Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

Unleashing mixed-reality capability in Deep Reinforcement Learning-based robot motion generation towards safe human–robot collaboration

Chengxi Li [a,b,c], Pai Zheng [a,b,*], Peng Zhou [d], Yue Yin [a], Carman K.M. Lee [a,b], Lihui Wang [e]

[a] *Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong Special Administrative Region of China*
[b] *Laboratory for Artificial Intelligence in Design, Hong Kong Science Park, Hong Kong Special Administrative Region of China*
[c] *Department of Aerospace and Mechanical Engineering, Viterbi School of Engineering, University of Southern California, LA, United States*
[d] *Department of Computer Science, The University of Hong Kong, Hong Kong Special Administrative Region of China*
[e] *Department of Production Engineering, KTH Royal Institute of Technology, Stockholm, Sweden*

ARTICLE INFO

Keywords:
Human–robot collaboration
Mixed reality
Deep Reinforcement Learning
Manufacturing safety
Smart manufacturing

ABSTRACT

The integration of human–robot collaboration yields substantial benefits, particularly in terms of enhancing flexibility and efficiency within a range of mass-personalized manufacturing tasks, for example, small-batch customized product inspection and assembly/disassembly. Meanwhile, as human–robot collaboration lands broader in manufacturing, the unstructured scene and operator uncertainties are increasingly involved and considered. Consequently, it becomes imperative for robots to execute in a safe and adaptive manner rather than solely relying on pre-programmed instructions. To tackle it, a systematic solution for safe robot motion generation in human–robot collaborative activities is proposed, leveraging mixed-reality technologies and Deep Reinforcement Learning. This solution covers the entire process of collaboration starting with an intuitive interface that facilitates bare-hand task command transmission and scene coordinate transformation before the collaboration begins. In particular, mixed-reality devices are employed as effective tools for representing the state of humans, robots, and scenes. This enables the learning of an end-to-end Deep Reinforcement Learning policy that addresses both the uncertainties in robot perception and decision-making in an integrated manner. The proposed solution also implements policy simulation-to-reality deployment, along with motion preview and collision detection mechanisms, to ensure safe robot motion execution. It is hoped that this work could inspire further research in human–robot collaboration to unleash and exploit the powerful capabilities of mixed reality.

1. Introduction

With the advancement of the Industry 5.0 paradigm, human–robot collaboration (HRC) has emerged as a fundamental component of modern smart manufacturing systems [1]. In an ideal HRC scenario, it is not only necessary to offload repetitive, low-skill, and ergonomically unfavorable tasks to robot partners, thus alleviating the physical burden on humans, but also to emphasize the significance of human intelligence and robots' dexterity manipulation and cognitive skills [2]. Currently, HRC should not only be limited to work cells in structured traditional industrial settings, but also be expanded to complete open and shared manufacturing scenes. Humans and robots can collaborate in unstructured environments, engaging in personalized small-batch and high-variety manufacturing activities such as quality control inspection, and product assembly/disassembly [3].

However, unstructured environments and unpredictable movements of humans lead most existing HRC solutions with pre-programmed robot motion not feasible anymore. To promote HRC in adapting to the unstructured environment, collaborative robotic platforms (cobots) are broadly adopted to assure physical safety with force-limited/speed-limited operations, emergency stops, etc. Additionally, the robot is also required to be able to adaptively plan movements in a safe manner when co-existing with humans [4,5]. Recently, Deep Reinforcement Learning (DRL) has demonstrated great potential for solving robotic safe motion-planning tasks in unstructured HRC environments [6–8]. However, insufficient state representation acquisition, over-complex scene settings, low transferability, and the lack of safety assurances prevent the efficient deployment of most DRL methods in human working environments currently.

[*] Corresponding author at: Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong Special Administrative Region of China.
E-mail address: pai.zheng@polyu.edu.hk (P. Zheng).

https://doi.org/10.1016/j.jmsy.2024.03.015
Received 4 October 2023; Received in revised form 19 March 2024; Accepted 30 March 2024
Available online 12 April 2024

At the same time, Mixed Reality (MR)-Head Mounted Display (HMD) serves as a portable and integrated human-machine interaction tool that has gained increasing interest in HRC [9,10]. On the one hand, MR-HMD could provide humans with visualization and communication towards geometric models, animations, and rich data flow based on the tasks, which assist humans in better decision-making [11]. On the other hand, spatial and visual computing provided by MR-HMD can digitally represent humans, perceive/localize scene information, and eventually align the virtual world with physical objects. Thus, MR-HMD could well afford the deployment of the DRL-based robot policy in an unstructured HRC environment without other external instruments. Moreover, it supplies safe assurance of deploying DRL via pre-execution like a simulator, without causing any damage to the physical entity. Currently, only few works have discussed the possibilities of integrating MR into the DRL learning loop [12].

To address the challenges associated with HRC, this study investigates the use of MR-HMD in DRL-based robot motion generation to realize safe HRC practice settings. Our study makes a threefold contribution, as follows:

- An MR-augmented robot control framework is proposed, which incorporates visual-aided gesture control, coordinate transformations, policy generation, and support for simulation-to-reality motion policy transfer in HRC scenarios. This framework facilitates intuitive collaboration between humans and robots, seamless transition between virtual and physical environments, and efficient transfer of pre-trained policies from simulation to reality.
- An MR-augmented DRL algorithm is developed and explored to enhance safe robot motion planning in HRC scenes. Leveraging MR-HMD as a source of spatial–temporal perception features, the algorithm utilizes a neural network to orchestrate and fuse the simultaneous capture of MR semantic vectors and image vector data through parallel data acquisition in the learning stage. By aligning spatial–temporal semantic vectors with temporal image sequences, an end-to-end policy is proposed, facilitating effective decision-making for robot motion generation in dynamic and unstructured environments.
- An on-site reactive collision detection approach is developed for executing robot motion within HRC scenarios, aimed at ensuring safety through the utilization of the spatial computing capabilities inherent in MR-HMD. This approach integrates on-site spatial-aware motion constraints, collision detection based on motion preview, and robust safety constraints to guarantee the feasibility of robot motion while prioritizing human safety.

The remainder of the paper is structured as follows. Section 2 briefly recaps the related work on MR applications in HRC. Section 3 states the task settings and the preliminary notation of the DRL. Section 4 formulates the proposed approaches regarding MR-augmented DRL for safe HRC. Section 5 discusses the experimental results for evaluating the proposed framework. Section 6 presents the discussion of Section 7 draw the conclusions of this work.

2. Related works

In this section, a concise overview of MR-assisted safe HRC applications is provided at the beginning. Additionally, the state-of-the-art robot control methods in the HRC safe domain are summarized, with focusing on the utilization of DRL.

2.1. Mixed-reality in safe HRC

MR technology could not only fuse virtual information into reality but also provide humans with the ability to perceive and interact with the surrounding environment. The immersive nature of MR helps to reduce human worker cognition workload and improve task processing efficiency. Malik et al. [13] designed a unified framework of a human–machine model for production system design based on virtual reality technology to meet the flexibility, adaptability, and safety of workpiece assembly. The system facilitates estimation of man-machine cycle times, process planning, layout optimization, and robot control programs, and allows end users to experience future production systems in an immersive environment. Choi et al. [14] proposed a safe distance calculation method based on the 3D offset of the robot digital twin and the human skeleton. Among them, 3D point cloud registration and deep learning instance segmentation are used in the process for environment and human detection, which are used to accurately measure the minimum safety distance in real-time, and provide operators with MR-based tasks and safety assistance. Focusing on safety, legibility, and efficiency, Chadalavada et al. [15] studied how robots (forklifts) in industrial logistics applications can communicate their navigational intent using spatial augmented reality so that humans can intuitively understand the robot's intent and feel safe near the robot. Aivaliotis et al. [16] and Hietanen et al. [17] both proposed an augmented reality-based way to show information about the process and the status of production and increase safety awareness by superimposing an active safety zone around the robot. Similarly, Li et al. [12] also adopted AR devices to propose a bidirectional human–robot safe interaction framework with the extraction of human states and robot states, including velocity control and prediction of potential collisions. Khatib et al. [18] proposed a system of multiple sensors for human–robot coordinated contactless motion tasks with maintaining a desired relative position and integrated the human–robot communication module within a mixed reality interface using an augmented reality system.

