

Combining Learning from Demonstration with Learning by Exploration to Facilitate Contact-Rich Tasks

Yunlei Shi^{1,2}, Zhaopeng Chen^{2,1}, Yansong Wu³, Dimitri Henkel²,
Sebastian Riedel², Hongxu Liu³, Qian Feng^{3,2}, Jianwei Zhang¹

Abstract— Collaborative robots are expected to work alongside humans and directly replace human workers in some cases, thus effectively responding to rapid changes in assembly lines. Current methods for programming contact-rich tasks, particularly in heavily constrained spaces, tend to be fairly inefficient. Therefore, faster and more intuitive approaches are urgently required for robot teaching. This study focuses on combining visual servoing learning from demonstration (LfD) and force-based learning by exploration (LbE) to enable the fast and intuitive programming of contact-rich tasks with minimal user efforts. Two learning approaches were developed and integrated into a framework, one relying on human-to-robot motion mapping (visual servoing approach) and the other relying on force-based reinforcement learning. The developed framework implements the noncontact demonstration teaching method based on the visual servoing approach and optimizes the demonstrated robot target positions according to the detected contact state. The developed framework is compared with two most commonly used baseline techniques, i.e., teach pendant-based teaching and hand-guiding teaching. Furthermore, the efficiency and reliability of the framework are validated via comparison experiments involving the teaching and execution of contact-rich tasks. The proposed framework shows the best performance in terms of the teaching time, execution success rate, risk of damage, and ease of use.

I. INTRODUCTION

The ability to rapidly setup and reprogram newly-introduced products in factories is an increasingly essential requirement for adaptive robotic assembly systems [1], [2]. Position-controlled robots can handle known objects within well-structured assembly lines with high efficiency and achieve highly accurate position control. However, they require considerable setup time and tedious reprogramming to fulfill new tasks, and cannot adapt to any unexpected variations in assembly processes [3].

Collaborative robots offer the promise of closing the gap between onerous reprogramming and unexpected variations by combining the capabilities of position-controlled robots with dexterity and flexibility. For example, the hand-guiding method enables unskilled users to interact with collaborative robots and facilitates quick programming [4]. However, during assembly line reconfiguration, a long time is still required to remove and reinstall the robot arms and various

*This research has received funding from the German Research Foundation (DFG) and the National Science Foundation of China (NSFC) in project Crossmodal Learning, DFG TRR-169/NSFC 61621136008, partially supported by European projects H2020 STEP2DYNA (691154) and ULTRACEPT (778602).

¹TAMS (Technical Aspects of Multimodal Systems), Department of Informatics, Universität Hamburg, ²Agile Robots AG, ³Technische Universität München.

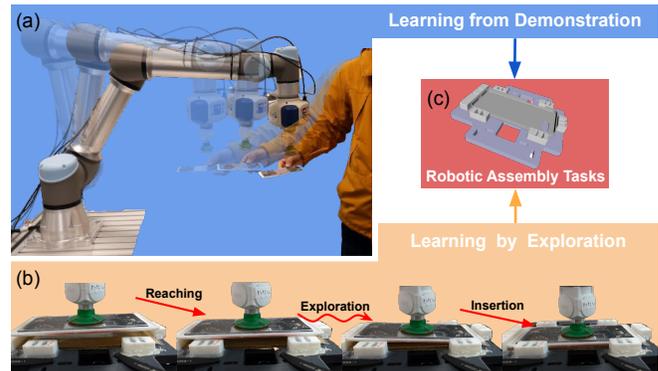


Fig. 1. A robot arm and a suction gripper performing a contact-rich tending task. (a) Gross motion is learned from human demonstration. (b) Fine motion is learned from exploration. (c) Example of a contact-rich tending task

attachments. Mobile manipulators (where robotic arms are mounted on mobile bases) were introduced to expand the productivity and adaptive capacity of manufacturing automation, particularly during the setting up phase when production lines must be reconfigured [5]. Because mobile manipulators can only be placed beside production lines and cannot be installed on production lines as collaborative robots, which occupying space previously provided for human workers. However, programming a robot in a constrained space is very difficult [6]. Overall, the ease-of-programming has been identified as an open challenge in robot assembly [2], [7].

Additionally, collaborative robots equipped with force control functions can perform certain hybrid position/force operations for contact-rich tasks [8], [9], [10], [11], [12]; however, their effectiveness and variation adaptive capacity in assembly processes are still unsatisfactory [13], [14].

Herein, we propose an intuitive programming method to decrease the setup time of mobile manipulators and introduce a reinforcement learning (RL) algorithm to overcome unexpected variations in assembly tasks (Figure 1). Our primary contributions are:

- **C1:** An approach is presented that learns the trajectories of robots from demonstrations based on visual servoing for fast, easy, and accurate robot setup in heavily constrained spaces.
- **C2:** A region-limited residual RL (RRRL) policy based on force-torque information is trained to overcome pose uncertainty.
- **C3:** C1 and C2 are combined into a robot tending skill and compared with two most commonly used baseline techniques using an UR5e robot.

The rest of the paper is organized as follows. In Section II, we provide an overview of the development of industrial mobile manipulators and robot programming methods, as well as their pros and cons. Moreover, the force controller is briefly introduced. In Section III, we outline some key problems of assembly tasks and corresponding ideas of our method. Details on our proposed method are provided in Section IV, and a quantitative experiment of our methods is presented in Section V. Experimental results are presented and discussed in Section VI. Finally, potential future research areas are suggested in Section VII.

