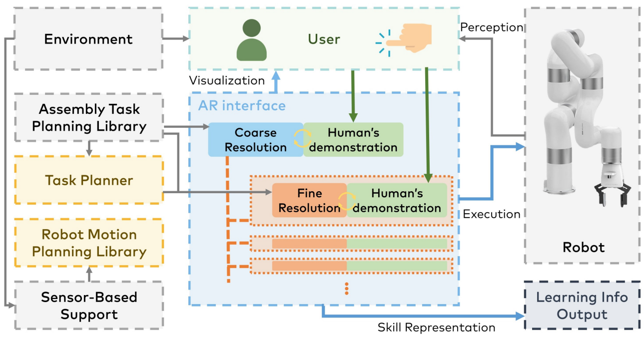


Fig. 2: The overview of RADAR system. The RADAR system supports robot learning from demonstration using AR for both coarse and fine level tasks. With assistance from the planning library, task results are executed, and skills are represented.

### A. System Design

RADAR system builds upon previous research [39], [56] by developing a framework for robot learning from demonstration using AR, as depicted in Fig. 2. A typical workflow of the RADAR system starts with a *Task Planner* supported from the *Assembly Task Planning Library*. Subsequently, the *Task Planner* divides the tasks into *coarse-level* tasks, requiring fewer human demonstrations and less precise human manipulations, and *fine-level* tasks, necessitating more human demonstrations and greater precision in manipulations. Both the coarse and fine-level tasks are completed by humans through the *AR Interface*, which visualizes the planned tasks, the perception of the environment, and the robot, and enables human interactions to complete demonstrations, provide constraints, and handle failures. The *Robot Motion Planning Library* generates the motion trajectories and kinematic information for the robot to move, subsequently verified by humans through the AR interface. Finally, once humans verify and confirm the current plan, the robot executes the planned motion to complete the task. This process is repeated until the whole task is completed. RADAR also collects human demonstrations, constraints, and failure handling, and stores them as the *Learning Info Output* for potential artificial intelligence (AI)-based LfD methods.

### B. System Implementation

Following the system design, we deploy the RADAR system depicted in Fig. 3. A Video See-Through (VST) AR is applied in RADAR, utilizing a stereo camera (ZED mini) affixed to the front of a Virtual Reality (VR) headset (Meta Quest). The main system operates on a Personal Computer (PC), with the Unity 3D platform serving as the graphical processing engine and ROS (Robot Operating System) for driving an xArm 6 robotic arm with its two-finger gripper. The laptop is equipped with an Intel i7-9850H @ 2.60GHz processor, 16GB of RAM, an NVIDIA Quadro RTX 4000 GPU, and the Windows 10 Enterprise operating system.

### C. Interactions & UIs

Two levels of interaction are enabled in the RADAR system (see Fig. 4): coarse and fine. Once the *Task Planner* has divided the tasks into both coarse and fine levels, users are asked to interactively complete these tasks. At the coarse resolution level, operators can readily demonstrate subtask organization, specify constraints, and take over in failure scenarios. In subtask organization mode, they can specify and modify the sequence of subtasks using mid-air hand gestures to drag the subtask widget. In constraint definition mode, operators can define constraints to precisely specify the orientations of gripper for subsequent pick-and-place actions. In the event of a failure, users can mark the subtask failure and take over control and manually command the robot's operations, which makes the subtask into a fine-level. At the fine resolution level, a standard learning-from-demonstration category [19] is implemented within the AR interface, including teach pendant, teleoperation, and passive observation methods. Teach pendant enables users to guide the robot through desired motions using a virtual teaching

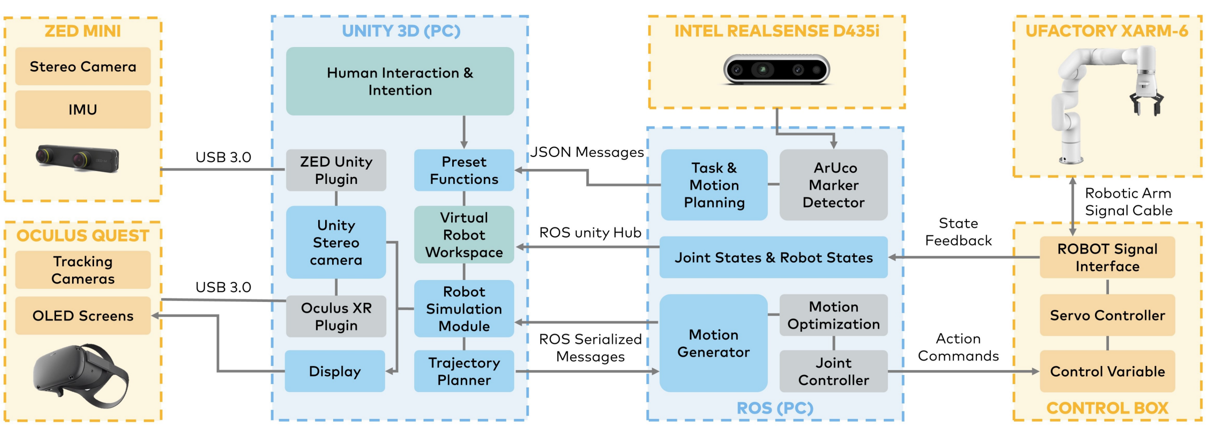


Fig. 3: The RADAR system deployment structure. The RADAR system is deployed using a VST AR setup with a ZED mini stereo camera on a Meta Quest VR headset, running on a PC with Unity 3D and ROS to control an xArm-6 robotic arm.