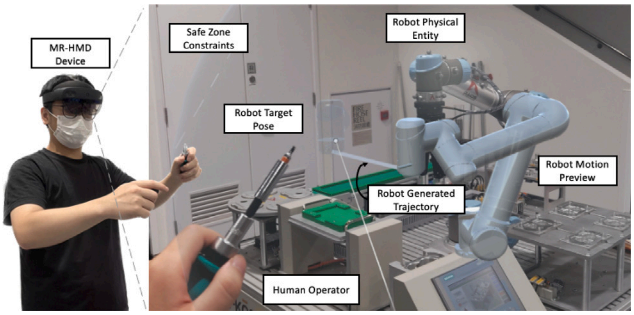


Fig. 1. Demonstrative setup of our MR-augmented safe human–robot collaboration scene.

Previous applications have demonstrated the effectiveness of MR technology in various safe applications. Currently, the utilization of MR devices has been primarily limited to visualization, feedback monitoring, and programming simple functions. Only a few studies integrate MR devices into the whole human–robot collaboration control process, particularly in state perception and decision-making capabilities.

2.2. Deep reinforcement learning in safe HRC

DRL, as a popular learning paradigm, has been broadly explored and adopted in robotic-related safe applications [19]. From robot manipulation policy exploration [20] to autonomous driving [21], the robot agent could mostly obtain outstanding performance. Regarding motion planning in safe HRC, DRL takes responsibility for collision avoidance and dodging moving obstacles to ensure human safety. El-Shamouty [6] proposed a safe HRC system based on DRL, which encodes the task and safety requirements and the context of applicability in RL settings. Then, adopting DRL with hindsight experience replay(HER)'s learning capabilities to improve the intelligence level and comprehensibility of the environment for robots. Similarly, Thumm et al. [7] and Schepp et al. [22] proposed a safe shield mechanism that combines ISO-verified human safety and DRL with HER algorithms on manipulators. The mechanism guarantees that the manipulator comes to a complete stop before a human is within its range by utilizing a fast reachability analysis of humans and manipulators. Liu et al. [23] designed the reward function by combining the external reward function and the intrinsic reward function with the Deep Deterministic Policy Gradient (DDPG) algorithm. The DDPG-based policy could be used to assess the risk and also enable the robot to dynamically avoid the human arm component. Chen et al. [24] and Li [12] et al. presented a DRL-based collision-avoidance trajectory planning for uncertain environments. The researchers created a simulated human–robot coexistence environment using the PyBullet and Unity physics engine. The performance of the environment was then assessed using the Soft Actor Critic (SAC), DDPG, and Proximal Policy Optimization (PPO) algorithms.

Hence, prior research has showcased the efficacy of DRL in enhancing robot intelligence, thereby enhancing their adaptability and flexibility in safe HRC activities. Nevertheless, the majority of these studies were carried out in simulated environments, featuring abundant observations and ideal experimental conditions that may not faithfully represent real-world settings. Consequently, there has been limited discourse on the direct implementation of the obtained policies in practical scenarios with safe assurance. Inspired by that, the integration of MR-HMD technology can used for bridging these gaps, provide the feasibility of deploying DRL policies in safe HRC activities.

3. Notations and problem formulation

3.1. Deep reinforcement learning

DRL is a decision-making optimization method that aims to improve performance by reinforcing an agent's interaction with the environment [25]. The decision-making processes are formulated as Markov decision processes (MDP) consisting of the following tuples: (S, A, R, P, γ). The elements of the tuple are the observation space S, action space A, reward function $R : S \times A \to \mathbb{R}$, a discount factor γ, and a state probability transition function $P : S \times A \times S' \to [0, 1]$ separately. The agent chooses an action $a_t \in A$ from the policy $\pi(a_t \mid s_t)$ according to the current state $s_t \in S$. Then, the state of the environment transitions to the next state s' with the probability $P(s_{t+1} \mid s_t, a_t)$, and the agent receives a reward R_{t+1} satisfying $\mathbb{E}[R_{t+1}] = R(s_t, a_t)$. The objective of DRL is to search for the optimal policy $\pi^*(a_t \mid s_t)$ to maximize the expected cumulative reward trajectory $J(\tau) = \int R(\tau)P\tau(\tau)d\tau$, where $\tau = (s0, a_0, \dots, s_t, a_t)$ denotes the trajectory under a randomly sampled task distribution $P(s)$ and policy function $\pi(a_t \mid s_t)$, resulting in $P\tau(\tau)$. The reward function $R(\tau) = \mathbb{E}\tau \left[\sum_{i=t}^{T} \gamma^{i-t} r_t\right]$ represents the cumulative reward of the expected trajectory in an episode from time step t to the terminal time step T.

3.2. Task & problem formulation

Task Configuration In the HRC task scenario, a human operator wearing MR-HMD device is responsible for performing various random assignment manufacturing tasks. These tasks are accomplished with the help of a robot attached to the workbench and may include activities such as assembly, disassembly, or inspection, among other things. In each iteration of a task, the robotic agent is required to navigate from a random starting pose, denoted as c_{start}, to a target pose specified by the operator c_{end} in a collision-free manner. It describes the condition in which motion execution occurs without any physical contact or unintended interference between human and robot, which implies that there are no instances of unwanted contact with the robot body, or potentially leading to safety hazards, physical harm, or disruption of the intended task. Subsequently, the robot triggers the task primitive to initiate the corresponding manufacturing procedure. Concurrently, the robot learns a continuous motion generation function, denoted as σ, which generates a path, denoted as τ, in the free configuration space C_{free} to avoid collisions with both the dynamic moving operator and the work station, which has a layout randomly generated as C_{obs}. For our practical experimental scene setup, please refer to Fig. 1.

Problem Definition The goal of safe motion task in this work is to find a collision-free trajectory of waypoints for a robot arm from the initial joint configurations c_{start} to reach the target pose c_{end}. The C denotes the configuration space, where $C \in \mathbb{R}^d$ and d refers to the dimension of the space. The obstacle set is denoted by $C_{obs} \subset C$. The collision-free configuration space set is defined as $C_{free} = C/C_{obs}$, the set of all configurations is not in collision. The solution of collision-free motion generation is provided by a parameterized continuous function $\sigma : [c_{start}, c_{end}] \to C_{free}$, which maps the waypoints during the interval $[c_{start}, c_{end}]$ to a specific configuration in C_{free}. With the given elements $C, C_{obs}, c_{start}, c_{end}$, finding a feasible optimal path σ^* is formulated as:

$$\sigma^*(c_{start}, c_{end}) = \overset{\sigma\ (c_{start},c_{end})\in C_{free}}{\arg\max}\ R(\tau) \tag{1}$$

where $R(\tau) \to \mathbb{R}$ is the reward function, which values all the state of the whole motion process τ and is detailed in Section 4.2.3.

4. Methodology

The integration of MR-HMD technology plays a pivotal role in facilitating safe HRC processes. It not only captures crucial data pertaining to human and robotic interactions but also offers a means to perceive the surrounding environmental context. These integrated functionalities render MR-HMD an optimal choice for supporting the Deep Reinforcement Learning (DRL) process in state representation learning, policy generation, and deployment. Through the utilization of MR-HMD devices, DRL can be effectively employed across the entire spectrum of robot motion generation in safe HRC activities, spanning pre-learning, learning, and execution stages. In the initial pre-learning phase, the MR-augmented approach encompasses MR-HMD-based data collection, assignment of robot target poses, input of task primitives, and the execution of coordinate system transformations. Subsequently, the focus transitions to the policy learning phase, where the primary objective is to efficiently and effectively generate an MR-augmented DRL-based policy in an end-to-end manner. Finally, the third stage involves the safe deployment of the DRL-based motion generation policy, facilitated by MR devices. To illustrate the systematic flow of the entire approach, please refer to Fig. 2.

4.1. MR-augmented pre-learning phase

The pre-learning phase of this work, as shown in Fig. 2, not only simply performs as a data collection and processing tool for the image and vector data utilizing its own characteristics, but also incorporates MR-HMD to support intuitive interactions with robots and tasks. Via MR-HMD, it enhances the human-operator experience by providing augmented visual feedback, enabling intuitive control to control the robot. In addition to that, MR-HMD enables coordinate system transformations via spatial computing to provide robot-usable data. Inspired by that, the following modules are proposed to address robot command transmission, and relative coordinate transformations.

