IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 25, 2019, accepted August 1, 2019, date of publication August 9, 2019, date of current version January 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2934174

Manipulation Skill Acquisition for Robotic Assembly Based on Multi-Modal Information Description

FENGMING LI[1], (Member, IEEE), QI JIANG[1], WEI QUAN[1], SHIBO CAI[2], RUI SONG[1], (Member, IEEE), AND YIBIN LI[1], (Member, IEEE)

[1]School of Control Science and Engineering, Shandong University, Jinan 250061, China
[2]School of Mechanical Engineering, Zhejiang University of Technology, Hangzhou 310023, China

Corresponding author: Rui Song (rsong@sdu.edu.cn)

This work was supported in part by the Integration Fund Project of China NSF and Zhejiang province, China, under Grant U1509212, in part by the Major Program of Shandong Province Natural Science Foundation, China, under Grant ZR2018ZC0437, and in part by the Key Research and Development Plan of Shandong Province, China, under Grant 2017CXGC0915.

ABSTRACT Automatic assembly of elastic components is difficult because of the potential deformation of parts during the assembly process. Consequently, robots cannot adapt their manipulation to dynamic changes. Designing systems that learn assembly skills can help in alleviating the uncertain factor for industrial-grade assembly robots. This study proposes a skill acquisition method based on multi-modal information description to realize the assembly of systems with elastic components. This multi-modal information includes two-dimensional images, poses, forces/torques, and robot joint parameters. In this method, robots acquire searching, location determination, and pose adjustment skills using these multi-modal information parameters. As a result, robots can reach the assembly target by analyzing two-dimensional images with no position constraint. While acquiring pose adjustment skills, the reward function with depth and assembly steps is used to improve the learning efficiency. The deep deterministic policy gradient (DDPG) algorithm is applied for acquiring skills. Experiments using a KUKA iiwa robot demonstrated the effectiveness and conciseness of our method. Our results indicate that the robot acquired searching, location determination, and pose adjustment skills that allowed it to successfully complete elastic assembly.

INDEX TERMS Industrial robots, acquisition of manipulation skills, deep reinforcement learning, multi-modal information description.

I. INTRODUCTION

A. MOTIVATION

Robots are used for more efficient production assembly. Research on automation of assembly systems has focused on intelligent, automatic, and flexible robots. During an actual assembly process, force measurement data are not available owing to elastic deformation between workpieces. Elastic deformation can be neglected when dealing with assembly of rigid parts. However, in the case of weakly rigid components, such deformation causes a series of uncertain changes, such as position, posture, and depth changes. There are methods based on force-control and compliance control that are already facing the problem of assembly of elastic components, to compensate for assembly uncertainties. The assembly state is difficult to measure, and reliable estimation of parametric coupling is extremely difficult. The randomness and nonlinearity are particularly obvious during the assembly process. The success of assembly is closely related to the coordination strategy, contact mechanism, and control strategy. Requirements are especially demanding for robotic manipulation.

Force and position information is usually needed for successfully addressing the robotic assembly problem. Traditional robotic assembly manipulation is realized using a force/position hybrid control and impedance control [1]–[6]. Most methods are suitable for the assembly of rigid parts. These methods require the ability to accurately measure the pose, and they also require a detailed knowledge of the system model. However, in a system with low-stiffness components

The associate editor coordinating the review of this article and approving it for publication was Okyay Kaynak.

and parts, deformations make measurements and modeling uncertain. Loris et al. have extensively studied the impedance control issue [1]–[3]. In [2], a rigid robot base was considered, and the system learned to tune the robot controller for assembly of big elastic parts. In [3], a model-free adaptive controller was proposed that takes into account a compliant robot base. Nowadays, acquisition of manipulation skills has become the mainstream method to solve this problem. Traditional methods mainly rely on vendor-specific robot programming languages [7]. In this method, classical programming is used to define position and orientation using the ''teaching pendant'' paradigm, enabling robots to complete assembly tasks. However, this method is usually tedious and time-consuming. In particular, when a robot faces a new assembly environment, it requires a long time to tune its parameters, even after programming. Experienced staff are needed to program these tools. This method also increases the cost of automation, especially in complex flexibility assembly tasks of low-stiffness components.

Motivated by data-driven reasoning, learning of manipulation skills has become an important research area. Machine learning methods are applied for acquisition of manipulation skills by robots. Many academic and industrial leaders, such as Deep Mind [8], [9], Open AI [10], [11], University of California Berkeley [12], [13] and Google Brain [14], have strongly contributed to the development of methods for robotic learning of manipulation skills. However, robotic acquisition of manipulation skills in the context of assembly remains very challenging. In particular, it is critically important to delineate suitable assembly parameters to describe the assembly procedure.

B. RELATED WORK

Learning of assembly manipulation skills typically involves several machine-learning algorithms. Collection of assembly state data is very challenging. The data generation method determines the specific method toward robotic learning of skills. Two methods have been identified: 1) expert provision and 2) generation of interactions with environment.

By imitating the given expert data for learning manipulation skills, the complexity of the robot's searching strategy space can be reduced. An important method involves human demonstrations [19], [20]. Existing human demonstration systems emphasize the extraction of information from human actions or operational objects. A human skill demonstration platform has been built, which used multiple cameras for object recognition and data gloves to capture human motion [17]. In [18], Yang et al. proposed a new learning framework. Assembly skills were acquired based on human demonstration. The objects and the human motion during the demonstration were important for learning the assembly task. However, skill representation, trajectory alignment, and skill segmentation could not be addressed satisfactorily using such skill transfer modeling. In addition, the proposed method was tedious.

Assembly data can be generated through interactions using reinforcement learning. In the reinforcement learning-based robotic skill acquisition method, a robot interacts with its environment by trial and error, and learns the optimal manipulation skill strategy by maximizing the overall reward. Reinforcement learning has been previously applied to manipulation learning [21]–[24]. Recently, much research has focused on using deep reinforcement learning to make robots play building block games [25], [26]and significant advances were made. In addition, significant amount of work has been done on robot grasbing [27], [28], door opening [29], navigating [30]. Schulman *et al.* [31] proposed the trust region policy optimization algorithm, which has been successfully applied to the robotic learning of operation skills in virtual scenes. However, additional engineering is required in real-world applications of reinforcement learning, such as determining the representation for the policy or value function [32]. Inoue et al. used deep reinforcement learning for peg-hole-based high-precision assembly [33] in the deep Q-learning framework, which can handle discrete action spaces. A deep Q-learning network was used to model the learning process of assembly skills [15]. Hou et al. proposed the knowledge-driven deep deterministic policy gradient algorithm for robotic multiple peg-in-hole assembly tasks [34]. The present work deals with the peg-hole-based assembly of rigid parts. The searching and insertion phases are discontinuous. The agent observes the current state of the system, parameterized in terms of the force/moment/position and angle. In the present work, the robotic motion parameters were also monitored in the observation state. Although the data dimensionality is relatively high, it allows to better describe the current assembly state.

Unlike the above-described previous work, the current study has extended the treatment to the learning of complex manipulation policies from assembly without user-provided demonstrations. Location was estimated based on visual guidance. After contact, assembly was completed through pose adjustment. To demonstrate the effectiveness of the proposed method, we considered learning fastening assembly skills of a circuit breaker. Description of the assembly state increased the number of robotic motion-related parameters. Continuous control with deep reinforcement was used for successful robotic acquisition of assembly skills.

C. PAPER CONTRIBUTION

In this paper, a skill acquisition framework based on multi-modal information description and deep reinforcement learning is proposed. The main contributions of this paper are as follows:

- In addition to the force and pose, the motion parameters of the manipulator are added to represent the assembly contact state, and these parameters include joint angles and torque during assembly.
- A learning framework with multi-modal information description is proposed for autonomous acquisition of assembly skills, and the framework includes estimation

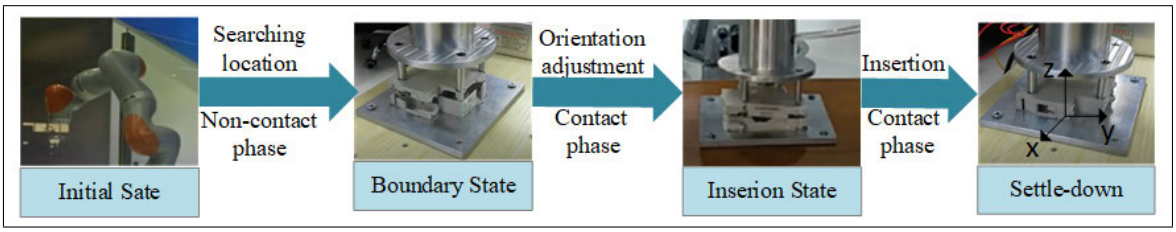


FIGURE 1. The fasten assembly process.

of object position during assembly and accounting for elastic deformations during assembly.

