Hindawi
Journal of Sensors
Volume 2021, Article ID 5073689, 12 pages
https://doi.org/10.1155/2021/5073689

Research Article

Alignment Method of Combined Perception for Peg-in-Hole Assembly with Deep Reinforcement Learning

Yongzhi Wang,[1] **Lei Zhao**,[1] **Qian Zhang**,[1,2] **Ran Zhou**,[1] **Liping Wu**,[1] **Junqiao Ma**,[1] **Bo Zhang**,[1] **and Yu Zhang**[1]

[1]*Department of Mechanical Engineering, Shenyang University of Technology, Shenyang 110000, China*
[2]*College of Science, Shenyang University of Chemical Technology, Shenyang 110000, China*

Correspondence should be addressed to Yu Zhang; zhangyu@sut.edu.cn

Received 29 July 2021; Revised 16 August 2021; Accepted 4 September 2021; Published 22 September 2021

Academic Editor: Kelvin Wong

The method of tactile perception can accurately reflect the contact state by collecting force and torque information, but it is not sensitive to the changes in position and posture between assembly objects. The method of visual perception is very sensitive to changes in pose and posture between assembled objects, but they cannot accurately reflect the contact state, especially since the objects are occluded from each other. The robot will perceive the environment more accurately if visual and tactile perception can be combined. Therefore, this paper proposes the alignment method of combined perception for the peg-in-hole assembly with self-supervised deep reinforcement learning. The agent first observes the environment through visual sensors and then predicts the action of the alignment adjustment based on the visual feature of the contact state. Subsequently, the agent judges the contact state based on the force and torque information collected by the force/torque sensor. And the action of the alignment adjustment is selected according to the contact state and used as a visual prediction label. Whereafter, the network of visual perception performs backpropagation to correct the network weights according to the visual prediction label. Finally, the agent will have learned the alignment skill of combined perception with the increase of iterative training. The robot system is built based on CoppeliaSim for simulation training and testing. The simulation results show that the method of combined perception has higher assembly efficiency than single perception.

1. Introduction

It is an important challenge for the intelligent robot to fully observe environmental information in the complex unstructured environment. However, the perception capacity of the robot will directly affect the robot's performance in the task [1–5]. It is difficult to meet current complex work demands only relying on a single type of sensor to perceive the environment. Besides, traditional programming methods in assembly tasks require technicians with a high technical level and rich work experience to complete a large amount of code compilation and parameter deployment. This not only takes time and effort but also limits the flexibility of the production line. The traditional programming method in the structured environment can no longer meet the production requirements that require frequent upgrades. The programming model of the robot has changed from hard coding to teaching-playback for the rapid changes in the production line [6–10]. The teaching-playback method greatly reduces the workload of programming. Nevertheless, the teaching method still requires a large number of parameter deployments like the traditional programming method. Therefore, more research has focused on training robots to acquire work skills independently with the learning-based method. The trained robot can autonomously interact with the environment to complete work. Robots mainly rely on visual and tactile perception methods to perceive the environment in the interacting process.

Tactile sensation is very important for humans to perceive the environment, and it is also one of the important perception means for robots. The method based on force control is mostly used to solve the task of precision assembly. The force sensor, position sensor, and force/torque (F/T) sensor are the most commonly used sensors based on force control. They can accurately feedback the contact force when the assembly parts are in contact with each other. When three-point contact occurs in the peg-in-hole assembly, the three degrees of freedom of the peg are restricted by the hole, which makes it difficult to complete the insertion for the peg with the traditional method. A novel alignment method based on geometric and force analysis is developed to deal with this dilemma [11]. This method uses the F/T sensor to measure the contact force information to estimate the relative pose of the pile and hole.

The alignment between the peg and the hole is accomplished by compensating motion based on attitude estimation. To address the assembly failure caused by the large friction resistance and poor contact situations, a screw insertion method was developed for peg-in-hole assembly [12]. The proposed method analyzes the point contact and surface contact to reduce axial friction in the assembly process. And it is still valid in the case of transition fit. For high-precision assembly tasks, a large number of parameters often need to be deployed, which technicians need to spend a lot of time on programming deployment. Therefore, an easy to deploy teach-less method is proposed to complete precise peg-in-hole assembly [13]. Whereafter, an easy to deploy teach-less method is proposed to complete precise peg-in-hole assembly. The low accuracy of conventional programming is compensated without artificial parameter tuning by training based on deep reinforcement learning. Moreover, a variable compliance control method based on deep reinforcement learning is proposed for the peg-in-hole of the 7-DOF with torque sensor robot to improve the efficiency and robustness of the assembly task in the uncertain initial state and complex environment [14]. The trained robot can select passive compliance or active regulation to dispose of the current environment, which makes the variable compliance fewer adjustment steps than the fixed compliance. In addition, the method of combined learning-based algorithm and force control strategy is proposed [15]. It contains the hybrid force/position controller and the variable impedance controller. The hybrid force/position controller was designed to ensure the safe and stabilization of the searching hole. The variable impedance controller based on fuzzy Q-learning is used to conduct compliance action. The proposed method improves the stability and adaptability of the peg-in-hole assembly. Many high-precision assembly tasks mostly choose the method based on force control. However, the appearance characteristics and related location information of the environment cannot be well perceived for the force sensors.

Visual perception plays an important role in the robotic perception of the environment. Visual perception can quickly perceive the appearance characteristics and relative position information of the object. It is difficult for visual perception to process the occluded part when the target is partially occluded. Human beings often rely on touch, hearing, and smell to perceive the environment when their vision is obscured. And the visual perception is interfered with by environmental factors such as lighting, which leads to the robot needing to work in a specific working environment [16].

In recent years, the field of visual perception has also made numerous research progress with the vigorous development of deep learning and deep reinforcement learning. The robot of the combined system uses a two-level vision measurement method in robot automatic assembly [17]. This technique has developed an accurate coordinate transformation for the calibration of the dynamic coordinate system. Whereafter, the hole was 3D reconstructed for the hole edge point selection. This method makes the cost of the pose determination become lower. And it also extends the visual measurement range and improves the positioning accuracy. In addition, the method of uncalibrated visual servoing is used in peg-in-hole assembly, which is a three-phase assembly strategy [18].

The designed system first uses an eye-to-hand mono camera to perform attitude alignment, which makes the assembly object and the predefined transition location parallel to each other. Then, the system aligns the assembly object and the predefined transition position collinearly. Finally, the assembly object completed the longitudinal alignment. Besides, a learning-based visual servoing method was used to quicken the speed of the searching hole [19]. This method uses the concept of domain randomization based on deep learning to predict the position of the hole. The deep neural network uses synthetic data for training to predict the hole's quadrant. Whereafter, the peg moves towards the hole through visual servoing iteration. The diameter and the length of the assembly are, respectively, 10 mm and 70 mm. The assembly clearances between the peg and the hole are 0.4 mm. It still can quickly complete the peg-in-hole assembly when facing different surfaces with various colors and textures in the real world. And the assembly time is less than 70 s. Whereafter, in order to peg-in-hole alignment, a visual servoing based on learning was developed to faster align with the hole [20]. The deep neural network for peg and hole point estimates uses purely synthetic data to train. The assembly system is equipped with two cameras and a special lighting system, which can align the peg with the holes covered by different materials and then complete the insertion of the peg through compliance control with force-feedback. Moreover, the method of the dynamic position-based servo can perform the microassembly with the micropeg of diameter 80 μm and the hole of 100 μm [21].