4.1.1. Robot pose generation

In accordance with the task settings described in the preceding section, human workers assign goal end-effector poses to the robot for facilitating subsequent collaboration tasks, such as reaching, transporting and assembly, as needed. As depicted in Fig. 3, our implementation differs from the robot visualization of previous methods [12,26] in that only the end-effector anchor is visualized and manipulated during the interaction, rather than the entire robot model superimposing with the real one. This approach enables the generation and adjustment of a clear set of waypoints using workers' hand gestures through MR devices. By superimposing a non-intrusive robot anchor, not only the potential safety risks associated with the occlusion of objects is reduced but also alleviates cognitive load caused by past virtual-physical

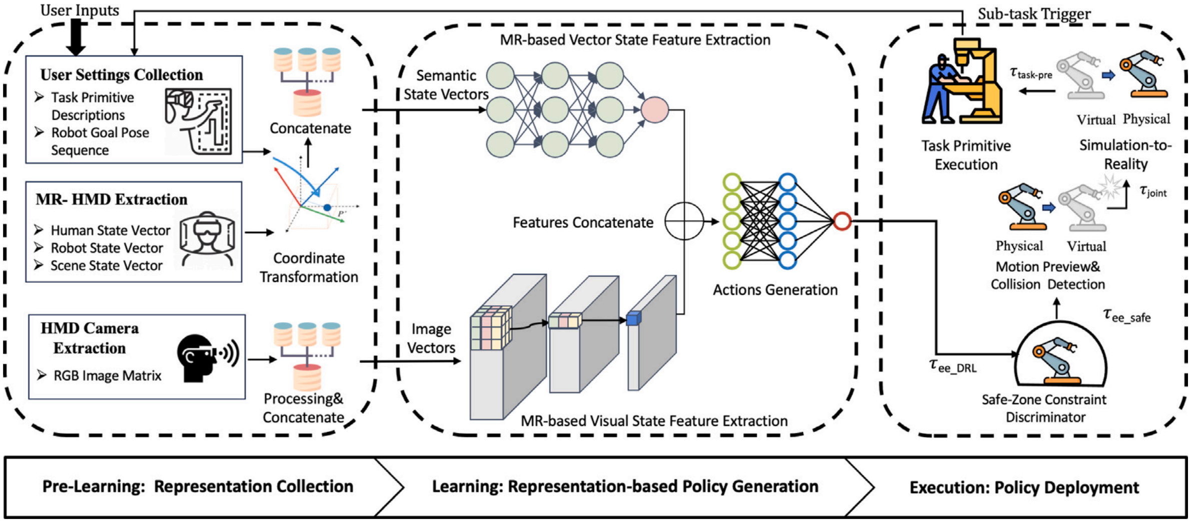


Fig. 2. MR-augmented robot safe motion planning and generation work flow in human-robot collaboration.

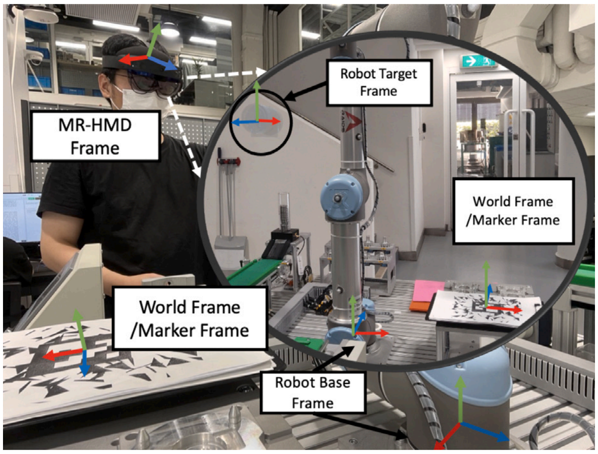


Fig. 3. Coordinate transformation in MR-HMD safe human-robot collaboration.

overlay scenes and animations. Moreover, the visual representation of waypoints prior to physical execution provides valuable guidance for workers. This process does not require prior knowledge of robot operation and can be carried out without causing any damage, as in simulated environments. It empowers workers with complete flexibility until the appropriate robot posture is determined based on different task requirements. The target robot pose frame Ψ_{tar} under world frame Ψ_0 is denoted as: $\begin{bmatrix} {}^0R_{tar} & {}^0T_{tar} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}$.

4.1.2. System coordinate transformation

Towards safe human–robot collaborative activities, the MR-acquired scene information (robot goal pose, human body info) must be transformed into the robot base frame for generating executable commands. To illustrate this process, five coordinate systems (frames) are involved in the transformation process, which are the world frame Ψ_0, the MR-HMD frame Ψ_{hmd}, the marker frame Ψ_m, the robot base frame Ψ_{base}, and the robot target pose frame Ψ_{tar} respectively. The HRC sample scene marked with coordinate systems is shown in Fig. 3

With the help of MR-HMD hand capture and spatial localization capability, the homogeneous transformation of each robot target pose frame Ψ_{tar} assigned from the MR-HMD frame Ψ_{hmd} to the world frame Ψ_0, is given by:

$$ {}^0H_{tar} = \begin{bmatrix} {}^0R_{tar} & {}^0T_{tar} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}; \tag{2}$$

$$ {}^0T_{tar} = {}^0T_{hmd} + {}^0R_{hmd} \cdot {}^{hmd}T_{tar} \tag{3}$$

$$ {}^0R_{tar} = {}^0R_{hmd} \cdot {}^{hmd}R_{tar} \tag{4}$$

where ${}^0T_{tar}$ is the translation vector of the target pose in the world frame Ψ_0, which consists of the translation vectors of MR-HMD and robot target pose in their local parent frames ${}^0T_{hmd}$ and ${}^{hmd}T_{tar}$. In addition to translation, the rotation matrix ${}^0R_{tar}$ is obtained in same way. For these, the elements in translation and rotation matrix could be obtained (via integrating depth camera and inertial measurement unit sensors) and calculated by the MR-HMD computing devices.

Similarly, to transform the robot to the target pose, the target pose also need be represented in the robot base frame Ψ_{base}. The homogeneous transformation matrix is presented ${}^{base}H_{tar}$:

$$ {}^{base}H_{tar} = {}^{base}H_m \cdot {}^mH_0 \cdot {}^0H_{tar} \tag{5}$$

$$ {}^{base}H_m = \begin{bmatrix} I_{3\times3} & {}^{base}T_m \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{6}$$

$$ {}^mH_0 = \begin{bmatrix} {}^mR_0 & {}^mT_0 \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{7}$$

where ${}^0H_{tar}$ is the homogeneous transformation of robot target pose in the world frame Ψ_0, denoted with Eq. (2). The transformation ${}^{base}H_m$, from the robot base frame Ψ_{base} to the marker frame Ψ_m, which is manually set and only own relatively constant translation without rotation. To simplify, the transformation ${}^mH_0$ between the world frame Ψ_0 and the marker frame Ψ_m is set to coincide with origin and orientation. Meanwhile, the marker frame Ψ_m could be dynamically detected by MR-HMD, thus the whole transformation could be carried out straightforwardly solely on an MR device. Thus, with homogeneous transformation ${}^{base}H_{tar}$, the target pose could be represented in a robot base frame and joints are configured by the inverse kinematic solver. Moreover, with the advantage of unifying the coordinates of all objects through straightforward calculations into a common-world coordinate. This streamlined approach not only provides information about the objects themselves but also enables the derivation of distance information. This feature could significantly enhances the efficiency of robot motion planning and provide more guidance in the next learning stage.

4.2. MR-augmented learning phase

After introducing the RL notations in Section 3.1, the SAC DRL trainer is adopted to train an optimal motion generation policy. SAC is an off-policy actor-critic method in the maximum-entropy reinforcement learning framework. Compared to the vanilla Actor-Critic method, SAC incorporates an entropy regularization term in the objective function, which encourages exploration in the learning process. It utilizes an actor-critic architecture with separate policy and value networks, the off-policy learning paradigm that enables the reuse of previously collected data, and entropy maximization to enable effective exploration, which makes the learning process more sample-efficient and allows for better utilization of collected experience. Finally, due to the soft update mechanism for both the actor and critic networks, SAC leads to more stable training and improved convergence properties, which helps in avoiding policy collapse and enhances learning performance. The goal of SAC is to automatically balance exploration and exploitation by maximizing the accumulated reward and the information content of the policy function. Therefore, the entropy $Entr$ of the policy is contained as part of the optimization objective:

$$J(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(s_i,a_i)\sim\rho_\pi}[r(\boldsymbol{s}_i,\ \boldsymbol{a}_i) + \alpha Entropy(\pi(\cdot|s_i))] \tag{8}$$

Compared to other mainstream DRL algorithms, SAC obtained stable and state-of-the-art performance on a range of classical benchmarks, and more details could be referred to the original paper [27]. Therefore, to verify the hypothesis that MR-augmented state representation could enhance agent performance and improve representation learning, our work is derived based on the SAC trainer.