pendant interface. Teleoperation is integrated with hand-tracking control, offering six degrees of freedom (DOF) for manipulating the end-effector. Passive observation involves collecting user actions during task execution, focusing on specific aspects such as target selection and loading area identification, even when the user's attention is only partially engaged. All aforementioned demonstrations are facilitated through the AR interface and virtual widgets. A user-centric

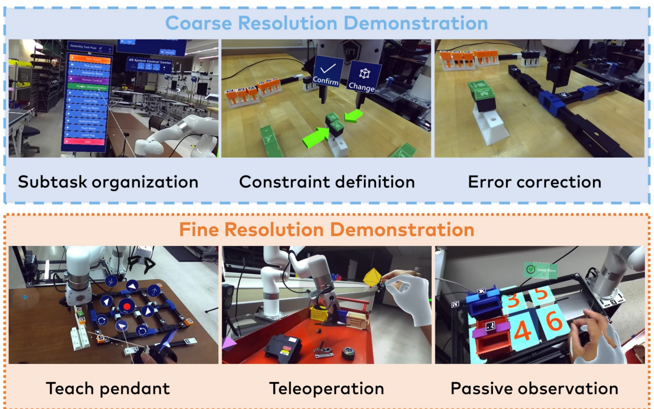


Fig. 4: AR demonstration methods in RADAR system: Coarse resolution for organizing subtasks, defining constraints, and correcting errors; Fine resolution for robot control via teach pendant, teleoperation or passive observation.

AR interface is developed to enable efficient human-robot interaction for the assembly task, as shown in Fig. 1(b-e). This interface provides users with visual guidance for subtasks and robot attempts. Users can physically demonstrate the virtual robot, such as peg grasping with correct orientation (Fig. 1(c)), and precise peg insertion (Fig. 1(d)). To simplify user demonstrations, virtual widgets and buttons are created for user's interaction (Fig. 1(e)). To achieve precise control, we first align the virtual robot workspace with the real-world workspace using fiducial markers, as described in our previous work [56]. The ArUco markers placed on pre-assembled beams serve the dual purpose of synchronizing their virtual counterparts for the headset camera and guiding the precise robot operation through the wrist camera. To ensure the visual display's accuracy, we conducted a global calibration [57] to align the headset with the camera.

### D. Learning Information Collection

The RADAR system collects information from diverse demonstration methods. This information includes *policies* (representing low-level robot behaviors), *dynamic plans* (representing high-level assembly sequences), and *other human demonstration information*. State-based policies incorporate data from various demonstration methods, such as end-effector pose and joint angles. Planned trajectories and motor actions can also be collected as policy outputs. Dynamic plans involve task abstractions, structured by the task planner into subtasks or actions. Other human demonstration information, including step organization, failure correction, and user-added constraints, is captured by the RADAR system as well. Note that this work does not focus on collecting and training learning information; future research will explore its application in learning from demonstration.

## IV. RADAR Task Planner

### A. Overview

The dual-level reasoning concept serves as the fundamental, encompassing both coarse and fine-grained resolution [58]. Considering the complex control factors from both algorithms and human operators, there is a need for a robust algorithmic framework for robotics assembly. This framework must be capable of managing interactions, updates, and component replacements effectively.

With a given task goal, the reasoning process at the coarse resolution phase involves computing a sequence of *Abstract Actions*, and evaluating associated uncertainties. Subsequently, each *Abstract Action* undergoes a refinement process at the fine resolution phase, wherein it is tailored to *Concrete Actions*. To facilitate human-in-the-loop interactions, human subjective awareness is empowered to engage at both resolution layers. Humans can prioritize critical *Abstract Actions*, suggest modifications to the coarse-resolution sequence, and actively participate in fine resolution to perform intricate operations or address algorithmic failures. The RADAR system extends the action language $\mathcal{AL}_d$ [59] to describe action signatures and dynamic domains.