- In the proposed framework, robots acquire position searching and posture adjustment skills through visual guidance and multi-modal information learning based on the deterministic policy gradient algorithm.
- The proposed method was verified using a KUKA iiwa robot with seven degrees of freedom, on a plastic fastening assembly task, and the assembly task was successfully completed after learning.

D. PAPER STRUCTURE

This remainder of this paper is organized as follows. Section II introduces the assembly system and formulates the problem to be solved. Section III contains the description of the proposed method. Experiments were performed to validate the proposed method, and the experimental results are presented and discussed in Section IV. Finally, in Section V we summarize the results of the current work and discuss future directions.

II. SYSTEM OVERVIEW

A. ASSEMBLY SYSTEM

The problem of the plastic fastening assembly is studied here. Two phases (non-contact and contact) and four states (initial, boundary, insertion, and settle-down) completely characterize the assembly process, as shown in Fig.1. The assembly process is described in terms of the assembly object state, robotic motion state, and end effort state, which is defined as the assembly state. In the non-contact phase, the pose of the assembly target is estimated by visual guidance, so that the upper cover can move quickly to the base. When contacting the assembly base, the pose of the cover part is changed, for insertion based on force sensors. The contact state is important for assembly. Jasim and Plapper [35] used Gaussian mixture models based on expectation maximization to identify the contact resistance state with a spiral search path. A cylindrical shaft-hole assembly experiment was successfully applied to a KUKA robot to validate the algorithm effectiveness. Huang *et al.* [36] presented a visual compliance strategy to deal with the problem of fast peg-and-hole alignment with large position and attitude uncertainties. Wan et al. trained robots to perform object assembly tasks using multi-modal three-dimensional (3D) vision [37]. In our previous work [38], [39], force/torque signals were used to represent the contact state. In the present paper, the position/posture, the force/torque of the end effector and the robot joint parameters were adopted to describe the contact state.

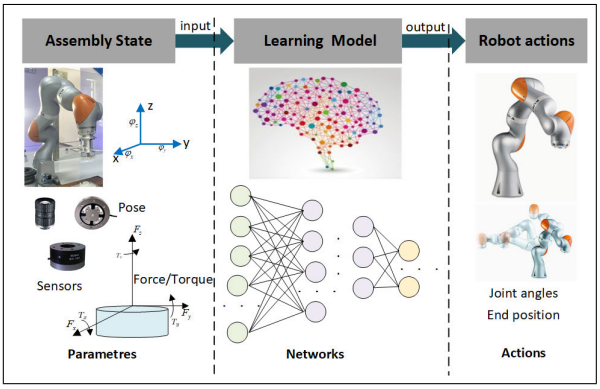


FIGURE 2. The problem to be solved.

B. PROBLEM SETUP

In addition to their irregular shapes and small internal parts, plastic components' shells are likely to deform in the insertion assembly. When the assembled plastic shells interact, small changes in their relative positions create large contact forces. The assembly components might be damaged owing to the imprecise pose. Compared with the peg-hole task [33], [34], the object complexity increases the difficulty of robotic skills acquisition. Given the above, the main problem to be solved is to make robots acquire skills of object location determination and pose adjustment. As shown in Fig.2, the overall problem is to determine an appropriate description and acquire skills using learning methods, to enable flexible robotic manipulation during different assembly stages. In general, object localization can be treated as the posture calculation problem in the complex assembly task in the non-contact phase. When the positions of the assembled parts change, the robot can still reach the assembled parts quickly. This process for vision-based robotic reaching usually relies on an RGB image acquired by a camera. The robot is given time for the pre-insertion manipulation, or whether it immediately moves into a good position. The robot has the ability to adjust the posture to ensure the completion of insertion during assembly, notwithstanding elastic deformations. The goal of the pose adjustment skill acquisition is to find a mapping function f between the assembly state s and the robot actions a. It is difficult to describe the fastening assembly process using a physical model. Thus, we considered the mapping function as

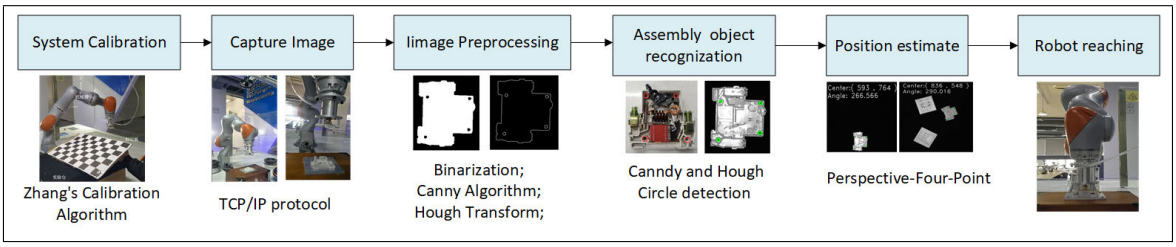


FIGURE 3. The step to acquire object searching located skill.

an unknown mathematical model ($s = f(a)$) which the system has to learn. This method entails solving some problems, such as determining suitable parameters for describing the assembly process, choosing the robot actions, evaluating the success of the assembly process, and training the network.

In our method, robots acquire object localization skills based on the perspective-n-point in vision guiding and pose adjustment using deep reinforcement learning. As a result, industrial robots are able to handle unforeseen events in unstructured environments after learning manipulation skills.

III. METHODOLOGY

This section describes object searching and localization, as well as pose adjustment methods, in the context of skill acquisition in fastening assembly tasks. The robot learns skills based on the multi-modal parametric description (e.g., images for the visual modality, force for the tactile modality), and joint information representations. The proposed skill acquisition framework allows to alleviate uncertainty factors during the assembly process, including assembly object localization and elastic deformation during assembly.

A. OBJECT SEARCHING AND LOCALIZATION

During the non-contact phase of assembly, visual modality plays a crucial role. A robot can recognize and localize objects to be assembled and quickly reach the target quickly using vision-based guiding. We define the associated skill set as object searching and localization skill. The proposed method for acquiring this skill is shown in Fig.3. The steps for acquiring the searching and localization skill are as follows:

1) The camera, the object of interest, and the robot are in the same Cartesian system of coordinates. camera parameters and rotation matrix are used for object localization [40], [41].

2) Thresholding and morphological closure operations are used for image preprocessing.

3) Feature extraction using the Canny and Hough-transform-based circle detectors is employed for detection of assembly objects [42], [43].

4) Pin-hole imaging is used for position estimation [44].

5) Reach Planning on the robot.

B. POSE ADJUSTMENT SKILL LEARNING

After a robot successfully contacts objects for assembly, adjusting the robot's pose is critical for successful assembly. To address the uncertain factors owing to the components' elastic deformation, a skill learning method based on deep reinforcement learning is proposed here. The key ingredients of the reinforcement learning setup include environment, observations (state space), action space and reward design, i.e. quadruple (s, a, r, s'). The state variable s represents the state of the system after action a. The system's environment can be seen as an assembly system, which consists of an actuator of a 7 degrees of freedom manipulator (a KUKA iiwa robot), and assembly components. Note that the environment may be stochastic. The description of the assembly state is the first step in perceiving the environment, especially in the contact phase. In this paper, we propose multi-modal parameters for describing the assembly state. The current state features space s including the angles $(\theta_1, \theta_2 \cdots \theta_7)$ and torque $(\tau_1, \tau_2 \cdots \tau_7)$ of seven joints, and the end-effector position and orientation $(p_x, p_y, p_z, \alpha, \beta, \gamma)$, the contact force/torque $(F_x, F_y, F_z, T_x, T_y, T_z)$. The axis of the robot base coordinate system are x, y and z. The state s_t is described by a 26-dimensional vector

$$s_t = \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6, \tau_7, p_x, p_y, p_z, \alpha, \beta, \gamma, F_x, F_y, F_z, T_x, T_y, T_z\} \quad (1)$$

The 7-dimensional continuous action vector a_t is defined as

$$a_t = \{\Delta\theta_1, \Delta\theta_2, \Delta\theta_3, \Delta\theta_4, \Delta\theta_5, \Delta\theta_6, \Delta\theta_7\} \quad (2)$$

where Δ is the joint angle offset. The algorithm starts with a random exploration of actions. The overall reward is defined as the sum of discounted future rewards

$$R_t = r_k + \lambda r_{k+1} + \lambda^2 r_{k+2} + \cdots + \lambda^{n-k} r_n = r_k + \lambda R_{k+1} \quad (3)$$

where the discounting factor $\lambda \in [0, 1]$, r is the current reward assigned to the action, k is the step number. The reward depends on the actions chosen. In the proposed method, one reward is computed at the end of an episode. The learning framework is shown in Fig.4.