4.2.1. MR-augmented spatial-temporal representation learning

As adopting DRL in robotic relevant applications, one commonly encountered challenge is to handling of sparse environmental information. This arises when the agent is tasked with achieving a difficult goal without any prior knowledge and guidance. To overcome this issue of task exploration, how to enrich observation information has been extensively discussed and proposed in the field. In particular, distance-based properties play a pivotal role in guiding the generation of the robot's trajectory in the realm of robot-safe motion planning. However, the acquisition of these distance terms can be challenging due to computational limitations, deployment investment and device constraints. Furthermore, in tasks with long horizons, historical information from previous scenes becomes crucial, but leveraging such information poses difficulties due to partial observability. To address the listed challenges, with MR-HMD's spatial computing capability and data buffer features, an MR-augmented DRL has been proposed for tackling the aforementioned issues within the algorithmic framework, offering a solution to enhance task exploration and overcome the limitations posed by sparse information, computational constraints, and partial observability.

For observation spaces, the outstanding performance of deep neural networks (DNN) has been proven in extracting a highly efficient representation (e.g. computer vision, natural language processing) and also in end-to-end control problems (i.e. mapping states to output actions). To improve learning efficiency and performance, MR-HMD devices in safe HRC activities not only obtain the visual pixel signals, but also support directly acquiring information about objects such as robots, human workers, and the environment. Thus, the MR-augmented state representation of DRL in this work consists of two streams: One is the visual input stream S_i, which could be acquired by the MR-HMD camera sensors and used for extracting the representation via DNN. It is used for extracting representations that humans may not be aware of or MR devices could not abstract. The other stream S_{MR} is an input of a pre-set abstracted semantic vector detected by the nature of MR devices, which consists of the robot state (joint, end effector), human state (hand, body, head), and environment state (layout, obstacles). These objects (position, pose, relevant distance) spatial information is all transformed into robot-usable data via MR-HMD's coordinate transformation feature. While the training process, all the objects come along with different semantic categories for safety constraint purposes, e.g. obstacles, human body, robots, etc. Moreover, only current state alone may not contain sufficient information to support optimal decisions, especially in dynamic scenes. By incorporating MR devices' data buffer features, MR devices maintain an internal memory that retains information about past observations or states including image and semantic vectors. The temporal information of the past 4 frames including all the representations also stored in the memory for further learning and control.

In the context of decision-making based on dual-stream temporal data, the structure of the network is shown in Fig. 2. Two data preprocess network components are introduced first to process the semantic vector and image vector of the dual-stream spatial–temporal information. Then, the concatenation component processes the fused data by concatenating the filtered information from both the semantic vector stream and the image stream. Then, the representation concatenation component combines the relevant representations from both streams into a single input representation information from the semantic and image streams. Lastly, the representation enables the action generation component to generate the corresponding appropriate action. In all, with the addition of MR-HMD, dual-stream spatial–temporal information is introduced, and by integrating the filtered dual-stream data in a concatenated manner, the robot could effectively leverage the complementary representation from both streams to support the decision-making process.

Therefore, each total state vector consists of 4 individual image vectors and semantic vectors in the past 4 frames. In each individual vector, the collected state S_t consists of an image vector S_i and a MR extracted information S_{MR} as follows:

$$S_t = [[S_{i_t}, \dots, S_{i_{t-3}}], [S_{MR_t}, \dots, S_{MR_{t-3}}]] \tag{9}$$

$$S_{MR_t} = [P_{r_t}, \dots, D_{h_t}] \in \mathbb{R}^{N\times 3} \tag{10}$$

where $P_t = [p_t^x, p_t^y, p_t^z]$ denote the positions of target goal, robot end-effector, fixed obstacle (table), and dynamic obstacle positions (random obstacle, human workers). The $D_t = [d_t^x, d_t^y, d_t^z]$ presents the relevant distance of the listed objects (target goal, static/dynamic obstacle) to the robot end effector, respectively.

4.2.2. IK-enabled action space

In safe HRC scenarios, robots are required to have complex interactions with human workers by using the end-effector. In the past, some work applied DRL to learn the robot safe motion planning policy with action spaces either in joint-level commands or continuous joint space [7], and the settings in DRL are intuitive. However, if the task involves reaching specific positions or manipulating objects in the environment, controlling the robot in Cartesian space can be more natural and intuitive. Instead of directly specifying joint angles, the RL agent can directly specify the desired Cartesian position in the global coordinate system. This simplification can make the RL agent's learning process more straightforward and potentially reduce the complexity of the action space. Thus, to reduce the exploration space of generating policy, in this work, an Inverse Kinematics (IK)-based action space is implemented. The bottom-level controller leverages an existing IK solver with existing robot description models (i.e. URDF) for configuring the robot joints to complete translating and rotating movement. The DRL-based controller mainly focuses on generating feasible trajectories for the robot. With the IK-controller, the action space is not only simplified from continuous to discrete but also reduces the dimensions of actions from 6/7 dimensions (determined by robot DoF) joint configuration space $q = \{q_j, j \in [1, \cdots, N]\}$ to 3-dimensional actions considering only translation in Cartesian space $A = [\Delta x, \Delta y, \Delta z]$ with fixed orientations. Moreover, it could also significantly improve the adaptability of ready-trained policy (policy optimized by the DRL through the process of training) among different platforms for further spread in safe HRC applications, and such generality can be useful when deploying the ready-trained policy on robots with different physical characteristics or when adapting the policy to new tasks.

4.2.3. Composite reward space

In the realm of robot applications, target-only settings have long been regarded as the most intuitive and convenient approach for defining task goals within the framework of DRL. However, these reward configurations present certain challenges due to the delayed nature of reward signals. As a result, the feedback provided by such reward functions tends to be sparse, leading to a significantly larger search space and impeding the learning efficiency. Furthermore, in the absence of constraints, the ready-trained policy may converge towards unintended behaviors. To address these issues and enhance the overall performance of DRL in robot applications, it is crucial to augment the reward function by incorporating specific conditions or limitations into the reward function to guide the learning process (i.e. constraints), and constraints can be used to shape the behavior of the RL agent and encourage desired actions or discourage undesired actions. For example, in a robotic arm control task, a constraint could be to penalize the agent for hitting obstacles or exceeding joint limits in order to encourage safe and collision-free movements. It not only can the task settings be more accurately described, but also help to shape the learned policy towards desired behaviors and ensure that the agent operates within certain bounds or guidelines.

In the context of the specific task of safe HRC, a composite reward function has been designed. This function is composed of several distinct components that account for the task's goal state, safety requirements, and the measurement of progress in terms of task completion. Notably, a distance-based reward is assigned to assess the advancement made towards achieving the task objectives. Considering the aforementioned considerations, our task reward settings involve the decomposition of the reward function into the following components:

$$R_t = \begin{cases} R_{goal} : W_{goal} \cdot (d(r\_ee, goal) < \delta) \\ R_{f\_obs} : W_{obs} \cdot (collision(robot, obstacle) > 0) \\ R_{h\_obs} : W_{human} \cdot (collision(robot, human) > 0) \\ R_{dist} : W_{dist} \cdot \sum \left[d(p_{\mathrm{goal}}, p_{\mathrm{r\_ee}}), \dots, d(p_{\mathrm{h}}, p_{\mathrm{r\_ee}}) \right] \end{cases} \tag{11}$$

In this context, R_{goal} represents whether the robot has reached its target, and δ is the distance threshold used to determine goal attainment. $R_{f_{obs}}$ and $R_{h_{obs}}$ indicate the penalty for collision occurrence with either fixed obstacles or humans. To guide the robot towards successful completion of the task, the Euclidean distance between objects is calculated using the function $d(\cdot)$, and the reward function R_{dist} is designed accordingly. The collision detection function provided by the simulator, i.e. $collision(\cdot)$, trigger and outputs 1 for collision occurrence. It also triggers the termination signal, which denotes the end state of the process. The weights W_i are used to balance the risks and benefits of reaching the goal and avoiding collisions.