The assembly system is equipped with encoders for position servo, light source, and three CCD cameras to automatically align, grasp, transport, and assemble. The process of the microassembly has not the contact adhesion force. The average time and the success rate of the assembly are 4 mins and 80%, respectively. In summary, the control method based on the vision for the assembly has higher assembly efficiency than force, but the assembly accuracy is not as good as the method based on force. If the system based on the vision method needs to improve the assembly accuracy, the system needs to be equipped with a high-precision vision sensor, a special lighting source, and spend more assembly time. The control methods based on vision or force have their own advantages and disadvantages. If they can complement each other, the robot will have higher assembly efficiency while ensuring assembly accuracy.

Humans often use the means of visual observation and tactile perception to complete the peg-in-hole assembly. It is possible to complete the peg-in-hole assembly of the minuscule clearance under the condition of clear observation and sensitive tactile perception. On the one hand, we can also use only visual observation to complete the assembly. However, there needs to be sufficient clearance when the state of the peg and the hole can be clearly observed. Otherwise, it will cause the assembly to fail. On the other hand, we can also use only the tactile perception to achieve a successful assembly. However, it may take more time. So the assembly speed of a robot using multiple perception methods is often better than that of a single perception method. Therefore, the current research of peg-in-hole assembly mostly adopts the hybrid control method of visual observation and tactile perception [22–26]. For instance, a guidance algorithm based on geometrical information and force control is proposed to improve the success rate of the peg-in-hole assembling with complex shapes [7].

The proposed method makes a 6-DOF industrial robot with the eye-in-hand camera chooses assembly direction through spatial arrangement and geometric. And it determines the magnitude of force through kinesthetic teaching. Besides, the dual-arm coordination robot adopts a hybrid assembly strategy based on vision/force guidance for peg-in-hole assembly [27]. This method can be used in round, triangle, and square assembly parts with 0.5 mm maximum clearance. Baxter research robot has three vision sensors placed on the left hand and right hand head, respectively. The robot first uses visual guidance to achieve rough adjustment. Afterward, the robot uses the force feedback mechanism with the F/T sensor to perform precise adjustments. The proposed method can ensure a high assembly success rate for assembly parts of different shapes. Furthermore, the modalities with different characteristics were designed based on deep reinforcement learning for different geometry peg-in-hole tasks with tight clearance [28]. The robot has three sensors to collect the data of RGB images, F/T sensor, and end-effector as input.

Our technique uses multiple inputs to establish a compact multimodal representation to predict contact and alignment in the peg-in-hole assembly. And then, the robot controller with haptic and visual feedback was realized through the self-supervision training without the manual annotation. Moreover, a novel method was proposed to find the right inserting pose through trials with force feedback and vision [23]. The adjustment times of the assembly were minimized by the reinforcement learning training, which uses force and visual feature design. In addition, the combined method of learning-based algorithms and force control strategies were proposed to improve the efficiency and safety of the assembly process [15]. This method takes advantage of the MLP network to generate the action trajectories during the hole-searching and uses the force/position controller to ensure the safety and stability in the contact. The variable impedance controller based on fuzzy Q-learning was designed to insert the peg into the hole. The proposed method improves the efficiency and effectiveness of the assembly.

The current research of the peg-in-hole assembly uses mostly multiple perception methods, but most of them use a single perception method to adjust the alignment between the peg and the hole. However, humans often use the method of visual and tactile perception to complete this work. The robot's visual and force perception should be well combined to better intelligent performance and higher assembly efficiency.

In this paper, a hybrid control method of vision and tactility is proposed based on deep reinforcement learning to improve alignment efficiency for the peg-in-hole tasks. The mapping relationship between visual features and tactile signals will be established by trial and error with the self-supervised. Firstly, the RGB-D image is obtained by the visual sensor. Secondly, the deep neural network extracts visual features from the image and predicts the contact state. Thirdly, the agent receives the force signal by the tactile sensor to determine the current contact state as a visual prediction label. Finally, the network of the visual prediction uses this label to conduct the backpropagation calculation for correcting the network weights. We introduce the working principle of the peg-in-hole assembly in Section 2, and a quick hole-searching strategy is designed. In Section 3, the hybrid control method is proposed for the peg-in-hole assembly to improve assembly efficiency. In Section 4, the simulation results in CoppeliaSim and analysis results are presented. Section 5 elaborates the conclusions and future work.

2. Working Principles and Analysis of Peg-in-Hole Assembly

2.1. Analysis of the Contact State between the Peg and the Hole. The task of peg-in-hole assembly is mainly divided into the grasping stage, the hole-searching stage, the alignment stage, and the insertion stage. The task of the grasping stage is to grasp the peg and move it to the vicinity of the hole. The task of the hole-searching stage is to visually detect the edge and the center of the hole and then move the peg to the center position of the hole. The task of the alignment stage is to adjust the posture of the peg, so that the posture alignment is completed between the peg and the hole. The task of the insertion stage is to insert the peg into the hole after alignment. In the assembly process, there are three vital contact states as shown in Figure 1. The bottom of the peg makes surface contact with the upper surface of the hole after moving the peg. This contact is called surface contact, as illustrated in Figure 1(a). The point contact will occur between the inside of the hole and the surface of the peg if the peg is close enough to the center of the hole. Two-point contact and three-point contact are shown in Figures 1(b) and 1(c), respectively. The plane contact only occurs in the hole-searching stage. It means that the position of the hole has been found when the point contact has occurred. It means that the robot has completed the task of the hole-searching stage and entered the alignment stage.