1) REWARD FUNCTION

In fastening assembly, the condition for successful assembly is defined relative to the displacement and force along the z axis, as shown in Eq.(4).

$$f_{min} \leq |f_z| \geq f_{max} \quad and \quad z \geq l + z_0 \quad (4)$$

where f_z is the Z-axis force, f_{min} and f_{max} is threshold value of force. l is assembly depth along the z-axis, z_0 is distance from the target at initial position, z the current displacement

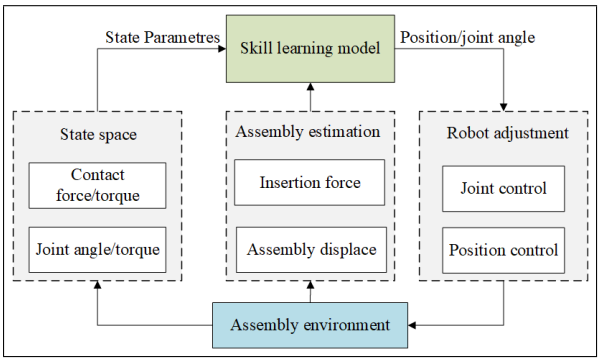


FIGURE 4. Framework of learning the pose adjustment skill.

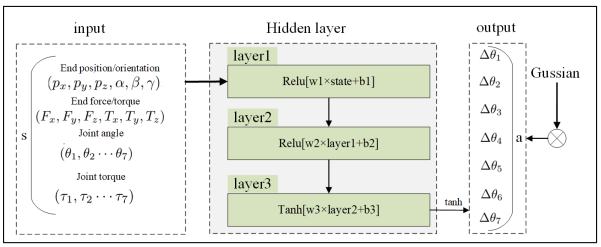


FIGURE 5. Actor-network architecture.

along the z axis. For successful assembly, the positive reward r is defined as:

$$r = 1 - \frac{step}{step_{max}} \tag{5}$$

where $step_{max}$ is the maximal number of steps in each episode, and $step \in (0, step_{max})$. From Eq.5, we can see that the learning target, which is objective of learning, is to successfully perform the task using a minimal number of steps. If the assembly tasks cannot be finished, the negative reward is defined as:

$$r = -\frac{\rho \mid (p_{z_0} - p_z) \mid}{p_{z_0}} \tag{6}$$

where ρ is the balance coefficient and $\rho > 0$, p_{z_0} is the position along the z axis when the assembly is successful, i.e. the insertion depth. p_z is the current position along the z axis. The reward takes on values within the range $-1 \leq r \leq 1$. No task can be finished in zero steps, in addition $r_{max} < 1$. If the cover is stuck at the entry of the base bottom, then $r_{min} = -1$. A suitable reward function is often non-obvious and may require considerable effort and experimentation.

2) NETWORK ARCHITECTURE

The goal is to maximize the cumulative reward as defined by Eq.(3). The variant reinforcement learning in the actor-critic framework(Fig.5and Fig.6) was used, which is based on the deep deterministic policy gradient (DDPG) algorithm [8]. It can realize robot continuous action space tasks. It can realize robot continuous action space tasks. The algorithm features a critic network and an actor network, both of which include a target-net and an eval-net. The actor network maps states to

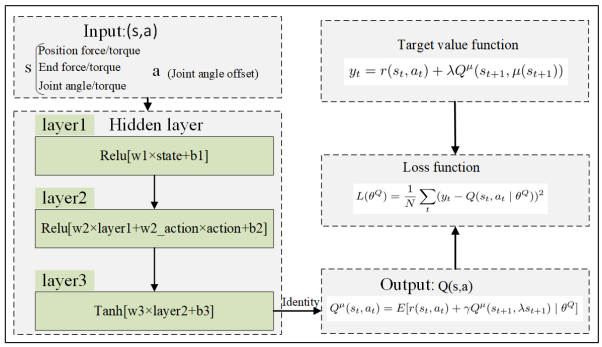


FIGURE 6. Critic-network architecture.

the deterministic actions. The actor networks is updated with the parameter θ^{μ}:

$$\begin{aligned}\nabla_{\theta_\mu} J &\approx E[\nabla_{\theta_\mu} Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t|\theta^\mu)}] \\ &= E[\nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t)} \nabla_{\theta_\mu} \mu(s \mid \theta^\mu) \mid_{s=s_t}]\end{aligned} \tag{7}$$

This was proved in [45]. The critic networks with parameters θ^Q are learned by the Temporary Difference (TD) algorithm to approximate the action-value function, which is defined as:

$$Q^{\mu}(s_t, a_t) = E[r(s_t, a_t) + \gamma Q^{\mu}(s_{t+1}, \lambda s_{t+1}) \mid \theta^Q] \tag{8}$$

It describes the expected return after taking an action a_t. The critic network is optimized by minimizing the loss function:

$$L(\theta^Q) = \frac{1}{N}\sum_t (y_t - Q(s_t, a_t \mid \theta^Q))^2 \tag{9}$$

where the target network

$$y_t = r(s_t, a_t) + \lambda Q^{\mu}(s_{t+1}, \mu(s_{t+1})) \tag{10}$$

is computed using by Bellman equation [46] and $\mu(s_{t+1})$ is the policy acquired from the actor network in the state s_{t+1}.

Balancing the exploration and exploitation strategies is the main challenge associated with application of the DDPG algorithm to continuous action spaces. A hybrid exploration strategy was implemented here to explore better actions efficiently and steadily during the different stages of the learning process. In the early stage, the Ornstein Uhlenbeck (OU) process is used for to action noise generation, as shown in [47]. After the agent learns a stable assembly policy, Gaussian noise is applied to the parametric space [48]:

$$\mu'(s_t) = \mu(s_t|\theta^\mu) + OU \tag{11}$$

$$\theta^{\mu'} = \theta^\mu + N(0, \sigma^2 I) \tag{12}$$

where $\mu(s_t|\theta^\mu)$ is the policy generated from the actor network, $\mu'(s_t)$ and $\theta^{\mu'}$ is the perturbed policy and parameters. The pseudo-code for the algorithm is provided in Algorithm1.

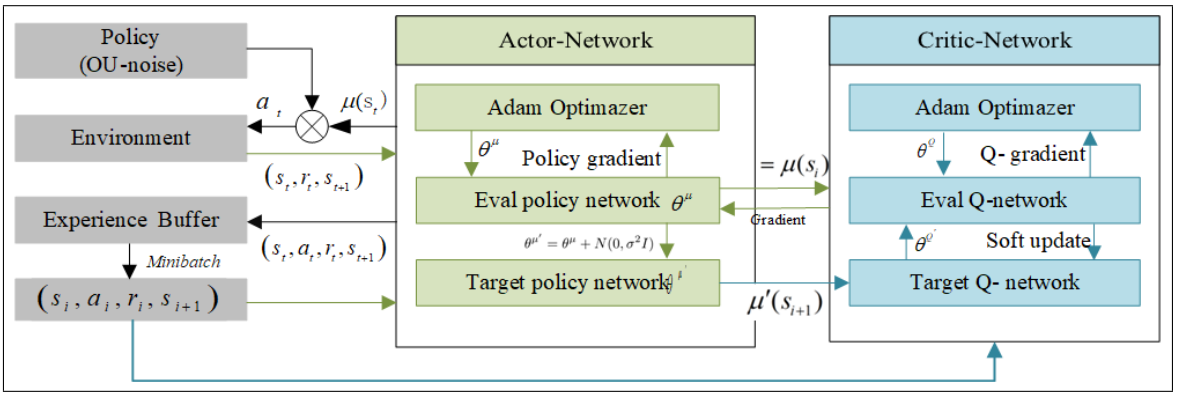


FIGURE 7. Flowchart of the deep deterministic policy gradient.