II. RELATED WORK AND BACKGROUND

A. Industrial Mobile Manipulator

Mobile manipulator systems, also known as mobile manipulator hybrid robots¹ or collaborative cells², were designed to reduce the setup time of industrial robots; all the required task devices are integrated into a mobile platform, and it is therefore a “plug and play” solution. Various mobile manipulators have been developed, such as Rob@Work [15], Little Helper [16], KMR iiwa [17], KUKA flexFELLOW³ and MiR UR⁴, which are the most widely-deployed industrial mobile manipulators. However, the traditional magnetic tracking approach utilized by commercial mobile manipulators (i.e., automatic guided vehicle) can be accompanied by position uncertainty problem. This does not satisfy the requirements of many industrial applications, particularly when mobile manipulators are required to move between picking and placing locations.

B. LfD Under Visual Guidance

LfD has recently been recommended as an effective technique for accelerating the programming of learning processes, spanning from low-level control to high-level assembly planning [18], [19]. Guiding robots by means of visual feedback [20], [21], [22] during assembly tasks is an effective way to overcome position uncertainties introduced by mobile manipulators. For robotic assembly tasks, performing high precision measurements is important. However, visual errors can be introduced by lenses and the imaging sensors, as well as the calibration of intrinsic and extrinsic parameters [23]. Some researchers believe that humans should focus more on execution tasks than vision sensors [23]. Based on this notion, other approaches have been developed, such as intelligent assembly algorithms, in an effort to lower the necessity of vision sensors for given tasks.

C. LbE for Contact-rich Tasks

LbE has been suggested as an effective method for reducing the programming time, and recent studies have introduced artificial intelligence methods into robotics [1], [3], [24], [25]. Moreover, RL offers a set of tools for



Fig. 2. Examples of heavily constrained spaces in real factories⁵.

designing sophisticated robotic behaviors that are difficult to engineer. RL and its derivative methods have previously been successfully used to address various robotic manipulation problems [12], [26], [9], [13], [27], [28]. Exploration behavior entails interactions between robots and their operational environment. Therefore, a force or impedance controller is required.

D. Operational Space Force Control

Interaction control has attracted widespread scholarly interest following pioneering works on impedance/admittance control and passivity by Hogan [29] and Colgate [30]. However, the first mention of admittance control concepts dates back to the literature by Whitney [31], where these concepts were used to address hard contact issues for industrial manipulation functions and indirect force control purposes. Overall, operational space force control has proven the effectiveness of RL-based assembly tasks [9], [13].

III. PROBLEM STATEMENT AND METHOD OVERVIEW

A. Programming with Mobile Manipulators

1) *Problem details:* Mobile manipulators can considerably reduce robots and devices installation time [32]. Programming based on demonstration approaches has been proposed to address variations in geometry and configurations for assembly, placement, handling, and picking tasks [32], which can reduce the programming time and user training requirements [3]. The use of mobile manipulators introduces a **positioning error at the ± 5 mm level** [33], and errors as small as ± 1 mm can induce large **huge contact forces and consistent failures** in typical assembly tasks [14]. In the present study, we address the more typical cases of mobile bases that involve the repositioning of mobile manipulators according to task requirements.

Teach pendants are still used for precision positioning (position and orientation of the end effector (EE) in many tasks [4]. However, these devices limit the intuitiveness of teaching processes and are time-consuming. Hand-guiding is a typical physical contact kinesthetic teaching solution, where programming is embodied using demonstration concepts, enabling users to quickly and intuitively program robots. However, it has drawbacks in terms of accuracy, locational separations, and operations involving dangerous objects [3]. Moreover, neither the hand-guiding nor the teach pendant

¹<http://std.samr.gov.cn/>

²<https://blog.robotiq.com/>

³www.kuka.com

⁴<https://mobile-industrial-robots.com/>

⁵www.kuka.com/en-de/products/robot-systems/industrial-robots/lbr-iiwa

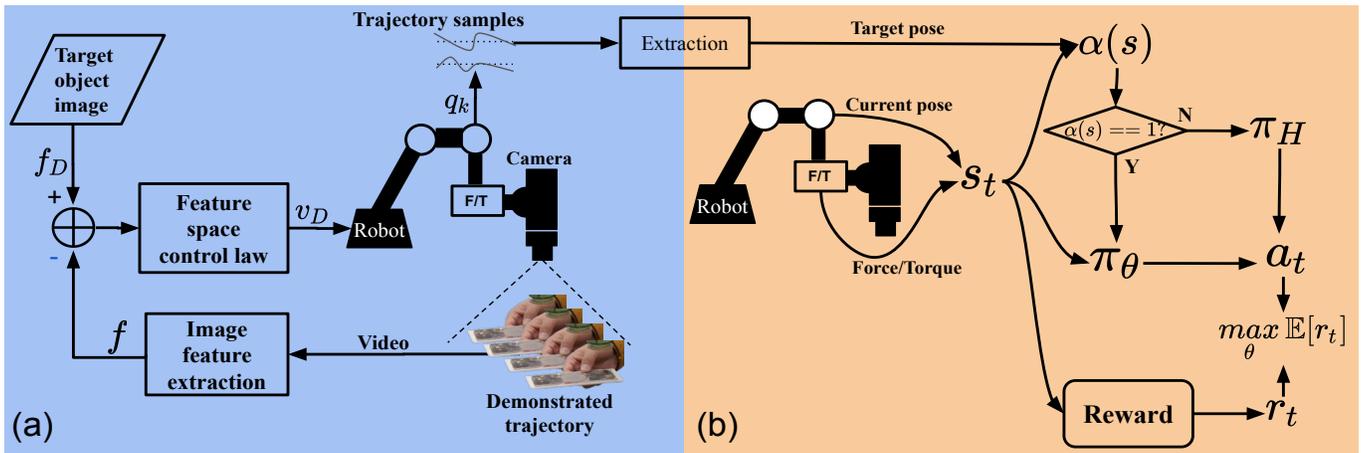


Fig. 3. Combination of LfD policy (a) and RRRL policy (b)

programming methods can compensate for the positioning errors that accompany mobile units [33], and can result in the generation of huge contact force that can damage objects [4].