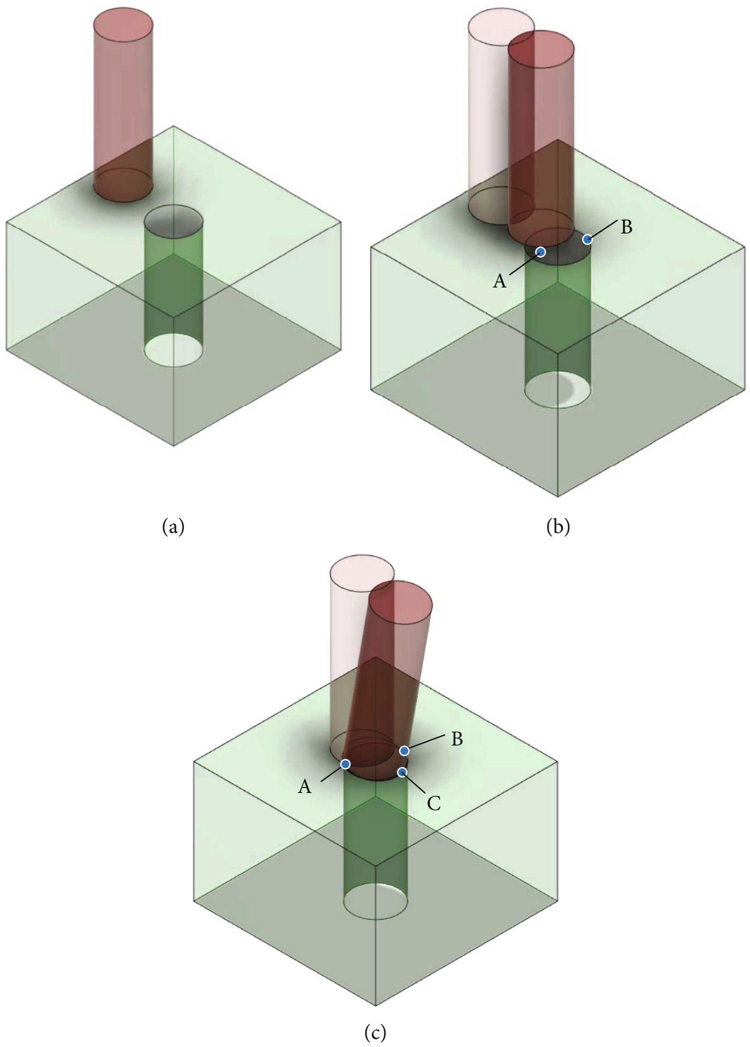


Figure 1: Schematic diagram of contact state based on (a) plane contact, (b) two-point contact, and (c) three-point contact.

The key stages that affect the efficiency of peg-in-hole assembly are the hole-searching stage and the alignment stage. Their details are introduced in Section 2.2 and Section 2.3, respectively.

2.2. *Working Principles of Searching Hole*

2.2.1. The Method of Force-Based Searching Hole. Firstly, the peg will be moved to the surface of the hole, which produces a plane contact state between the peg and the hole. At this time, the peg situates the outside of the hole. Subsequently, the peg searches for the position of the hole with an Archimedes spiral trajectory. During the search process, the center of the peg gradually approaches the center of the hole. The peg will be inserted into the hole or tilted in the inside of the hole under the action of the assembly force when the position of the shaft and the hole are close enough. The peg went into the inside of the hole by this time, that is, the work of the searching hole is completed and the adjustment phase is entered. The method of force-based searching hole often spends more time than the vision-based.

2.2.2. The Method of Vision-Based Searching Hole. The image data expressing the current environment information is obtained through the vision sensor. And then, it is applied edge detection with the Canny operator. But the edge detection is susceptible to interference from image noise. Therefore, image noise removal must be performed with Gaussian filtering before the edge detection. The image noises will be eliminated by the Gaussian smoothing filter, and the Gaussian kernel used by the filter is described as follows:

$$K = \frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}. \tag{1}$$

And after that, the system calculates the intensity gradients and direction with the Sobel operator. The convolution

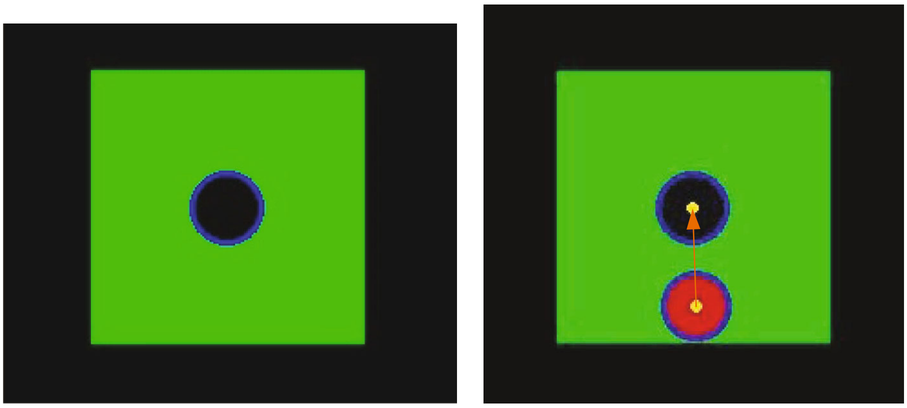


Figure 2: Schematic diagram of visual recognition with (a) edge detection of the hole and (b) position detection of the center for the hole.

arrays need to be applied to the x and y directions, respectively, to calculate the gradient magnitude and direction. The convolution arrays are shown as follows:

$$d_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad (2)$$

$$d_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \quad (3)$$

The intensity gradients S determine whether the point is an edge point. The large gradient value indicates that the gray value around the point changes quickly and is an edge point. The small gradient value indicates that the point is not an edge point. The gradient direction θ indicates the direction of the edge. The calculation formula of the intensity gradients S and direction θ is described as follows:

$$S = \sqrt{{d_x}^2 + {d_y}^2}, \quad (4)$$

$$\theta = \arctan \frac{d_y}{d_x}. \quad (5)$$

Subsequently, the system performs the nonmaximum suppression operation for each pixel to filter out nonedge pixels. First of all, the gradient direction θ is approximated as one of 0, 45, 90, 135, 180, 225, 270, and 315. That is, the gradient direction θ is defined as eight directions in a two-dimensional space. And then, it compares the intensity gradients S of each pixel. Finally, the pixel would be retained if the intensity gradients S of the pixel is the largest; otherwise, it is suppressed to 0. The purpose of this process is to make the blurred boundary become sharp. There are still many image noises in the image after the process of nonmaximum suppression. This method is more sensitive to noise, so it is necessary to filter for image blurring and denoising. Thereafter, the hysteresis threshold will be used to further process the noise. The method sets the upper bound and the lower bound of the threshold. It is considered to be an edge if the intensity gradients of the pixel are greater than the upper bound of the threshold, which is called a strong edge. It must not be an edge if its intensity gradients are less than the lower bound of the threshold, which will be removed. When the intensity gradients of the pixel are in threshold interval, it is considered as the weak edge. At this time, these pixels can only be considered as the candidate of the edge. They will be retained if it is connected to the edge; otherwise, it will be removed. The upper bound of the threshold is to distinguish the contour of the object from the environment, which determines the contrast between the object and the environment. The lower bound of the threshold is used to smooth the contour of the edge. The contour of the edge may be discontinuous or not smooth enough when the upper bound of the threshold is set too large. The detected edges of the contour may not be closed at this time. The lower bound of the threshold can make up for this; it can smooth the contour or connect the discontinuous parts.

In this way, a complete outline can be obtained, as illustrated in Figure 2(a). When the edge detection has been completed, the Hough gradient method is used to detect the center of the hole. This method will draw straight lines along the gradient direction of the pixels for all edge pixels. The straight line is perpendicular to the tangent line of the boundary pixel, which is the normal line.

The system will accumulate votes in the Hough two-dimensional accumulator space after the normal line of all contour pixels is drawn. The pixel with more votes is more likely to be the center of the hole. The robot gradually moves the peg to the inside from the outside of the hole after determining the center of the hole, as shown in Figure 2(b). However, during the peg approaches the center of the hole, it will slide down to the center of the hole under the action of the assembly force if the peg is close enough to the center of the hole. Subsequently, the peg will convert from plane contact to two-point contact or three-point contact. At this time, the work tasks of the hole-searching stage have been completed and the alignment stage has been entered.

