Towards safe motion planning for industrial human-robot interaction: A co-evolution approach based on human digital twin and mixed reality

Bohan Feng , Zeqing Wang , Lianjie Yuan , Qi Zhou , Yulin Chen , Youyi Bi [*]

University of Michigan – Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai, China

ARTICLE INFO

ABSTRACT

Advanced human-robot interaction (HRI) is essential for the next-generation human-centric manufacturing mode such as "Industry 5.0". Despite recent mutual cognitive approaches can enhance the understanding and collaboration between humans and robots, these methods often rely on predefined rules and are limited in adapting to new tasks or changes of the working environment. These limitations can hinder the popularization of collaborative robots in dynamic manufacturing environments, where tasks can be highly variable, and unforeseen operational changes frequently occur. To address these challenges, we propose a co-evolution approach for the safe motion planning of industrial human-robot interaction. The core idea is to promote the evolution of human worker's safe operation cognition as well as the evolution of robot's safe motion planning strategy in a unified and continuous framework by leveraging human digital twin (HDT) and mixed reality (MR) technologies. Specifically, HDT captures real-time human behaviors and postures, which enables robots to adapt dynamically to the changes of human behavior and environment. HDT also refines deep reinforcement learning (DRL)-based motion planning, allowing robots to continuously learn from human actions and update their motion strategies. On the other hand, MR superimposes rich information regarding the tasks and robot in the physical world, helping human workers better understand and adapt to robot's actions. MR also provides intuitive gesture-based user interface, further improving the smoothness of human-robot interaction. We validate the proposed approach's effectiveness with evaluations in realistic manufacturing scenarios, demonstrating its potential to advance HRI practice in the context of smart manufacturing.

1. Introduction

Human-Robot Interaction (HRI) is becoming increasingly important to manufacturing as human-centric is a pivotal characteristic of next-generation manufacturing paradigm such as Industry 5.0 [1]. Unlike Industry 4.0, which emphasizes automation and digitalization, Industry 5.0 extends these advancements by prioritizing human-machine collaboration, well-being, and sustainability [2,3]. Advanced HRI combines robots' precision and robustness with human dexterity, enhancing collaborative efforts and productivity in manufacturing settings. Although the International Organization for Standardization (ISO) has established protocols for HRI safety (e.g., ISO 10,218–2, ISO/TS 15, 066, etc.), including working distance and force regulations, these strategies fall short in dynamic manufacturing environments [4]. For example, a flexible assembly line needs to handle products with varying specifications, and each assembly process may differ. Emergent issues such as mis-installed components, tool failures, or program errors

require rapid intervention of human experts.

Given these complexities, current research on HRI safety strategies focuses on integrating advanced vision, artificial intelligence and sensing technologies to enhance the safety and efficiency of HRI [5]. Particularly, researchers proposed the concept of mutual cognition to enhance the shared understanding and collaboration between humans and robots through the system-level integration of various technologies [1,6,7]. These existing studies often rely on predefined rules to guide human-robot interactions. However, this reliance is limited in dynamic and complex environments. Once the interaction rules are set, they often hardly update according to new situations or environmental changes. This limitation is dangerous when unforeseen issues or changes occur in practical scenarios such as changes in worker operating habits or equipment positions. If the system cannot quickly adjust and adapt, the human's safety in interaction is highly risky, not to mention the decreased interaction efficiency.

To address these issues, we propose a co-evolution approach for the

---

safe motion planning of industrial human-robot interaction. Inspired by the co-evolution concept from biology [8], we expect robots can actively learn from human behaviors to optimize their operations and get continuously evolved safety planning strategies. Concurrently, humans can gain a deeper understanding of robot's working patterns and behaviors through interaction, leading to persistently evolved cognition on the safe operations in the context of industrial human-robot interaction. This bidirectional and collaborative learning strategy allows both humans and robots to adapt together to dynamically changing environments, thus improving the efficiency and safety of human-robot interaction.

The key mechanisms of our co-evolution approach, including information sharing and understanding, real-time perception and modeling, and dynamic adaptation and learning, are supported by human digital twin (HDT) and mixed reality (MR) technologies. Specifically, HDT captures and simulates human actions and postures in real time, providing detailed information about human operators. This allows robots to better understand human behavior and adjust their actions correspondingly to avoid collisions or other hazardous situations. Additionally, HDT provides rich data support for learning-based motion planning algorithm, enhancing the system's adaptability and robustness. Meanwhile, MR allows humans to intuitively view robot status and acknowledge its behavior, helping them better understand and anticipate robot actions, thereby reducing misunderstandings and uncertainties in human-robot interaction. The MR system also provides operational guidance, assisting humans in performing complex tasks or avoiding potential hazards, thereby enhancing their control over and trust in the robot system. The main contributions of this study include:

1. **A co-evolution approach for safe motion planning in dynamic industrial human-robot interaction environment is proposed.** Our approach can facilitate the evolution of human worker's safety operation cognition as well as the evolution of robot's safety planning strategy in a unified and continuous framework, enhancing the adaptability of HRI for complex and dynamic industrial environments.
2. **A vision-based HDT capturing and modeling method is developed.** This method effectively captures and models human behavior with limited computing resources, providing precise human reference data for human-robot interaction.
3. **An HDT-enabled deep reinforcement learning-based motion planning method is designed.** This method supports robots to dynamically adjust their motion paths and adapt to various task scenarios, continuously optimizing algorithmic performance through learning from past interaction experiences, thereby enhancing their ability to handle complex industrial environments.

The rest of the paper is structured as follows: Section 2 reviews related work in HRI and the applications of human digital twin and extended reality in HRI. Section 3 introduces the overall architecture of our approach and explains the key techniques. Section 4 validates the performance of the proposed approach with a set of simulation and physical experiments. Section 5 summarizes the findings of our study and highlights potential directions for future research.

2. Related works

2.1. Safe human-robot interaction

In HRI, safety-oriented robot control strategies mainly include two categories: passive and active. For passive robot control, ISO has issued two technical specifications, ISO 10,218–2 and ISO/TS 15,066 [4]. The former specifies the limits on the power and force of robots, addressing dangerous situations when collaborative robots directly contact with human operators. The latter defines standards for the motion speed and status monitoring of robots to eliminate potential contact hazards.

In contrast, active robot control strategies focus on real-time adaptation and planning [9]. The key of safe HRI in active control is adaptive motion planning, which allows robots to dynamically adjust their movements in real-time based on environmental changes, obstacles, and task requirements, ensuring robust and reliable robotic operations [10]. For instance, Shaoul et al. [11] proposed a method for accelerating multi-robot manipulation planning by reusing online-generated experiences to improve heuristic search efficiency and quality. Feng et al. [12] introduced a multi-RRT method combined with information gain analysis to enhance exploration quality and adaptability in highly constrained settings. Wang et al. [13] presented a real-time motion generation method using a multi-objective optimization framework with relaxed barrier functions to improve the computational efficiency.

Besides the search, sampling, and optimization-based methods mentioned above, learning-based methods have attracted considerable attention recently, emerging as a class of novel and versatile motion planning solutions for robotic systems. Characterized by their high adaptability and real-time decision-making ability, these methods harness the power of machine learning, particularly imitation learning (IL) and deep reinforcement learning (DRL), to model and solve high-dimensional, non-linear planning problems that are computationally intractable for traditional algorithms [14].

One typical IL-based model is DeepMotion [15], a human-aware navigation model that combines convolutional neural networks and long short-term memory layers to process laser data for safe navigation. Wang et al. [16] proposed a novel imitation learning framework for coordinated human-robot collaboration derived from hidden state-space models. Zhang et al. [17] introduced a human-robot collaboration method based on a human skill imitation model, which utilizes electromyographic signals and joint angles to directly translate human arm forces into robot speed adjustments, enabling the robot to fulfill human motion intentions. However, IL depends heavily on high-quality demonstration data. Errors, biases, or inadequacies in the training data can impair the model's performance and limit its adaptability to unfamiliar environments.

Compared to IL, DRL demonstrates superior capabilities in generalizing from limited or sparse data and adapting to new environments. DRL agents achieve this by interacting directly with environment and receiving feedback in the form of rewards or penalties. This process enables them to develop robust strategies capable of handling a wide variety of scenarios. For example, Sangiovanni et al. [18] proposed a model-free DRL method based on the normalized advantage function to complete tasks. However, this method only considered collisions between human and robot end-effector, neglecting human's potential contact with the entire robotic arm. Liu et al. [19] designed a reward function-based obstacle avoidance solution using DRL. They introduced a reward function optimization method, combining external and internal reward functions, allowing the robot to dynamically avoid human arm. Mohamed et al. [20] proposed a safe HRI method using DRL to enhance the intelligence and safety of interaction, providing a systematic approach for encoding safety requirements and applicability contexts in RL settings. Li et.al. [21] proposed a reinforcement learning algorithm for human-robot collaboration, and it only modeled the human body as two cylinders, failing to fully capture the complexity of human motion.

DRL can effectively enhance a robot's adaptability and flexibility, enabling robots to perform collision-avoidance motion planning efficiently. However, significant challenges remain, including the gap between simulation and reality, and the complexities of extracting accurate environmental information in real-world settings. These challenges hinder the practical deployment of DRL-based motion planning systems. In this study, we expect to utilize human digital twin to train and deploy reinforcement learning algorithms. By continuously collecting real-time human motion data, adaptive motion planning algorithms can better learn and evolve, thereby enhancing the performance of DRL in dynamic human-robot interaction scenarios.

2.2. Human digital twin in human-robot interaction

Human digital twin (HDT) is a multidimensional representation of human, facilitating bidirectional interaction between the physical and digital worlds. In addition to human individuality and characteristics, HDT also considers human interactions with other objects, such as machines and environments, to provide a comprehensive depiction of a person. This representation may include various models related to interactive attributes, such as motion and perception models. For example, external sensors can capture human operational commands to recognize human actions and instructions. Simao et al. [22] used wearable sensors to identify static and dynamic human gestures. Robots receive commands from these human gestures, such as stopping movement, rotating the robot end-effector, and opening/closing grippers. Lanini et al. [23] estimated upper body movements from force sensor signals as commands to robots. He et al. [24] estimated human lower limb motion intentions using surface electromyography signals and developed a coupled dynamics model for exoskeleton robot control. To ensure human safety, Buerkle et al. [25] used mobile electroencephalogram sensors to detect potential emergencies in industrial tasks, such as dropped workpieces, crushed workpieces on the workbench, and operation malfunctions. Awareness of potential emergencies enables robots to quickly take actions to prevent harmful consequences for humans. Mohammad et al. [26] studied the application of HRI in dynamic environments with moving obstacles/humans using lidars and inertial measurement units. Humans and robots were represented by capsules, allowing for real-time calculation of the minimum safe human-robot distance.

Although wearable sensors have been widely recognized for their accuracy in posture capture, they pose several challenges in practical industrial applications. Firstly, wearing these sensors can be uncomfortable for workers and cause fatigue, especially during long hours of work. Secondly, deploying these sensors incurs high costs, and limited noise immunity and data processing speed. Moreover, in actual manufacturing workshops, wearing sensors may affect the operational flexibility of workers and increase the complexity of maintenance and management. To overcome these issues, another research trend is to utilize computer vision technology to non-invasively analyze human posture and actions through RGB or depth cameras. Classic methods include the work of Liu et al. [27], in which RGB-D sensors and deep learning models are utilized to accurately capture human skeletal posture and spatial occupancy, thereby achieving real-time collision avoidance in HRI systems. Parsa et al. [28] proposed a spatiotemporal convolutional neural network for recognizing human behaviors and associated ergonomic risks from RGB-D video streams. Fan et al. [29] proposed a method for constructing digital humans, which while capable of recognizing various human actions, can be further improved to meet real-time modeling requirements in complex human-robot interaction scenarios. Yi et al. [30] reported a vision-based digital twin system for human-robot collaborative assembly primarily focusing on human skeletal recognition, but did not achieve accurate 3D human pose estimation.

In summary, existing computer vision methods often struggle to ensure real-time and accurate human modeling under limited computational resources. Additionally, they fail to effectively integrate human models with safe motion planning. To overcome these limitations, we expect to enhance human state perception to ensure real-time and effective human recognition in dynamic, resource-constrained scenarios. Additionally, we plan to advance the training of motion planning algorithms with HDT, thereby improving the overall effectiveness and safety of HRI.

2.3. Extended reality in human-robot interaction

Extended reality (XR) technology encompasses augmented reality (AR), virtual reality (VR), and mixed reality (MR), offering a suite of capabilities that integrate virtual and real-world experiences. AR displays virtual information in the real environment, enhancing user's perceptual dimension of reality. VR generates a completely virtual environment, isolating the user from the physical world. MR combines elements of both AR and VR, enabling human's interaction with both virtual and real-world objects.

In manufacturing, the immersive characteristics of VR can significantly reduce manual workload and enhance information processing efficiency. For instance, Galambos et al. [31] demonstrated high-fidelity virtual collaboration for the design, programming, and orchestration of heterogeneous manufacturing systems through VR-powered remote collaboration. Vogel et al. [32] introduced a projection- and camera-based method to visualize rectangular safety zones on the floor for humans during human-robot interaction.

With advancements in vision technology, current AR/MR devices possess improved computational capabilities and flexible deployment options, extending their use beyond mere visualization and monitoring [33]. For example, Hietanen et al. [34] developed a shared spatial awareness model to compute the robot's workspace, presenting it through AR glasses in the form of 2D projections and 3D models. Papanatasiou et al. [35] proposed an AR-based human-robot collaborative assembly system that utilizes visual and auditory signals to warn operators of potential risks. Siew et al. [36] introduced haptic support to enhance AR system interaction, combining visual, auditory, and tactile feedback to provide comprehensive maintenance information and instructions, thereby improving user efficiency and safety. Jost et al. [37] suggested an AR-based human-machine safety interaction method for factory logistics, enabling AR glasses to inform humans about the locations of nearby robots in automated warehouses. Beyond visualization and monitoring, AR can also serve as a perception and control device for hazard prediction and management. Li et al. [6] utilized the sensing and computational capabilities of AR head-mounted displays (HMDs) to assist in digital twin (DT)-based collision detection and path planning. Similarly, Li et al. [38] proposed a human-robot collaboration (HRC) safety control framework that leverages VR and DT simulations along with vision-based human-robot distance perception to ensure human safety during the design phase.