Additionally, the base of a mobile manipulator requires considerable space within work cells. Figure 2 shows some real-world factory examples of constrained spaces. Sometimes, a user must teach a robot with a highly **awkward body posture** owing to anthropometric limitations [6]. Moreover, the delicate movement required by a user may be difficult to realize due to the resistance of robots in the drag mode [6]. Hence, because of these two issues, the use of hand-guiding for accurate assembly is rather difficult and yields low quality (e.g., excessive contact force and low accuracy).

2) *Method overview*: We propose a method that is simple and fast to implement and reduce the physical contact force to solve the problems described in Section III-A. We introduce visual guiding into the teaching phase as shown in Figure 3(a). The proposed visual guiding teaching is performed using two steps:

- **Grasping pose definition**: First, a user guides the robot to achieve a grasping pose under the target object frame. Then, the robot moves up and uses an eye-in-hand camera to capture a photo of the object as a reference.
- **Trajectory generation**: Second, the robot follows the moving object (the object can be moved by the user) to achieve a new target pose using a vision-based control algorithm (e.g., visual servoing [34], [35]), and records the entire moving trajectory.

Comparing to utilizing a global camera [36], the method proposed in this paper uses an eye-in-hand camera to effectively avoid the occlusion of the target, and ensure that the robot follows the object to achieve the proper trajectory, thus achieves better performance in real industrial scenarios. Trajectory errors have little effect on final assemblies because uncertainties can be generally ignored in gross motion planning, and a fine motion planner can solve uncertainties during the assembly process [37].

B. Assembly Under Position Uncertainty

1) *Problem details*: Visual sensors can be used for target recognition, pose estimation, measurement, and positioning using traditional methods [23]. However, for visual sensors, lenses, imaging sensors and the calibration of intrinsic/extrinsic parameters considerably influence the precision of visual guidance, as well as reflection, shadow, and occlusion may fail to extract the edges and features of objects owing to lights changes and object textures [23].

2) *Method overview*: Residual RL [14], [38] is a novel method that exploits the efficiency of conventional controllers and the flexibility of RL. Residual RL attempts to introduce prior information in an RL algorithm to accelerate the training process, rather than performing random explorations from scratch. For example, the estimated position can be set as prior information and even can have errors. The guided uncertainty-aware policy optimization (GUAPO) [27] showed better performance than residual RL, soft actor critic (SAC), and pure model-based method such as deep object pose estimator (DOPE). However, the force and torque information is not considered in the GUAPO policy. This information can provide observations regarding current contact conditions between objects and their environments for accurate localization [20]. Moreover, it can ensure manipulation safety [13]. However, pure force-based learning policies may lead to substantial deviations from goals and reduce learning efficiency. Thus, we combine the “region limitation” idea from GUAPO and the residual RL policy to develop a force-based approach called “**Region-limited Residual RL**” (RRRL) (Figure 3(b)). In the RRRL policy, the rough target pose is obtained using by the teaching phase as the residual part and a function $\alpha(s) = 1[s \in \mathbb{S}_u]$ is used to switch between the fixed $\pi_H(s)$ and the parametric $\pi_\theta(s)$ policies [27]:

$$\pi(a|s) = (1 - \alpha(s)) \cdot \pi_H(a|s) + \alpha(s) \cdot \pi_\theta(a|s). \quad (1)$$

\mathbb{S}_u is the region containing the goal position with uncertainty. Because force control is more safe and reliable in the fine motion/manipulation phase than position control and impedance control in assembly tasks [9], the RRRL

policy $\pi_\theta(s)$ takes the operational force controller as the desired force/torque in operational spaces. thus, our goal is to maximize expected return (i.e., $\max_\theta \mathbb{E}[r_t]$) through the RRRL policy. The fixed policy u_H is used to move the object back to the initial target pose when the function $\alpha(s) = 0$.

IV. POLICY DESIGN

The processes in LfD can be divided into three steps: observing, representing, and reproducing an operation [3]. We perform our entire policy following these processes, and then add the RRRL policy at the end of the operations. Our method comprises two learning policies:

- **Lfd policy:** The robot learns gross motions via human demonstrations, wherein a human demonstrates the target object images and grasping position to the robot.
- **RRRL policy:** The robot learns fine motions based on the RRRL policy, as described in Algorithm 1. The RRRL policy can be trained in advance to save the setup time.

First, we define the terminology and notation required to represent the coordinate transformations. We represent the task space of the robot as \mathcal{T} , which constitutes a set of positions and orientations that can be attained by the robot EE (i.e., suction gripper). \mathcal{T} is a smooth m -manifold in which $m = 6$ and $\mathcal{T} = SE^3 = \mathcal{R}^3 \times SO^3$. The superscripts/subscripts of the coordinate frames are listed in Table I.

TABLE I
COORDINATE FRAMES

e	Coordinate frame attached to the robot EE
c	Coordinate frame of the camera
b	Base coordinate frame of the robot
o	Coordinate frame attached to the target object

A. Lfd Policy

In our policy, we equip an eye-in-hand camera to avoid the occlusions caused by the robot's links and other industrial devices [3], [39] during the demonstration. The relative homogeneous transformation ${}^e\mathbf{x}_c$ and intrinsic camera parameters are determined by means of the hand/eye calibration method [40].