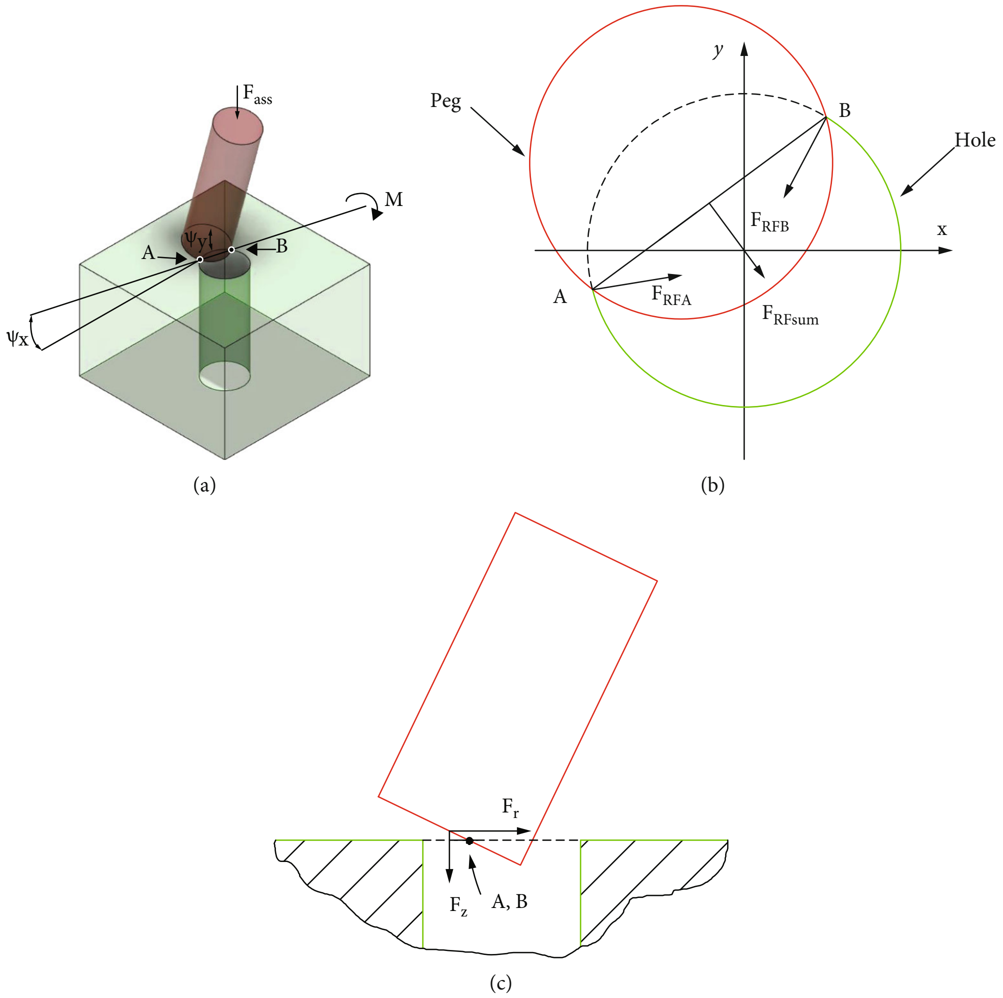


Figure 3: Schematic diagram of contact force analysis illustrating (a) contact force. (b) Top view of contact force analysis. (c) Side view of contact force analysis.

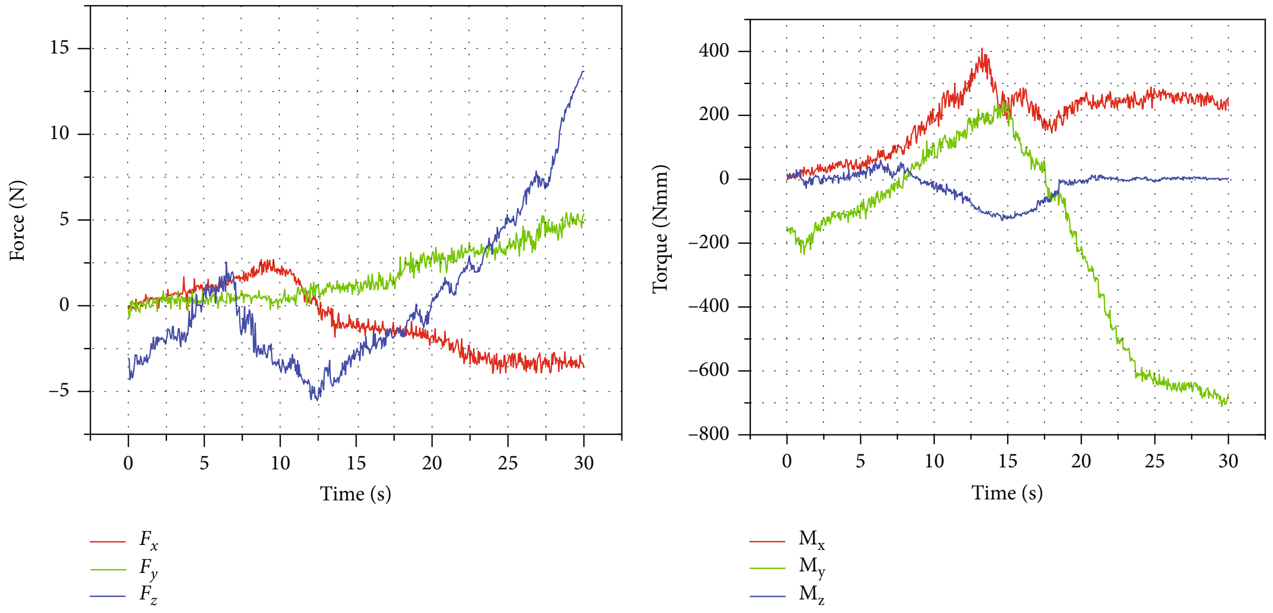
2.3. Working Principles of Alignment. The adjusting posture for the peg usually uses the method of compliance-based with force feedback to align the hole whether the assembly control method is force-based or hybrid control based on vision and force. When the point contact occurs, the peg will overcome the contact friction force between the peg and the hole under the action of the assembly force F_{ass} and slide to the center of the hole, as shown in Figure 3(a).

This phenomenon of sliding to the center of the hole is called the "natural attraction" of compliance-based peg-in-hole assembly. For instance, the assembly force exerted by the robot on the peg causes a corresponding reaction force at the contact point A and B between the peg and the hole. The sum of the reaction forces F_{RFsum} on the contact points always points to the center of the hole, as illustrated in Figure 3(b). The projections of the assembly force F_{ass} on the xy-plane and the z-axis are F_r and F_z, respectively, as shown in Figure 3(c). F_z is always vertically downward, but the direction of F_r is uncertain. They will counteract each other when the directions of F_{RFsum} and F_r are inconsistent. In this case, the peg cannot overcome the friction at the contact point and will keep the peg stationary. When the direction of F_{RFsum} and F_r are consistent, the peg will overcome the friction at the contact point to slide to the center of the hole.