Due to its powerful virtual and real interaction capabilities, integrating MR technology into HRI has significant potential for enhancing the safety, efficiency, and user experience of interaction. For instance, Choi et al. [39] introduced a novel integrated MR system for safety-aware HRI, utilizing deep learning and DT. This method accurately measures safety distances through robot digital twins and 3D offset human skeletal models, providing real-time MR-based task assistance to human operators. Su et al. [40] proposed a mixed reality integrated 3D/2D vision and motion mapping method for immersive and intuitive remote operation of complex mobile manipulators. Khatib et al. [41] suggested a multi-sensor control system for safe, non-contact human-robot collaboration, coordinating constrained motion tasks through a mixed reality interface.

While prior research has made significant progress in the field of industrial human-robot interaction, it has generally not fully considered the enhancement effects of HDT models on MR systems within dynamic industrial environments. Existing studies mostly focus on single-direction improvements, such as real-time information display or robot motion planning optimization, but lack an approach that integrates human operator cognition with robot adaptability in a collaborative manner.

Against this backdrop, our study aims to leverage HDT models displayed through MR glasses to provide real-time operational information and historical trajectory data, enabling operators to better understand robot behaviors and predict potential risks. This approach not only facilitates timely adjustments in operator actions but also records historical interaction data, which serves as a foundation for further optimization of motion planning algorithms. Specifically, the historical data recorded through MR functionality can support the training of

motion planning algorithms at various stages, allowing for the progressive evolution of these algorithms to better adapt to complex and dynamic industrial environments.

3. Methodology

3.1. Overall architecture

Fig. 1 illustrates the overall architecture of the proposed approach, which primarily consists of modules of mixed reality (MR), object detection, human detection, motion planning, and digital twins. The MR module serves as an information bridge between human workers and robots by providing real-time visualization of critical information about robot status, task details, human presence, and safety warnings. Additionally, the MR module allows workers to issue commands through the MR interface.

The object detection module processes images captured by cameras to identify objects and determine their poses. The human detection module uses vision-based techniques to detect and estimate human skeletal information, identifying and tracking the presence of humans within the workspace. This data is crucial for planning safe human-robot collaborative movements.

The motion planning module integrates multi-source data from HDT and MR, utilizes reinforcement learning (RL) and geometry-based planning methods to compute safe paths for the robot, and ensures obstacle avoidance and safe interaction with humans. It considers both static and dynamic elements in the environment to generate efficient, collision-free trajectories and makes real-time adjustments based on continuous feedback from the MR and detection modules. The incremental learning mechanism continuously updates and optimizes the motion planning algorithm's performance.

A virtual digital twin of the physical robot and environment is created, allowing for real-time simulation and monitoring. The virtual layer is continuously updated with external data to ensure high fidelity and accuracy. Communication between the virtual and physical layers is achieved through TCP protocol sockets, enabling the mapping of workspace information from the physical layer to the virtual layer and the real-time transmission of motion commands from the virtual layer to the physical robot.

The core of co-evolution lies in the bidirectional flow of information within the architecture, enabling humans and robots to continuously learn from and adapt to each other. For human workers, real-time visualization and instructions from various modules enhance their understanding of robot behavior, allowing them to predict robot actions more accurately. This dynamic information exchange and continuous feedback improve operational awareness, reducing misunderstandings and potential accidents. As a result, workers can perform tasks more safely and efficiently, adjusting their actions swiftly in response to robot behavior, thereby evolving their cognitive strategies for safe operations. For robots, human behavior data obtained from the HDT module and processed within the digital twin, along with real-time instructions from the MR system, allow the robot to better understand human actions and adapt to the worker's movements. The motion planning module not only supports dynamic adjustments to the robot's movement paths but, through an incremental learning mechanism, also enables continuous policy updates. This adaptive capability allows the robot to refine its motion planning strategies over time, ensuring dynamic avoidance and smoother interaction with humans in complex environments. In a word, the co-evolution of human and robot behaviors continually enhances both the robot and human's decision-making abilities and action strategies, which drives the human-robot interaction system to operate safely and intelligently.[1]

---

[1] For the code files of our system, please visit: https://github.com/bohan feng/Towards-Safe-Motion-Planning-for-Industrial-Human-Robot-Interaction.

The following sections will delve into the key techniques involved in our approach. Section 3.2 focuses on the construction of the digital twin, detailing the detection of environmental and human information. Section 3.3 discusses the principles of the DRL-based robot motion planning algorithm, including its incremental learning feature. Section 3.4 introduces the design and use of the MR module.

3.2. Digital twin construction

Digital twin (DT) construction involves the creation of a virtual replica of physical entities, allowing real-time monitoring, simulation, and optimization of interactions between robots and humans. Two key techniques in DT construction are object detection and human detection. The former ensures real-time tracking of objects, and the latter provides continuous updates of human skeletal data, both essential for maintaining synchronization between physical and digital environments. With accurate representation of objects and humans, DT enables precise and adaptive robot motion planning in dynamic HRI scenarios. Section 3.2.1 and Section 3.2.2 detail the methodologies of object detection and human detection, respectively.

3.2.1. Object detection

The initial step in robot motion planning within dynamic environments is the real-time acquisition of positions for both obstacles and operational objects. In this study, we developed an object detection module leveraging OVE6D [42], an algorithm that estimates six-dimension information of object position and orientation. OVE6D exhibits outstanding generalizability in object detection tasks, capable of managing unseen objects without the need for a fine-tuning process. Moreover, the algorithm's rapid response speed is particularly beneficial for real-time detection in dynamic settings. Fig. 2 depicts the workflow of the object detection module, which comprises four stages. Initially, a 3D model of the target object undergoes preprocessing to generate codebooks, which are encoded with uniformly distributed views of the model and serve as references for subsequent pose estimation. Next, the environmental images captured by an RGB-D camera, along with their corresponding codebooks, are then processed through the OVE6D algorithm to provide pose predictions for each object. Finally, the resulting pose outputs and their respective object identifiers, are transmitted to Unity to update the digital model. The last three stages are iteratively executed to continuously track changes of the environment in real time.

3.2.2. Human detection

In dynamic industrial environments, effective human-robot interaction relies on real-time visual processing models that accurately capture human skeletal information. However, achieving fast and accurate 3D Human Pose Estimation (HPE) with limited computational resources remains a significant challenge. Recent advancements in transformer-based architectures have shown favorable performance in video-based HPE [43–46]. These video pose transformers (VPT) treat each video frame as a pose token and use transformers to capture spatial and temporal information from long video sequences. However, these methods are computationally demanding, as the complexity of self-attention in VPT grows quadratically with the number of tokens (i. e., frames), making their deployment on devices with limited computational resources challenging. Although some researchers have suggested pruning pose tokens from video frames [47], these methods still require predefined parameters (e.g., human action classifications), limiting their application in diverse human-robot interaction scenarios.

To address these challenges, we propose an Adaptive Clustering-based 3D HPE (AC-HPE) method to achieve efficient and accurate human 3D pose estimation under constrained computational resources. The workflow of the AC-HPE method is illustrated in Fig. 3, which begins by embedding spatial and temporal information from 2D pose sequences, followed by a domain adaptive density clustering algorithm
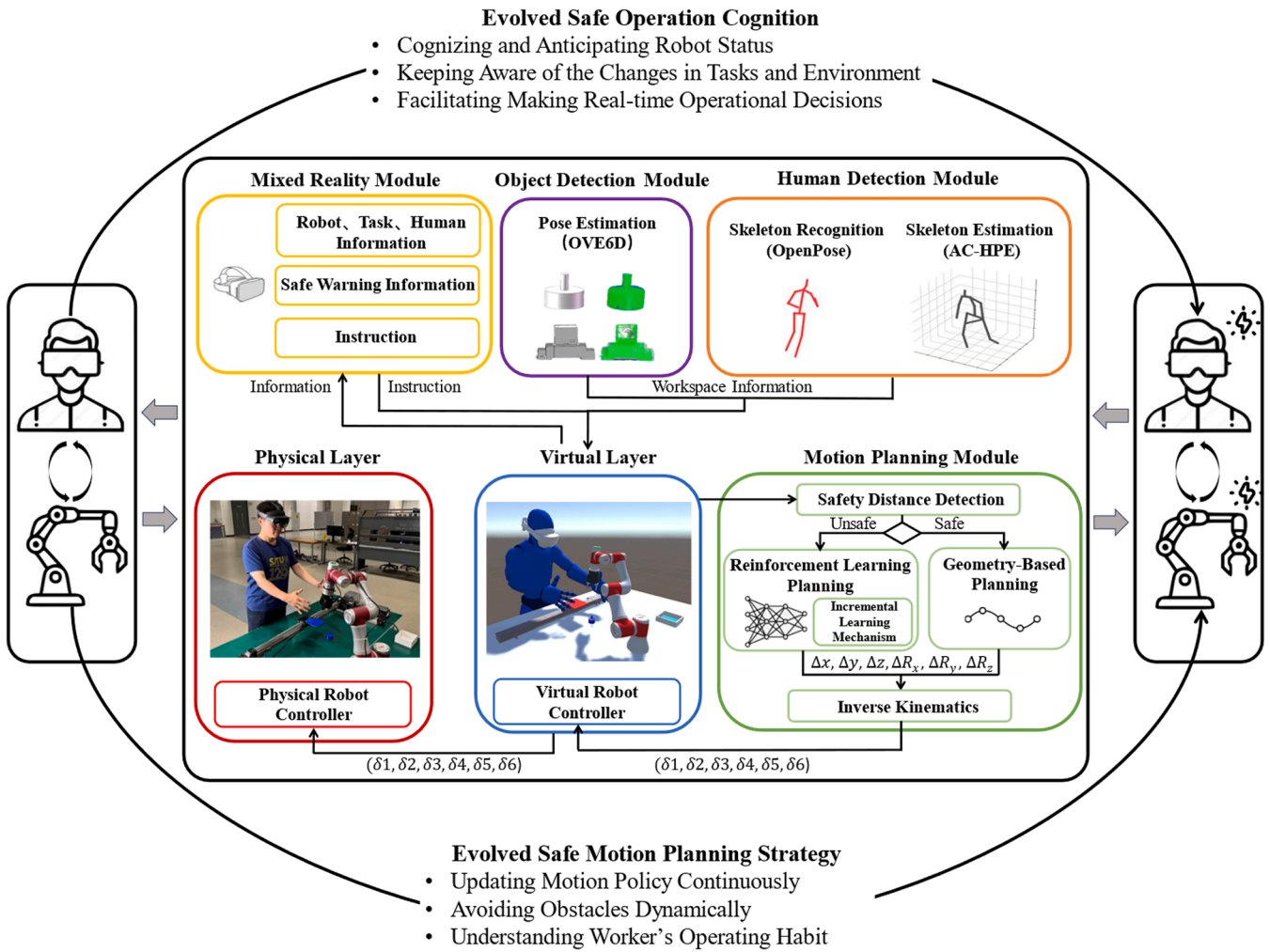
Evolved Safe Operation Cognition
- Cognizing and Anticipating Robot Status
- Keeping Aware of the Changes in Tasks and Environment
- Facilitating Making Real-time Operational Decisions

Fig. 1. Overall architecture of the co-evolution approach for safe motion planning of industrial human-robot interaction based on human digital twin and mixed reality.
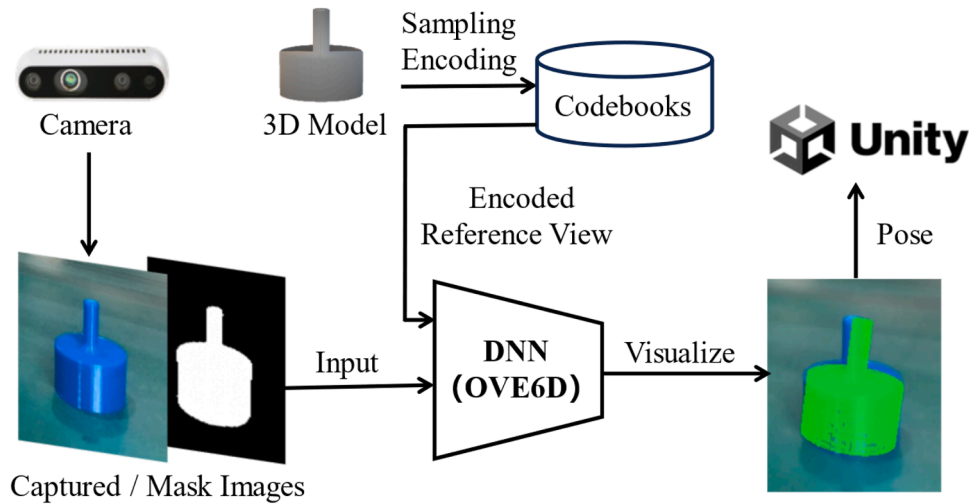

Fig. 2. Workflow of the object detection module based on OVE6D algorithm.

(DADC) that selects keyframes to reduce computational redundancy. These representative tokens are then processed through a transformer-based architecture for 3D pose estimation, culminating in the synchronization of real-time skeletal data with a digital twin model. The detailed processes are explained as follows.

Given the input 2D pose sequence $p \in \mathbb{R}^{F \times J \times 2}$ detected using a pretrained 2D human skeleton detector (OpenPose [48]), we first input them into the pose embedding module to embed the spatial and
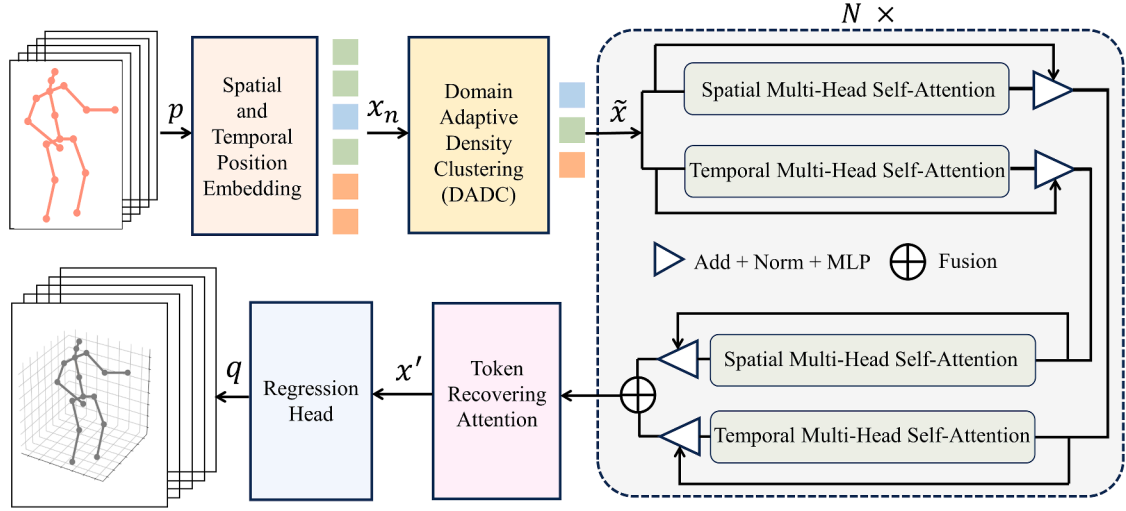
Fig. 3. Workflow of the adaptive clustering-based 3D human pose estimation.

temporal information of the pose frames, resulting in tokens $x_n \in \mathbb{R}^{F \times J \times C}$. Here, F, J, and C represent the number of input frames, body joints, and the feature dimension.