In the demonstration phase, teaching was categorized into the three following steps:

- 1) Robot EE was moved to the **desired grasping pose (DGP)**, which was then recorded as ${}^b\mathbf{x}_{DGP}$.
- 2) Robot EE was moved to the **desired visual servoing pose (DVSP)**, which was recorded as ${}^b\mathbf{x}_{DVSP}$; here, the object must be kept in view of the camera. The **first reference photo (RF1)** was captured, and a fixed **relative pose (RP)** was calculated as follows:

$${}^c\mathbf{x}_o = {}^{DVSP} \mathbf{x}_{DGP} = ({}^b\mathbf{x}_{DVSP})^{-1} ({}^b\mathbf{x}_{DGP}), \quad (2)$$

where ${}^c\mathbf{x}_o$ is the coordinate transformation of the object frame o with respect to the camera frame c .

Algorithm 1 RRRL

Require: Model based policy π_H , learning frequency C_1 , and target action-value update frequency C_2 .

- 1: Initialize replay memory \mathcal{H} to capacity N
 - 2: Initialize action-value function Q with random weights θ
 - 3: Initialize target action-value function Q_{target} with weights $\theta^- = \theta$
 - 4: **for** episode = 1 to M **do**
 - 5: Sample state s_0
 - 6: **while** NOT EpisodeEnd **do**
 - 7: Calculate $\alpha(s)$ using Equation (8)
 - 8: Select action a_H from $\pi_H(s_t)$
 - 9: With probability ϵ , select a random action a_{RL}
 - 10: Otherwise select $a_{RL} \sim \pi_\theta(s_t)$
 - 11: Obtain action $a_t = (1 - \alpha) * a_H + \alpha * a_{RL}$
 - 12: Execute a_t , and observe reward r_t and state s_{t+1}
 - 13: Store transition (s_t, a_t, r_t, s_{t+1}) in \mathcal{H} with priority $p_t = \max_{i < t} p_i$
 - 14: **for** $j = 1$ to C_1 **do**
 - 15: Sample minibatch of transitions with priority from \mathcal{H}
 - 16: Update transition priority
 - 17: Update θ using the method proposed in [41]
 - 18: **end for**
 - 19: In each C_2 step, reset $Q_{target} = Q$
 - 20: **end while**
 - 21: **end for**
-

3) The visual servoing strategy [35] was activated, and the following system constraints [42] were applied to the robot during the teaching process:

$$\mathbf{q} \in \mathbb{Q}_c, \quad (3)$$

$$\mathbf{q} \in [\mathbf{q}^{min}, \mathbf{q}^{max}], \quad \mathbf{q}^{min}, \mathbf{q}^{max} \in \mathbb{R}^N, \quad (4)$$

$$\dot{\mathbf{q}} \in [\dot{\mathbf{q}}^{min}, \dot{\mathbf{q}}^{max}], \quad \dot{\mathbf{q}}^{min}, \dot{\mathbf{q}}^{max} \in \mathbb{R}^N, \quad (5)$$

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \delta \dot{\mathbf{q}}_k, \quad (6)$$

where \mathbb{Q}_c is the set of configurations that do not cause any part of the arm to collide with obstacles that are difficult to model. Equation (4) and Equation (5) describe the robot's joint positions and velocity constraints, respectively. The object was then moved from the DVSP to the **desired final pose (DFP)** by the user, the robot EE followed the trajectory \mathbf{q}_k from the DVSP to the DFP under the aforementioned constraints. Further, the trajectory could be recorded. At DFP, the camera automatically captured the **second reference photo (RF2)**, and the DFP could be easily calculate at the end of the trajectory by Equation (7):

$$DFP = ({}^b\mathbf{x}_e)({}^e\mathbf{x}_c)({}^c\mathbf{x}_o) \quad (7)$$

In this study, the image-based visual servoing method was introduced based on specified observed feature positions (Figure 3(a)). With the human expert (user) in the teaching loop, the trajectory-based representation output using LfD

could avoid collisions even if the robot operated in a heavily constrained operation space. Additionally, our noncontact guiding method can avoid the resistance force of the robot in the drag mode, making it easier and more accurate for the users to perform teaching.

B. RRRL Policy

In this section, we explain the RRRL algorithm in detail as shown in Figure 3(b) and Algorithm 1 to learn the assembly tasks.

1) *Limited Region and Policy Structure*: In most assembly tasks, the aim is to minimize the distance between objects and their goal positions [13]. The limited region is used to constrain the exploration area. Many methods can describe the uncertainty region with respect to a nonparametric distribution [27] or a parametric equation. In this study, the Euclidean distance was used to indicate the switch signal between the fixed $\pi_H(s)$ and parametric $\pi_\theta(s)$ policies in the hybrid RRRL policy:

$$\alpha(s) = \begin{cases} 1, & \text{if } \|CP - DFP\|_2 < D \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

CP and DFP denote the current and desired final poses, respectively. D is an engineering hyperparameter determined based on experience, and we suggest that D should be at least twice the final positioning error introduced by the fixed policy $\pi_H(s)$. When $\alpha(s) = 1$, the **force-based** learning policy $\pi_\theta(s)$ is used, otherwise the **position-based** fixed policy $\pi_H(s)$ is used to the object back to its initial uncertain target pose. The same switch algorithm was used during the training and execution phases. A double DQN with proportional prioritization [41] was selected as the learning policy $\pi_\theta(s)$ in this study.

2) *Action Design*: The actions in the assembly task can be either a position [20], [27] or force/torque command [9], [13]. Because we aim to reduce the contact force between the object and environment to ensure safety, the force/torque command action in the operational space (i.e. under frame x_e) was selected [13].