4.2.4. Parallel training data collection

As stated, distinguished by its facilitation of off-policy learning, SAC empowers agents to learning from historical policy collected data, thus amplifying sample efficiency and optimizing utilization of accrued experiences. Noteworthy is that SAC is a policy-based framework, typically operationalized through a single-threaded architecture wherein an agent interfaces with the environment to accumulate experiential data. In light of the fixed environment dynamics and continuous action, these experiences evince conspicuous temporal correlations, with only a subset of the state and action space explored within finite temporal bounds. To address these challenges, a parallel training environment is adopted characterized by a multi-threaded architecture. By this training approach, the sampled experiences become independent and effectively decouple the interdependence between experiences. Through the deployment of multiple workers simultaneously exploring replicas of the environment, the training agents enable the concurrent exploration of diverse segments of the environment. This innovative strategy not only enhances the efficacy of computational resource utilization but also serves to augment training efficiency significantly.

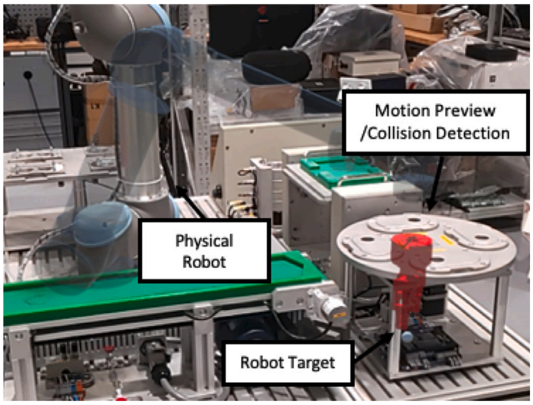


Fig. 4. Demonstration of MR-augmented safe motion preview and collision detection.

4.3. MR-augmented execution phase

In the previous section, leveraging the advantages of MR-HMD, the robot motion policy is generated with the MR-augmented SAC algorithm. After training, the other crucial problem lies in how to transfer the policy derived from the simulator to a real robot (Sim-to-Real) for the benefit of safe HRC deployments. During policy deployment in practice, MR-HMD could help the process in these two steps, policy Sim-to-Real deployment, and verification of robot safe execution. The detailed process and the demonstrative cases are presented in Algorithm 1 and Fig. 4.

4.3.1. Policy deployment

To deploy the policy gained from the simulator on a real robot, the MR-HMD first projects the whole area of the scene and maps the constraints, attributes of the environment, and robot configurations from the physical scene to a simulated scene. With that, the ready-trained optimal control policy could publish the robot actions and solve the joint configuration. The whole process runs in the following steps:

- The goal poses Ψ_{tar} via MR-HMD Ψ_{hmd} firstly determined and feed into the homogeneous transformation matrix ${}^{base}H_{tar}$ to transform into the pose under the robot base coordinate system.
- Via the MR-HMD, the image and vector state representations are collected and concatenated. The representation is fed into the previous ready-trained DRL-based policy π^{*}_{DRL} and generates the action trajectory.
- Each waypoint in the trajectory could be solved by an IK solver to calculate the joint configuration from Cartesian space. With the calculated the goal joint values, the physical robots are synchronized to perform motions.

4.3.2. Safe execution assurance

In DRL settings, the penalty term of reward functions are usually soft penalties. Thus, it may lead the policy cannot totally guarantee the safety of the output actions. Regarding the on-site safe assurance of motion execution, two MR-augmented safe execution assurance measurements are proposed:

- Safe Zone Discriminator: By harnessing space transformation techniques and leveraging MR-HMD depth sensors, a human worker can manually configure a safe workspace for robots. This process allows determining the position and distance relationship between the end-effector position of the target and the feasible working space. In practice, if the DRL policy outputs actions that would cause the robot to move outside the defined area, the robot's actions are blocked. This prompts a re-planning of the trajectory to ensure safe collaboration within the constrained workspace. Through the combined use of space transformation and MR-HMD depth sensors, the manual configuration of the safe workspace empowers human workers to establish boundaries for robot motion. By enforcing these boundaries and triggering trajectory adjustments when necessary, the safe collaboration between humans and robots can be maintained.
- Space-aware Motion Preview: The spatial computing capabilities of MR devices facilitate the reconstruction of objects within the work area, which can serve as potential obstacles to collision during the motion execution phase. To mitigate unexpected and unsafe actions, a virtual robot motion preview, denoted Ψ_{anchor}, is performed before the physical robot is executed. This preview occurs in advance within a time interval Δt between each waypoint, aligned with the planned movement derived from DRL. By leveraging the space-aware motion preview, the robot can proactively identify potential collision risks involving obstacles and humans. This enables the robot to trigger collision avoidance measures and re-plan its trajectory, ensuring both safety and operational efficiency. For a visual representation, refer to Fig. 4.

In DRL-based safe motion generation, reactive on-site collision detection plays a crucial role in robot motion driven by DRL for execution safe guarantee. Firstly, collision detection helps ensure the safety of both the robot and its environment. By detecting potential collisions with obstacles or other objects in the environment, the robot can take appropriate actions to avoid collisions and prevent damage to itself or its surroundings. This is particularly important in scenarios where robots operate in close proximity to humans or objects. Meanwhile, collision detection is essential for generating collision-free paths for the robot to follow. It allows the robot to analyze the environment and identify obstacles that need to be avoided. By incorporating collision detection into the motion planning process, the robot can re-plan its trajectory to navigate around obstacles and reach its desired goal while avoiding collisions. Inspired by this, spatial computing capability is well leveraged to enable space-aware motion preview collision detection. In working scenes, objects in working areas may become potential obstacles during the motion execution phase. The spatial computing capabilities of MR devices facilitate the reconstruction of these objects for the detection of motion collisions. Moreover, the planned robot motion path will be pre-run and previewed based on virtual robots. By leveraging the space-aware motion preview, the robot can identify potential collision risks involving obstacles and humans. This also enables the robot to trigger path replanning to ensure safety. Finally, a hard constraint is added based on the distance threshold; Once the trajectory is planned, which is not in the feasible workspace, the MR devices also could terminate the planning process and replan it. These significantly reduce the cost of deploying DRL in the real scene and improve efficiency. Additionally, the system also uses the physical safety assurance of the cobot to protect human operators from being harmed. Cobots are specifically designed to operate alongside humans in shared workspaces. Thus, it employs features such as force-limited/speed-limited operation and emergency stop. These mechanisms prevent the cobots from exerting excessive force that could potentially harm human workers. In all, incorporating these features listed above during the execution stage could better ensure that the robot's actions properly complete the task and provide the fundamental safety protection in HRC.

Algorithm 1: Deep Reinforcement Learning-based Robot Motion Policy Deployment

```
Initialization:
current pose frame $\Psi_{anchor}$;
target pose frame $\Psi_{tar}$;
while $\Psi_{tar} \neq \Psi_{anchor}$ do
    environment state $S_t \leftarrow [[s_{i_t}, \cdots, s_{i_{t-3}}], [s_{MR_t}, \cdots, s_{MR_{t-3}}]]$
    robot action $a_{optimal} \leftarrow \arg\max \pi^*_{DRL}(\boldsymbol{a}_t \mid S_t)$;
    $\Psi_{anchor} \leftarrow a_{optimal}$
    if $\Psi_{anchor}$ not in safe zone; then
        $safe \leftarrow$ False;
        return $safe$ ;
    end
    $q = \{q_j, j \in [1, \cdots, N]\} \leftarrow$ InverseKinematic($\Psi_{anchor}$)
    if $\Psi_{anchor}$-based mesh collide; then
        $safe \leftarrow$ False;
        return $safe$ ;
    end
    return $safe$
end
```

5. Experiment & results

In this section, the DRL algorithm is first adapted to the safe HRC scene with randomly generated obstacles using the Unity and Gym environments. Meanwhile, the performance of different MR-augmented settings for DRL towards the proposed safe HRC tasks is evaluated separately. Lastly, the policy is deployed on an MR-HMD and to corporate with a real assembly platform to demonstrate its effectiveness and safety concerns.