This adjustment method based on compliant control can smoothly complete the peg-in-hole assembly. However, it will also have some difficult situations, such as the peg slipping out of the hole and larger position errors or posture errors. Humans often rely on the cooperation of vision and tactile to deal with this dilemma. Therefore, this research improves the work efficiency of the peg-in-hole assembly by training the vision and tactile cooperation of the robot. The training details will be introduced in Section 3.

3. Alignment Method of Combined Perception for Peg-in-Hole

The assembly system for peg-in-hole is mainly composed of hole-searching module and alignment module. The performance of the alignment module determines the alignment efficiency. The current research usually uses alignment



(a)

(b)

Figure 4: Interaction between peg and hole based on (a) contact forces and (b) contact torques.

methods based on force control. This method performs well when dealing with smaller position and posture errors, but it performs poorly when dealing with larger position and posture errors. This is because the force-based control method can only perceive the change of the contact state but cannot intuitively perceive the change of the spatial position and posture of the peg. Therefore, we propose a multiperception alignment method with vision and tactility based on the analysis in Section 2.3.

3.1. Working Principles of Combined Perception with Deep Reinforcement Learning. Tactile perception with a force/torque sensor can accurately perceive the information of the contact state, but it is not sensitive to changes in spatial position and posture. Visual perception can intuitively reflect the change of spatial position and posture. However, when the perceived object is in contact with other objects, visual perception cannot accurately perceive the contact state. If visual perception and tactile perception can be combined, the intelligence of the robot will be further improved. In this work, the robot perceives the relative position and posture of the peg and the hole through the visual sensor to make adjustment action decisions. Then, the robot perceives the information of the contact state through the force/torque sensor, and the information of contact force and torque is shown in Figure 4.

Afterward, the robot gives the adjustment action based on tactile information as a prediction label of the current state [29]. Subsequently, if the predicted action is inconsistent with the label, the backpropagation calculation is performed on the neural network to modify the weight [30]. Finally, the robot can establish a mapping relationship between visual perception and tactile perception after training, so that the robot is sensitive to changes in position, posture, and contact force. The training process is shown in Figure 5.

The proposed method enables the robot to learn the alignment skills for peg-in-hole assembly through training based on self-supervised deep reinforcement learning. Thus, the decision-making problem of the alignment adjustment process is transformed into a probabilistic problem of the Markov decision processes. At the time t, the robot chooses action a_t according to the observed environment state s_t. The environment state s_t transitions to s_{t+1}, which has obtained the reward $R_{t+1} = r$. The transition probability of the state can be expressed as follows:

$$p(s_{t+1} \mid s_t, a_t) \doteq P_r\{s_{t+1} \mid s_t, a_t \} = \sum_{r \epsilon R} p(s_{t+1}, R_{t+1} \mid s_t, a_t). \quad (6)$$

The state-action-reward chain is saved as a sample D_i:

$$D_i = (s_t, a_t, s_{t+1}, R_{t+1}). \quad (7)$$

The agent uses the strategy $\pi(s)$ to choose executable actions a_t from action space $A(s)$. The process of training robots to learn skills can also be seen as maximizing the reward of the agent. The agent also obtains the optimal strategy $\pi^*(s)$ when the total reward G_t is maximized.

$$G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}, \quad (8)$$

where $\gamma = 0.5$ is the future discount factor.

The proposed method uses off-policy Q-learning, and its action-value function is to evaluate the expected value Q for the action in the current state:

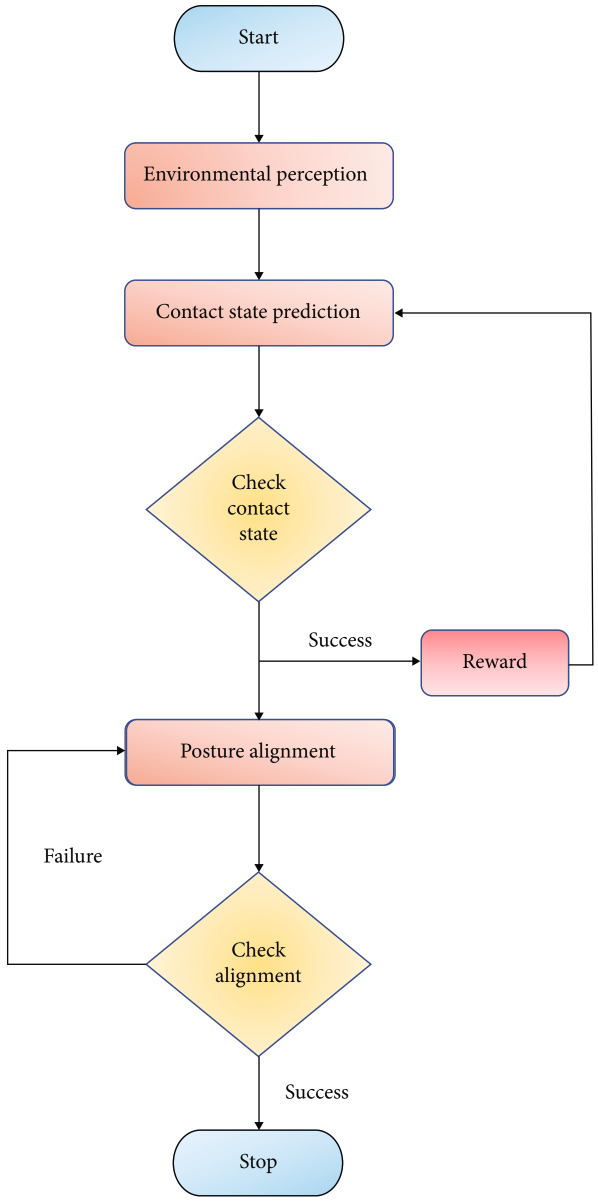


Figure 5: Flowchart of peg-in-hole procedure.

$$Q_\pi(s_t, a_t) \doteq \mathbb{E}_\pi[G_t \mid s_t, a_t] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \mid s_t, a_t\right]. \tag{9}$$

This greedy strategy will select the optimal action ${a^*}_t$ with the highest Q value; the agent obtains the optimal policy $\pi^*(s_t) = {a^*}_t = \underset{a \in A(s)}{\operatorname{argmax}} Q_{\pi^*}(s_t, a_t)$ and the optimal action-value function $Q_{\pi^*}(s_t, a_t)$ after the completion of training:

$$\begin{aligned} Q_{\pi^*}(s_t, a_t) &= \mathbb{E}_\pi\left[R_{t+1} + \gamma \max_{a_{t+1}} Q_{\pi^*}(s_{t+1}, a_{t+1}) \mid s_t, a_t\right] \\ &= \sum_{r \epsilon R} p(s_{t+1}, R_{t+1} \mid s_t, a_t)\left[R_{t+1} + \gamma \max_{a_{t+1}} Q_{\pi^*}(s_{t+1}, a_{t+1})\right]. \end{aligned} \tag{10}$$

3.2. Neural Network Architecture.

3.2. Neural Network Architecture. The alignment module builds the neural network based on deep Q-networks by modeling Q-function. It has two convolutional neural networks with the same structure, namely, the target network and the evaluation network. The agent observes the environment to obtain RGB-D images as the input of the neural network. Initially, the RGB-D image is processed by the convolutional layer with the 5×5 convolution kernel and then performed batch normalization. Whereafter, it uses the ReLU activation function for nonlinear activation. Subsequently, max-pooling is used to reduce the deviation of the estimated mean value caused by the parameter error of the convolutional layer. The unit composed of convolutional layer, batch norm, ReLU, and max-pooling layer is defined as a convolution unit.