Carefully exploiting the natural constraints in the design of the learning policy was essential for the assembly tasks considered herein. It is obvious that the task is simplified if the motion is constrained in “wrong” directions. We utilized comparative experiments to investigate the ways in which different force-based actions utilize natural constraints. We set the same initial positional error of $\delta P \in [2, 4]$ mm in a random direction, and then performed a random strategy to select the actions. We tested each action 200 times with a maximum steps of 20 and a maximum force amplitude of 10N, the success rates of which are as follows:

	Force control action	Success rate
1	Operational space controller [13]	32%
2	Fz with the force of another one dimension [9]	60%
3	Fz with the forces of other two dimensions	69%

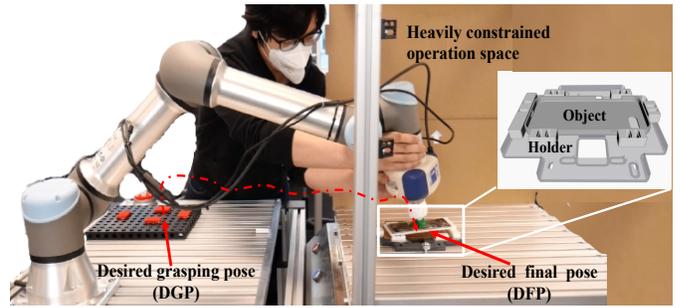


Fig. 4. Hand-guiding teaching in a heavily constrained operation space. Both the user and robot arms are close to their operating boundaries (near singular configurations).

The third discrete action as Equation (9) performs the best in the comparison experiment:

$$\begin{aligned} 1 &: [+^e F_x, +^e F_y, +^e F_z, 0, 0, 0] \\ 2 &: [+^e F_x, -^e F_y, +^e F_z, 0, 0, 0] \\ 3 &: [-^e F_x, +^e F_y, +^e F_z, 0, 0, 0] \\ 4 &: [-^e F_x, -^e F_y, +^e F_z, 0, 0, 0] \end{aligned} \quad (9)$$

Here, we set the orientation space as the position mode to utilize the flexibility of the suction cup. Then, all the torque commands in the operation space were set to 0 Nm, and all the force amplitudes were set to 10 N, which is half of force amplitudes set in [9].

3) *State Design*: Forces and torques feature the most direct information that characterizes contact states during an operation. Thus, the 6-dimensional force-torque vector $s = [F_x, F_y, F_z, M_x, M_y, M_z]$ under the robot EE’s frame x_e were sent to the RRRL network as the input state. The state sampling frequency was 10 Hz, which could guarantee the observation of the contact states.

4) *Reward Design*: We employed the precise target position of the hole as a reference for the reward during the learning phase. Unlike the execution phase, the precise desired position was easy to obtain because the robot exhibited high positional repeatability.

$$r = \begin{cases} 1 - k_{steps}/k_{max}, & \text{success} \\ -\|CP - DFP\|_2, & \text{otherwise.} \end{cases} \quad (10)$$

5) *Parameters Setup*: In order to gain more contact experience, a positional error $\delta P \in [2, 4]$ mm was added in a random direction during the training phase. The transitions (s_t, a_t, r_t, s_{t+1}) sampled from the environment were stored in a replay buffer [41]. The size of the experience’s replay memory P_{replay} was 20, 000, the maximum number of training episodes M was 200, and the maximum number of steps k_{max} for the search phase was 50. The batch size P_{batch} was set to 64 to select random experiences from P_{replay} , and the discount factor λ was 0.5.

V. EXPERIMENTS

We evaluated our methods in the **teaching** and **execution** phases. In our experiments in both phases, we aimed to answer the following questions: **(Q1)** Can our method maintain fast and easy programming abilities even in constrained

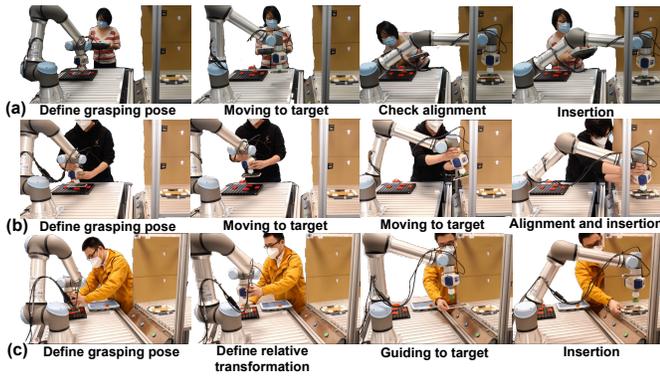


Fig. 5. (a) Teach-pendant teaching: aligning the object with the target holder by eye is difficult. (b) Hand-guiding teaching: moving the robot EE with singular configurations and align with target holder are difficult. (c) Our LfD method teaching: contact-free guiding that is not physically demanding, and is easy to align without robot resistance.

operation spaces? (Q2) Can our method retain the execution success rate against positional uncertainty? (Q3) Can our method reduce the risk of damage during operation of an object?

A UR5e robot⁵ was used to implement our novel approach to facilitate a tending task. The UR5e features a 6-axis and 5 kg payload, a working radius of 850 mm. It is equipped with a 6-DOF force/torque sensor on the EE. UR5e robot uses admittance controller [29] to achieve operational space force control. A Schmalz CobotPump ECBPi suction cup was installed beyond the force/torque sensor in order to ensure the detection of the contact force with the environment. An Intel RealSense Depth Camera D435i was attached to the EE to conduct the visual servoing process. Our policy was run using a Dell Precision 5510 laptop, and the updated position was sent to the UR5e controller. We used ur-rtde⁶ as the Python interface for controlling and receiving data from the UR robot. A 6-DoF ATI Axia80 force sensor⁷ was mounted under the holder in order to measure the operating force, and a low-pass filter with a cutoff frequency of 9.37 Hz was used to mitigate the force noise.