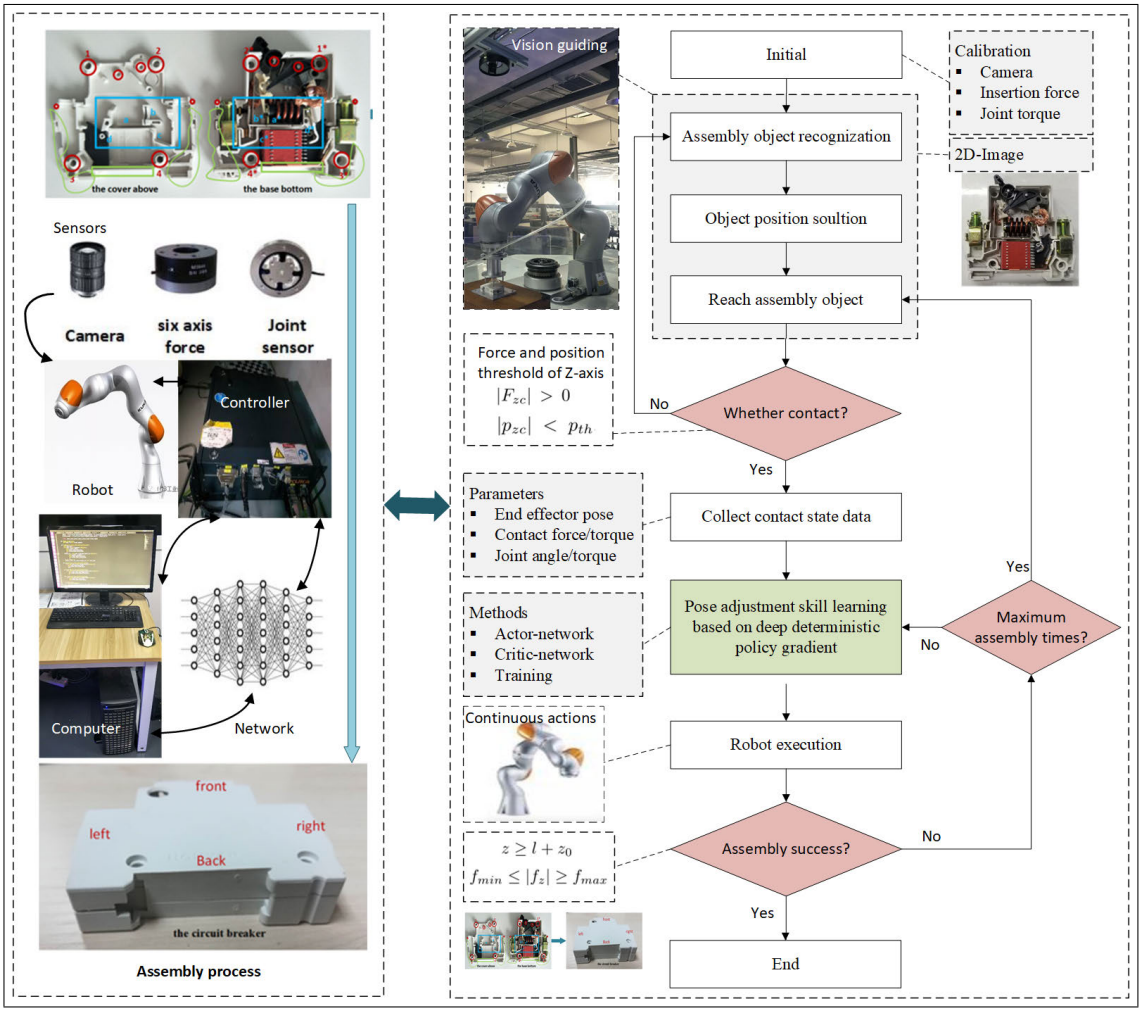


FIGURE 8. Overview of the assembly skill acquisition framework.

3) NETWORK TRAINING

To achieve fast convergence and stable learning, sufficient amount of data is necessary. The experience buffer $\{s_t, a_t, r_t, s_{t+1}\}$ is stored in a finite-size memory buffer. which is modeled as a first-in-first-out structure. When the experience buffer fills up, the oldest samples are discarded, which ensures data quality improvement over time. The actor and critic networks are trained by sampling data in mini-batches from the buffer. The training chart is shown in Fig.7. A copy of the critic target-net $Q'(s, a \mid \theta^{Q'})$ is created. The parameters are updated by soft strategy as follows

$$\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'} \tag{13}$$

Algorithm 1 Manipulation Skill Acquisition for Robotic Assembly

Initialize parameters:
Discount factors λ, Learning rate τ,
Episodes M,Maximum times $step_{max}$,
Initialize: Critic network $Q(s, a \mid \theta^Q)$ with θ^Q
Actor network $\mu(s \mid \theta^\mu)$ with θ^μ
Target network Q' and μ' with $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$
Initialize replay buffer D to capacity
Repeat $e = 0, \quad e = e + 1$
  Repeat $step = 0, \quad step = step + 1$
  With random process for action exploration
  Receive initial state s_1
  for t=1, T **do**
    select action a_t
    Execute action a_t and reward r_t and new state s_{t+1}
    Store $\{s_t, a_t, r_t, s_{t+1}\}$ in D
    Repeat $i = 0, i = i + 1$
    Sample a random minibatch
    N of $\{s_i, a_i, r_i, s_{i+1}\}$ from D
    Set $y_i = r_i + \lambda Q'(s_{i+1}, \mu'(s_{i+1}) \mid \theta^{\mu'}) \mid \theta^{Q'})$
    Update critic by minimizing Loss:
      $L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i \mid \theta^Q))^2$
    Update actor policy using gradient:
    $\nabla_{\theta_\mu} J \approx$
      $\frac{1}{N} \sum_i \nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_i, a=\mu(s_i) \nabla_{\theta_\mu} \mu(s \mid \theta^\mu) \mid s_i}$
    Update target networks:
      $\theta^{Q'} \leftarrow \tau\theta^Q + (1 - \tau)\theta^{Q'}$
      $\theta^{\mu'} = \theta^\mu + N(0, \sigma^2 I)$
  end for **Until** $step = step_{max}$
Until $e = M$

In general, $\tau = 0.001$. The training steps are as follows:

- Actor chooses a_t for the robot, and network returns r_t, s_{t+1}.
- State (s_t, a_t, r_t, s_{t+1}) is stored in the relay buffer.
- Eval network is trained by sampling from the relay buffer.
- Eval Q-network gradient is calculated.
- Eval Q-network is updated and parameters are optimized using the Adam optimizer.
- Eval policy network gradient is calculated.
- Eval policy network is updated and parameters are optimized using the Adam optimizer..
- Update the target network using soft strategy.

C. ACQUISITION OF ASSEMBLY SKILLS

The framework for the assembly skill acquisition is shown in Fig.8. In real-world experiments, a significant challenge is to ensure safe exploration. Assembly system calibration is performed at the initial stage, including joint torque, minimal/maximal terminal force, and camera parameter calibration. The maximal commanded velocity and strict position allowed per joint are set. The range of the end-effector is also defined. The ranges of the torque parameters for each joint are also set to ensure safety during the contact process. The contact position P is calculated by applying forward kinematics to the joint angles measured by the robot encoders. In the assembly process, the position is not precise. To ensure robustness against position errors [33], the rounded values $\widetilde{p_x}$ and $\widetilde{p_y}$ represent position data p_x and p_y using the grid shown in Fig.9. The contour center $\{x, y\}$ can be defined at $\{-c, c\}$ instead of $\{0, 0\}$, where c is the margin of the positional

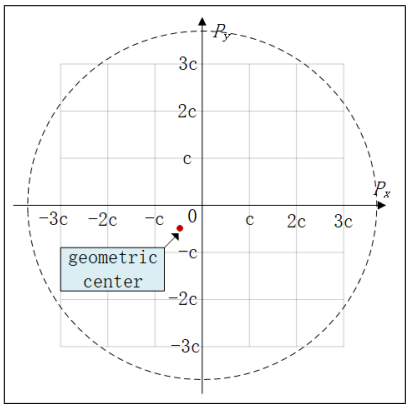


FIGURE 9. Overview of the assembly skill acquisition coordinate system.

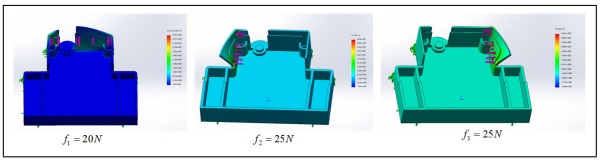


FIGURE 10. Stiffness of the circuit breaker housing.