### B. Implementation

The RADAR's reasoning tasks, including inference, planning, and diagnostics, are performed based on a domain description developed using Answer Set Prolog (ASP) [60], which encompasses a dual-resolution system description denoted as $\mathcal{D}$, a collection of statements of $\mathcal{AL}_d$, and a historical record $\mathcal{H}$. The coarse-resolution system description $\mathcal{D}_c$ functions as a specification for the transition diagram $\mathcal{T}_c$, which defines the complete set of feasible solutions for the robotics assembly task. Both dual-resolution system descriptions are equipped with a sorted signature $\sum$, featuring customized sorts $\mathcal{C}$, sort hierarchy $\mathcal{S}$, and functions $\mathcal{F}$, as in:

$$\sum = < \mathcal{C}, \mathcal{S}, \mathcal{F} > \quad (1)$$

For example, the basic sorts of assembly domains include $direction(dir)$, $thing$, $robot(rob)$, $object(obj)$, $beam(bm)$, $joint(jnt)$, and $peg$. $\mathcal{S}$ describes the hierarchy of sorts, e.g. $\{rob, obj\} \in thing$ and $\{bm, jnt, peg\} \in obj$. $\mathcal{F}$ is partitioned into two distinct components: $\mathcal{A}$ and $\mathcal{DA}$, where they are respectively denoted as *actions* and *domain attributes*. Every function symbol $f \in \mathcal{F}$ is denoted using lowercase letters or lowercase words, along with sorts $\{c_0, ..., c_n\} \in \mathcal{C}$ representing parameters. Note that the result of *action* $\mathcal{A}$ is always in boolean form. The dual-level actions $\mathcal{A}$ include several reasoning descriptions in Tab. I.

$\mathcal{DA}$ is subdivided into: $\mathcal{DA}_s$, referred to as *statics*, which remains unchanged by actions, and $\mathcal{DA}_f$, referred to as *fluents*, which can be changed by actions. The *statics* within the

TABLE I: Actions Formulation and Description

| Actions | Parameters | Description |
|---|---|---|
| **Abstract Actions** | | |
| $move$ | $(rob, dir)$ | Navigate the robot gripper to a target pose. |
| $pickup$ | $(rob, obj)$ | Grasp an object using the robot's gripper. |
| $putdown$ | $(rob, obj, dir)$ | Place a grasped object to a target location. |
| $assemble$ | $(rob, obj, jnt)$ | Insert a beam fitting into an existing joint. |
| $fasten$ | $(rob, jnt, peg)$ | Secure pegs in connecting joints. |
| $push$ | $(rob, bm)$ | Adjust the position of existing beams. |
| **Concrete Actions** | | |
| $move$ | $(rob, dir)$ | Navigate the robot gripper to a target pose. |
| $shift$ | $(rob, dir)$ | Manipulate robot in Cartesian space. |
| $rotate$ | $(rob, dir)$ | Rotate the robot end-effector at the task level. |
| $face$ | $(rob, dir)$ | Quickly reset the gripper orientation. |
| $grasp$ | $(rob, obj, dir)$ | Grasp an object with desired orientation. |
| $gripper$ | $(rob)$ | Trigger the gripper to open or close. |

RADAR system for the beam assembly task encompasses relationships like $next\_to(bm_i, jnt_i)$ and $next\_to(jnt_i, jnt_j)$, which encode the spatial arrangement of joints on the pre-assembled beams. Furthermore, $next\_(dir, jnt)$ denotes the joint position from an $object$ sort a to $direction$ sort. *Fluents* are properties that can be changed by *actions*, including *inertial fluents* and *defined fluents*. Specifically, the following *fluents* of the domain are applied: $fix(jnt, peg)$, $assemble(bm, bm, jnt)$, $loc(thing, dir)$, and $inhand(rob, obj)$

$\sum$ also encompasses relations like $holds(f, step)$ and $occurs(action, step)$, signifying that a specific function is true and a specific action occurs within a plan at a particular step. Additionally, axioms are defined within both $\mathcal{D}_c$ and $\mathcal{D}_f$, which encompass the *Dynamic Causal Law*, *State Constraint*, and *Executability Condition*, such as:

$$\begin{aligned} &\neg hold(inhand, I+1) \leftarrow hold(inhand, I), \\ &\quad occurs(putdown, I) \\ &loc(obj, dir, I) \leftarrow loc(R, dir, I), inhand(I) \\ &\textbf{impossible}\ pickup(rob, obj)\ \textbf{if}\ inhand(rob, obj) \end{aligned} \tag{2}$$

The history $\mathcal{H}_c$ typically comprises a log of observations concerning basic functions $obs(f, boolean, step)$, and action executions $aeh(action, step)$. An answer set $\sum$ represents a feasible sequence of dual-resolution actions to achieve the goal based on the robot's beliefs. These sequences are computed using the SPARC system [61] by the clingo solver [62]. The results are stored and visualized in the AR interface. The control loop is represented in Algorithm 1. The function *human_involvement* applies the AR interface for task management, target object selection, and fine manipulation.