5.1. Experimental design

5.1.1. Device and development

In this work, the algorithms and relevant improvements are derived in C# and Python with the assistance of Mixed Reality Toolkit and Pytorch. The physical implementations are mostly based two hardware devices, which are Microsoft HoloLens2 MR-HMD and a 6-DoF UR5 collaborative manipulator separately. Regarding HoloLens2 MR-HMD, the development workload is mainly focused on realizing the whole process of hand gesture-based command assignment, multiple coordinate transformation, integration of the DRL algorithm, and collision detection measurements of robots. For the UR5, the development mainly adopted a series of Python packages used in the Ubuntu 18.04 environment for joint command transmission and relevant robot command execution.

5.1.2. Experimental settings

In the experiments, the environmental setup of the test scene consisted of robots, workstation, part-like static obstacles, and human-like dynamic obstacles. Regarding the presented obstacles, two parts and one working station are static but take up space, and two dynamic human-like ones randomly move around the robots. The end-effector target position is randomly assigned and the orientation is vertical ground. The task is to learn a planning policy that generates a sequence to drive the robot tuneable end-effector area (red sphere) to the target position for task primitive execution in collision-free to obstacles. In the scene, the policy is deployed on the 6-DoF UR5 robot agent with a suction gripper (see Fig. 5).

In the training process, in each task episode, the robot starting poses are randomly configured within the reachable area (radius $0.5m$) of the robot. Robots driven by DRL policy are allowed to reach each target position within a maximum trajectory length of T in each episode, where T is set to 200 RL decision steps to allow the robot to evade around

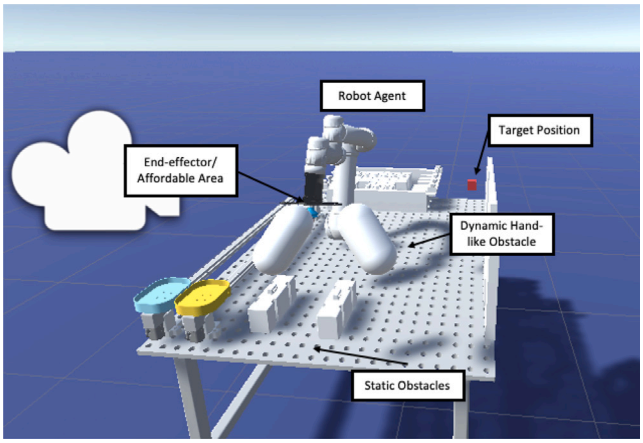


Fig. 5. Simulated scene setup of experimental evaluation.

Table 1
Hyperparameters of DRL training.

DRL common hyperparameters	Value	SAC hyperparameters	Value
network hidden layers	3	replay buffer size	50 000
learning rate	0.0003	buffer initialization step	1024
batch size	512	soft update coefficient τ	0.5
network hidden units	1024	entropy regularization coefficient	0.005
learning steps	500 000	initial entropy coefficient	0.5
discount factor γ	0.99	batch for model updates	1

Table 2
Comparative results of DRL algorithms in safe HRC setting.

Scene	Metrics	Synchronous advantage actor-critic [28]	PPO [29]	SAC(ours)
Static obstacle	Reward	−4.6	5.704	7.399
	Success Rate	1.6%	86.45%	98.04%
	Episode Length	7.964	14.33	12.73

the human and obstacle to any goal. Meanwhile, during training, an episode may be terminated earlier due to collisions and successful goal achievements, leading to a maximum of 500 000 RL steps and with CPU Intel I9-10900K and Nvidia GTX 3070TI GPU.

In the experiment, the task success distance threshold is settled to 0.1 cm and the collision detection function adopted the Unity integrated. The position of the goal, considering the randomness of the obstacle locations, is sampled in a uniform distribution around the robot target with a radius of 0.2–0.5 m. Similarly, dynamic obstacles are generated uniformly at random within the 0.1–0.6 m boundary around the robot. The moving direction of the dynamic obstacle can be randomly varied to move away from or towards the target location with a speed of 0.15 m/s. In performance validation, starting points are randomly sampled around the robot, and the experiment was 5000 steps for each goal position to allow for different obstacle positions and velocities for each goal to determine the policy performance. In robot morphologies adaptation evaluation, the approach is ported to the 4 different robots for evaluation to identify the feasibility of our proposed system.

5.2. Results

With respect to performance evaluation, it is important to consider not only the success rate but also the enhancements brought about by the MR-augmented measures. In this section, the experiments employed SAC as the fundamental learning algorithm to assess the effectiveness of the proposed approaches towards the static obstacle and dynamics obstacle avoidance scenes. In the experiments, a test scene is first initialized as a DRL environment with action space and composite reward shaping. Then, the task success criteria are set, where the robot reaches the target position (deviation ≤ 10 mm) within a finite task time horizon (i.e., execution time ≤ 30 s, 200 decision intervals) and under task safety assurance (i.e., human–robot distance ≥ 10 mm, robot-obstacle distance ≥ 10 mm). Moreover, a visual-only policy is employed to perform as the benchmark and introduce the temporal features. The performance evaluations were conducted by integrating MR-augmented state representation to enrich spatial relationships. Subsequently, temporal sequence enhancements were incrementally introduced to gauge the efficacy of the MR components. Furthermore, within the optimal policy framework, parallel data collection mechanisms and various robot platforms were incorporated during the rollout stage and re-training process to assess learning efficiency and transferability.

5.2.1. MR-augmented policy performance

In this part, the objective is to compare the performance changes caused by agents that target different combinations of state representations brought by MR features and also to determine the effectiveness of adding features enhanced with MR. Before that, a comparative experiment with the classical actor-critic reinforcement learning algorithms is carried out for the pilot study reason to evaluate the effectiveness of the benchmark algorithm (i.e. SAC), the experiment executed based on the static obstacle scene with pure MR-augmented semantic vector observation and the rest of the settings remains the same. The adopted hyperparameters are listed in Table 1. Owing to the listed advantages in Section 4.2, SAC achieves best performance among those classical algorithms in the safe motion generation task settings as well, as shown in Table 2. In addition to that, in the evaluation stage, the SAC method comes as a single visual perception as a baseline to reflect the roles of the MR-augmented components and utilize the unified reward functions to fairly compare the improvement of the learning capabilities. The experiment initializes with the state representations containing visual information only, and the perception algorithm only adopts the vanilla convolutional neural network proposed by Mnih et al. [30] to identify the feasibility while considering hardware constraints.

Due to the intricate nature of the scene and the limited capacity to learn representation, the only visual-based policy did not converge satisfactorily, rendering the learning curves and success rates depicted in the figure irrelevant. Subsequently, with the integration of MR devices, the inclusion of semantic vectors markedly enhances representation learning capabilities, enabling successful convergence of the policy within a static obstacle scenario. However, in dynamic scenes characterized by expansive exploration space and human-induced uncertainties, extracting representations becomes considerably challenging. Subsequently, by leveraging the buffering function of MR-HMD, temporal visual and vector sequences are incorporated into the observations of SAC, thereby augmenting temporal awareness and decision-making capabilities. Comparative analysis with baseline reveals notable performance improvements, particularly in dynamic obstacle scenarios. This performance enhancement is further bolstered by aligning vector and visual inputs, resulting in success rates of 98.6% and 83.8% in static and dynamic scenes, respectively. Furthermore, the adoption of parallel data collection mechanisms not only enhances learning efficiency but also reduces training time. The performance metrics are summarized in Table 3 and are depicted in Fig. 6. Specifically, the training time of parallel data collection-based approach demonstrates a reduction of approximately 21% and better convergence compared to single-threaded Visual-Temporal approaches but remain same performance level.

Table 3
Result of MR-augmented policy in safe HRC.