Then, we propose a domain adaptive density clustering algorithm (DADC) for keyframe selection. This algorithm extracts key frames from temporal data to reduce redundant information, decreasing computational load while enhancing responsiveness to dynamic changes. DADC processes the full-length pose tokens $x_n \in \mathbb{R}^{F \times J \times C}$ from the n-th transformer block and outputs a subset of representative tokens $\tilde{x} \in \mathbb{R}^{f \times J \times C}$ (where $f \ll F$). Here, f denotes the representative tokens, respectively. DADC computes the similarity matrix $S \in \mathbb{R}^{F \times F}$ for all pose tokens. The similarity is computed using the Euclidean distance:

$$S_{ij} = \|x_i - x_j\|^2 = \sum_{c=1}^{C} \left(x_i(c) - x_j(c) \right)^2, \tag{1}$$

where x_i and x_j are the input pose tokens. To capture local density information, we determine the K-nearest neighbors (KNN) and compute the KNN distance:

$$KDist_i = (1/K) \cdot \sum_{j \in N(x_i)} d_{ij}, \tag{2}$$

where K denotes the number of neighbors for each data point. The KNN density, reflecting the local neighborhood's density distribution, is then calculated as follows:

$$KDen_i = K \Bigg/ \sum_{j \in N(x_i)} d_{ij}. \tag{3}$$

Next, we compute the domain density to incorporate both the local and neighboring densities:

$$\partial_i = KDen_i + \sum_{j \in N(x_i)} \left(KDen_j \cdot (1/d_{ij}) \right) \tag{4}$$

To identify high-density points, we determine the maximum domain density:

$$\partial_{max} = max(\partial). \tag{5}$$

To distinguish significant high-density points and ensure that cluster centers are representative, we introduce a Delta distance calculated as follows:

$$\delta_i = \begin{cases} max(d_{ij}) & \text{if } \partial_i = \partial_{max} \\ min_{j:\partial_j > \partial_i}(d_{ij}) & \text{otherwise} \end{cases} \tag{6}$$

The domain adaptive density, which captures both local and global data distribution, is then defined by:

$$\partial_i = \partial_i \cdot \delta_i. \tag{7}$$

To identify cluster centers, the critical point is computed as:

$$C_p(x, y) = (\partial_{max}/2, \delta_{max}/4) \tag{8}$$

To identify cluster centers, we compute the critical point:

$$\Lambda_{center} = x_i \big| \partial_i > C_p(x) \text{ and } \delta_i > C_p(y) \tag{9}$$

$$\Lambda_{outlier} = \left\{ x_i \big| \partial_i < C_p(x) \text{ and } \delta_i > C_p(y) \cdot \partial_i / C_p(x) \right\} \tag{10}$$

$$\Lambda_{remaining} = \left\{ x_i \big| x_i \notin \Lambda_{center} \text{ and } x_i \notin \Lambda_{outlier} \right\} \tag{11}$$

By establishing these critical points and classification criteria, we ensure that the generated clusters exhibit high intra-cluster point densities and large inter-cluster distances, thereby enhancing the representativeness and accuracy of the clustering process. The remaining data points are subsequently assigned to the nearest cluster center, guaranteeing the completeness of the clustering.

These cluster centers exhibit high semantic diversity, containing more informative data than other tokens, and serve as the representative tokens $\tilde{x} \in \mathbb{R}^{f \times J \times C}$ for efficient estimation. To leverage these representative tokens effectively, we input $\tilde{x}$ into a state-of-the-art transformer-based 3D HPE architecture. Specifically, we employ MotionBert [46], a sequence-to-sequence model known for its efficacy in 3D HPE. The architecture includes dual-stream spatiotemporal multi-head self-attention blocks, with DADC pruning tokens before these blocks and restoring the full token sequence afterward. After the token recovering attention, we recover the original temporal resolution, producing the recovered tokens $x' \in \mathbb{R}^{F \times J \times C}$. A regression head is then added to estimate the 3D pose sequence $q \in \mathbb{R}^{F \times J \times 3}$. Finally, by binding skeletal information to the human model, the skeletal point data transmitted to Unity is used to update the digital twin model. This step is crucial as it synchronizes the real-time detected skeletal information with the virtual model, ensuring the real-time accuracy of human-robot interactions.

Overall, our method enables the model to adaptively refine its keyframe selection based on observed data to improve its computational efficiency and maintain its responsiveness to environmental changes. By reducing computational redundancy through keyframe selection, the method also infers occluded joint positions, compensating for brief occlusions by capturing both the temporal progression of poses and spatial relationships among joints. By integrating real-time skeletal data with the digital twin model, the system allows for continuous, robust pose

estimation in dynamic environments, and can continually improves its understanding of human movement, facilitating more accurate and efficient human-robot interactions.

3.3. Deep reinforcement learning-based safe robot motion planning

Fig. 4 illustrates the workflow of our motion planning method. Once the pose of objects and the real-time modeling and mapping information of the human body are constructed in the digital twin environment, the safety distance detection module monitors the distance between the robot and obstacles (d_{ro}) and the minimum distance between the robot and the human (d_{rh}). If these distances exceed the safety threshold, the robot is sufficiently distant from obstacles, thereby triggering a geometry-based motion planning algorithm. Otherwise, the reinforcement learning (RL) algorithm is employed to generate actions for the robot (Δx, Δy, Δz, ΔRx, ΔRy, ΔRz) to avoid collisions. The output pose differences allow for direct control adjustments in task space, keeping the system efficient and in line with real-time control needs. This adaptive approach enhances the flexibility and efficiency of motion planning, as the geometry-based method handles straightforward path planning when the robot is in a safe zone, while the RL-based method takes over near obstacles to ensure sophisticated collision avoidance. Additionally, the system is configured with a fixed control rate (e.g., 10Hz) and a buffer to stabilize pose transitions, thus reducing system jitter and enabling smooth operations within real-time constraints.

Subsequently, the inverse kinematics (IK) module is used to compute the joint angles required for the robot's end-effector to achieve the desired position and orientation. Specifically, we employ the Cyclic Coordinate Descent (CCD) algorithm [49] as the IK solver, which iteratively adjusts the robot's joints by prioritizing those with the smallest deviation from the target pose, gradually converging towards the desired configuration. The IK module calculates the joint angles based on the current and target poses of the robot arm, using cubic spline interpolation. The resulting joint angles are then transmitted to both the virtual and physical robots to execute the corresponding motions.

3.3.1. Safety distance detection

To accurately determine the minimum distance between the robot and obstacles (d_{ro}) as well as between the robot and the human model (d_{rh}) at each step, we implemented a vertex-based distance detection mechanism as illustrated in Fig. 5. This mechanism relies on the collision detection and coordinate transformation functions of the Unity3D engine. Based on our previous work [50], we first select key vertices prone to collision from the mesh models of the robot and obstacles (e.g., joint surfaces and end-effectors of the robot, and surfaces of obstacles). At each step of the RL training process, the spatial distances between these selected vertices on the robot and the obstacles are calculated. The
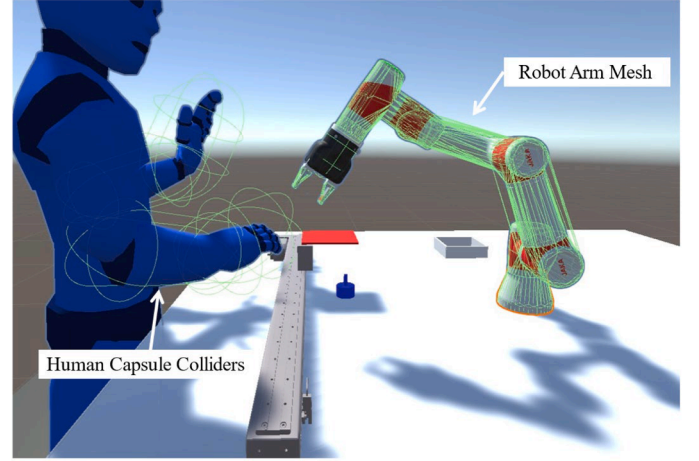

Fig. 5. Example of human capsule colliders and robot arm mesh model.

shortest distance between the robot and obstacles is then set as d_{ro}. Simultaneously, we add capsule colliders to the human model and calculate the distance between the centers of these colliders and the key vertices of the robot as the minimum distance, which is set as d_{rh}. Fig. 5 provides an illustrative example of the human capsule colliders and the robot arm mesh model.

Given that our study focuses on a desktop-level scenario, where human interaction primarily involves hands and arms, we prioritize collision avoidance for these specific configurations. This simplifies the problem by detecting human poses using vision-based methods. Such a scenario allows us to validate the feasibility and effectiveness of our framework while ensuring real-time performance.

Additionally, we introduce Gaussian noise (d_{noise}) to smooth out inevitable errors during the measurement process, mitigating the impact of distance measurement errors due to sparse vertices on the robot, human, and obstacles, thereby enhancing safety.

3.3.2. Reinforcement learning

If the distance between the robot and an obstacle falls below a predefined safety threshold, a RL algorithm is triggered to prevent a collision. At the heart of RL lies the concept of sequential decision-making, formalized as a Markov Decision Process (MDP). Within MDP, the decision-making entity is termed the agent, while the entity interacting with the agent is the environment. At each discrete time step, the agent observes the current state of the environment s_t and selects an action a_t based on a policy $\pi(a|s)$. Subsequently, the agent receives a reward R_{t+1} and transitions to a new state s_{t+1}.
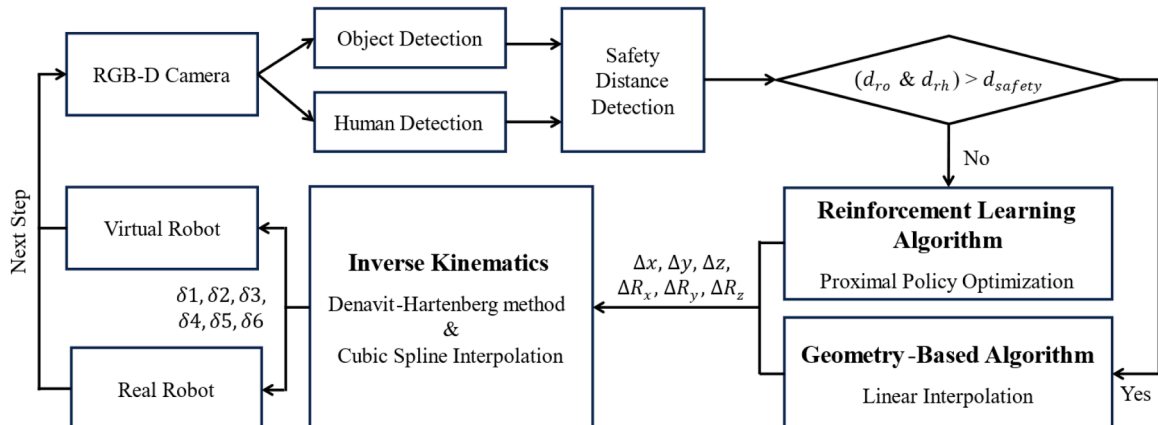

Fig. 4. Workflow of deep reinforcement learning-based planning.

In the domain of robotic motion planning, the robotic arm, acting as the agent, iteratively interacts with its environment to learn an optimal motion strategy that maximizes cumulative rewards, specifically aiming to reach the target position without collisions. Our approach leverages the Proximal Policy Optimization (PPO) algorithm [51] as the RL model, which offers notable advantages in terms of stability and sample efficiency in continuous action spaces and high-dimensional state spaces.

Proximal Policy Optimization:

PPO is a cutting-edge deep reinforcement learning method grounded in policy gradient techniques. Unlike other DRL methods such as Deep Deterministic Policy Gradient (DDPG) [52] and Soft Actor-Critic (SAC) [53], PPO enhances stability and sample efficiency by employing a novel objective function known as the clipped surrogate objective. This function aims to maximize the expected return of the new policy while constraining the divergence between the old and new policies. By avoiding excessively large or small policy updates, PPO achieves a more stable performance. The core principle of the PPO algorithm revolves around maximizing a constrained objective function, as delineated in the following equations:

The clipped objective function $L^{CLIP}(\theta)$ is central to the PPO algorithm:

$$L^{CLIP}(\theta) = \mathbb{E}_t[min(r_t(\theta) \cdot A_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \cdot A_t)]. \tag{12}$$

Here, $r_t(\theta)$ represents the probability ratio between the new and old policies, defined as $r_t(\theta) = (\pi_\theta(a_t|s_t))/(\pi_{\theta_{old}}(a_t|s_t))$. The term A_t is the advantage function, estimating the relative benefit of taking a_t in state s_t. The clipping operation $clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)$ restricts the probability ratio to the range $1 - \epsilon$ to $1 + \epsilon$, preventing excessive updates to the policy. This clipped objective aims to maximize the expected return of the new policy while ensuring the updates remain within a safe boundary, thereby enhancing training stability.

The total objective function $L^{PPO}(\theta)$ is defined as:

$$L^{PPO}(\theta) = \mathbb{E}_t\left[L^{CLIP}(\theta) - c_1 \cdot \mathbb{E}_t\left[\left(V_\theta(s_t) - V_t^{target}\right)^2\right] + c_2 \cdot \mathbb{E}_t[S[\pi_\theta](s_t)]\right]. \tag{13}$$

Here, $c_1 \cdot \mathbb{E}_t\left[\left(V_\theta(s_t) - V_t^{target}\right)^2\right]$ is the value function error term, representing the mean squared error between the value function $V_\theta(s_t)$ and the target value V_t^{target}. This term aims to minimize the value function's estimation error. $c_2 \cdot \mathbb{E}_t[S[\pi_\theta](s_t)]$ is the policy's entropy term. This term encourages exploration by preventing the policy from prematurely converging to suboptimal solutions.