## V. Experimental Results

The performance of the RADAR system is evaluated using an open-source RAMP benchmark [17], which consists of various beam assembly sub-tasks designed to simulate the offsite construction of steel-rolled beams. We deploy RAMP system on xArm 6 robot arm (Fig. 1(a)). The control algorithms utilized in RAMP serve as fundamental baseline references for evaluation, referred to as RAMP. We utilize ArUco markers [63] instead of the April tag for better detection. Comparing overall task completion times and accuracy in the RAMP benchmark poses challenges due to the non-replicability of experimental scenarios and hardware conditions. The automated planning algorithm in RAMP imposes environmental constraints, such as strict beam placement along the robot's coordinate and known placement heights for all parts, simplifying the task but not accurately representing real-world intricacies. In addition, variations in robotic arms, operational speeds, and gripper types further affect execution time and success rates. Therefore, a direct comparison is not feasible. Instead, we decompose the entire task into several subtasks and compare the completeness and efficiency of the RADAR system and the RAMP benchmark across them. We evaluate nine subtask-specific metrics using data from the four scenarios shown in Fig. 5. Descriptions of each action are provided in Table II.

For each action, two conditions, RAMP and RADAR, are individually tested in 15 trials, except for 45 trials of Task 2 (15 trials for each position). Completeness is quantified as the ratio of successful executions to total attempts, as presented in Tab. II. The results indicate that when humans are involved in the loop, RADAR achieves higher completeness rates for fine actions. The execution times are derived from successful executions, as depicted in Fig. 6. Each dataset from various tasks and conditions has passed the Kolmogorov-Smirnov normality test, indicating that they follow a normal distribu-

**Algorithm 1:** Control Loop of RADAR Architecture

**Input:** $\mathcal{G}$: Task goal; $\prod(\mathcal{D},\mathcal{H})$: core ASP program; $\mathcal{HM}$: human intension and input; $\mathcal{P}$: other path planning algorithms

**Result:** task_states: statistics of assembly task

Create environment, load $\mathcal{P}$, initialize environment;
$s \leftarrow$ state of environment;
**while** $\neg$ *task_done(s)* **do**
  $hi \leftarrow$ human_intention $(s, \mathcal{HM})$;
  $tp \leftarrow$ task_planner$(s, \prod(\mathcal{D},\mathcal{H}), \mathcal{P})$;
  $a = hi \cup tp$;
  **if** $\neg human\_confirm$ **then**
    $a \leftarrow$human_demo$(\mathcal{HM}, a)$
  **else**
    $s' = s$ ;
    **while** $\neg$ *subtask_done$(s')$* **do**
      $tpf \leftarrow$ task_planner$(s, \prod(\mathcal{D}_f,\mathcal{H}_f), \mathcal{P})$;
      $a' \leftarrow$ human_demo$(s, tpf, a, \mathcal{HM})$;
      $s' =$ execute$(s, a')$;
      **if** $\neg fail\_exe(s')$ **then**
        $s' \leftarrow$ human_demo$(s', a', \mathcal{HM})$;
      **end**
    **end**
    $\mathcal{H}_c \leftarrow$ add_history$(s')$;
  **end**
  $s = s'$
**end**
return $task\_states$

TABLE II: Subtasks Completeness

| Task | Actions | Description | RAMP | RADAR |
|---|---|---|---|---|
| 1 | Pick up Single Peg | Grasp a single peg from a random position and place it in box A. | 66.67% | 86.67% |
| 2 | Pick up Peg Set (Single) | Grasp a single target peg from a full peg set and place it in box B. | 35.56% | 91.11% |
| 3 | Pick up Peg Set (Group) | Grasp 3 pegs with a designed sequence from a full peg set and place them in box B. | 20.00% | 66.67% |
| 4 | Pick up Beam (Short) | Grasp a target short beam from a random orientation and place it horizontally in area C. | 86.67% | 93.33% |
| 5 | Pick up Beam (Long) | Grasp a target long beam from a random orientation and place it horizontally in area C. | 53.33% | 93.33% |
| 6 | Insert Peg (Close, 1 Marker) | Grasp a target peg and insert it into the hole on joint D, guided by one joint marker. | 46.67% | 86.67% |
| 7 | Insert Peg (Far, 1 Marker) | Grasp a target peg and insert it into the hole on joint E, guided by one joint marker. | 33.33% | 93.33% |
| 8 | Insert Peg (Far, 2 Markers) | Grasp a target peg and insert it into the hole on joint F, guided by two joint markers. | 60.00% | 86.67% |
| 9 | Assemble & Push Beam | Grasp a target short beam, assemble it, and push it into joint G of a long beam. | 53.33% | 80.00% |

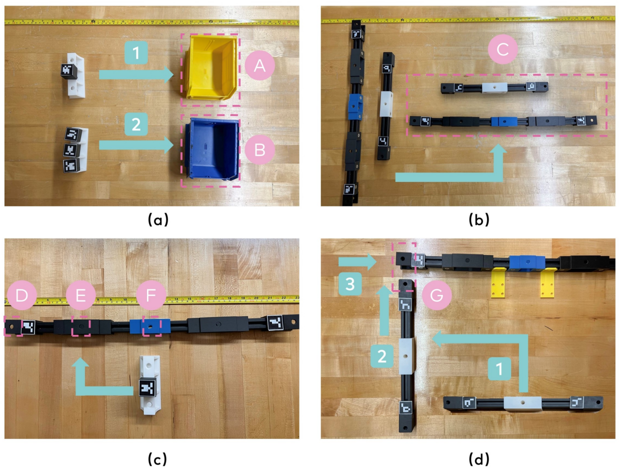


Fig. 5: The experimental setup for various sub-tasks: (a) pick up peg and peg set; (b) pick up short and long beam; (c) insert peg to different joints; (d) assemble and push beam to a joint.