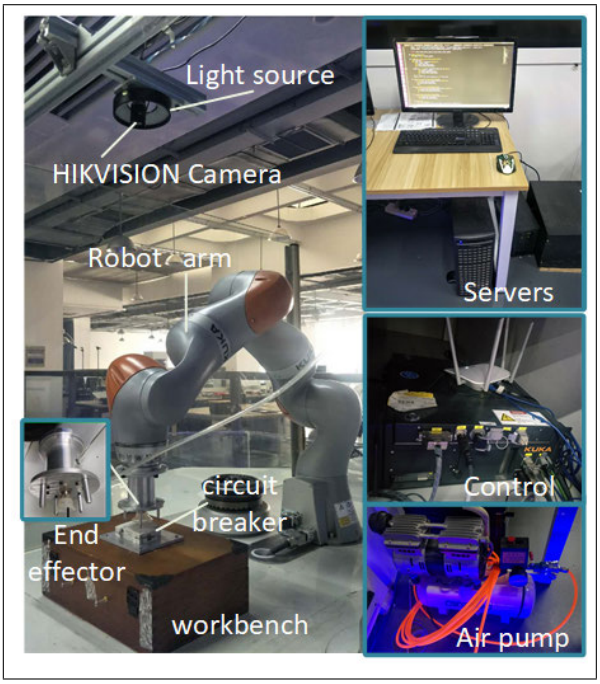


FIGURE 11. Platform of the assembly system.

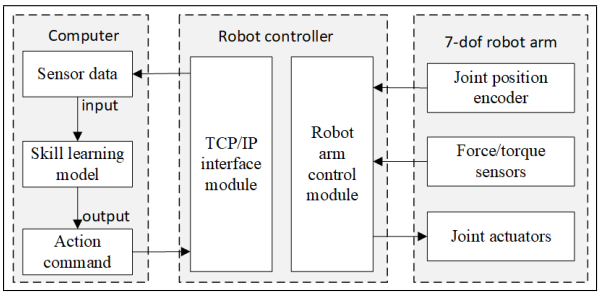


FIGURE 12. Architecture of the experimental platform.

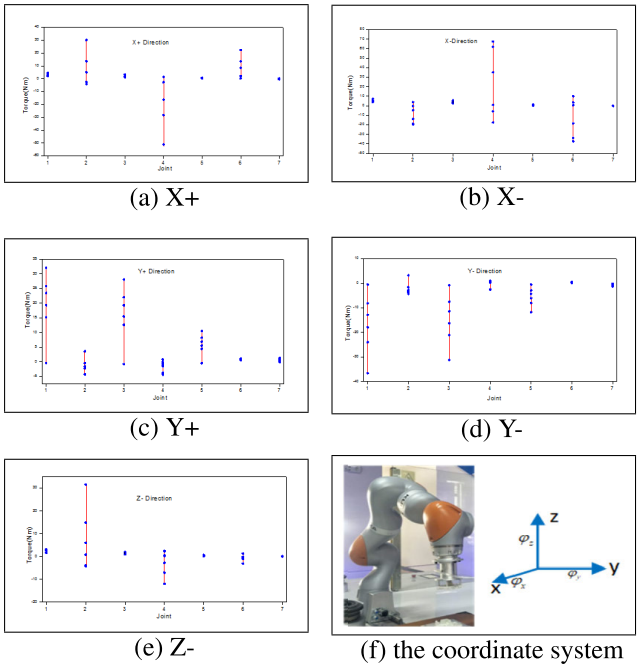


FIGURE 13. Joint torque of the robot.

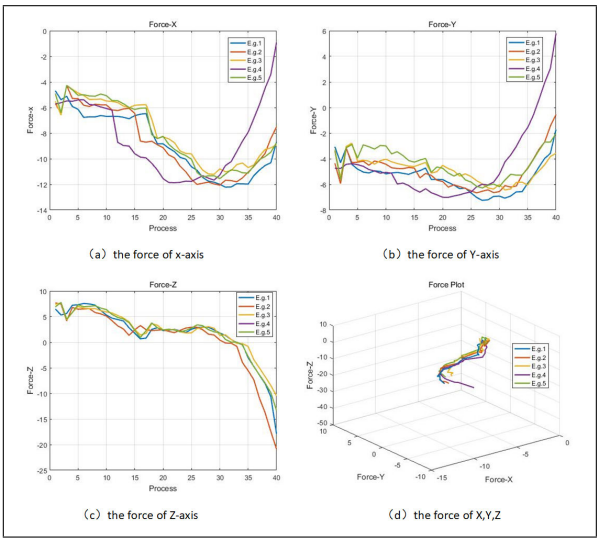


FIGURE 14. Force during the training process.

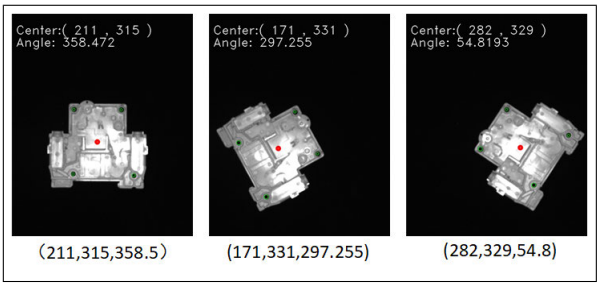


FIGURE 15. Results for different positions.

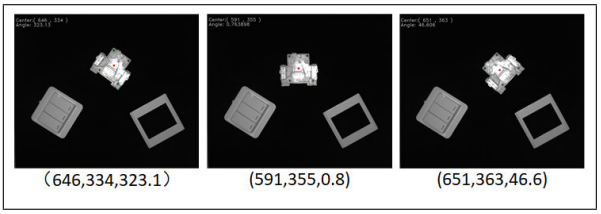


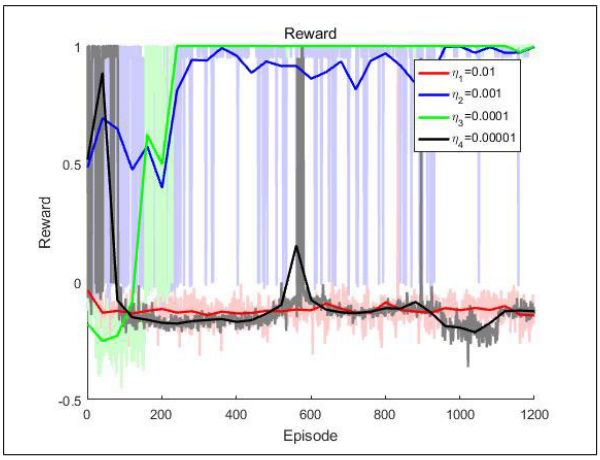
FIGURE 16. Assembly of objects in cluttered scenes.

error. When the cover reaches the base, the critical state is estimated. The force threshold F_{th} and the displacement threshold p_{th} values along the z-axis F_{zc} and p_{zc}, are measured in experiments, considered by $|F_{zc}| > F_{th}$ and $|p_{zc}| < p_{th}$. This provides the transition condition from the non-contact to the contact state. After confirming contact, the contact state parameters are obtained. The next step amount to learning the pose adjustment skill. The maximal assembly times N is set to avoid getting caught in a cycle.

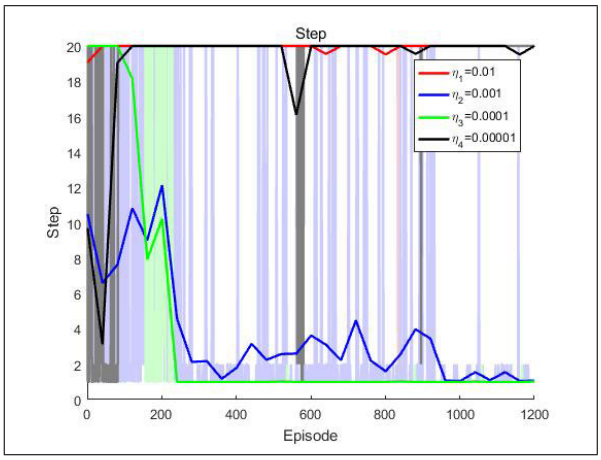
IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

Experiments were performed using fastening assembly of a circuit breaker (HYB1-63) as an experimental system, to valid the proposed method. The object to be assembled was a circuit breaker, which is characterized by a compact structure, small size, and many parts, and a complex shape. In addition, different models are possible. The circuit breaker housing was made from the ABS plastics. Stresses in three directions provided an estimation of the stiffness of the proposed parts, as shown in Fig.10. The safety factor was 2. The experimental system consisted of iiwa7 R800, vision sensor with HIKVISION, a server, an air pump and a workbench, as shown in Fig.11. The vision sensor captureed the assembly object information. A vacuum suction tool with four small iron props was used to pick up the upper cover. Fig.12 shows the architecture of the experimental platform. The camera, the robot and the server communicated via the TCP/IP, and read files from a socket. The system was programmed in C++, Python and Java, using VS, Tensorflow and Sunrise. Sunrise.FRI was used with Java to program the iiwa robot. Our assembly algorithm was implemented and trained on devices equipped with NVIDIA GTX1070 graphics cards and NVIDIA K80 GDDR5 384Bit 10Gbps GPUs. Before the algorithm validation, calibration and security constraints were set.