A. Baseline Techniques

The commonly available methods for collaborative robots were selected as baselines [6]. We compared our proposed method with the following baselines:

Baseline 1: Teach-pendant + spiral searching. The UR5e teach pendant with a UR PolyScope GUI was held in one hand by a user, who pressed the on-screen buttons to map the rate control of the EE’s translation and rotation in the task space \mathcal{T} with the other. At the DFP, a spiral search function similar to that outlined in the literature [43] was added.

Baseline 2: Hand-guiding + spiral searching. The UR5e “Freedrive” mode was utilized, and users physically grabbed and exerted force to move the robot arm using one or two hands. At the DFP, a spiral search function was also added.

⁵<https://www.universal-robots.com/products/ur5-robot/>

⁶<https://pypi.org/project/ur-rtde/>

⁷<https://www.ati-ia.com/Products/ft/sensors.aspx>

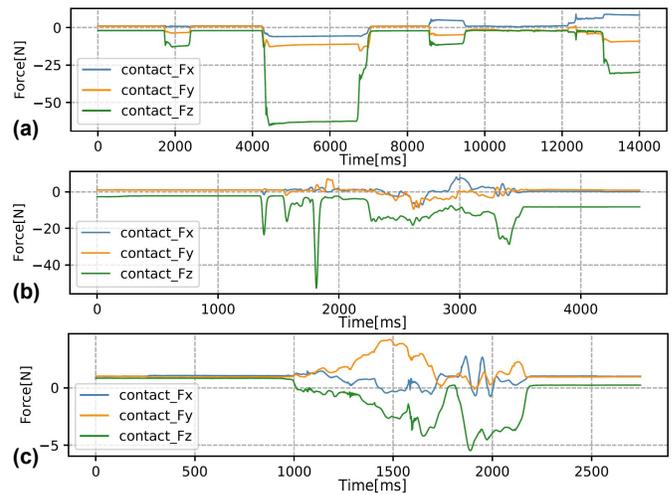


Fig. 6. Contact force during teaching. (a) This curve shows the contact characteristics of the teach-pendant teaching method: the user uses an “observe-move” strategy and only makes adjustment after finding unsuccessful insertions, hence, the contact force is maintained during the observation. (b) The hand-guiding method: the object frequently collides with the holder when it is not inserted (impact force in the figure), and there is a continuous contact force after inserting the object. (c) Our method produces a small contact force (less than 5 N) in the teaching phase.

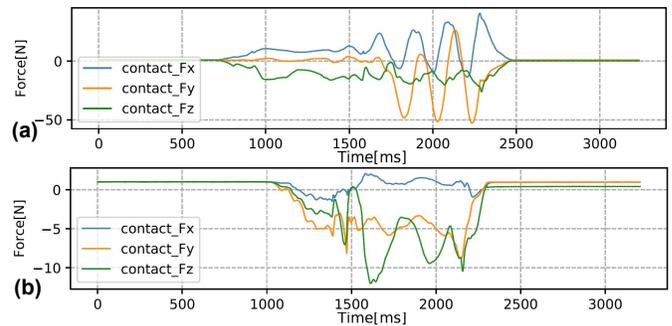


Fig. 7. Contact force during execution. (a) Spiral exploration method: the first half of the curve (700-1500ms) is not yet constrained by the holder, and the contact force is small and stable. After 1600ms, the object is inserted into the holder, immediately generating a larger contact force. Due to the accuracy issue of the UR5e force sensor, the spiral movement generates a larger contact force than the stopping threshold. (b) RRRL method: the maximum contact force is around 10N because of the force action amplitude limitation.

B. Task Setup

A tight clearance machine tending task (similar to the assembly task) was used to evaluate our method and the two baselines (Figure 4). One holder was installed in an opaque box to simulate a situation in which the field of view in the production line is obscured and an object is inserted into the holder. The users were provided a brief tutorial and allowed to practice until they felt ready. A group of four able-bodied volunteers (1 female, 3 male, aged: 24 to 35) participated in the experiment.

C. Measurement and Evaluation

- **Teaching time:** The task’s completion time was measured as the time costed by the user to move the robot from the DGP to the DFP (Table II).
- **Execution success rate:** After the teaching phase was

TABLE II
EVALUATION IN THE TEACHING PHASE

Teaching phase	Time cost	Maximum contact force
Teach-pendant	60–120 s	15–50 N
Hand-guiding	15–42 s	30–60 N
Our method	23–30 s	3–10 N

TABLE III
EVALUATION IN THE EXECUTION PHASE

Execution phase	Success rate		Maximum contact force
	Perfect	Uncertainty	
Only teach-pendant	55/100	17/100	15 N
Only hand-guiding	33/100	5/100	15 N
Teach-pendant + spiral searching	69/100	47/100	35 N
Hand-guiding + spiral searching	51/100	33/100	35 N
Our method	95/100	91/100	15 N

completed, the trajectory of the demonstration was executed and the insertion success rate was tested. Contrary to the “**perfect**” group, an error of $\delta P \in [2, 4]$ mm in a random direction was added on DFP to simulate the pose uncertainties in the “**uncertainty**” group (Table III).

- **Risk of damage:** The risk of damage caused by the operating force was ignored in previous studies [12], [26], [13], [27], [43]. In this study, the maximum absolute contact force during contact operation was employed to evaluate the risk of damage (Table III).

VI. RESULTS AND DISCUSSIONS

Overall, the results of 12 group robot teaching and 1000 group robot execution were obtained.