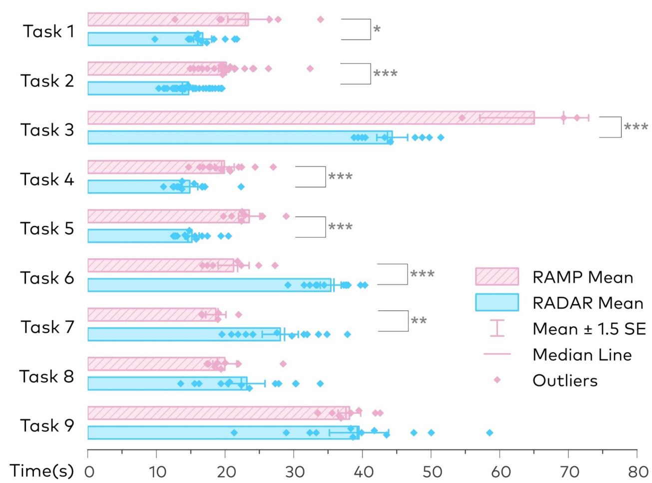


Fig. 6: Comparison of completion time for each action set: RADAR shows lower times for tasks 1-7, while RAMP performs better for inserting far pegs.

tion. Therefore, a two-tailed t-test is conducted to assess the comparative analysis between conditions in different tasks. Significant improvements in efficiency are observed in Task 1 ($p = 0.00325$), Task 2 ($p = 1.91 \times 10^{-9}$), Task 3 ($p = 0.000208$), Task 4 ($p = 3.56 \times 10^{-4}$), and Task 5 ($p = 8.97 \times 10^{-4}$). RAMP demonstrates quicker execution in inserting peg tasks, including Task 6 ($p = 3.38 \times 10^{-7}$) and Task 7 ($p = 0.00469$), but no significant differences in Task 8 ($p = 0.184$) and Task 9 ($p = 0.707$).

Results indicate that the involvement of humans significantly enhances the execution accuracy of long-horizon tasks. For certain coarse operations like target selection, orientation adjustments, or task sequencing, the RADAR system demonstrates superior and faster performance. For operations requiring high accuracy in environmental perception, integrating human input compensates for the limitations of sensor information, thereby achieving a comparable completion rate within a similar timeframe.

The slower peg insertion is attributed to the limitation of the robot's movement speed for a higher precision necessitating a pause for human coordination. Tasks 6 and 7 involve target destinations situated at a considerable distance from the initial position, resulting in extended robot travel times.

## VI. Conclusion

This research introduces the system design of RADAR, aiming to enhance precision and efficiency in complex, long-horizon assembly tasks from a systematic level. An AR-based interface and associated interactions are developed for human users to manage failures and fine-tune robot operations through demonstrations. A task planner and operational algorithm within the RADAR system are further developed for dual-level task planning. An extensive evaluation of the RADAR system is conducted, comparing it to the RAMP benchmark across various assembly subtasks, with a focus on completeness and execution time as performance metrics. Results consistently demonstrate significant accuracy improvements for fine-grained actions with human involvement. RADAR excels in specific coarse operations like target selection, orientation adjustments, and task sequencing, aided by AR visualization and interaction.

In the future, algorithms, and methodologies are expected to be developed to effectively utilize the collected demonstration data for enhancing robot capabilities in long-horizon assembly tasks. RADAR system can be potentially extended to other long-horizon tasks, such as disassembly, multi-robots and multi-human collaborations, and HMLV productions.

## References


[1] L. Gualtieri, I. Palomba, E. J. Wehrle, and R. Vidoni, "The opportunities and challenges of sme manufacturing automation: safety and ergonomics in human–robot collaboration," *Industry 4.0 for SMEs: Challenges, opportunities and requirements*, pp. 105–144, 2020. [Online]. Available: https://doi.org/10.1007/s43154-020-00006-5

[2] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "Substantial capabilities of robotics in enhancing industry 4.0 implementation," *Cognitive Robotics*, vol. 1, pp. 58–75, 2021. [Online]. Available: https://doi.org/10.1016/j.cogr.2021.06.001

[3] S.-K. Ong, A. Yew, N. K. Thanigaivel, and A. Y. Nee, "Augmented reality-assisted robot programming system for industrial applications," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101820, 2020. [Online]. Available: https://doi.org/10.1016/j.rcim.2019.101820

[4] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018. [Online]. Available: https://doi.org/10.1016/j.mechatronics.2018.02.009

[5] J. Arents and M. Greitans, "Smart industrial robot control trends, challenges and opportunities within manufacturing," *Applied Sciences*, vol. 12, no. 2, p. 937, 2022.