The network has six convolutional units, followed by three linear layers interleaved with two ReLU activation layers. Firstly, the target network outputs the adjustment action a_t of the current state with softmax after inputting the RGB-D image. Then, the evaluation network evaluates the output of the target network. Afterward, the state s_t transitions to s_{t+1} after performing a_t the action, and the reward value $R(s_t, a_t)$ is obtained. The evaluation network conducts the backpropagation calculation according to the reward $R(s_t, a_t) = r$ to update the parameters θ_i of the evaluation network:

$$\begin{aligned} \theta_{i+1} = \theta_i + \alpha\left[r + \gamma \max_{a_{t+1}} Q_\pi(s_{t+1}, a_{t+1}; \theta_i) - Q_\pi(s_t, a_t; \theta_i)\right] \\ \cdot \nabla Q_\pi(s_t, a_t; \theta_i), \end{aligned} \tag{11}$$

where the learning rate α is set as 10^{-4}.

The parameters θ_i of the evaluation network are updated in real-time; however, the parameters θ_i^- of the target network are fixed during a batch of iterative training. The target network does not conduct backpropagation calculations. The parameters θ_i^- of the target network are updated by copying parameters θ_i from the evaluation network after a batch of iterative training, that is, $\theta_i^- \longleftarrow \theta_i$. The predicted difference $\Delta Q = |{Q_E}^{\theta_i} - {Q_T}^{\theta_i^-}|$ gradually shrinks between the predicted value Q_T of the target network and the predicted value Q_E of the evaluation network as the number of iterative training increases. The Huber loss function $\mathscr{L}_i$ used for training is described as follows:

$$\mathscr{L}_i = \begin{cases} \dfrac{1}{2} \times \left({Q_E}^{\theta_i} - {Q_T}^{\theta_i^-}\right)^2, \text{for} \Delta Q = \left|{Q_E}^{\theta_i} - {Q_T}^{\theta_i^-}\right| < 1, \\ \left|\left({Q_E}^{\theta_i} - {Q_T}^{\theta_i^-}\right) - \dfrac{1}{2}\right|, \text{otherwise.} \end{cases} \tag{12}$$

The collected continuous sample in training with self-supervised deep reinforcement learning may always be correlated. However, the correlation of the continuous sample will make the variance of the parameter update relatively large. The prioritized experience replay is used to reduce

```
Initialize replay buff $D$
Initialize evaluation network parameters $\theta_i$
Initialize target network parameters $\theta_i^- = \theta_i$
**for** episode=1, $M$**do**
  **for**$t = 1$, $T$**do**
    Obtain image $s_t$ from environment
    With probability $\varepsilon$ select a random adjustment action $a_t$
    otherwise select adjustment action $a_t = \mathrm{argmax}Q(s_t, a_t\,; \theta_i^-)$
    Execute adjustment action $a_t$ in CoppeliaSim
     Obtain image $s_{t+1}$ and reward $R_{t+1} = r_t$ from environment
     Store transition $(s_t, a_t, R_{t+1}, s_{t+1})$ in $D$
     Sample random minibatch of transitions $(s_t, a_t, R_{t+1}, s_{t+1})$ from $D$
     Set $Q_{E j} = \begin{cases} r_j & \text{, for terminal } s_{j+1} \\ r_j + \gamma max\widehat{Q}(s_{j+1}, a_{j+1}\,; \theta_j^-) & \text{, for non − terminal } s_{j+1} \end{cases}$
     Perform a gradient descent step on $({Q_E}^{\theta_i} - {Q_T}^{\theta_i^-})^2$
   **end for**
 **end for**
```

Algorithm 1: System pipeline.

sample correlation and nonstationary distribution. Therefore, the training uses experience replay memory D_i to store each transition $(s_t, a_t, s_{t+1}, R_{t+1})$. Afterward, the training samples a minibatch of transitions from the replay buffer to minimize the loss function. The pseudocode is described in Algorithm 1.

4. Simulation Results and Analyses

4.1. Alignment Strategy Training with Visual and Tactile Perception. The alignment training of peg-in-hole assembly with self-supervised deep reinforcement learning will be conducted in CoppeliaSim, as illustrated in Figure 6. The assembly system in simulation is equipped with a UR5 robotic arm and RG2 gripper. The working space fixedly places two RGB-D vision sensors. The force/torque sensor is installed between the RG2 gripper and the UR5 robotic arm. The diameter and length of the assembly peg are $\phi30$ mm and 100 mm, respectively. The assembly clearance of the peg and the hole is 0.8 mm. The simulation workstation has configured the CPU of 3.80 GHz Intel(R) Xeon(R) Gold 522, the GPU of NVIDIA GeForce RTX 3090, and the RAM of 128 GB. The software version of CoppeliaSim on the Ubuntu 16.04 operating system is v4.0 with Bullet Physics 2.83 for dynamic and inverse kinematics modules.

The alignment strategy uses trial and error training based on self-supervised deep reinforcement learning. Firstly, the agent observes the environment through visual perception and obtains an RGB-D image. Then, the agent predicts the contact state and selects adjustment actions. Afterward, the robot recognizes the contact state based on the information of tactile perception, and it gives adjustment action as a prediction label for the visual prediction. Subsequently, visual prediction performs backpropagation calculations based on the prediction label.

Finally, the agent establishes the mapping relationship between visual perception and tactile perception through the iteration of training. The agent will autonomously train 14,000 times without human intervention. The exploration strategy of the agent uses the ε-greedy strategy, in which its initial value is set to 0.5, and then gradually annealed to 0.1. The agent is more likely to select exploration actions in the early stages of training.

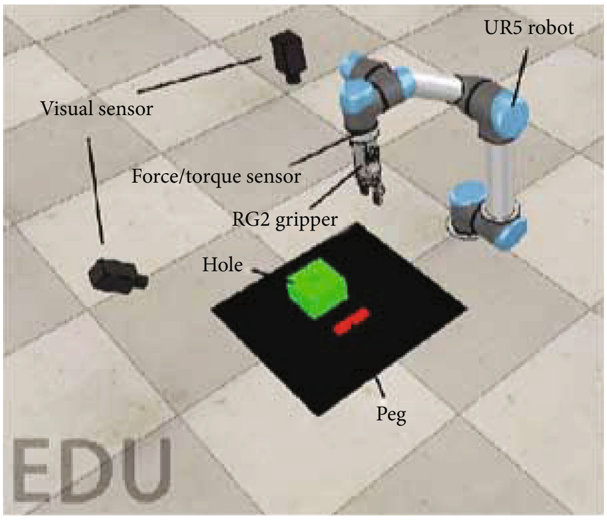


Figure 6: Schematic diagram of simulation scene.