Environment States and Actions:

The state parameters s_t of the robot arm agent are shown as follows:

$$s_t = \{P_g, P_t, P_o, P_p, d_{ro}, d_{rh}, d_{gt}, d_{gp}\}, \tag{14}$$

where

$P_g = \left(x_g, y_g, z_g, R_{x_g}, R_{y_g}, R_{z_g}\right)$	Position and orientation of the gripper
$P_t = (x_t, y_t, z_t, R_{x_t}, R_{y_t}, R_{z_t})$	Position and orientation of the target object center
$P_o = (x_o, y_o, z_o, R_{x_o}, R_{y_o}, R_{z_o})$	Position and orientation of the obstacle
$P_p = \left(x_p, y_p, z_p\right)$	Position of task area
d_{ro}	Minimum distance between the robot and obstacle
d_{rh}	Minimum distance between the robot and human
d_{gt}	Distance between the gripper and target object
d_{gp}	Distance between the gripper and task area

The action a is defined as a 7-dimensional vector as Eq. (15) shows:

$$a = \Delta x, \Delta y, \Delta z, \Delta R_x, \Delta R_y, \Delta R_z, \alpha \tag{15}$$

where $(\Delta x, \Delta y, \Delta z, \Delta R_x, \Delta R_y, \Delta R_z)$ are the translational and rotational displacements of gripper center in Cartesian space. α is a binary-value variable controlling the opening and closing of the gripper. An action step size factor β is introduced to control the amplitude of robot action,

where $(\Delta x, \Delta y, \Delta z) \in \beta \times [-1, 1]m$, $(\Delta R_x, \Delta R_y, \Delta R_z) \in \beta \times [-1, 1]rad$.

Design of Reward Functions:

To reduce the difficulty of motion planning, we divide it into three stages: (1) Approach, (2) Manipulate, and (3) Execute. The reward functions are designed as follows:

$$r(s) = R_a + R_o + R_p + R(s), \tag{16}$$

The components are defined as:

$$R_a = (\lambda_1 - d_{gt}) \cdot \mathbb{I}\{stage = (1) \vee stage = (2)\}, \tag{17}$$

where $\mathbb{I}$ is the indicator function that returns 1 if the condition is true and 0 otherwise. This term rewards the agent for reducing the distance to the target object during the approach and manipulate stages.

$$R_o = -\lambda_2 \cdot max(0, 1 - d_{ro}/d_{safety}), \tag{18}$$

where d_{safety} is a predefined safety threshold. These terms penalize the robot for proximity to the obstacle and the human, encouraging safe navigation.

$$R_p = (\lambda_3 - \lambda_4 \cdot d_{gp}) \cdot \mathbb{I}\{stage = 3\}. \tag{19}$$

This term incentivizes precise placement of objects during the final stage. In these formulas, $\lambda_{1 \to 4}$ are positive constants.

$$R(s) = \begin{cases} -1 \cdot \mathbb{I}\{collide\} \\ +1 \cdot \mathbb{I}\{approach\} \\ +5 \cdot \mathbb{I}\{manipulate\} \\ +10 \cdot \mathbb{I}\{execute\} \end{cases} \tag{20}$$

Specifically, $\mathbb{I}\{approach\}$ indicates whether the gripper has arrived at the manipulation area, which is defined as $d_{gt} < 3mm$. This term dynamically adjusts the reward based on the success of specific actions, thereby promoting the completion of the task.

Overall, these reward functions integrate both progressive rewards (R_a, R_o, R_p) and phase rewards $(R(s))$ to address issues of sparse rewards and local optima, which are common challenges in reinforcement learning model training. The progressive rewards provide dense feedback, offering gradient information that facilitates effective learning at each stage. In contrast, the phase rewards are sparse and are only applied when the agent reaches specific states, such as collision, arrival at the gripping area, successful gripping of the target object, and release of the target object in the placing area. This combination of dense and sparse rewards ensures a balanced learning process, helping the agent avoid local optima and achieve overall task success.

3.3.3. Incremental learning

In our approach, we also incorporate incremental learning to enhance the continuous evolution of the robot's motion planning algorithm. Incremental learning combines offline training with online adaptation, allowing the model to more accurately emulate worker behaviors during actual interactions, as illustrated in Fig. 6. Initially, the RL model undergoes offline training in a simulated environment where human movements are generated randomly. This stage provides a foundational understanding of various potential interaction scenarios, ensuring the model's robustness and its ability to handle unexpected behaviors.

During the offline training phase, the simulation environment generates diverse human movement patterns, including varied paths, speeds, and interactions with the environment. This diversity aids the RL model in generalizing its learning, thus preventing overfitting to specific movement patterns. Utilizing the RL module described in Section 3.3.2, the model develops a baseline policy $\pi_\theta(a|s)$ focused on avoiding collisions, accurately reaching target objects, and ensuring task completion.

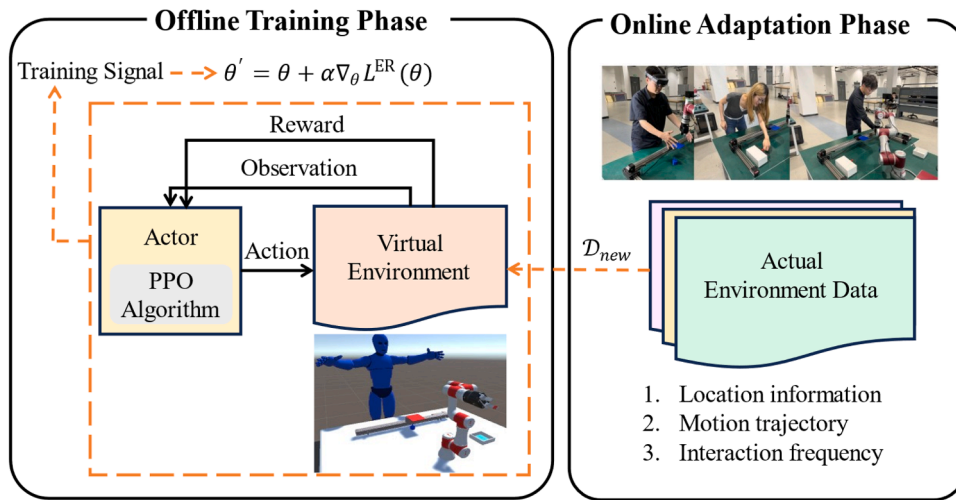Once deployed in a real-world setting, the model collects data on

Fig. 6. Incremental learning workflow for enhancing RL model performance.

actual human movements and interactions. This continuous data collection encompasses detailed positional information, movement trajectories, and interaction frequencies. The RL model then engages in incremental learning, adjusting its policy based on newly acquired data $\mathscr{D}_{new}$ while retaining previously learned policies. This ongoing adaptation phase ensures the model's policy θ closely aligns with the actual behaviors of the workers. The policy parameters θ' are updated incrementally as follows:

$$\theta' = \theta + \alpha \cdot \nabla_\theta L^{\mathrm{ER}}(\theta), \tag{21}$$

$$\nabla_\theta L^{\mathrm{ER}}(\theta) = \mathbb{E}_{(s,a,r,s') \sim \mathscr{B} \in \mathscr{D} \cup \mathscr{D}_{new}}[\nabla_\theta \log \pi_\theta(a|s) A_t], \tag{22}$$

where α is the learning rate and $\mathscr{B}$ is the experience replay buffer that has been newly updated with $\mathscr{D}_{new}$.

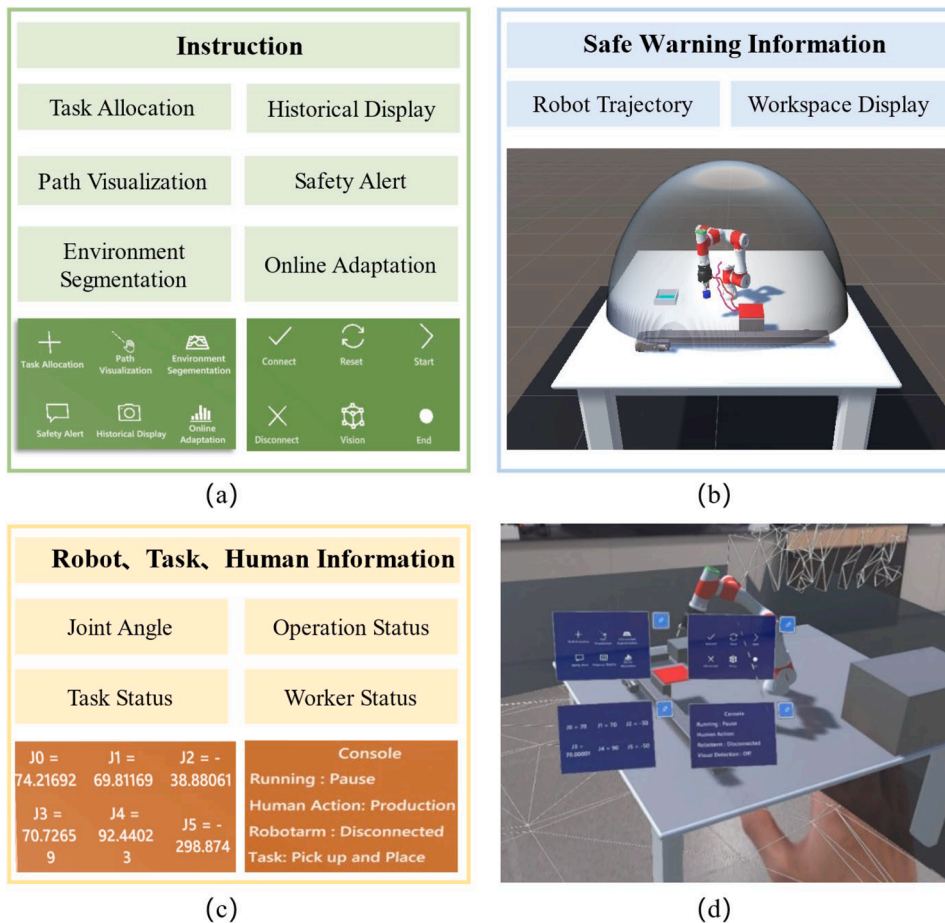During the online adaptation phase, the robot can consistently avoid

Fig. 7. The design of the MR system: (a) instruction interface; (b) safety warning information interface; (c) detailed operational status about robots, tasks, and workers; (d) overview of the MR system's user interface.

collisions and perform tasks efficiently. The incremental learning approach, characterized by continuous adaptation, results in a motion planning algorithm that evolves over time, improving the robot's safety and efficiency continuously. This evolutionary process enables the robot to better align its movements with the dynamic behaviors of human workers, leading to more seamless and productive human-robot interaction.

3.4. Mixed reality system

The mixed reality (MR) system is crucial for enhancing efficient HRI by providing an intuitive and informative user interface, as depicted in Fig. 7. The MR system's key functionalities include instruction, safety warning information, and detailed status on robots, tasks, and human workers. The system employs a gesture-based user interface, allowing operators to intuitively interact with the MR environment through natural hand movements.

Fig. 7 (a) illustrates the instruction functionality of the MR system, encompassing various aspects such as task allocation, historical display, path visualization, safety alerts, environment segmentation, and online adaptation. Task allocation dynamically assigns tasks to robots and human workers based on their real-time capabilities and availability, optimizing workflow and ensuring efficient resource utilization. The historical display feature allows operators to review past tasks and interactions, providing insights into performance trends and potential bottlenecks, thereby improving future task execution processes. Path visualization offers a graphical representation of the paths that robots will take, enabling operators to anticipate and mitigate potential collisions or interferences, thus enhancing operational safety and efficiency. The safety alert system monitors real-time operational data, detecting and warning potential hazards, ensuring that both human workers and robots are aware of unsafe conditions, thereby reducing the risk of accidents. Environment segmentation utilizes technology embedded in MR glasses to continuously scan and segment the workspace, identifying and highlighting potential hazards or unexpected changes in the environment in real time. Online adaptation records workers' interaction movements and incrementally updates the reinforcement learning-based motion planning algorithm based on the recorded operational trajectories, ensuring continuous improvement in task execution.

Fig. 7 (b) shows the safe warning information functionality, which is vital for maintaining a secure working environment. By tracking and displaying real-time trajectories of robots, this feature allows operators to monitor movements closely, predict potential collisions, and take preventive measures, ensuring safe interactions between robots and workers. The workspace display provides a comprehensive view of the entire robot workspace, emphasizing hazardous zone, thereby enhancing situational awareness and effective spatial management, which is crucial in dynamic industrial settings.

Additionally, the MR system includes a detailed information functionality about the operational status of robots, tasks, and human workers, as depicted in Fig. 7 (c). The robot information display shows critical data such as joint angles and the operational status of robotic arms, aiding in diagnosing issues, planning maintenance, ensuring that robots operate with defined parameters. Task information provides real-time status updates, helping tracking task progress and identifying areas requiring intervention. Human information offers data on workers' operational status, including current activities, facilitating effective integration of human workers within the MR system and promoting better coordination and safety.

Fig. 7 (d) presents the overall effect of MR system's user interface, showcasing how it integrates these functionalities to provide a comprehensive user experience. In summary, the multi-faceted design of the MR system aims to support seamless and safe human-robot interaction in industrial environments. The information presented in the MR environment influences both the robot's cognition of its operations and the worker's operating habits, helping workers better understand the robot's status. Simultaneously, real-time instructions are conveyed to both the worker and the robot through the mixed reality interface, providing immediate guidance and corrections. By integrating adaptive interaction capabilities, the MR system facilitates dynamic co-evolution between robots and human workers. This continuous feedback loop ensures that both robots and humans learn and adapt to each other's behaviors over time, leading to more harmonious and productive collaboration.

4. Experiments and results

To validate and demonstrate the proposed co-evolution approach for human-robot interaction, we conducted a series of experiments to evaluate the performance of HDT methods, DRL algorithms, and their combined use in improving the efficiency and safety of HRI. The experimental settings and results are detailed as follows.