[6] F. Muhammad, A. Hassan, A. Cleaver, and J. Sinapov, "Creating a shared reality with robots," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2019, pp. 614–615.

[7] J. Lambrecht, L. Kästner, J. Guhl, and J. Krüger, "Towards commissioning, resilience and added value of augmented reality in robotics: Overcoming technical obstacles to industrial applicability," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102178, 2021. [Online]. Available: https://doi.org/10.1016/j.rcim.2021.102178

[8] J. Delmerico, R. Poranne, F. Bogo, H. Oleynikova, E. Vollenweider, S. Coros, J. Nieto, and M. Pollefeys, "Spatial computing and intuitive interaction: Bringing mixed reality and robotics together," *IEEE Robotics & Automation Magazine*, vol. 29, no. 1, pp. 45–57, 2022.

[9] E. Rosen, D. Whitney, M. Fishman, D. Ullman, and S. Tellex, "Mixed reality as a bidirectional communication interface for human-robot interaction," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 11 431–11 438.

[10] K. Chandan, V. Kudalkar, X. Li, and S. Zhang, "Arroch: Augmented reality for robots collaborating with a human," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 3787–3793. [Online]. Available: https://doi.org/10.1109/ICRA48506.2021.9561144

[11] M. B. Luebbers, C. Brooks, C. L. Mueller, D. Szafir, and B. Hayes, "Arc-lfd: Using augmented reality for interactive long-term robot skill maintenance via constrained learning from demonstration," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 3794–3800.

[12] M. Kapinus, V. Beran, Z. Materna, and D. Bambušek, "Spatially situated end-user robot programming in augmented reality," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–8.

[13] D. Bambušek, Z. Materna, M. Kapinus, V. Beran, and P. Smrž, "Combining interactive spatial augmented reality with head-mounted display for end-user collaborative robot programming," in *2019 28th IEEE international conference on robot and human interactive communication (RO-MAN)*. IEEE, 2019, pp. 1–8.

[14] J. Krieglstein, G. Held, B. A. Bálint, F. Nägele, and W. Kraus, "Skill-based robot programming in mixed reality with ad-hoc validation using a force-enabled digital twin," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 612–11 618.

[15] H. Wu, W. Yan, Z. Xu, T. Cheng, and X. Zhou, "A framework of robot skill learning from complex and long-horizon tasks," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3628–3638, 2021.

[16] S. Cheng and D. Xu, "League: Guided skill learning and abstraction for long-horizon manipulation," *IEEE Robotics and Automation Letters*, 2023.

[17] J. Collins, M. Robson, J. Yamada, M. Sridharan, K. Janik, and I. Posner, "Ramp: A benchmark for evaluating robotic assembly manipulation and planning," *arXiv preprint arXiv:2305.09644*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.09644

[18] L. Nägele, A. Hoffmann, A. Schierl, and W. Reif, "Legobot: Automated planning for coordinated multi-robot assembly of lego structures," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9088–9095. [Online]. Available: https://doi.org/10.1109/IROS45743.2020.9341428

[19] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual review of control, robotics, and autonomous systems*, vol. 3, pp. 297–330, 2020. [Online]. Available: https://doi.org/10.1146/annurev-control-100819-063206

[20] V. Chu, T. Fitzgerald, and A. L. Thomaz, "Learning object affordances by leveraging the combination of human-guidance and self-exploration," in *2016 11th ACM/IEEE international conference on human-robot interaction (HRI)*. IEEE, 2016, pp. 221–228. [Online]. Available: https://doi.org/10.1109/HRI.2016.7451755

[21] W. Si, N. Wang, and C. Yang, "A review on manipulation skill acquisition through teleoperation-based learning from demonstration," *Cognitive Computation and Systems*, vol. 3, no. 1, pp. 1–16, 2021. [Online]. Available: https://doi.org/10.1049/ccs2.12005

[22] D. Vogt, S. Stepputtis, S. Grehl, B. Jung, and H. B. Amor, "A system for learning continuous human-robot interactions from human-human demonstrations," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2882–2889.

[23] Y. Lee, E. S. Hu, and J. J. Lim, "Ikea furniture assembly environment for long-horizon complex manipulation tasks," in *2021 ieee international conference on robotics and automation (icra)*. IEEE, 2021, pp. 6343–6349. [Online]. Available: https://doi.org/10.1109/ICRA48506.2021.9560986

[24] M. R. Bahubalendruni, B. B. Biswal, M. Kumar, and R. Nayak, "Influence of assembly predicate consideration on optimal assembly sequence generation," *Assembly Automation*, vol. 35, no. 4, pp. 309–316, 2015.

[25] Y. Zhang, S. Sreedharan, A. Kulkarni, T. Chakraborti, H. H. Zhuo, and S. Kambhampati, "Plan explicability and predictability for robot task planning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 1313–1320.