The purpose of exploration is that this can enable the robot to collect more contact state information at the beginning of training. Afterward, the agent selects the action with the highest Q value according to the strategy $\pi(s_t)$. As shown in Figure 7, the rewards obtained by the agent gradually increase to the convergence value as the accuracy of prediction increases.

4.2. Simulation Results for Peg-in-Hole Assembly. A series of simulation tests were performed to compare the performance

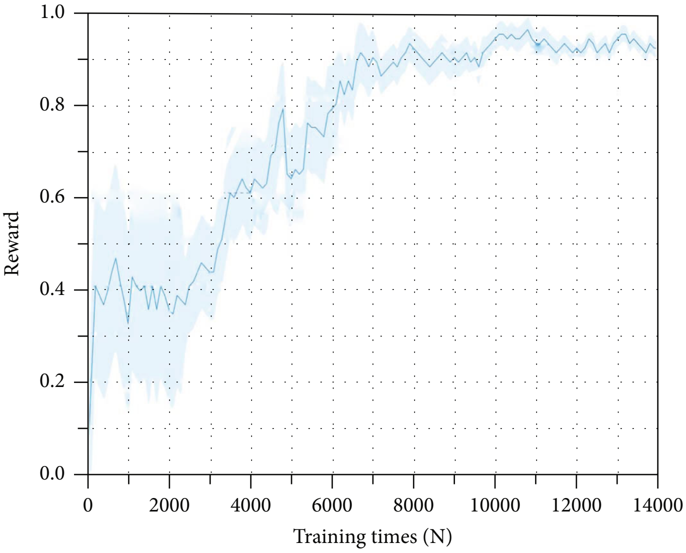


FIGURE 7: Iterative training reward for an agent.

of tactile perception, multiple perceptions in stages, and combined perceptions in peg-in-hole assembly. As analyzed in Section 2, the method of tactile perception (TP) refers to the peg-in-hole assembly using only the F/T sensor. The hole-finding stage uses visual perception, and the alignment stage uses tactile perception, and this method is called multiple perceptions in stages (MP). The proposed method in this work is called combined perceptions (CP). The robot will perform 1,000 peg-in-hole assembly tests after completing the training with self-supervised deep reinforcement learning. In addition, the robot will, respectively, use methods tactile perception and multiple perceptions in stages to perform 1,000 peg-in-hole assembly tests. The simulation test results are shown in Table 1.

The total time for peg-in-hole assembly using the method of tactile perception and multiple perceptions in stages is 38.46 hours and 34.31 hours, respectively. However, the total time of the combined perceptions is 32.15 hours. It can be seen that the method of combined perceptions takes 6.31 hours less than the method of tactile perception from the simulation results, and the assembly efficiency has improved by 16.41% compared with the method of tactile perception. Besides, the method of combined perceptions reduces 2.16 hours less than the method of multiple perceptions in stages, and the assembly efficiency has improved by 6.3% compared with the method of multiple perceptions in stages. This proves that the proposed method not only learns alignment skills but also improves assembly efficiency. Subsequently, 100 assembled samples are randomly selected for analysis and comparison, as illustrated in Figure 8.

Although the minimum assembly time and the maximum assembly time are relatively close among the three perception methods, the distribution area of the assembly time using the method of combined perceptions concentrates on a smaller time area. The total standard deviation of tactile perception (TP), multiple perceptions in stages (MP), and combined perceptions (CP) are 11.6926, 8.2279, and 5.1998, respectively. In addition, the standard error was also analyzed for the three methods, as shown in Figure 9. It can be seen that the method of the combined perceptions not only has better efficiency but also has smaller efficiency fluctuations.

TABLE 1: The total time of peg-in-hole assembly with three perception methods.

Method	Total assembly time (h)
Tactile perception (TP)	38.46
Multiple perceptions in stages (MP)	34.31
Combined perceptions (CP)	32.15

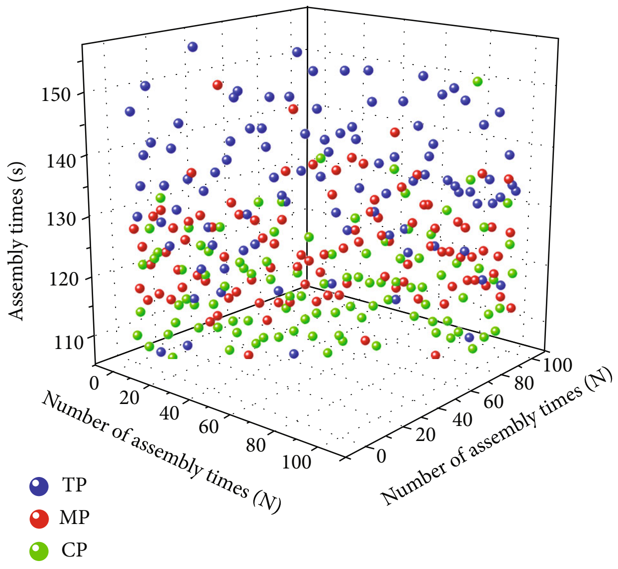


FIGURE 8: Scatter plot of assembly time.

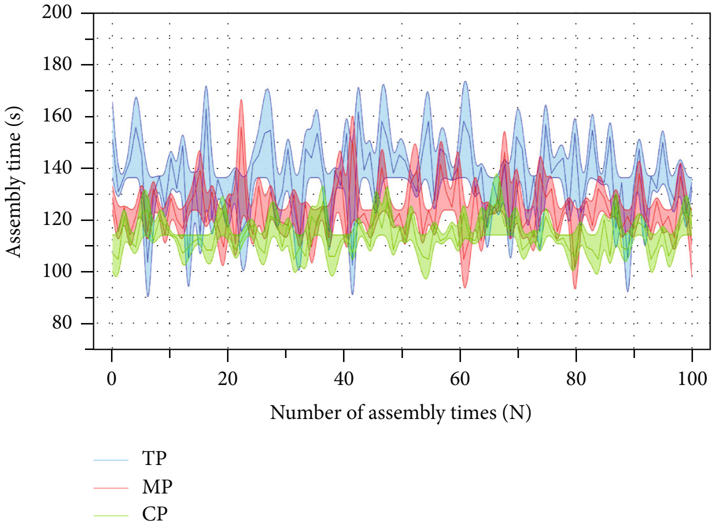


Figure 9: Comparison results of standard error for assembly time.

5. Conclusions and Future Work

In this paper, we proposed an alignment method of combined perception for peg-in-hole assembly with self-supervised deep reinforcement learning. The proposed method has combined tactile perception and visual perception to better perceive the environment information. The agent does not need human interference during the training process, which greatly reduces the difficulty and cost of data collection. In CoppeliaSim simulation, with the iterative training of the agent, visual perception and tactile perception have established a mapping relationship so that the robot can better perceive the changes of environmental information in the assembly.