4.1. Experiment settings

Fig. 8 illustrates the configuration of the physical workspace. A robotic arm (JAKA Zu3) equipped with an electric gripper (CTEK CTP2F50) is mounted on the workbench. Two RealSense D435i cameras are deployed: one to capture human skeletal data and the other to detect obstacles and target objects (not shown in the figure due to perspective constraints). For the experimental tasks, we designed dynamic scenarios, including randomly moving target objects and various work tasks, such as pick-and-place, drawer opening, and switch operation. To increase the complexity of the tasks, a 3D-printed board (10 cm×10 cm×0.3 cm) was placed on a linear track, moving linearly at random initial positions and speeds ranging from 5 to 10 cm/s. The physical scene and DT of the human body are constructed in Unity3D. We integrated the scene into MR system using HTC's equipment HoloLens 2. Our method is executed on a computer with an Intel i7–10,300 CPU, 32 GB RAM, and an NVIDIA 3090 GPU.

In Section 4.2.1, the HDT experiments evaluate the accuracy of capturing and modeling human movements. In Section 4.2.2, the DRL experiments examine the robot's motion planning capabilities in complex HRI task environments. In Section 4.2.3, the integrated demonstration experiments showcase the joint effects of HDT, MR, and DRL technologies in real-world industrial scenarios. These demonstrations provide a clear illustration of how humans and robots can achieve efficient and safe collaboration through information sharing and real-time feedback.

4.2. Experiment results

4.2.1. Human digital twin performance

We evaluated the effectiveness of our adaptive clustering-based 3D HPE (AC-HPE) method using a benchmark dataset, Human3.6M [54], the most widely used dataset in 3D HPE. It comprises 3.6 million video frames recorded at 50 Hz by four RGB cameras in an indoor environment.

Evaluation Metrics:

For Human3.6M dataset, we utilized the Mean Per Joint Position Error (MPJPE) as the primary evaluation metric. This metric measures the average Euclidean distance in millimeters, between the estimated 3D joint coordinates and the ground truth 3D joint coordinates, providing a direct measure of accuracy for pose estimation. In terms of computational efficiency, we employed Floating Point Operations (FLOPs) as the evaluation metric. FLOPs provide a standard measure of computational complexity, indicating the number of arithmetic operations required to process a single forward pass of the model.

Performance Validation: We verified the performance of our algorithm through several key aspects: joint position accuracy, computational efficiency, and robustness across different environments. By conducting thorough experiments and analyses, we demonstrate the
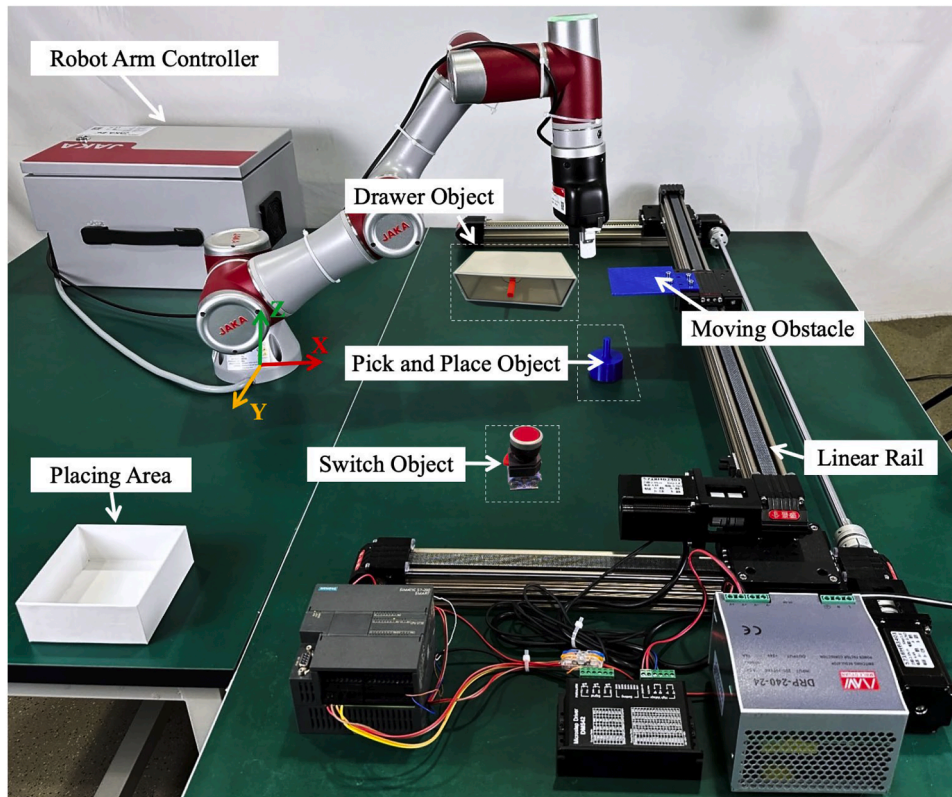
Fig. 8. Experimental setup of the physical workspace.

superior real-time perception and modeling capabilities of our method.

(1) Comparison with state-of-the-art methods

We first compared our proposed method, AC-HPE, with several state-of-the-art methods including Hossain & Little [44], VideoPose3D [45], MixSTE [43], and MotionBert [46]. Two key metrics were adopted, computational complexity, measured in FLOPs, and accuracy, assessed through MPJPE. The objective was to demonstrate that AC-HPE not only reduces computational cost but also enhances accuracy, thereby providing a more efficient and effective solution for human pose estimation in dynamic industrial environments.

Table 1 presents the comparison results. In terms of computational efficiency, AC-HPE achieves a significant reduction in computational complexity with 64.33 GFLOPs, which is substantially lower than MixSTE (277.25 GFLOPs) and MotionBert (131.09 GFLOPs). This reduction indicates that our method requires fewer computations, making it more suitable for real-time applications with limited computational resources. Regarding parameter efficiency, AC-HPE has 16.39 million parameters, which is comparable to Hossain & Little (16.96M) and MotionBert (16.00M), but significantly fewer than MixSTE (33.78M) and VideoPose3D (29.5M). This smaller model size helps in reducing memory usage and potentially leads to faster inference time. Even slight reductions in computational cost, as observed in our method, are meaningful in real-time systems where rapid and reliable processing is essential.

In terms of accuracy, AC-HPE achieves an MPJPE of 39.8 mm, which is better than Hossain & Little (41.6 mm), VideoPose3D (41.1 mm), and MixSTE (40.9 mm), and on par with MotionBert (39.8 mm). Though the difference in MPJPE values may seem marginal (e.g., 39.8 mm vs. 41.6 mm in Hossain & Little), even small improvements in accuracy are critical in dynamic human-robot interaction environments. Such improvements reduce potential errors in joint position estimation, enhancing the system's response to human movements and ensuring smoother collaboration. This result demonstrates that our method not only reduces computational cost but also maintains high accuracy.

The experimental results clearly highlight the advantages of our proposed AC-HPE method. It offers a balanced trade-off between computational efficiency and accuracy, making it a superior choice for human pose estimation in dynamic industrial environments. By significantly lowering the FLOPs while maintaining competitive accuracy, AC-HPE stands out as an effective solution that addresses the limitations of current state-of-the-art methods.

(2) Ablation studies

Our ablation studies focused on three aspects: the block index of representative tokens, frame pruning design choices, and visual assessment of 3D pose estimation. Each aspect was tested to evaluate its impact on accuracy and computational efficiency, with an emphasis on examining AC-HPE's core components rather than comparing with other methods. This approach allows for a clear assessment of how our design choices enhance efficiency and accuracy in real-time applications.

(a) Block index of representative tokens

We experimented with different block indexes (n) within the sequence-to-sequence pipeline to determine the optimal configuration for our method. The evaluation metrics for these experiments include MPJPE and computational complexity. The results of this study are summarized in Table 2, which highlights how varying the block index affects performance. Our method, AC-HPE, with a block index of $n=1$, achieved an MPJPE of 39.8 mm with a computational complexity of 64.33 GFLOPs, which is a 50.9 % reduction in FLOPs compared to the

Table 1
Comparison with state-of-the-art methods.

Method	Param (M)	FLOPs (G)	MPJPE
Hossain & Little [44]	16.96	33.88	41.6
VideoPose3D [45]	29.5	59.03	41.1
MixSTE [43]	33.78	277.25	40.9
MotionBert [46]	**16.00**	131.09	**39.8**
AC-HPE (Our method)	16.39	**64.33 (↓ 50.9 %)**	**39.8**

Table 2

Ablation study on the block index of representative tokens (n) under the sequence-to-sequence pipeline.

Method	Param (M)	FLOPs (G)	MPJPE
AC-HPE, $n=1$	16.39	**64.33 (↓50.9 %)**	**39.8**
AC-HPE, $n=2$	16.39	80.75 (↓38.4 %)	40.1
AC-HPE, $n=3$	16.39	97.14 (↓25.9 %)	40.4
AC-HPE, $n=4$	16.39	113.65 (↓13.3 %)	40.9

baseline. As the block index increased, we observed a trade-off between computational complexity and accuracy. Specifically, while increasing the block index to $n=2$ resulted in a slightly higher MPJPE of 40.1 mm, it also led to a FLOPs reduction to 80.75 GFLOPs. However, further increasing the block index to $n=3$ and $n=4$ increased the computational complexity to 97.14 GFLOPs and 113.65 GFLOPs, respectively, with corresponding MPJPE values of 40.4 mm and 40.9 mm. These results demonstrate that our method achieves the best balance of accuracy and computational efficiency with a block index of $n=1$, significantly outperforming higher block index configurations. This optimization highlights the efficiency of our approach, making it highly suitable for real-time applications where both accuracy and computational cost are critical factors.

(b) Frame pruning design choices

This study explored various design choices for frame pruning, assessing their impact on MPJPE and frame selection efficiency. Our goal was to identify the most effective frame pruning strategy. We compared our method's adaptive clustering against other methods, including Uniform Sampling, which evenly selects frames at set intervals to ensure balanced distribution. Attention Pruning leverages an attention mechanism to retain frames rich in information by focusing on their importance. Motion Pruning prioritizes frames based on motion amplitude, targeting areas with significant movement. Additionally, the Token Pruning Cluster (TPC) method [47] employs K-means clustering to select frames based on a predefined number of clusters.

The results detailed in Table 3, provide a comprehensive comparison of the different design choices. Our method achieves an MPJPE of 39.8 mm, demonstrating the highest accuracy among all strategies tested. Uniform Sampling achieved an MPJPE of 40.3 mm, while Attention Pruning and Motion Pruning resulted in MPJPE values of 41.0 mm and 41.7 mm, respectively. The TPC method, with an MPJPE of 39.9 mm, closely followed our method but did not surpass it. Our method outperforms the TPC method primarily due to its dynamic adaptation to the temporal structure and density of the data. While TPC uses a fixed number of clusters, which might not always represent the data's variability effectively, our method adapts to the varying density of the temporal data. This allows our method to better capture essential variations and reduce redundancy more efficiently. This flexibility in handling different data densities and distributions may contribute to its superior performance in keyframe selection, as evidenced by the lower MPJPE.

Additionally, Fig. 9 presents the statistical distribution of frames selected by our AC-HPE token pruning strategy. From a randomly chosen subset of 50 samples, we observed that the adaptive clustering approach effectively identifies key frames that capture significant temporal variations. This highlights its efficacy in both detecting essential changes and reducing redundancy.

Table 3

Ablation study on the design choices of frame pruning.

Method	MPJPE
Uniform Sampling	40.3
Attention Pruning	41.0
Motion Pruning	41.7
Token Pruning Cluster	39.9
Adaptive Clustering (Our method)	**39.8**

These results illustrate the effectiveness of our frame pruning strategy in maintaining high accuracy while optimizing frame selection efficiency. Our approach outperforms traditional methods by effectively balancing the retention of informative frames and reducing computational overhead, making it highly suitable for real-time applications in dynamic environments.

(3) Visual assessment of 3D pose estimation

To demonstrate the accuracy of our 3D pose estimation method, we provided visual showcases of its performance. Fig. 10 displays results from challenging industrial videos, showcasing the effectiveness of our method in accurately estimating 3D human poses in complex environments. Fig. 10 consists of four sub-images: (a) a worker stands and performs operations with the right arm partially occluded, (b) a worker sits on a chair with legs crossed, and (c) and (d) a worker operates on tabletop objects.

Specifically, Fig. 10 (a) illustrates the method's capability to accurately estimate poses even when parts of the body are occluded, highlighting its effectiveness in handling partial visibility. Fig. 10 (b) shows the system's ability to capture complex sitting postures, including crossed legs, which is critical for accurate human modeling in varied working conditions. Fig. 10 (c) and (d) depict the precision of our method during detailed tabletop operations, emphasizing its applicability in tasks requiring fine motor skills. Through these visual assessments, we demonstrate the method's reliability and precision in real-world industrial scenarios, where accurate 3D pose estimation is essential for improving human-robot interaction and ensuring worker safety. The ability to handle occlusions, complex postures, and detailed tasks underscores the method's potential to enhance operational efficiency and safety in dynamic industrial environments.

4.2.2. Deep reinforcement learning performance

In the experimental setup, we incorporated human motion scenarios, where human movements randomly appeared in the robotic arm's workspace, moving at random speed (10–20cm/s). In all subsequent tests, human motion was included in the scenes to simulate realistic dynamic industrial HRI environments. We compared the performance of our algorithm (A-IPPO) against Proximal Policy Optimization (PPO) [51], Soft Actor-Critic (SAC) [53], Improved SAC (I-SAC) [55], Improved PPO (I-PPO) [56], and Multi-Stage Curricula-based PPO (MSC-PPO) [6].

The metrics for performance evaluation include convergence speed, task success rate, episode length, collision rate, overtime rate, generalization performance in physical environments, and incremental performance. By comparing the convergence speeds during the 5 million steps of training for each method, we assessed the task success rates and episode lengths. We analyzed the collision rates and overtime rates during tasks, providing a comprehensive performance evaluation. Additionally, we tested each method's generalization ability in real-world physical environments and examined performance changes when changing human motion habit, further validating its adaptability and robustness in dynamic industrial settings. Through in-depth experiments and analysis, we aim to demonstrate our method's capability for safe motion planning with dynamic adaptation and learning.

(1) Comparison on convergence

We conducted a comparative study to evaluate the convergence of our method against other existing methods. The evaluation metrics included cumulative reward, convergence speed, and curve smoothness. Fig. 11 shows the comparison results, in which the proposed A-IPPO method achieved the highest cumulative reward, the smoothest curve, and the fastest convergence speed. This superior performance stems from our adaptive motion planning approach. In safe zones, geometry-based planning provides efficient, straightforward paths. When safety thresholds are approached, the system switches to RL, allowing the robot to handle more complex tasks, especially in dynamic environments with human motions.