[26] M. Stenmark and E. A. Topp, "From demonstrations to skills for high-level programming of industrial robots," in *2016 AAAI fall symposium series*, 2016.

[27] Y. Mollard, T. Munzer, A. Baisero, M. Toussaint, and M. Lopes, "Robot programming from demonstration, feedback and transfer," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 1825–1831.

[28] Z. Zhu and H. Hu, "Robot learning from demonstration in robotic assembly: A survey," *Robotics*, vol. 7, no. 2, p. 17, 2018.

[29] R. A. Knepper, T. Layton, J. Romanishin, and D. Rus, "Ikeabot: An autonomous multi-robot coordinated furniture assembly system," in *2013 IEEE International conference on robotics and automation*. IEEE, 2013, pp. 855–862. [Online]. Available: https://doi.org/10.1109/ICRA.2013.6630673

[30] N. Dhanaraj, S. V. Narayan, S. Nikolaidis, and S. K. Gupta, "Contingency-aware task assignment and scheduling for human-robot teams," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5765–5771. [Online]. Available: https://doi.org/10.1109/ICRA48891.2023.10160806

[31] J. Luo, E. Solowjow, C. Wen, J. A. Ojea, A. M. Agogino, A. Tamar, and P. Abbeel, "Reinforcement learning on variable impedance controller for high-precision robotic assembly," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3080–3087. [Online]. Available: https://doi.org/10.1109/ICRA.2019.8793506

[32] T. Jiang, H. Cui, X. Cheng, and W. Tian, "A measurement method for robot peg-in-hole prealignment based on combined two-level visual sensors," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2020. [Online]. Available: https://doi.org/10.1109/TIM.2020.3026802

[33] H. Fakhurldeen, F. Dailami, and A. G. Pipe, "Cara system architecture-a click and assemble robotic assembly system," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5830–5836. [Online]. Available: https://doi.org/10.1109/ICRA.2019.8794114

[34] B. Sadrfaridpour and Y. Wang, "Collaborative assembly in hybrid manufacturing cells: An integrated framework for human–robot interaction," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1178–1192, 2017. [Online]. Available: https://doi.org/10.1109/TASE.2017.2748386

[35] Z. Makhataeva and H. A. Varol, "Augmented reality for robotics: A review," *Robotics*, vol. 9, no. 2, p. 21, 2020.

[36] R. Fung, S. Hashimoto, M. Inami, and T. Igarashi, "An augmented reality system for teaching sequential tasks to a household robot," in *2011 RO-MAN*. IEEE, 2011, pp. 282–287.

[37] Y. E. Cogurcu and S. Maddock, "Augmented reality safety zone configurations in human-robot collaboration: A user study," in *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 360–363. [Online]. Available: https://doi.org/10.1145/3568294.3580106

[38] C. Wang, A. Belardinelli, S. Hasler, T. Stouraitis, D. Tanneberg, and M. Gienger, "Explainable human-robot training and cooperation with augmented reality," in *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, ser. CHI EA '23. New York, NY, USA: Association for Computing Machinery, 2023. [Online]. Available: https://doi.org/10.1145/3544549.3583889

[39] W. Yang, Q. Xiao, and Y. Zhang, "Ha r 2 bot: a human-centered augmented reality robot programming method with the awareness of cognitive load," *Journal of Intelligent Manufacturing*, pp. 1–19, 2023. [Online]. Available: https://doi.org/10.1007/s10845-023-02096-2

[40] R. C. Quesada and Y. Demiris, "Design and evaluation of an augmented reality head-mounted display user interface for controlling legged manipulators," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 950–11 956.

[41] C. P. Quintero, S. Li, M. K. Pan, W. P. Chan, H. M. Van der Loos, and E. Croft, "Robot programming through augmented trajectories in augmented reality," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1838–1844. [Online]. Available: https://doi.org/10.1109/IROS.2018.859370

[42] B. Akan, A. Ameri, B. Cürüklü, and L. Asplund, "Intuitive industrial robot programming through incremental multimodal language and augmented reality," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3934–3939. [Online]. Available: https://doi.org/10.1109/ICRA.2011.5979887

[43] J. E. Solanes, A. Munoz, L. Gracia, A. Martí, V. Girbés-Juan, and J. Tornero, "Teleoperation of industrial robot manipulators based on augmented reality," *The International Journal of Advanced Manufacturing Technology*, vol. 111, pp. 1077–1097, 2020. [Online]. Available: https://doi.org/10.1007/s00170-020-05997-17

[44] E. Rosen, N. Kumar, N. Gopalan, D. Ullman, G. Konidaris, and S. Tellex, "Building plannable representations with mixed reality," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 11 146–11 153.

[45] R. C. Quesada and Y. Demiris, "Proactive robot assistance: Affordance-aware augmented reality user interfaces," *IEEE Robotics & Automation Magazine*, vol. 29, no. 1, pp. 22–34, 2022.