Scene	Metrics	Visual-only	MR vector-only	Visual-MR vector	Visual-temporal +MR vector-temporal	Parallel visual-temporal +MR vector-temporal
Static obstacle	Reward	−5.119	**7.399**	7.344	7.358	7.398
	Success Rate	–	**98.06%**	96.24%	97.17%	97.2%
	Episode Length	–	12.73	12.08	**11.13**	10.5
Dynamic obstacle	Reward	−6.479	5.269	6.249	**6.784**	6.743
	Success rate	–	70.3%	81.9%	**83.8%**	82.64%
	Episode length	–	30.53	10.59	**17.0**	12.09

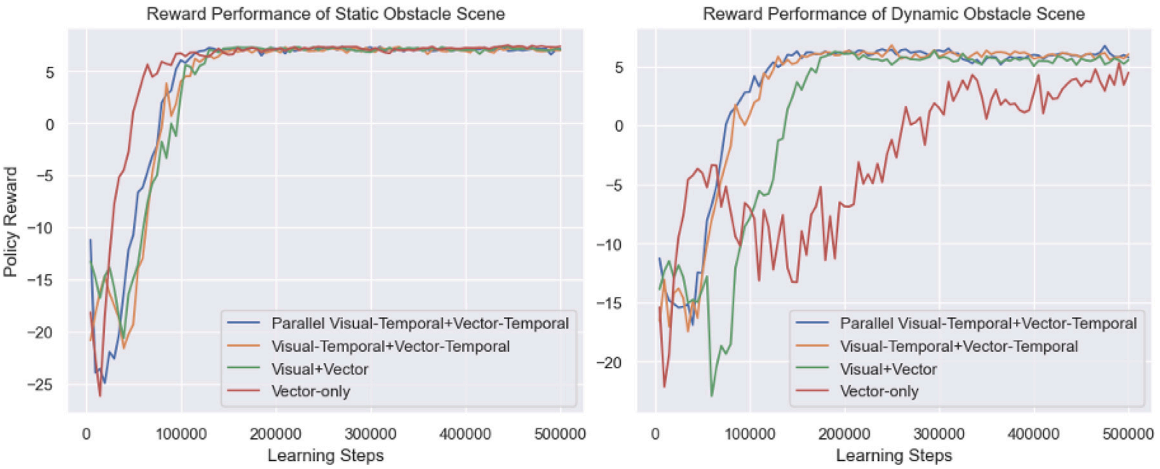


Fig. 6. Rewards learning curves of different DRL settings in static obstacle motion planning tasks (left) and dynamic obstacle motion planning tasks (right).

Table 4
System adaptability for various robots motion planning task in safe HRC.

Scene	Metrics	UR5	UR5e	UR3	UR10	IIWA7
Static obstacle	Reward	7.398	6.014	6.339	4.435	7.138
	Success Rate	97.2%	89.3%	99.8%	74.2%	99.8%
Dynamic obstacle	Reward	6.784	5.517	6.165	3.191	6.353
	Success Rate	83.8%	80.1%	94.2%	46.2%	86.3%

5.2.2. System adaptability

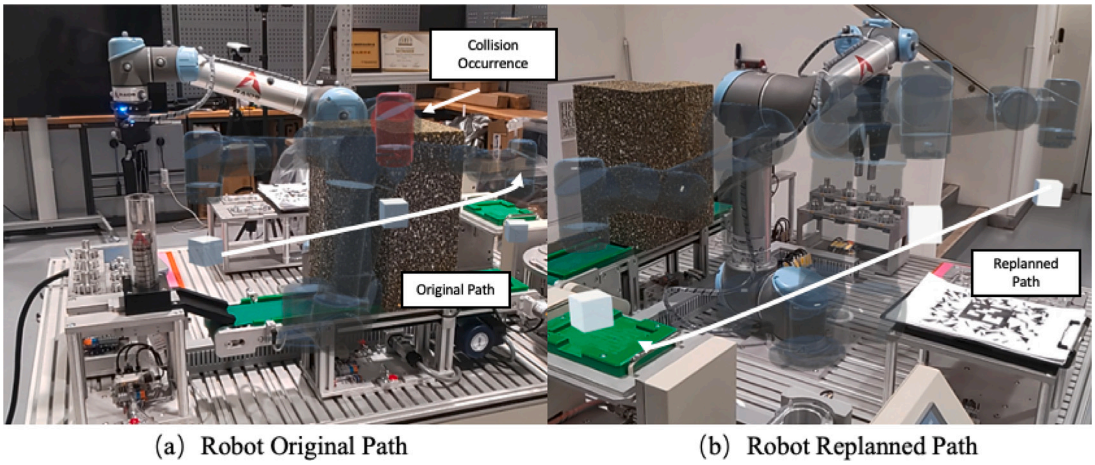
In terms of system adaptability, only the Cartesian positions of the robot end-effector are utilized as inputs to the existing IK solver during the training of a robot policy using DRL. Consequently, trained robot policies are expected to exhibit comparable performance between robots with different morphologies and demonstrate similar convergence characteristics for various types of robots. Thus, the current systematic approach demonstrates adaptability to all distributions of the HRC motion planning tasks. Table 4 illustrates that most robots achieve similar performance, albeit with slight deviations observed in UR3, UR10, and IIWA 7. Variations in performance arise from factors such as redundant joints (7 DoF) and appropriate morphological configurations of the robot (for example, size UR3). Conversely, performance declines can be attributed to inappropriate shapes and sizes that may restrict flexibility and lead to collisions (e.g., UR10). Furthermore, attempts have been made to directly transfer well-trained policies between robots, albeit without achieving satisfactory performance. This is primarily due to the distinct shapes and degrees of freedom inherent in each robot, encoded within the agent's latent space during the feature extraction stage.

5.2.3. Motion preview & collision detection

The collision verification is based on collision checking of the potential reachable sets of all possible human motions and the preview of the intended robot trajectory. In the implementation, the human body and hand are detected by the MR-HMD integrated camera modeled with a capsule set. The robot occupancies are modeled by the original robot model and the environment is reconstructed by triangle pieces as a convex set of original shapes. As shown in Fig. 4, the robot occupancy trajectory is described as a set of close sets during execution time. The motion of a robot is verified as safe if no collision is detected throughout the process before reaching the target. In practice, confined to the computing power of Hololens2, the collision detection based on the environment detection frequency is only up to 2 Hz, the collision detection frequency is 50 Hz and the collision detection success rate reaches around 95% in the experimental scene. The failures are affected by challenging lighting conditions, object occlusions, or material reflections in the working scene. The common poor lighting, shadows, or glare can significantly impact the accuracy and reliability of MR-HMD sensor-based perception systems, leading to potential errors or incomplete information. A group of the comparative sample execution trajectories is attached in Fig. 7 for reference. Meanwhile, for better demonstrative performance, the virtual robot is visualized.

6. Discussion

In this work, a systematic solution is proposed for safe robot motion generation and execution in collaborative human–robot manufacturing activities, leveraging MR-HMD technologies and DRL. It covers a series of works on robot-safe motion planning, from easing task assignment to supporting policy learning&execution. With validation, the approach has shown satisfactory and safe motion in various sample HRC scenarios, yielding encouraging and promising results. Moreover, in terms of the various industrial conditions, the solution could be efficiently plugged in by changing the safety settings, replacing the robot model, and adding additional constraints to keep the system working safely through the configurability of AR devices. In addition to that, with the flexibility and adaptation of DRL, the robot motion generation policy towards different robots could also be easily fine-tuned and adapted to the new robot settings. Meanwhile, the MR-based framework also brings a great benefit to the DRL control policy towards the practical deployment of HRC scenes (i.e. Sim2Real).



(a) Robot Original Path (b) Robot Replanned Path

Fig. 7. Sampled trajectory in practical manufacturing task scene.

Despite the significant contributions outlined in this research, several limitations should be acknowledged. One of such limitations is the potential for discomfort associated with the long-term use of V/AR HMD, which can result in nausea, headaches, and decreased productivity. However, it is important to note that our proposed system differs from a VR-like immersive operational environment. Instead, it is deployed and superimposed based on the physical world, providing realistic object references. To minimize cognitive load, the animation of robot movement can be reduced to avoid disruption of normal working activities. Based on brief interviews with MR-HMD users during trials, it is found that they rarely experienced symptoms of nausea and dizziness. However, due to the weight and design of the Hololens 2 helmet, users can become fatigued and uncomfortable after extended usage exceeding 30 min. To mitigate these factors, the system takes into account human behavior and carefully plans tasks to minimize fatigue and temporary emotional disturbances [31]. Furthermore, it is important to note that the research objective of this work is primarily focused on demonstrating the effectiveness of combining DRL with MR features in HRC safe motion planning scenarios, rather than conducting a direct performance comparison at this stage. Therefore, to achieve improved performance, advanced network architectures will naturally be incorporated to address the proposed motion planning problem formulation.