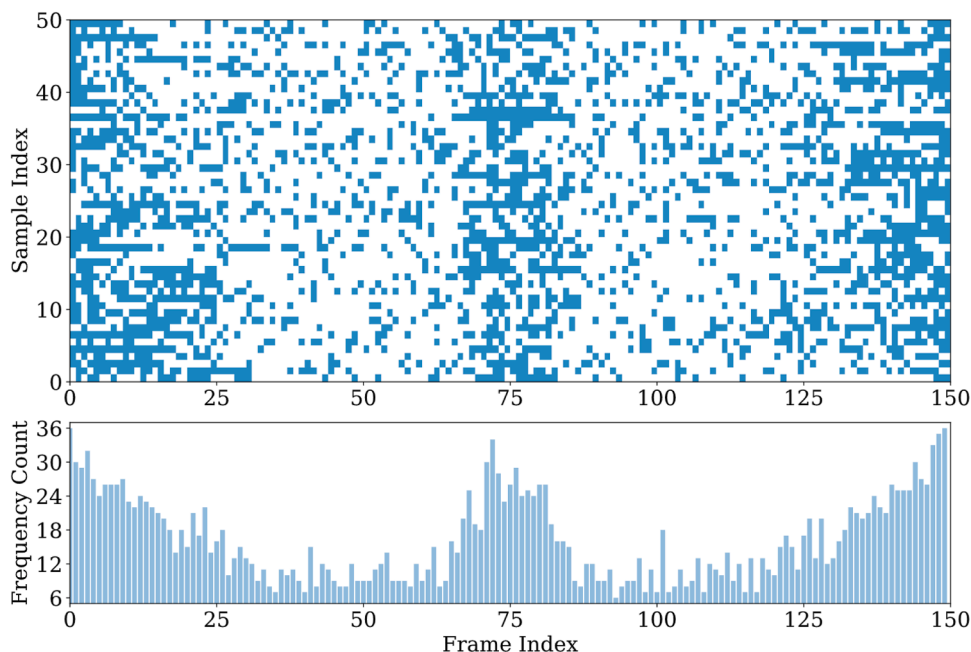Additionally, algorithms based on PPO displayed smoother reward

Fig. 9. Statistical visualization of selected frames for AC-HPE token pruning adaptive clustering strategy.
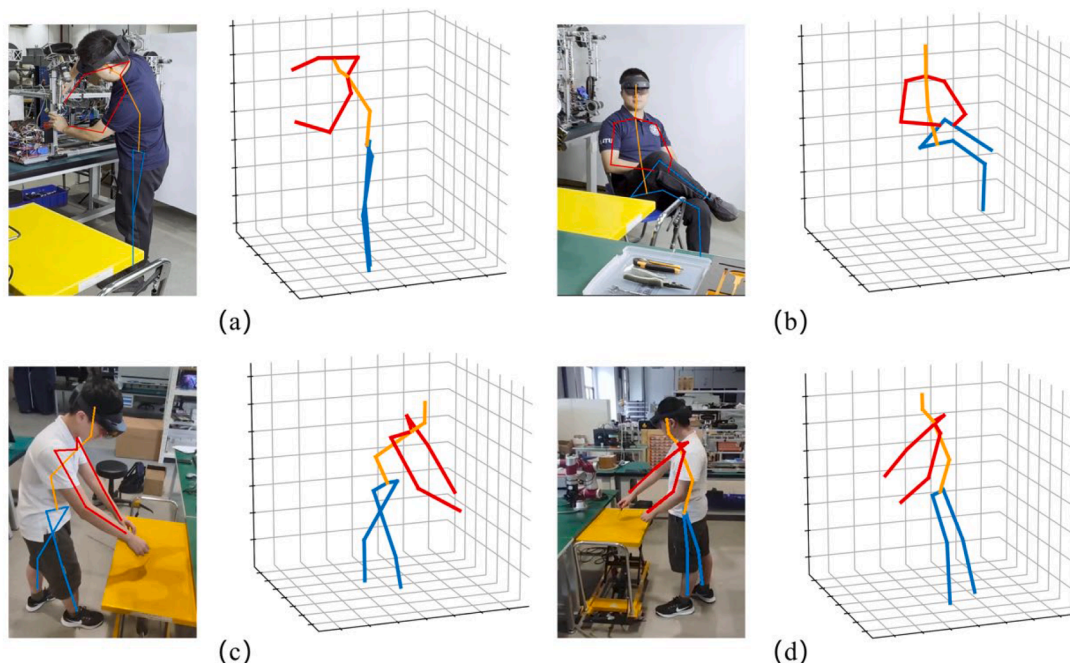

Fig. 10. 3D pose estimation in various industrial environments: (a) a worker is performing operations with right arm partially occluded; (b) a worker is seated with legs crossed; (c) and (d) a worker is engaged in detailed tabletop operations.

curves compared to those based on SAC. This difference can be explained by the inherent stability of the PPO algorithm. SAC algorithms, while powerful in many aspects, often suffer from higher variance in reward signals due to their reliance on entropy regularization, which encourages exploration. This characteristic can lead to less stable learning curves, especially in dynamic environments with frequent human interactions. Our choice of PPO as the RL model thus proves to be well-founded.

(2) Comparison on task success rate and episode length

To comprehensively assess model performance, we also compared our method with others regarding task success rate and episode length.

Each task was tested 1000 times in a virtual environment with dynamic human motions using six different methods. Table 4 shows that our A-IPPO method achieved the highest success rate in three tasks, with nearly 95.5 % success for button pressing, 94.0 % success for drawer opening, and 90.5 % for the pick and place task. It also required the fewest steps in most tasks.

In contrast, pure RL methods (i.e., SAC, PPO, I-SAC, MSC-PPO) without adaptive geometry-based combinations showed significantly poorer performance. These methods needed more exploration and trials to find collision-free paths, leading to higher collision rates and more steps. They also tended to fall into local optima, such as waiting above
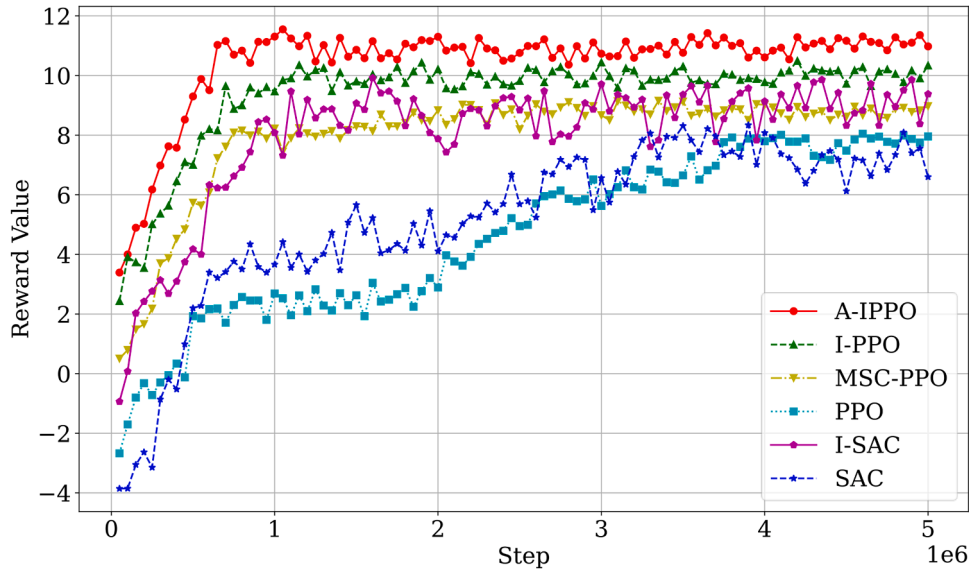
Fig. 11. Comparison of convergence performance across different methods.

Table 4
Comparison of task success rates and episode lengths across different methods.

Method	Button Press	Drawer Open	Pick and Place
SAC	80.5 % (534.8)	78.5 % (605.3)	63.0 % (986.3)
PPO	86.5 % (465.4)	73.0 % (679.8)	62.0 % (1074.5)
I-SAC	88.5 % (400.2)	87.0 % (544.3)	66.5 % (807.4)
MSC-PPO	89.4 % (351.7)	88.3 % (493.7)	77.3 % (820.2)
I-PPO	95.0 % (343.2)	92.5 % (598.6)	84.5 % (772.2)
A-IPPO (Our method)	**95.5 % (320.8)**	**94.0 % (412.3)**	**90.5 % (704.0)**

obstacles to avoid collisions with moving humans, which increased the steps needed to complete tasks.

When comparing A-IPPO with I-PPO, both methods use a combination of IK and RL. However, our algorithm employs a dynamic and multistage reward structure that promotes efficient path planning and task execution by providing continuous and stage-specific feedback. This approach can help the agent avoids local optima and adapts more effectively to dynamic environments, leading to superior performance than I-PPO.

(3) Comparison on collision rate and overtime rate

To further explore the impact of combining geometry-based methods in motion planning, we compared the collision rate and overtime rate of different methods as shown in Table 5. A-IPPO achieved the lowest collision and overtime rates across all tasks: 3.1 % and 1.4 % for button press, 4.0 % and 2.0 % for drawer opening, and 6.3 % and 3.2 % for pick and place, highlighting the superior performance of the A-IPPO algorithm.

In comparison, SAC exhibited the highest collision and overtime rates, particularly in the pick and place task with 19.5 % and 17.5 %, respectively. This underscores the instability associated with SAC's high

exploration tendency in dynamic environments. PPO, although more stable, also displayed relatively high collision (21.5 %) and overtime (16.5 %) rates in the same task, indicating a need for further optimization in complex settings. Both I-SAC and MSC-PPO showed improvements over their baseline counterparts, with reduced collision and overtime rates. I-SAC achieved notable reductions, particularly in the button press task (collision: 5.3 %, overtime: 6.2 %). MSC-PPO also performed better, with a collision rate of 6.1 % and an overtime rate of 4.5 % for the button press task. The I-PPO algorithm further reduced collision and overtime rates by incorporating inverse kinematics and a multi-stage reward structure, achieving 5.1 % and 2.4 % for collision and overtime rates in the drawer opening task, respectively.

The A-IPPO's success can be attributed to its adaptive planning mechanism, which identifies safe zones within the digital twin environment, reducing task difficulty and improving navigation efficiency. This results in significantly lower collision and overtime rates, emphasizing the importance of integrating advanced safety mechanisms and dynamic reward adjustments to enhance the performance of reinforcement learning algorithms in human-robot interaction scenarios.

(4) Evaluation of minimum distance direction in RL input

To assess the impact of incorporating the minimum distance direction between the robot and obstacles into the RL state input, we conducted additional tests using both a simulated and a real-world pick-and-place task. In these experiments, we modified the input configuration by adding the minimum distance direction alongside the original obstacle position, orientation, and minimum distance d_{ro}, to examine whether these additional inputs could enhance the task performance.

To assess the impact of incorporating the minimum distance direction between the robot and obstacles into the RL state input, we conducted additional tests using both a simulated and a real-world pick-and-place task. In these experiments, we modified the input configura-

Table 5
Comparison of collision rates and overtime rates across different methods.

Method	Button Press		Drawer Open		Pick and Place	
	Collision Rate	Overtime Rate	Collision Rate	Overtime Rate	Collision Rate	Overtime Rate
SAC	10.3 %	9.2 %	11.8 %	10.2 %	19.5 %	17.5 %
PPO	8.5 %	5.0 %	13.5 %	13.5 %	21.5 %	16.5 %
I-SAC	5.3 %	6.2 %	8.3 %	4.7 %	16.3 %	17.2 %
MSC-PPO	6.1 %	4.5 %	7.8 %	3.9 %	11.0 %	11.7 %
I-PPO	3.8 %	**1.2 %**	5.1 %	2.4 %	8.7 %	6.8 %
A-IPPO (Our method)	**3.1 %**	1.4 %	**4.0 %**	**2.0 %**	**6.3 %**	**3.2 %**

tion by adding the minimum distance direction alongside the original obstacle position , orientation, and minimum distance d_{ro}, to examine whether these additional inputs could enhance the task performance.

As shown in Table 6, experimental results indicate that adding the closest point and minimum distance direction had minimal impact on task success rates. In the virtual environment, task success rates were comparable between the two configurations, with 90.5 % for the original input configuration and a slightly higher 90.8 % with the added information. However, in real-world testing, the success rate for the original configuration was 80.6 %, while the configuration with the additional input parameters slightly decreased to 77.3 %. This suggests that the additional information did not contribute meaningfully to the RL agent's ability to handle physical variability in real environments.

In terms of computational efficiency, the configuration with the additional inputs required an average decision-making time of 96 ms, compared to 142 ms for the original input configuration. This increased computational load introduced additional processing time without yielding significant gains in task performance.

These results indicate that the RL agent can successfully learn effective avoidance behavior with the original input configuration, which includes only obstacle position, orientation, and minimum distance d_{ro}. The high task success rates in both the virtual and real environments suggest that the original input parameters provide sufficient spatial information for the agent to implicitly infer avoidance directions during training. In contrast, the closest point and minimum distance direction data did not enhance the agent's ability to avoid obstacles, as the agent could already learn such behaviors from the original inputs.

This outcome can be attributed to the agent's capacity for implicit learning, whereby it effectively uses position, orientation, and minimum distance d_{ro} to generalize collision avoidance across different obstacle configurations. Additional spatial information, such as the closest point and specific avoidance direction, may introduce redundancy, which can increase the computational overhead without necessarily enhancing performance. Moreover, in real-world environments with inherent sensor noise and dynamic variability, additional input parameters might reduce the system robustness, as more complex inputs can be more sensitive to inaccuracies. Simpler input configuration, therefore, is better at balancing computational efficiency and task performance, proving advantageous for dynamic industrial settings.

(5) Incremental learning

To assess the impact of the online adaptation (i.e., incremental learning) component, we conducted ablation experiments comparing the full A-IPPO algorithm with and without this feature. These tests were performed in both simulated and real environments, focusing on specific human motion habits and movement patterns. We introduced a new worker's motion habit, characterized by frequent and unpredictable stops and starts, to the robotic arm's working scenario to mimic real-world variability in human movement during collaborative tasks.

The evaluation metrics included success rate, collision rate, and overtime rate. We conducted 1000 trials in a simulated pick-and-place task environment, continuously updating the new worker's motion habits in the A-IPPO algorithm. The results in Table 7 indicate that the full algorithm with online adaptation significantly outperformed the ablation variant across all metrics. Specifically, the A-IPPO with online

Table 6
Comparison of task success rates and computational efficiency with varying RL input configurations.