[46] I. Soares, M. Petry, and A. P. Moreira, "Programming robots by demonstration using augmented reality," *Sensors*, vol. 21, no. 17, p. 5976, 2021.

[47] M. B. Luebbers, C. Brooks, M. J. Kim, D. Szafir, and B. Hayes, "Augmented reality interface for constrained learning from demonstration," in *Proceedings of the 2nd International Workshop on Virtual, Augmented and Mixed Reality for HRI (VAM-HRI)*, 2019.

[48] S. Makris, P. Karagiannis, S. Koukas, and A.-S. Matthaiakis, "Augmented reality system for operator support in human–robot collaborative assembly," *CIRP Annals*, vol. 65, no. 1, pp. 61–64, 2016.

[49] A. Luxenburger, J. Mohr, T. Spieldenner, D. Merkel, F. Espinosa, T. Schwartz, F. Reinicke, J. Ahlers, and M. Stoyke, "Augmented reality for human-robot cooperation in aircraft assembly," in *2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*. IEEE, 2019, pp. 263–2633.

[50] O. Danielsson, A. Syberfeldt, R. Brewster, and L. Wang, "Assessing instructions in augmented reality for human-robot collaborative assembly by using demonstrators," *Procedia CIRP*, vol. 63, pp. 89–94, 2017.

[51] N. Kousi, C. Stoubos, C. Gkournelos, G. Michalos, and S. Makris, "Enabling human robot interaction in flexible robotic assembly lines: An augmented reality based software suite," *Procedia CIRP*, vol. 81, pp. 1429–1434, 2019.

[52] F. De Pace, F. Manuri, A. Sanna, and C. Fornaro, "A systematic review of augmented reality interfaces for collaborative industrial robots," *Computers & Industrial Engineering*, vol. 149, p. 106806, 2020. [Online]. Available: https://doi.org/10.1016/j.cie.2020.106806

[53] Z. Wang, X. Bai, S. Zhang, M. Billinghurst, W. He, P. Wang, W. Lan, H. Min, and Y. Chen, "A comprehensive review of augmented reality-based instruction in manual assembly, training and repair," *Robotics and Computer-Integrated Manufacturing*, vol. 78, p. 102407, 2022. [Online]. Available: https://doi.org/10.1016/j.rcim.2022.102407

[54] W. Yang and Y. Zhang, "Visualization error analysis for augmented reality stereo video see-through head-mounted displays in industry 4.0 applications," in *International Manufacturing Science and Engineering Conference*, vol. 85819. American Society of Mechanical Engineers, 2022, p. V002T06A016. [Online]. Available: https://doi.org/10.1115/MSEC2022-85440

[55] W. Fang, L. Chen, T. Zhang, C. Chen, Z. Teng, and L. Wang, "Head-mounted display augmented reality in manufacturing: A systematic review," *Robotics and Computer-Integrated Manufacturing*, vol. 83, p. 102567, 2023. [Online]. Available: https://doi.org/10.1016/j.rcim.2023.102567

[56] W. Yang, Q. Xiao, and Y. Zhang, "An augmented-reality based human-robot interface for robotics programming in the complex environment," in *International Manufacturing Science and Engineering Conference*, vol. 85079. American Society of Mechanical Engineers, 2021, p. V002T07A003. [Online]. Available: https://doi.org/10.1115/MSEC2021-62468

[57] W. Yang and Y. Zhang, "A Global Correction Framework for Camera Registration in Video See-Through Augmented Reality Systems," *Journal of Computing and Information Science in Engineering*, pp. 1–17, 09 2023. [Online]. Available: https://doi.org/10.1115/1.4063350

[58] M. Sridharan, M. Gelfond, S. Zhang, and J. Wyatt, "Reba: A refinement-based architecture for knowledge representation and reasoning in robotics," *Journal of Artificial Intelligence Research*, vol. 65, pp. 87–180, 2019. [Online]. Available: https://doi.org/10.1613/jair.1.11524

[59] M. Gelfond and D. Inclezan, "Some properties of system descriptions of," *Journal of Applied Non-Classical Logics*, vol. 23, no. 1-2, pp. 105–120, 2013. [Online]. Available: https://doi.org/10.1080/11663081.2013.798954

[60] M. Gelfond and Y. Kahl, *Knowledge representation, reasoning, and the design of intelligent agents: The answer-set programming approach*. Cambridge University Press, 2014.

[61] E. Balai, M. Gelfond, and Y. Zhang, "Towards answer set programming with sorts," in *International Conference on Logic Programming and Nonmonotonic Reasoning*. Springer, 2013, pp. 135–147.

[62] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub, "Multi-shot asp solving with clingo," *Theory and Practice of Logic Programming*, vol. 19, no. 1, pp. 27–82, 2019. [Online]. Available: https://doi.org/10.1017/S1471068418000054

[63] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014. [Online]. Available: https://doi.org/10.1016/j.patcog.2014.01.005