7. Conclusion

In this work, an MR-augmented DRL-based robot approach is proposed to guide robots to achieve adaptive and safe motion generation towards potential safe collision hazards in human–robot collaborative manufacturing activities to maximize the utility of MR. The approach first utilizes MR-HMD devices and expert knowledge to collect user input regarding tasks and safety requirements for applying DRL algorithm settings. Second, supported by MR-HMD spatial and temporal features, a safe robot motion generation policy is carried out supported by DRL-based representation learning. Finally, the implementation of safe deployment of DRL control policies with the assistance of MR devices is discussed. In the evaluation, the DRL agents are deployed in various HRC motion planning tasks under different settings with various MR additions. The results show that the sample efficiency and performance of the MR-augmented DRL algorithm could outperform the DRL algorithms under conventional settings.

Despite these, in future work on HRC scene, the high granularity of human modeling and state representation techniques will be explored first, it could better support the close proximity and human ergonomics to optimize the robot motion planning and operator's interaction experience. Moreover, in the implementation of DRLs, the negative reward used in this work usually imposes a soft penalty, which does not guarantee safety in the generated behaviors. To enhance the safety from the theoretical way of DRL, hard constraints are planned to be introduced instead of relying solely on soft penalties, which aims to enforce stricter safety criteria and improve the overall safety assurance of the system [20,32,33]. Last but not least, regarding the robot control, the physical information (e.g. force, torque)-based compliance control and impedance control will be explored to support the interaction between humans and robots. These control mechanisms will facilitate the harmonization between humans and robots, ensuring safe human–robot collaboration [34].

CRediT authorship contribution statement

Chengxi Li: Conceptualization, Data curation, Methodology, Validation, Visualization, Writing – original draft. **Pai Zheng:** Funding acquisition, Project administration, Supervision, Writing – review & editing, Conceptualization, Resources. **Peng Zhou:** Formal analysis, Methodology, Validation. **Yue Yin:** Conceptualization, Methodology, Visualization. **Carman K.M. Lee:** Funding acquisition, Project administration. **Lihui Wang:** Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research work was partially supported by the grants from the General Research Fund (GRF) from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU15210222 and PolyU15206723), the Laboratory for Artificial Intelligence in Design (Project Code: RP2-1), Innovation and Technology Fund, Hong Kong Special Administrative Region, and the Research Committee of The Hong Kong Polytechnic University under Collaborative Departmental General Research Fund (G-UAMS), Hong Kong Special Administrative Region.

References

[1] Xu Xun, et al. Industry 4.0 and industry 5.0—Inception, conception and perception. J Manuf Syst 2021;61:530–5.

[2] Wang Lihui, et al. Symbiotic human-robot collaborative assembly. CIRP Ann 2019;68(2):701–26.

[3] Zheng Pai, et al. A collaborative intelligence-based approach for handling human-robot collaboration uncertainties. CIRP Ann 2023.

[4] Zhou Peng, et al. Neural reactive path planning with Riemannian motion policies for robotic silicone sealing. Robot Comput-Integr Manuf 2023;81:102518.

[5] Zhu Cheng, Yu Tian, Chang Qing. Task-oriented safety field for robot control in human-robot collaborative assembly based on residual learning. Expert Syst Appl 2024;238:121946.

[6] El-Shamouty Mohamed, et al. Towards safe human-robot collaboration using deep reinforcement learning. In: 2020 IEEE international conference on robotics and automation. ICRA, IEEE; 2020, p. 4899–905.

[7] Thumm Jakob, Althoff Matthias. Provably safe deep reinforcement learning for robotic manipulation in human environments. In: 2022 international conference on robotics and automation. ICRA, IEEE; 2022, p. 6344–50.

[8] Yu Tian, Chang Qing. Motion planning for human-robot collaboration based on reinforcement learning. In: 2022 IEEE 18th international conference on automation science and engineering. CASE, IEEE; 2022, p. 1866–71.

[9] Yin Yue, et al. A state-of-the-art survey on augmented reality-assisted digital twin for futuristic human-centric industry transformation. Robot Comput-Integr Manuf 2023;81:102515.

[10] Yang Wenhao, Xiao Qinqin, Zhang Yunbo. 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, V002T07A003.

[11] Yang Wenhao, Xiao Qinqin, Zhang Yunbo. HA R 2 bot: A human-centered augmented reality robot programming method with the awareness of cognitive load. J Intell Manuf 2023;1–19.

[12] Li Chengxi, et al. An AR-assisted deep reinforcement learning-based approach towards mutual-cognitive safe human-robot interaction. Robot Comput-Integr Manuf 2023;80:102471.

[13] Malik Ali Ahmad, Masood Tariq, Bilberg Arne. Virtual reality in manufacturing: Immersive and collaborative artificial-reality in design of human-robot workspace. Int J Comput Integr Manuf 2020;33(1):22–37.

[14] Choi Sung Ho, et al. An integrated mixed reality system for safety-aware human-robot collaboration using deep learning and digital twin generation. Robot Comput-Integr Manuf 2022;73:102258.

[15] Chadalavada Ravi Teja, et al. Bi-directional navigation intent communication using spatial augmented reality and eye-tracking glasses for improved safety in human–robot interaction. Robot Comput-Integr Manuf 2020;61:101830.

[16] Aivaliotis Sotiris, et al. An augmented reality software suite enabling seamless human robot interaction. Int J Comput Integr Manuf 2023;36(1):3–29.

[17] Hietanen Antti, et al. AR-based interaction for human-robot collaborative manufacturing. Robot Comput-Integr Manuf 2020;63:101891.

[18] Khatib Maram, Al Khudir Khaled, De Luca Alessandro. Human-robot contactless collaboration with mixed reality interface. Robot Comput-Integr Manuf 2021;67:102030.

[19] Li Chengxi, et al. Deep reinforcement learning in smart manufacturing: A review and prospects. CIRP J Manuf Sci Technol 2023;40:75–101.

[20] Pham Tu-Hoa, De Magistris Giovanni, Tachibana Ryuki. Optlayer-practical constrained optimization for deep reinforcement learning in the real world. In: 2018 IEEE international conference on robotics and automation. ICRA, IEEE; 2018, p. 6236–43.

[21] Krasowski Hanna, Wang Xiao, Althoff Matthias. Safe reinforcement learning for autonomous lane changing using set-based prediction. In: 2020 IEEE 23rd international conference on intelligent transportation systems. ITSC, IEEE; 2020, p. 1–7.

[22] Schepp Sven R, et al. Sara: A tool for safe human-robot coexistence and collaboration through reachability analysis. In: 2022 international conference on robotics and automation. ICRA, IEEE; 2022, p. 4312–7.

[23] Liu Quan, et al. Deep reinforcement learning-based safe interaction for industrial human-robot collaboration using intrinsic reward function. Adv Eng Inform 49:101360.

[24] Chen Lienhung, et al. Deep reinforcement learning based trajectory planning under uncertain constraints. Front Neurorobot 2022;16:883562.

[25] Sutton Richard S, Barto Andrew G. Reinforcement learning: An introduction. MIT Press; 2018.

[26] Li Chengxi, et al. AR-assisted digital twin-enabled robot collaborative manufacturing system with human-in-the-loop. Robot Comput-Integr Manuf 2022;76:102321.

[27] Haarnoja Tuomas, et al. Soft actor-critic algorithms and applications. 2018, arXiv preprint arXiv:1812.05905.

[28] Mnih Volodymyr, et al. Asynchronous methods for deep reinforcement learning. In: International conference on machine learning. PMLR; 2016, p. 1928–37.

[29] Schulman John, et al. Proximal policy optimization algorithms. 2017, arXiv preprint arXiv:1707.06347.

[30] Mnih Volodymyr, et al. Human-level control through deep reinforcement learning. Nature 2015;518(7540):529–33.

[31] Wang Baicun, et al. Human digital twin in the context of industry 5.0. Robot Comput-Integr Manuf 2024;85:102626.

[32] Yu Dongjie, et al. Reachability constrained reinforcement learning. In: International conference on machine learning. PMLR; 2022, p. 25636–55.

[33] Hsu Kai-Chieh, et al. Sim-to-lab-to-real: Safe reinforcement learning with shielding and generalization guarantees (abstract reprint). In: Proceedings of the AAAI conference on artificial intelligence, vol. 38, (no. 20):2024, p. 22699.

[34] Li Weidong, et al. Safe human–robot collaboration for industrial settings: A survey. J Intell Manuf 2023;1–27.