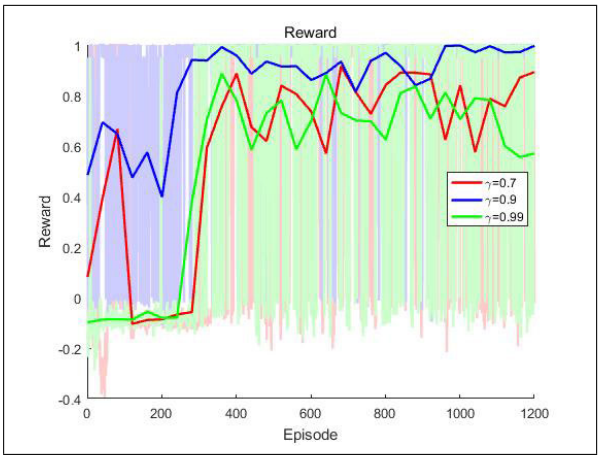


(a) Reward comparison

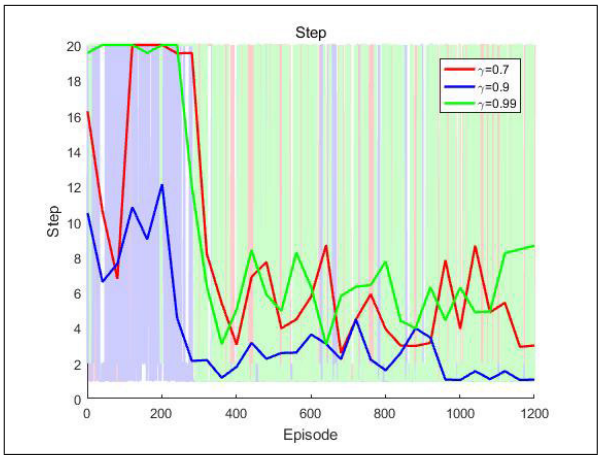


(b) Step comparison

FIGURE 17. Results for different learning rates.



(a) Reward comparison



(b) Step comparison

FIGURE 18. Results for different discount factors.

1) CAMERA CALIBRATION

The camera interior parameter M using the Zhang's calibration algorithm [40] and the coordinate transformation matrix R using target calibration method [41], we obtain

$$M = \begin{bmatrix} 113 & 0 & 1319 & 0 \\ 0 & 117 & 1061 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{14}$$

$$R = \begin{bmatrix} 0.321 & -0.024 & 0 & -993.2 \\ -0.005 & -0.317 & 0 & 344.7 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

2) THE ROBOT CRITICAL JOINT TORQUE

In all of the experiments, the maximal torque allowed per joint was set to protect the assembly components. The seven-axis joint torque affected the force and torque of the end effector through different weights. Constraints were applied for the fastening assembly of low-voltage appliances. Experiments were performed to determine the threshold values for different orientations. The parameter on the z axis is also be set. The assembly cover and base are just damaged for the limit value, and assembly is not successful under this condition. During the assembly execution, the reference force is approximately 3N along the z axis, and approximately 2N of x and y axes. From Fig.13, the torque τ_2 makes a big difference for the z and x axis. In addition, τ_4 also affects the x axis. The torques τ_1 and τ_3 mainly affect the y-axis. The range of the joint torque was set to $(-10, 10)$ for τ_1, τ_3, τ_5, τ_6 and it was $\tau_2 \in (-10, 25)$ and $\tau_4 \in (-7, 5)$. The force plots of assembly produce were shown in Fig.14. The x axis data is the amount of force information collected during assembly.

B. SEARCHING AND LOCALIZATION

The position and recognition of the target base are highlighted in red. It takes less than one second for target localization using the vision algorithm. In addition, the robustness of our method was evaluated for different

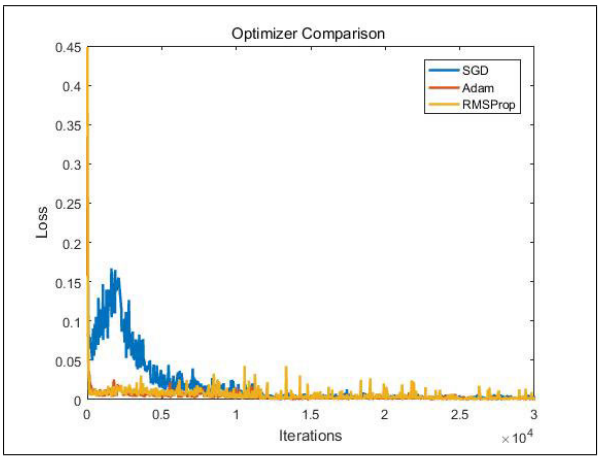


FIGURE 19. Comparison of the Adam and SGD optimizer results.

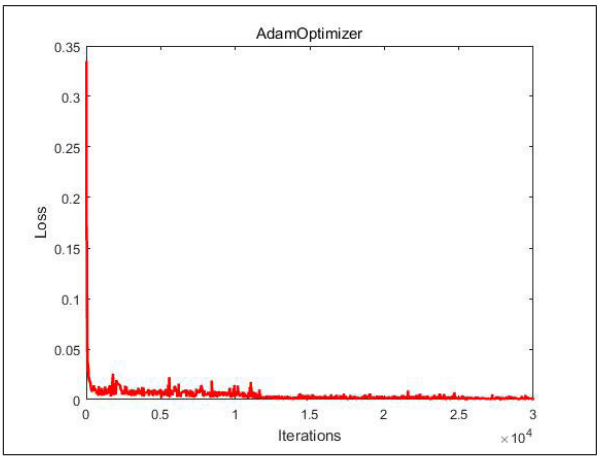


FIGURE 20. Convergence of the loss function during training.

positions and for cluttered scenes. Fig.15 show that for different positions, the base can still be recognized and localized. Fig.16 shows that our method works correctly on cluttered scenes. The two plastic parts in a cluttered scene are similar to the base. To evaluate the recognition and localization accuracy, experiments were conducted three times, for three different positions. The experiments were conducted under the condition of constant illumination.

C. POSE ADJUSTMENT

The networks were trained to learn assembly skills. The maximum size is set to 20000 in the replay buffer D. The hidden nodes of the networks were fully connected layers with (300,200). The activation functions ware modeled by Relu units, in both the critic and actor networks. All assembly scenarios, as episodes M, were set to 1200. The maximal adjustment time for each assembly procedure was $step_{max} = 20$. The size of one mini-batch was 32 (critic-net) and 64 (actor-net), to select random experiences from D.

1) HYPER-PARAMETERS

The model hyper-parameters, such as the learning rate and discount factor, affect the assembly performance. We conducted experiments for different learning rates $\eta(0.01, 0.001, 0.0001, 0.00001)$ for the critic network, while the values were tenfold smaller for the actor network ($\gamma = 0.9$ and Adam optimizer) and the results are shown in Fig.17. The learning rate was $\eta = 0.0001$ and the Adam optimizer was used. Three experiments were performed, for three different discount factors $\gamma(0.99, 0.9, 0.7)$.

As shown in Fig.18, the results for the discount factor $\gamma = 0.9$ were better than for other factors. Compared with the SGD optimizer, the Adam and RMS optimizers achieved better training. From the loss function in the Fig.19, the Adam and RMS optimizers performed better than the SGD optimizer. The Adam optimizer is more robust. The Adam optimizer was more robust. The iterations were adjustment times in all assembly scenarios.

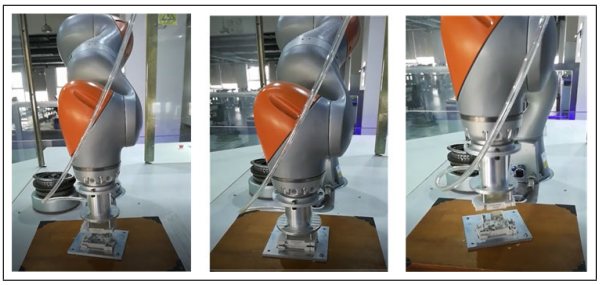


FIGURE 21. Three assembly base positions.

2) MODEL TRAINING RESULTS

After localization using visual guidance was complete, the training process started. The loss function dynamics during learning is shown in Fig.20. As shown by the green line in Fig.17, the reward begins to converge from 230 episodes (less than 100 min). It takes 535 min to learn the necessary skills, which is equivalent to 1200 episodes. As the number of the training steps increases, the reward finally stabilizes in the (0.98, 1) interval, and the step approximates to 1. The average reward is 0.95 and the average step is 1. A training video is provided in the training process.