The teaching phase test scenarios can be observed in Figure 5, and the results are presented in as Table II. Although it is more generalized, our method features a similar time cost to the hand-guiding method. Using the hand-guiding method, male volunteers always required less teaching time than females owing to physical demands. All the volunteers noted that the robot “required excessive force to move, particularly near the boundaries”. Our method does not require physical contact with the robot, therefore, it is not physically demanding. All the volunteers required considerably longer teaching times when the teach-pendant teaching method was used. considerable amount of time was spent to align the object with the target, and the tricky sight angle made this even more difficult. In contrast, our method does not require a human to align the object with the target, but enables the robot arm to actively track the object to attain the target position. Thus, the setup is performed quickly. Hence, the answer to **Q1** is yes. Furthermore, our method produces minimal contact force on the environment (Figure 6), as our schematic approach is identical to the

human tending one. The other two methods resulted in considerably less transparency during the interactions with the environment owing to the resistance of the robot arm itself or the inability to interact with the environment in terms of force.

The execution phase results are presented in Table III. The success rate of our method far exceeded those of the other reference baselines. We found that **the elasticity of the suction cup** had a major influence on the accuracy of the demonstrated target position. The contact force at the end of the demonstration could induce the **deformation of the suction cup** and thus affect the actual target position. Using our method, the robot EE did not contact the target environment at the end of the teaching phase; hence, there was no contact force and therefore no deformation of the suction cup. Finally, high target position accuracy was achieved. Additionally, the RRRL policy could determine force actions based on the contact state of the object and holder, thus greatly improving the insertion success rate. Our method can guarantee small contact forces owing to the amplitude limitation of the force actions (fig. 7(b)). Therefore, the answers to **Q2** and **Q3** are also yes.

VII. CONCLUSIONS

Collaborative robots are rapidly making advances in new work environments. However, there is a control and teaching bottleneck in the programming of contact-rich tasks.

In this study, we combined visual servoing based LfD and force-based LbE to facilitate the rapid and intuitive execution of the assembly tasks that require minimal user expertise, involvement, and physical exertion. The efficiency of the proposed method was validated via a series of experiments that involved the execution of a tending task using a robot arm and a suction cup system.

In a challenging setting designed to simulate heavily constrained operation spaces, which is very common in actual factories, experiments that compared our method with two commonly used baselines, namely teach pendant-based and hand-guiding teaching, were performed. Our method realized the best feedback in terms of both subjective and objective evaluations

We noted the negative effect of the elasticity of the suction cups on the accuracy of the position demonstrations. In the future, we hope to investigate ways to utilize this elasticity using a learning approach. We also plan to analyze more contact-rich tending tasks and more types of grippers to refine our method and improve its generalizability.

ACKNOWLEDGMENT

We would like to thank Chengjie Yuan, Chunyang Chang, Jingjing Sun, and Alexander Daimer for their contributions as volunteers and for their valuable feedback during the discussion of the experimental protocol. We also thank mechanical engineer Ningxin Lu for building the hardware mockup system.