Input Configuration	Task Success Rate (%)		Computational Efficiency (ms)
	(Simulated)	(Real-world)	
Position, Orientation, d_{ro} (Original)	90.5	**80.6**	**96**
Position, Orientation, d_{ro}, and Minimum Distance Direction	**90.8**	77.3	142

Table 7
Performance comparison of our algorithm with and without online adaptation.

	A-IPPO (with online adaptation)	A-IPPO (without online adaptation)
Success Rate	**90.4 %**	83.3 %
Overtime Rate	**6.7 %**	7.6 %
Collision Rate	**2.9 %**	9.1 %

adaptation achieved a success rate of 90.4 %, compared to 83.3 % without adaptation. Collision rates were 2.9 % with adaptation and 9.1 % without, while overtime rates were 6.7 % and 7.6 %, respectively.

The superior performance with online adaptation can be attributed to the algorithm's ability to swiftly adjust to unique movement patterns and environmental changes. By continuously learning and adapting based on newly acquired data, the model maintains high performance even when encountering new or varying task conditions. This results in more efficient and safer task execution, demonstrating the robustness and versatility of the proposed method in dynamic industrial settings.

(6) Generalization performance in physical environment

To validate the practicality of the proposed A-IPPO method, we conducted physical experiments in dynamic environments with human movements. It includes three tasks: button press, drawer open, and pick and place. The visual localization accuracy of obstacles, including target objects, was within ±2 mm, and the human digital twin environment updated at 10 frames per second. Each task was tested 50 times in these dynamic obstacle environments, with performance results summarized in Table 8.

The A-IPPO method demonstrated high success rates and efficient task completion times across all tasks. Specifically, the success rates were 92.0 % for button press, 90.0 % for drawer open, and 86.0 % for pick and place. The few failures were primarily due to task timeout, often occurring when the robot hesitated to grasp the target object due to rapid obstacle movement. The overtime rates were 6.0 % for button press, 6.0 % for drawer open, and 8.0 % for pick and place. Collision rates were minimal, with no collisions in the button press task, 4.0 % in drawer open, and 6.0 % in pick and place. Completion times averaged 7.2 s for button press, 7.9 s for drawer open, and 12.3 s for pick and place. These results underscore the effectiveness of the A-IPPO method in real-world environments with dynamic human motion, validating its practical application and robustness. The high success rates and low collision and overtime rates indicate that the A-IPPO method can maintain performance and safety standards in complex and unpredictable environments, highlighting its potential for widespread industrial applications.

4.2.3. Integrated demonstration

To demonstrate the overall feasibility and effect of the proposed approach, we tested a practical HRI scenario in a typical manufacturing environment with dynamic assembly tasks, as illustrated in Fig. 12. In this scenario, the control frequency of the DRL-based robot controller and the update frequency of the HDT environment are both set to 10Hz, which is a common treatment in HRI studies [6,57] striking a balance between system performance and computational load.

Fig. 12 (a) depicts a worker using MR glasses to operate a robot and assign tasks, such as retrieving a screwdriver. Fig. 12 (b) shows the worker viewing the robot's motion planning trajectory through the MR

Table 8
Task performance of A-IPPO method in real environment.

	Button Press	Drawer Open	Pick and Place
Success Rate	92.0 % (46/50)	90.0 % (45/50)	86.0 % (44/50)
Overtime Rate	6.0 % (4/50)	6.0 % (3/50)	8.0 % (4/50)
Collision Rate	0 % (0/50)	4.0 % (2/50)	6.0 % (6/50)
Completion Time (s)	7.2	7.9	12.3
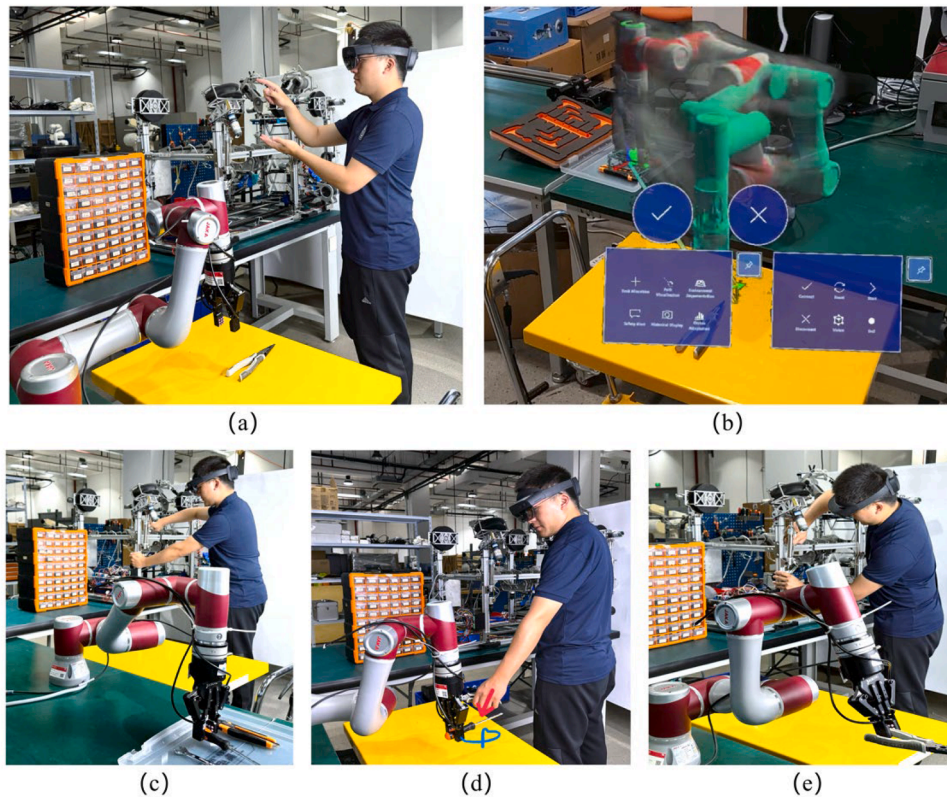
Fig. 12. Practical manufacturing HRI scenario: (a) a worker is assigning tasks using MR glasses; (b) real-time robot motion planning trajectory visualization through MR glasses; (c) simultaneous worker's assembly operations and robot's tool retrieval; (d) robot arm is dynamically avoiding worker's arm; (e) robot arm is returning tools while worker continues assembly task.

glasses. This visualization enables the worker to monitor the robot's status and expected trajectory in real-time, and helps the worker better comprehend and anticipate the robot's behavior, thereby preventing potential conflicts. In Fig. 12 (c), the worker uses pliers to assemble complex components in the assembly area, while the robot retrieves the screwdriver from the tool area. This shared workspace scenario requires the robot to perform precise movements without interfering with the worker's tasks. The MR system provides detailed instructions and guidance directly displayed in the worker's field of view, assisting the worker in locating the correct tools and assembly parts, and offering step-by-step assembly instructions. Fig. 12 (d) demonstrates the dynamic adjustment capabilities of the robotic arm. Utilizing a safe motion planning algorithm, the robot can deftly avoid the worker's hand as he/she places the pliers, ensuring no collisions occur during task execution. The blue path in the figure represents the end-effector avoidance trajectory. This capability showcases the system's ability to maintain safe and efficient operations amidst real-time changes. Finally, Fig. 12(e) illustrates the robotic arm returning the used pliers to their original position while the worker continues with the complex assembly task.

Throughout the task execution, the MR system's online adaptive features enable the reinforcement learning-based motion planning algorithm to update based on the real-time recorded operational trajectories. Additionally, the MR glasses play a crucial role in post-task inspection and quality control, highlighting potential issues or assembly errors to ensure timely correction and resolution. In summary, the practical demonstration of the human-robot interaction scenario in a manufacturing environment, as illustrated in Fig. 12, highlights the significant advantages of the proposed approach in fostering safe and efficient human-robot collaboration. This result validates the potential application of our approach in complex and dynamic environments.

4.2.4. System validation via operator performance evaluation

To evaluate the feasibility and effectiveness of the proposed MR-based HRI system, we recruited 15 participants (college students with engineering background, including 8 males and 7 females) to perform dynamic assembly tasks using the system. Prior to the experiment, participants received a brief training session to familiarize themselves with the MR system's basic operations and interaction workflows. This training covered key aspects, including the reception of real-time guidance, response to visual cues, and handling of the robot's dynamic adjustments.

The experimental tasks consisted of three stages: tool retrieval, component assembly, and dynamic obstacle avoidance. In the tool retrieval stage, operators were required to locate specific tools accurately and efficiently under MR guidance, enabling us to assess the system's impact on task efficiency. During component assembly, operators followed MR-provided instructions to complete assembly tasks, which allowed us to evaluate the system's ability to enhance operational accuracy and reduce assembly errors. In the dynamic obstacle avoidance task, operators collaborated with the robot while assembling components, which can help with validating the system's capability to ensure operator safety and optimize real-time obstacle avoidance.

Following task completion, participants provided feedback via a questionnaire-based survey. This survey collected quantitative data on various performance indicators, including task support effectiveness, response timeliness, efficiency improvement, error reduction, system adaptability, and overall user satisfaction. The following subsections present a detailed analysis of both operator and system performance, supported by quantified results.

(1) Operator performance metrics

Table 9 presents a comparison of task completion time and accuracy between the MR system and traditional method. Under traditional method, tool retrieval involved locating tools using static pictorial

Table 9
Comparison of task completion time and accuracy between MR system and traditional methods.

Task	MR System Time (seconds)	Traditional Time (seconds)	Accuracy with MR System (%)	Accuracy with Traditional Method (%)
Tool Retrieval	**15.2**	18.5	**98**	94
Component Assembly	**120.3**	138.7	**97**	91
Obstacle Avoidance	**10.4**	12.8	**100**	95

Table 10
Operator satisfaction ratings on MR system usability and performance.

Question	Rating
How helpful was the MR system in understanding and completing tasks?	4.7/5
How satisfied were you with the real-time feedback provided by the MR system?	4.5/5
Did the MR system improve your task efficiency and reduce errors?	4.6/5
Did you encounter any technical issues (e.g., delays or glitches)?	1.2/5
How would you rate your overall satisfaction with the MR system?	4.8/5

guides, and component assembly relied on manual references and printed instructions. Additionally, obstacle avoidance lacked dynamic planning when using traditional method, causing the system to halt whenever someone approached closely to the robot. Task completion time was measured in seconds, while accuracy was determined based on task precision (e.g., tool retrieval and assembly accuracy).

The data shows that the MR system significantly reduced task completion time by an average of 18 %, particularly in tasks requiring dynamic adjustments like tool retrieval and obstacle avoidance. Furthermore, the system improved accuracy by an average of 10 % in component assembly, reflecting the effectiveness of the MR guidance system. The MR system demonstrated a response time of 0.5 s, which is well within acceptable limits for real-time human-robot interaction.

(2) Operators' satisfaction on using the System

Table 10 presents the operator satisfaction levels, based on a subjective assessment of the system's impact on task performance, real-time feedback, and overall usability.

The subjective feedback indicates a high level of operator satisfaction, with the majority reporting that the MR system improved their efficiency and reduced errors. A very small proportion of operators (approximately 10 %) noted minor technical issues, primarily related to system latency.

The results of operators' feedback provide strong quantitative evidence supporting the effectiveness of the proposed MR-based human-robot interaction system. The MR system not only improved task completion efficiency and accuracy but also enhanced operator satisfaction and system performance. These results validate the feasibility of applying the proposed approach in real-world, dynamic environments, demonstrating its potential for improving safety and collaboration efficiency in industrial tasks.[2]

5. Conclusion

In this study, we propose a co-evolution approach to enhance the safety of industrial human-robot interaction (HRI). By integrating advanced human digital twin (HDT) and mixed reality (MR) technologies, our approach enables bidirectional information flow, allowing for mutual learning and adaptation between humans and robots. For human workers, the system provides real-time information visualizations and working instructions, helping them better understand and anticipate robot's actions, thereby increasing situational awareness and reducing misunderstandings and safety accidents. For robots, human behavior data obtained through HDT and MR, along with real-time instructions and operational habits, enhance the robots' understanding of human's working processes and patterns. This enables robots to progressively optimize their own behavior strategies, improving operational capabilities in complex environments and refining safety planning strategies for both parties. The HDT, DRL, and integrated demonstration experiments demonstrate the effectiveness of the proposed approach in dynamic

manufacturing settings. Our study contributes to the development of novel human-robot interaction framework for the next-generation human-centric smart manufacturing mode.

One limitation of this study is that, while MR glasses provide comprehensive visual guidance, their reliance on real-time data of environment, human and robot can be a bottleneck in environments with severe network latency or data transmission issues. The user experience and usability of the MR interface also require further optimization to ensure broad applicability across different operators.

Future research will address these limitations. We plan to further improve the computational efficiency of the HDT and motion planning algorithms to reduce dependency on high-frequency updates, thereby increasing system applicability and robustness. Additionally, exploring edge computing and advanced communication protocols will enhance the system's performance under varying network conditions. Through iterative user testing and feedback, we will also refine the MR interface to ensure it meets the needs of diverse industrial operational environments.

CRediT authorship contribution statement

Bohan Feng: Writing – original draft, Methodology, Conceptualization. **Zeqing Wang:** Validation, Software. **Lianjie Yuan:** Validation, Software. **Qi Zhou:** Data curation. **Yulin Chen:** Visualization. **Youyi Bi:** Writing – review & editing, Project administration, Funding acquisition.

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.

Data availability

Data will be made available on request.

References

[1] E. Commission, D.-G. for Research, Innovation, M. Breque, L. De Nul, A. Petridis, Industry 5.0 – towards a sustainable, human-centric and resilient European industry, Publications Office of the European Union, 2021. 10.2777/308407.
[2] X. Xu, Y. Lu, B. Vogel-Heuser, L. Wang, Industry 4.0 and Industry 5.0—inception, conception and perception, J. Manuf. Syst. 61 (2021) 530–535, https://doi.org/10.1016/j.jmsy.2021.10.006.
[3] J. Leng, W. Sha, B. Wang, P. Zheng, C. Zhuang, Q. Liu, T. Wuest, D. Mourtzis, L. Wang, Industry 5.0: prospect and retrospect, J. Manuf. Syst. 65 (2022) 279–295, https://doi.org/10.1016/j.jmsy.2022.09.017.
[4] D. Mukherjee, K. Gupta, L.H. Chang, H. Najjaran, A survey of robot learning strategies for human-robot collaboration in industrial settings, Robot. Comput. Integr. Manuf. 73 (2022) 102231, https://doi.org/10.1016/j.rcim.2021.102231.
[5] N. Dimitropoulos, T. Togias, N. Zacharaki, G. Michalos, S. Makris, Seamless human–robot collaborative assembly using artificial intelligence and wearable devices, Appl. Sci. 11 (2021) 5699, https://doi.org/10.3390/app11125699.