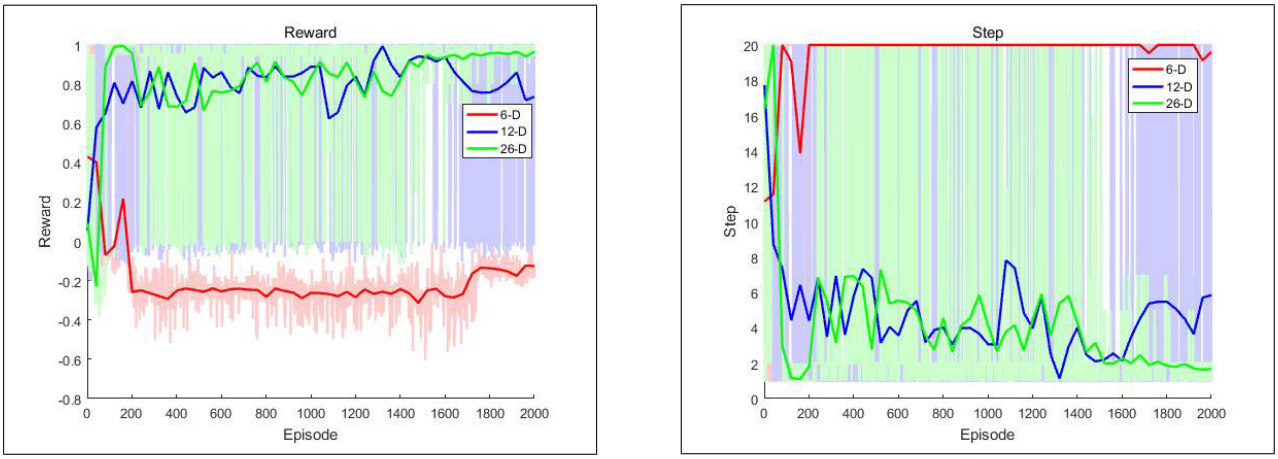
3) PERFORMANCE EVALUATION

To test the generalization capability of the proposed method, the positions of the objects to be assembled were changed. The same architecture was used to train the network. The experiments were performed for three different base positions, and the results are shown in Fig.21. For each position, the assembly process was rerun 1000 times. The results of these experiments (Table.1) show that the success rate exceeded 90%.

D. COMPARISON RESULTS

1) STATE REPRESENTATION AND ITS EFFECTS

State representation richness strongly affects the robot performance. Our results (Table.2) show that the state representation captured the force/torque, position and robot joint parameters better than the other representations.



(a) different state reward (b) different state step

FIGURE 22. Results for different state representations.

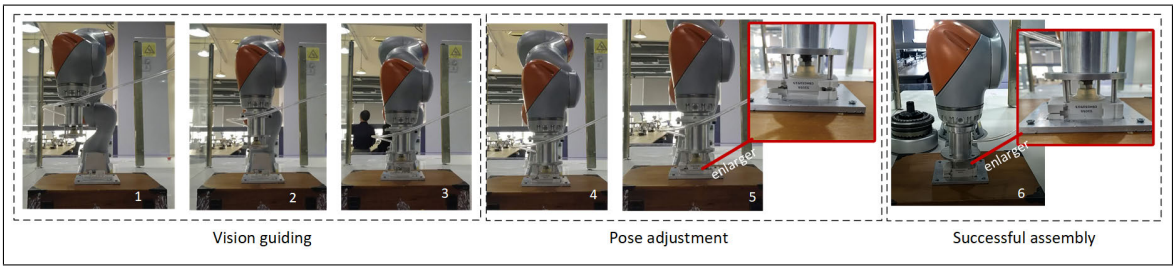


FIGURE 23. Full assembly process.

TABLE 1. Successful times for different positions.

Position	Success times	Success rate
1	932	93.2%
2	941	94.1%
3	924	92.4%

TABLE 2. Performance for different state representation.

State representation	Performance	Training time
Force/torque only	68%	1320 min
Force/torque + position	89%	550min
Force/torque + position + joint parameters	94%	535min

These experiments indicate that while the contact force/torque can effectively capture the assembly state, explicitly adding more state features improves performance. The training time was reduced by more than 700 min. After adding the joint parameters, the success rate increased significantly. The reward and step were smooth for the 26-dimensional DDPG framework, as shown in Fig.22.

TABLE 3. Performance for different frameworks.

Frame	Success rate
[38]based on ELM	56%
[39]based on SVM	54%
[15]based on DQN	81%
proposed based on DDPG	94%

2) DIFFERENT FRAMES

To demonstrate the effectiveness of the proposed method for skill acquisition, we compared the proposed method with knowledge-based methods [38], [39] and with the deep Q-learning network framework [15]. The robot could acquire pose adjustment skills in the frameworks of DQN and DDPG, with little prior knowledge. Table2 shows that the success rate improved by 10% compared with previously suggested methods [15]. Knowledge base significantly affected the success rate. When knowledge was not sufficient, the success rate was under 40%.

The full assembly process is shown in Fig.23, including vision guiding, pose adjustment, and successful assembly. It takes 3-5 seconds to perform the assembly. The initial state assumed that the camera was not occluded. A video was provided in the assembly process.

V. CONCLUSION

This paper proposed a method for acquiring manipulation skills during assembly using deep reinforcement learning. The ability to acquire skills could be considered an improved behavior of the industrial robots. The DDPG algorithm was used to realize the continuous space. We analyzed the assembly environment, assembly state space, continuous action space, and reward system. The proposed method to acquire skills is very time efficient and allows the collection of many more samples in comparison with the human-demonstration-based method.

Real-world experiments were performed in the fastening assembly process of a circuit breaker to demonstrate the efficiency of the proposed method. The results of these experiments show that robots can complete the fastening assembly tasks by imitating human learning. Further studies are necessary to optimize the training time and increase the success rate. We envision two directions for further studies. First, the return function is well-designed according to the specific assembly task. On the other hand, it is considered that prior knowledge should be embedded in the robot learning process, similar to what is believed to occur for humans. More studies must be conducted to explore the structures of the deep network in the algorithm that are more suitable for complex assemblies. The proposed method using deep reinforcement can improve robot intelligence.

REFERENCES

[1] L. Roveda, N. Iannacci, F. Vicentini, N. Pedrocchi, F. Braghin, and L. M. Tosatti, "Optimal impedance force-tracking control design with impact formulation for interaction tasks," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 130–136, Jan. 2016.

[2] L. Roveda, G. Pallucca, N. Pedrocchi, F. Braghin, and L. M. Tosatti, "Iterative learning procedure with reinforcement for high-accuracy force tracking in robotized tasks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 4, pp. 1753–1763, Apr. 2018.

[3] L. Roveda, "Adaptive interaction controller for compliant robot base applications," *IEEE Access*, vol. 7, pp. 6553–6561, 2018.

[4] L. Peternel, T. Petrič, and J. Babič, "Human-in-the-loop approach for teaching robot assembly tasks using impedance control interface," in *Proc. Int. Conf. Robot. Automat.*, May 2015, pp. 1497–1502.

[5] H. Park, J. Park, D.-H. Lee, J.-H. Park, M.-H. Baeg, and J.-H. Bae, "Compliance-based robotic peg-in-hole assembly strategy without force feedback," *IEEE Trans. Ind. Electron.*, vol. 64, no. 8, pp. 6299–6309, Aug. 2017.

[6] J. Bos, A. Wahrburg, and K. D. Listmann, "Iteratively learned and temporally scaled force control with application to robotic assembly in unstructured environments," in *Proc. Int. Conf. Robot. Automat.*, May/Jun. 2017, pp. 3000–3007.

[7] Z. Pan, J. Polden, N. Larkin, S. Van Duin, and J. Norrish, "Recent progress on programming methods for industrial robots," *Robot. Comput. Integr. Manuf.*, vol. 28, no. 2, pp. 87–94, 2012.

[8] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *Comput. Sci.*, vol. 8, no. 6, p. A187, 2015.

[9] N. Heess, D. Tb, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. Riedmiller, and D. Silver, "Emergence of locomotion behaviours in rich environments," Jul. 2017, *arXiv:1707.02286*. [Online]. Available: https://arxiv.org/abs/1707.02286

[10] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," Aug. 2017, *arXiv:1707.06347*. [Online]. Available: https://arxiv.org/abs/1707.06347

[11] M. Al-Shedivat, T. Bansal, Y. Burda, I. Sutskever, I. Mordatch, and P. Abbeel, "Continuous adaptation via meta-learning in nonstationary and competitive environments," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, 2018.