REFERENCES

- [1] A. Kramberger, B. Nemeč, M. Gams, A. Ude *et al.*, “Learning of assembly constraints by demonstration and active exploration,” *Industrial Robot: An International Journal*, 2016.
- [2] C. Sloth, A. Kramberger, and I. Iturrate, “Towards easy setup of robotic assembly tasks,” *Advanced Robotics*, vol. 34, no. 7-8, pp. 499–513, 2020.
- [3] Z. Zhu and H. Hu, “Robot learning from demonstration in robotic assembly: A survey,” *Robotics*, vol. 7, no. 2, p. 17, 2018.
- [4] M. Safeea, R. Bearee, and P. Neto, “End-effector precise hand-guiding for collaborative robots,” in *Iberian Robotics conference*. Springer, 2017, pp. 595–605.
- [5] J. Marvel and R. Bostelman, “Towards mobile manipulator safety standards,” in *2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*. IEEE, 2013, pp. 31–36.
- [6] J. H. Lee, Y. Kim, S.-G. An, and S.-H. Bae, “Robot telekinesis: application of a unimanual and bimanual object manipulation technique to robot control,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9866–9872.
- [7] J. A. Marvel, R. Bostelman, and J. Falco, “Multi-robot assembly strategies and metrics,” *ACM Computing Surveys (CSUR)*, vol. 51, no. 1, pp. 1–32, 2018.
- [8] A. Albu-Schäffer, C. Ott, and G. Hirzinger, “A unified passivity-based control framework for position, torque and impedance control of flexible joint robots,” *The international journal of robotics research*, vol. 26, no. 1, pp. 23–39, 2007.
- [9] T. Inoue, G. De Magistris, A. Munawar, T. Yokoya, and R. Tachibana, “Deep reinforcement learning for high precision assembly task,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 819–825.
- [10] N. Lin, L. Zhang, Y. Chen, Y. Zhu, R. Chen, P. Wu, and X. Chen, “Reinforcement learning for robotic safe control with force sensing,” in *2019 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*. IEEE, 2019, pp. 148–153.
- [11] C. Gaz, M. Cognetti, A. Oliva, P. R. Giordano, and A. De Luca, “Dynamic identification of the franka emika panda robot with retrieval of feasible parameters using penalty-based optimization,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, 2019.
- [12] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, “Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks,” *arXiv preprint arXiv:1810.10191*, 2018.
- [13] J. Luo, E. Solowjow, C. Wen, J. A. Ojea, A. M. Agogino, A. Tamar, and P. Abbeel, “Reinforcement learning on variable impedance controller for high-precision robotic assembly,” *arXiv preprint arXiv:1903.01066*, 2019.
- [14] G. Schoettler, A. Nair, J. Luo, S. Bahl, J. A. Ojea, E. Solowjow, and S. Levine, “Deep reinforcement learning for industrial insertion tasks with visual inputs and natural rewards,” *arXiv preprint arXiv:1906.05841*, 2019.
- [15] E. Helms, R. D. Schraft, and M. Hagele, “rob@ work: Robot assistant in industrial environments,” in *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*. IEEE, 2002, pp. 399–404.
- [16] M. Hvilshøj, S. Bøgh, O. Madsen, and M. Kristiansen, “The mobile robot little helper: concepts, ideas and working principles,” in *2009 IEEE Conference on Emerging Technologies & Factory Automation*. IEEE, 2009, pp. 1–4.
- [17] C. Wurrll, T. Fritz, Y. Hermann, and D. Hollnacher, “Production logistics with mobile robots,” in *ISR 2018; 50th International Symposium on Robotics*. VDE, 2018, pp. 1–6.
- [18] N. Krüger, A. Ude, H. G. Petersen, B. Nemeč, L.-P. Ellekilde, T. R. Savarimuthu, J. A. Rytz, K. Fischer, A. G. Buch, D. Kraft *et al.*, “Technologies for the fast set-up of automated assembly processes,” *KI-Künstliche Intelligenz*, vol. 28, no. 4, pp. 305–313, 2014.
- [19] T. Davchev, K. S. Luck, M. Burke, F. Meier, S. Schaal, and S. Ramamoorthy, “Residual learning from demonstration: adapting dynamic movement primitives for contact-rich insertion tasks,” *arXiv preprint arXiv:2008.07682*, 2020.
- [20] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, “Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.
- [21] Y. Zheng, X. Zhang, Y. Chen, and Y. Huang, “Peg-in-hole assembly based on hybrid vision/force guidance and dual-arm coordination,” in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 418–423.
- [22] Y. Huang, X. Zhang, X. Chen, and J. Ota, “Vision-guided peg-in-hole assembly by baxter robot,” *Advances in Mechanical Engineering*, vol. 9, no. 12, p. 1687814017748078, 2017.
- [23] R. Li and H. Qiao, “A survey of methods and strategies for high-precision robotic grasping and assembly tasks: some new trends,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2718–2732, 2019.
- [24] A. Al-Zabt and T. A. Tutunji, “Robotic arm representation using image-based feedback for deep reinforcement learning,” in *2019 IEEE Jordan International Conference on Electrical Engineering and Information Technology (JEEIT)*. IEEE, 2019, pp. 168–173.
- [25] R. Bogue, “The role of artificial intelligence in robotics,” *Industrial Robot: An International Journal*, 2014.
- [26] J. Luo, E. Solowjow, C. Wen, J. A. Ojea, and A. M. Agogino, “Deep reinforcement learning for robotic assembly of mixed deformable and rigid objects,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2062–2069.
- [27] M. A. Lee, C. Florensa, J. Tremblay, N. Ratliff, A. Garg, F. Ramos, and D. Fox, “Guided uncertainty-aware policy optimization: Combining learning and model-based strategies for sample-efficient policy learning,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 7505–7512.
- [28] Q. Feng, Z. Chen, J. Deng, C. Gao, J. Zhang, and A. Knoll, “Center-of-mass-based robust grasp planning for unknown objects using tactile-visual sensors,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 610–617.
- [29] N. Hogan, “Impedance control: An approach to manipulation: Part i: theory,” 1985.
- [30] J. E. Colgate and N. Hogan, “Robust control of dynamically interacting systems,” *International journal of Control*, vol. 48, no. 1, pp. 65–88, 1988.
- [31] D. E. Whitney, “Force feedback control of manipulator fine motions,” 1977.
- [32] E. Matheson, R. Minto, E. G. Zampieri, M. Faccio, and G. Rosati, “Human–robot collaboration in manufacturing applications: A review,” *Robotics*, vol. 8, no. 4, p. 100, 2019.
- [33] S. Su, X. Zeng, S. Song, M. Lin, H. Dai, W. Yang, and C. Hu, “Positioning accuracy improvement of automated guided vehicles based on a novel magnetic tracking approach,” *IEEE Intelligent Transportation Systems Magazine*, 2018.
- [34] D. Kragic, H. I. Christensen *et al.*, “Survey on visual servoing for manipulation,” *Computational Vision and Active Perception Laboratory, Fiskartorpsv*, vol. 15, p. 2002, 2002.
- [35] S. Hutchinson, G. D. Hager, and P. I. Corke, “A tutorial on visual servo control,” *IEEE transactions on robotics and automation*, vol. 12, no. 5, pp. 651–670, 1996.
- [36] Q. Wei, C. Yang, W. Fan, and Y. Zhao, “Design of demonstration-driven assembling manipulator,” *Applied Sciences*, vol. 8, no. 5, p. 797, 2018.
- [37] S. Gottschlich, C. Ramos, and D. Lyons, “Assembly and task planning: A taxonomy,” *IEEE Robotics & Automation Magazine*, vol. 1, no. 3, pp. 4–12, 1994.
- [38] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine, “Residual reinforcement learning for robot control,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6023–6029.
- [39] V. Lippiello, B. Siciliano, and L. Villani, “Position-based visual servoing in industrial multirobot cells using a hybrid camera configuration,” *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 73–86, 2007.
- [40] R. Y. Tsai, R. K. Lenz *et al.*, “A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration,” *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 345–358, 1989.
- [41] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, “Prioritized experience replay,” *arXiv preprint arXiv:1511.05952*, 2015.
- [42] A. Chan, E. A. Croft, and J. J. Little, “Constrained manipulator visual servoing (cmvs): Rapid robot programming in cluttered workspaces,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 2825–2830.
- [43] H. Park, J. Park, D.-H. Lee, J.-H. Park, and J.-H. Bae, “Compliant peg-in-hole assembly using partial spiral force trajectory with tilted peg posture,” *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4447–4454, 2020.