[2] For a demonstration of our system, please see the video: https://www.youtube.com/watch?v=iZl7r_tehIk.

[6] C. Li, P. Zheng, Y. Yin, Y.M. Pang, S. Huo, An AR-assisted deep reinforcement learning-based approach towards mutual-cognitive safe human-robot interaction, Robot. Comput. Integr. Manuf. 80 (2023) 102471, https://doi.org/10.1016/j.rcim.2022.102471.

[7] S. Li, P. Zheng, S. Liu, Z. Wang, X.V. Wang, L. Zheng, L. Wang, Proactive human–robot collaboration: mutual-cognitive, predictable, and self-organising perspectives, Robot. Comput. Integr. Manuf. 81 (2023) 102510, https://doi.org/10.1016/j.rcim.2022.102510.

[8] J.N. Thompson, The Coevolutionary Process, University of Chicago Press, Chicago, 1994, https://doi.org/10.7208/9780226797670.

[9] E. Yoshida, F. Kanehiro, Reactive robot motion using path replanning and deformation, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2011, pp. 5456–5462, https://doi.org/10.1109/ICRA.2011.5980361.

[10] A. Gasparetto, P. Boscariol, A. Lanzutti, R. Vidoni, Path planning and trajectory planning algorithms: a general overview. https://doi.org/10.1007/978-3-319-14705-5_1.

[11] Y. Shaoul, I. Mishani, M. Likhachev, J. Li. Accelerating search-based planning for multi-robot manipulation by leveraging online-generated experiences, AAAI Press, 2024. https://doi.org/10.1609/icaps.v34i1.31513.

[12] B. Feng, X. Jiang, B. Li, Q. Zhou, Y. Bi, An adaptive multi-RRT approach for robot motion planning, Expert. Syst. Appl. 252 (2024) 124281, https://doi.org/10.1016/j.eswa.2024.124281.

[13] Y. Wang, P. Praveena, D. Rakita, M. Gleicher, RangedIK: an optimization-based robot motion generation method for ranged-goal tasks, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2023, pp. 9700–9706, https://doi.org/10.1109/ICRA48891.2023.10161311.

[14] J. Wang, T. Zhang, N. Ma, Z. Li, H. Ma, F. Meng, M.Q.-H. Meng, A survey of learning-based robot motion planning, IET Cyber-Syst. Robot. 3 (2021) 302–314, https://doi.org/10.1049/csy2.12020.

[15] M. Hamandi, M. D'Arcy, P. Fazli, DeepMoTIon: learning to navigate like humans, in: Proceedings of the 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), IEEE, 2019, pp. 1–7, https://doi.org/10.1109/RO-MAN46459.2019.8956408.

[16] L. Wang, G. Wang, S. Jia, A. Turner, S. Ratchev, Imitation learning for coordinated human–robot collaboration based on hidden state-space models, Robot. Comput. Integr. Manuf. 76 (2022) 102310, https://doi.org/10.1016/j.rcim.2021.102310.

[17] T. Zhang, H. Sun, Y. Zou, H. Chu, An electromyography signals-based human-robot collaboration method for human skill learning and imitation, J. Manuf. Syst. 64 (2022) 330–343, https://doi.org/10.1016/j.jmsy.2022.07.005.

[18] B. Sangiovanni, A. Rendiniello, G.P. Incremona, A. Ferrara, M. Piastra, Deep reinforcement learning for collision avoidance of robotic manipulators, in: Proceedings of the European Control Conference (ECC), 2018, pp. 2063–2068, https://doi.org/10.23919/ECC.2018.8550363.

[19] Q. Liu, Z. Liu, B. Xiong, W. Xu, Y. Liu, Deep reinforcement learning-based safe interaction for industrial human-robot collaboration using intrinsic reward function, Adv. Eng. Inform. 49 (2021) 101360, https://doi.org/10.1016/j.aei.2021.101360.

[20] M. El-Shamouty, X. Wu, S. Yang, M. Albus, M.F. Huber, Towards safe human-robot collaboration using deep reinforcement learning, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 4899–4905, https://doi.org/10.1109/ICRA40945.2020.9196924.

[21] C. Li, P. Zheng, Y. Yin, Y.M. Pang, S. Huo, An AR-assisted deep reinforcement learning-based approach towards mutual-cognitive safe human-robot interaction, Robot. Comput. Integr. Manuf. 80 (2023) 102471, https://doi.org/10.1016/j.rcim.2022.102471.

[22] M.A. Simão, O. Gibaru, P. Neto, Online recognition of incomplete gesture data to interface collaborative robots, IEEE Trans. Ind. Electron. 66 (2019) 9372–9382, https://doi.org/10.1109/TIE.2019.2891449.

[23] J. Lanini, H. Razavi, J. Urain, A. Ijspeert, Human intention detection as a multiclass classification problem: application in physical human–robot interaction while walking, IEEE Robot. Autom. Lett. 3 (2018) 4171–4178, https://doi.org/10.1109/LRA.2018.2864351.

[24] Y. He, F. Li, J. Li, J. Liu, X. Wu, An sEMG based adaptive method for human-exoskeleton collaboration in variable walking environments, Biomed. Signal. Process. Control 74 (2022) 103477, https://doi.org/10.1016/j.bspc.2021.103477.

[25] A. Buerkle, W. Eaton, N. Lohse, T. Bamber, P. Ferreira, EEG based arm movement intention recognition towards enhanced safety in symbiotic human-robot collaboration, Robot. Comput. Integr. Manuf. 70 (2021) 102137, https://doi.org/10.1016/j.rcim.2021.102137.

[26] M. Safeea, P. Neto, Minimum distance calculation using laser scanner and IMUs for safe human-robot interaction, Robot. Comput. Integr. Manuf. 58 (2019) 33–42, https://doi.org/10.1016/j.rcim.2019.01.008.

[27] H. Liu, L. Wang, Collision-free human-robot collaboration based on context awareness, Robot. Comput. Integr. Manuf. 67 (2021) 101997, https://doi.org/10.1016/j.rcim.2020.101997.

[28] B. Parsa, E.U. Banerjee, R. Samani, C. Hendrix, S.M. Devine, S. Singh, A.G. Devasia, Toward ergonomic risk prediction via segmentation of indoor object manipulation actions using spatiotemporal convolutional networks, IEEE Robot. Autom. 4 (2019) 3153–3160, https://doi.org/10.1109/LRA.2019.2925305.

[29] J. Fan, P. Zheng, C.K.M. Lee, A vision-based human digital twin modeling approach for adaptive human–robot collaboration, J. Manuf. Sci. Eng. 145 (2023), https://doi.org/10.1115/1.4062430.

[30] S. Yi, S. Liu, X. Xu, X.V. Wang, S. Yan, L. Wang, A vision-based human-robot collaborative system for digital twin, Procedia CIRP 107 (2022) 552–557, https://doi.org/10.1016/j.procir.2022.05.024.

[31] P. Galambos, Á. Csapó, P. Zentay, I.M. Fülöp, T. Haidegger, P. Baranyi, I.J. Rudas, Design, programming and orchestration of heterogeneous manufacturing systems through VR-powered remote collaboration, Robot. Comput. Integr. Manuf. 33 (2015) 68–77, https://doi.org/10.1016/j.rcim.2014.08.012.

[32] C. Vogel, C. Walter, N. Elkmann, Safeguarding and supporting future human-robot cooperative manufacturing processes by a projection- and camera-based technology, Procedia Manuf. 11 (2017) 39–46, https://doi.org/10.1016/j.promfg.2017.07.127.

[33] D.K. Baroroh, C.-H. Chu, L. Wang, Systematic literature review on augmented reality in smart manufacturing: collaboration between human and computational intelligence, J. Manuf. Syst. 61 (2021) 696–711, https://doi.org/10.1016/j.jmsy.2020.10.017.

[34] A. Hietanen, R. Pieters, M. Lanz, J. Latokartano, J.K. Kämäräinen, AR-based interaction for human-robot collaborative manufacturing, Robot. Comput. Integr. Manuf. 63 (2020) 101891, https://doi.org/10.1016/j.rcim.2019.101891.

[35] S. Papanastasiou, N. Kousi, P. Karagiannis, C. Gkournelos, A. Papavasileiou, K. Dimoulas, K. Baris, S. Koukas, G. Michalos, S. Makris, Towards seamless human robot collaboration: integrating multimodal interaction, Int. J. Adv. Manuf. Technol. 105 (2019) 3881–3897, https://doi.org/10.1007/s00170-019-03790-3.

[36] C.Y. Siew, S.K. Ong, A.Y.C. Nee, A practical augmented reality-assisted maintenance system framework for adaptive user support, Robot. Comput. Integr. Manuf. 59 (2019) 115–129, https://doi.org/10.1016/j.rcim.2019.03.010.

[37] J. Jost, T. Kirks, P. Gupta, D. Lünsch, J. Stenzel, Safe Human-robot-interaction in highly flexible warehouses using augmented reality and heterogenous fleet management system, in: Proceedings of the IEEE International Conference on Intelligence and Safety for Robotics (ISR), 2018, pp. 256–260, https://doi.org/10.1109/IISR.2018.8535808.

[38] H. Li, W. Ma, H. Wang, G. Liu, X. Wen, Y. Zhang, M. Yang, G. Luo, G. Xie, C. Sun, A framework and method for human-robot cooperative safe control based on digital twin, Adv. Eng. Inform. 53 (2022) 101701, https://doi.org/10.1016/j.aei.2022.101701.

[39] S.H. Choi, K.-B. Park, D.H. Roh, J.Y. Lee, M. Mohammed, Y. Ghasemi, H. Jeong, An integrated mixed reality system for safety-aware human-robot collaboration using deep learning and digital twin generation, Robot. Comput. Integr. Manuf. 73 (2022) 102258, https://doi.org/10.1016/j.rcim.2021.102258.

[40] Y. Su, X. Chen, T. Zhou, C. Pretty, G. Chase, Mixed reality-integrated 3D/2D vision mapping for intuitive teleoperation of mobile manipulator, Robot. Comput. Integr. Manuf. 77 (2022) 102332, https://doi.org/10.1016/j.rcim.2022.102332.

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

[42] D. Cai, J. Heikkia, E. Rahtu, OVE6D: object viewpoint encoding for depth-based 6D object pose estimation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2022, pp. 6793–6803, https://doi.org/10.1109/CVPR52688.2022.00668.

[43] J. Zhang, Z. Tu, J. Yang, Y. Chen, J. Yuan, MixSTE: Seq2seq Mixed Spatio-Temporal Encoder for 3D Human Pose Estimation in Video. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 13222–13232. https://doi.org/10.1109/CVPR52688.2022.01288.

[44] M.R.I. Hossain, J.J. Little, Exploiting Temporal Information for 3D Human Pose Estimation, in: Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part X, Springer-Verlag, Berlin, Heidelberg, 2018, pp. 69–86. https://doi.org/10.1007/978-3-030-01249-6_5.

[45] D. Pavllo, C. Feichtenhofer, D. Grangier, M. Auli, 3D Human Pose Estimation in Video With Temporal Convolutions and Semi-Supervised Training. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7745–7754. https://doi.org/10.1109/CVPR.2019.00794.

[46] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, Y. Wang, MotionBERT: A Unified Perspective on Learning Human Motion Representations. 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 2023, pp. 15039–15053. https://doi.org/10.1109/ICCV51070.2023.01385.

[47] W. Li, M. Liu, H. Liu, P. Wang, J. Cai, N. Sebe, Hourglass Tokenizer for Efficient Transformer-Based 3D Human Pose Estimation. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 604–613. https://doi.org/10.1109/CVPR52733.2024.00064.

[48] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, Y. Sheikh, OpenPose: realtime multi-person 2D pose estimation using part affinity fields, IEEE Trans. Pattern Anal. Mach. Intell. 43 (2021) 172–186, https://doi.org/10.1109/TPAMI.2019.2929257.

[49] B. Kenwright, Inverse kinematics – cyclic coordinate descent (CCD), J. Graph. Tools 16 (2012) 177–217, https://doi.org/10.1080/2165347X.2013.823362.

[50] Q. Zhou, S. Li, J. Qu, J. Wu, H. Xu, Y. Bi, An adaptive path planning approach for digital twin-enabled robot arm based on inverse kinematics and deep reinforcement learning, in: volume 3: advanced manufacturing, Am. Soc. Mech. Eng. (2023), https://doi.org/10.1115/IMECE2023-113131.

[51] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, ArXiv abs/1707.06347 (2017). https://api.semanticscholar.org/CorpusID:28695052.

[52] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N.M.O. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, CoRR abs/1509.02971 (2015). https://api.semanticscholar.org/CorpusID:16326763.

[53] T. Haarnoja, A. Zhou, P. Abbeel, S. Levine, Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor, ArXiv abs/1801.01290 (2018). https://api.semanticscholar.org/CorpusID:28202810.

[54] C. Ionescu, D. Papava, V. Olaru, C. Sminchisescu, Human3.6M: large scale datasets and predictive methods for 3D Human sensing in natural environments, IEEE

Trans. Pattern. Anal. Mach. Intell. 36 (2014) 1325–1339, https://doi.org/10.1109/TPAMI.2013.248.

[55] P. Chen, J. Pei, W. Lu, M. Li, A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance, Neurocomputing. 497 (2022) 64–75, https://doi.org/10.1016/j.neucom.2022.05.006.

[56] D. Luipers, N. Kaulen, O. Chojnowski, S. Schneider, A. Richert, S. Jeschke, Robot control using model-based reinforcement learning with inverse kinematics, in:

Proceedings of the IEEE International Conference on Development and Learning (ICDL), 2022, pp. 244–249, https://doi.org/10.1109/ICDL53763.2022.9962215.

[57] C. Li, P. Zheng, P. Zhou, Y. Yin, C.K.M. Lee, L. Wang, Unleashing mixed-reality capability in deep reinforcement learning-based robot motion generation towards safe human–robot collaboration, J. Manuf. Syst. 74 (2024) 411–421, https://doi.org/10.1016/j.jmsy.2024.03.015.