[12] S. Levine and P. Abbeel, "Learning neural network policies with guided policy search under unknown dynamics," in *Proc. 28th Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, 2014, pp. 1071–1079.

[13] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1334–1373, 2016.

[14] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," in *Proc. 25th Int. Symp. Exp. Robot.* Cham, Switzerland: Springer, 2016, pp. 173–184.

[15] F. Li, Q. Jiang, S. Zhang, M. Wei, and R. Song, "Robot skill acquisition in assembly process using deep reinforcement learning," *Neurocomputing*, vol. 345, pp. 92–102, Jun. 2019.

[16] Y. Gu, W. H. Sheng, C. Crick, and Y. S. Ou, "Automated assembly skill acquisition and implementation through human demonstration," *Neurocomputing*, vol. 9259, pp. 85–93, 2017.

[17] R. Dillmann, "Teaching and learning of robot tasks via observation of human performance," *Robot. Auton. Syst.*, vol. 47, nos. 2–3, pp. 109–116, Jun. 2004.

[18] C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang, "A learning framework of adaptive manipulative skills from human to robot," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 1153–1161, Feb. 2019.

[19] M. Skubic and R. A. Volz, "Acquiring robust, force-based assembly skills from human demonstration," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 772–781, Dec. 2000.

[20] Y. Gu, W. Sheng, C. Crick, and Y. Ou, "Automated assembly skill acquisition and implementation through human demonstration," *Robot. Auton. Syst.*, vol. 99, pp. 1–16, Jan. 2018.

[21] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Netw.*, vol. 21, no. 4, pp. 682–697, 2008.

[22] E. Theodorou, J. Buchli, and S. Schaal, "Reinforcement learning of motor skills in high dimensions: A path integral approach," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, May 2010, pp. 2397–2403.

[23] J. Peters, K. Mülling, and Y. Altun, "Relative entropy policy search," in *Proc. AAAI Conf. Artif. Intell.*, 2010.

[24] M. Kalakrishnan, L. Righetti, P. Pastor, and S. Schaal, "Learning force control policies for compliant manipulation," in *Proc. Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2011, pp. 4639–4644.

[25] I. Popov, N. Heess, T. Lillicrap, R. Hafner, G. Barth-Maron, M. Vecerik, T. Lampe, Y. Tassa, T. Erez, and M. Riedmiller, "Data-efficient deep reinforcement learning for dexterous manipulation," 2017, *arXiv:1704.03073*. [Online]. Available: https://arxiv.org/abs/1704.03073

[26] N. Fazeli, M. Oller, J. Wu, Z. Wu, J. B. Tenenbaum, and A. Rodriguez, "See, feel, act: Hierarchical learning for complex manipulation skills with multisensory fusion," *Sci. Robot.*, vol. 4, no. 26, 2019, Art. no. eaav3123.

[27] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, "QT-Opt: Scalable deep reinforcement learning for vision-based robotic manipulation," 2018, *arXiv:1806.10293*. [Online]. Available: https://arxiv.org/abs/1806.10293

[28] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with large-scale data collection," in *Proc. Int. Symp. Exp. Robot.*, 2016, pp. 173–184.

[29] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May/Jun. 2017, pp. 3389–3396.

[30] D. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May/Jun. 2017, pp. 3357–3364.

[31] J. Schulman, S. Levine, P. Moritz, M. Jordan, and P. Abbeel, "Trust region policy optimization," in *Proc. 31st Int. Conf. Mach. Learn.*, Lille, France, 2015, pp. 1889–1897.

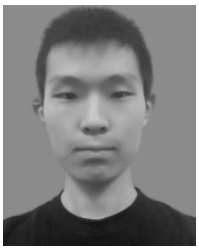
[32] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, Sep. 2013.

[33] T. Inoue, G. D. Magistris, A. Munawar, T. Yokoya, and R. Tachibana, "Deep reinforcement learning for high precision assembly tasks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 820–825.

[34] Z. Hou, H. Dong, K. Zhang, Q. Gao, K. Chen, and J. Xu, "Knowledge-driven deep deterministic policy gradient for robotic multiple peg-in-hole assembly tasks," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2018, pp. 256–261.
[35] I. F. Jasim and P. W. Plapper, "Contact-state modeling of robotic assembly tasks using Gaussian mixture models," *Procedia CIRP*, vol. 23, pp. 229–234, Jan. 2014.
[36] S. Huang, K. Murakami, Y. Yamakawa, T. Senoo, and M. Ishikawa, "Fast peg-and-hole alignment using visual compliance," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 287–292.
[37] W. W. Wan, F. Lu, Z. P. Wu, and K. Harada, "Teaching robots to do object assembly using multi-modal 3D vision," *Robot. Auton. Syst.*, 2018.
[38] S. Zhang, Q. Jiang, Y. Li, F. Li, and R. Song, "Contact state classification in industrial robotic assembly tasks based on extreme learning machine," in*Proc. IEEE 8th Annu. Int. Conf. Cyber Technol. Automat., Control, Intell. Syst. (CYBER)*, Jul. 2018, pp. 617–622.
[39] F. Li, Q. Jiang, Y. Li, M. Wei, and R. Song, "Modeling contact state of industrial robotic assembly using support vector regression," in *Proc. IEEE 8th Annu. Int. Conf. Cyber Technol. Automat., Control, Intell. Syst. (CYBER)*, Jul. 2018, pp. 646–651.
[40] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
[41] W.-L. Li, H. Xie, G. Zhang, S.-J. Yan, and Z.-P. Yin, "Hand–eye calibration in visually-guided robot grinding," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2634–2642, Nov. 2016.
[42] Y. Xiao and J. Li, "Crack detection algorithm based on the fusion of percolation theory and adaptive canny operator," in *Proc. 37th Chin. Control Conf. (CCC)*, Jul. 2018, pp. 4295–4299.
[43] Z. Yiming and W. Jun, "Research on iris recognition algorithm based on Hough transform," *Proc. IOP Conf. Series, Mater. Sci. Eng.*, vol. 439, no. 3, 2018, Art. no. 032007.
[44] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 930–943, Aug. 2003.
[45] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. Int. Conf. Int. Conf. Mach. Learn. (ICML)*, Jun. 2014, pp. I-387–I-395.
[46] R. E. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1957.
[47] G. E. Uhlenbeck and L. S. Ornstein "On the theory of the Brownian motion," *Phys. Rev.*, vol. 36, no. 5, p. 823, Sep. 1930.
[48] M. Plappert, R. Houthooft, P. Dhariwal, S. Sidor, R. Y. Chen, X. Chen, T. Asfour, P. Abbeel, and M. Andrychowicz, "Parameter space noise for exploration," 2017, *arXiv:1706.0190*. [Online]. Available: https://arxiv.org/abs/1706.01905

FENGMING LI received the B.S. degree in automation from Weifang University, Weifang, China, in 2007, and the M.S. degree from the School of Control Science and Engineering, Shandong University, Jinan, China, in 2010, where she is currently pursuing the Ph.D. degree. Her research interests include intelligent control, machine learning, and optimization theory.

QI JIANG received the Ph.D. degree from Tianjin University, Tianjin, China, in 2003. He is currently a Professor with the School of Control Science and Engineering, Shandong University, Jinan, Shandong, China. His major research interests include novel inspection and sensors, and FBG sensors.



WEI QUAN received the B.S. degree in automation from Qingdao University, China, in 2018. He is currently pursuing the M.S. degree with the School of Control Science and Engineering, Shandong University, Jinan, China. His research interests include machine learning and intelligent robot.

SHIBO CAI received the B.E., M.S., and Ph.D. degrees from the Zhejiang University of Technology, in 2003, 2010, and 2018, respectively, where he is currently an Associate Research Fellow. His main research interests include robot and intelligent electromechanical equipment.

RUI SONG received the B.E. degree in industrial automation, in 1998, the M.S. degree in control theory and control engineering from the Shandong University of Science and Technology, in 2001, and the Ph.D. degree in control theory and control engineering from Shandong University, in 2011. He is currently an Associate Professor with the School of Control Science and Engineering, Shandong University, Jinan, China, and one of the directors of the Center of Robotics of Shandong University. He is engaged in research on intelligent sensor networks, intelligent robot technology, and intelligent control systems. His research interests include medical robots, industrial robots, and the quadruped robots.

YIBIN LI received the bachelor's and doctor's degrees from Tianjin University, China, in 1982 and 2006, respectively, and the master's degree from the Shandong University of Science and Technology, China, in 1988. He is currently a Professor with the School of Control Science and Engineering, Shandong University, China. His research interests include robotics, mechatronics, intelligent control, intelligent vehicles, etc.