From the simulation results, it can be seen that the assembly efficiency is improved after the agent learns the combined perception, and the stability of the assembly efficiency is better than the single perception method. The combined perception increases the perception ability of the robot, which will enable the robot to complete more complex tasks in an unstructured environment. In future research work, we hope to be able to apply the combined perception method to more tasks. In addition, we will still have committed to the research work about improving the efficiency of the peg-in-hole assembly.

Data Availability

The data is available at https://github.com/Bensonwyz/Alignment-Method-of-Combined-Perception.

Conflicts of Interest

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work; there is no professional or other personal interest of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

References

[1] A. Zeng, “Learning visual affordances for robotic manipulation[D],” Princeton University, 2019.

[2] Z. Tang, G. Zhao, and T. Ouyang, “Two-phase deep learning model for short-term wind direction forecasting,” *Renewable Energy*, vol. 173, pp. 1005–1016, 2021.

[3] S. Fong, W. Song, K. Cho, R. Wong, and K. Wong, “Training classifiers with shadow features for sensor-based human activity recognition,” *Sensors*, vol. 17, no. 3, p. 476, 2017.

[4] T. Li, S. Fong, K. K. L. Wong, Y. Wu, X. S. Yang, and X. Li, “Fusing wearable and remote sensing data streams by fast incremental learning with swarm decision table for human activity recognition,” *Information Fusion*, vol. 60, pp. 41–64, 2020.

[5] K. Lan, S. Fong, L. S. Liu et al., “A clustering based variable sub-window approach using particle swarm optimisation for biomedical sensor data monitoring,” *Enterprise Information Systems*, vol. 15, no. 1, pp. 15–35, 2021.

[6] F. J. Abu-Dakka, B. Nemec, A. Kramberger, A. G. Buch, N. Krüger, and A. Ude, “Solving peg-in-hole tasks by human demonstration and exception strategies,” *Industrial Robot: An International Journal*, vol. 41, no. 6, pp. 575–584, 2014.

[7] H. C. Song, Y. L. Kim, and J. B. Song, “Guidance algorithm for complex-shape peg-in-hole strategy based on geometrical information and force control,” *Advanced Robotics*, vol. 30, no. 8, pp. 552–563, 2016.

[8] 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 Robotics and Automation Letters*, vol. 1, no. 1, pp. 130–136, 2016.

[9] L. Roveda, N. Iannacci, and L. M. Tosatti, “Discrete-time formulation for optimal impact control in interaction tasks,” *Journal of Intelligent & Robotic Systems*, vol. 90, no. 3, pp. 407–417, 2018.

[10] J. Yuan, R. Guan, L. Du, and S. Ma, “A robotic gripper design and integrated solution towards tunnel boring construction equipment,” in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 2650–2655, Las Vegas, NV, USA, Oct 2020.

[11] Te Tang, H.-C. Lin, Y. Zhao, W. Chen, and M. Tomizuka, “Autonomous alignment of peg and hole by force/torque measurement for robotic assembly,” in *2016 IEEE international conference on automation science and engineering (CASE)*, pp. 162–167, Fort Worth, TX, USA, Aug 2016.

[12] Z. Liu, L. Song, Z. Hou, K. Chen, S. Liu, and J. Xu, “Screw insertion method in peg-in-hole assembly for axial friction reduction,” *IEEE Access*, vol. 7, pp. 148313–148325, 2019.

[13] T. Inoue, G. De Magistris, A. Munawar, T. Yokoya, and R. Tachibana, “Deep reinforcement learning for high precision assembly tasks,” in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 819–825, Vancouver, BC, Canada, Sept 2017.

[14] T. Ren, Y. Dong, D. Wu, and K. Chen, “Learning-based variable compliance control for robotic assembly,” *Journal of Mechanisms and Robotics*, vol. 10, no. 6, 2018.

[15] P. Zou, Q. Zhu, J. Wu, and R. Xiong, “Learning-based optimization algorithms combining force control strategies for peg-in-hole assembly,” in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS).*, pp. 7403–7410, Las Vegas, NV, USA, 24 Oct.-24 Jan. 2021.

[16] J. Xu, Z. Hou, Z. Liu, and H. Qiao, “Compare contact model-based control and contact model-free learning: a survey of robotic peg-in-hole assembly strategies,” arXiv preprint arXiv: 1904.05240, 2019.

[17] T. Jiang, H. Cui, X. Cheng, and W. Tian, “A measurement method for robot peg-in-hole prealignment based on combined two-level visual sensors,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[18] Y. Liao, W. Chen, H. Wang, and R. Wu, “A peg-in-hole assembly strategy using uncalibrated visual servoing,” in *2019 IEEE international conference on robotics and biomimetics (ROBIO),*, pp. 1845–1850, Dali, China, Dec 2019.

[19] J. C. Triyonoputro, W. Wan, and K. Harada, “Quickly inserting pegs into uncertain holes using multi-view images and deep network trained on synthetic data,” in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 5792–5799, Macau, China, Nov 2019.

[20] R. L. Haugaard, J. Langaa, C. Sloth, and A. G. Buch, “Fast robust peg-in-hole insertion with continuous visual servoing,” arXiv preprint arXiv: 2011.06399, 2020.

[21] R. J. Chang, C. Y. Lin, and P. S. Lin, “Visual-based automation of peg-in-hole microassembly process,” *Journal of Manufacturing Science and Engineering*, vol. 133, no. 4, 2011.

[22] Y. Liu, D. Romeres, D. K. Jha, and D. Nikovski, “Understanding multi-modal perception using behavioral cloning for peg-in-a-hole insertion tasks,” ar Xiv preprint ar Xiv: 2007.11646, 2020.

[23] J. Ding, C. Wang, and C. Lu, “Transferable trial-minimizing progressive peg-in-hole model,” in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 5862–5868, Macau, China, Nov 2019.

[24] A. Owens and A. A. Efros, “Audio-visual scene analysis with self-supervised multisensory features[C],” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 631–648, Munich, Germany, 2018.

[25] C. Finn and S. Levine, “Deep visual foresight for planning robot motion,” in *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2786–2793, Singapore, 29 May-3 June 2017.

[26] J. Sinapov, C. Schenck, and A. Stoytchev, “Learning relational object categories using behavioral exploration and multimodal perception,” in *2014 IEEE international conference on robotics and automation (ICRA),*, pp. 5691–5698, Hongkong, China, 31 May-7 June 2014.

[27] 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),*, pp. 418–423, Macau, Macao, Dec 2017.

[28] M. A. Lee, Y. Zhu, K. Srinivasan et al., “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)*, pp. 8943–8950, Montreal, QC, Canada, May 2019.

[29] X. Zhang, Y. Zheng, J. Ota, and Y. Huang, “Peg-in-hole assembly based on two-phase scheme and f/t sensor for dual-arm robot,” *Sensors*, vol. 17, no. 9, p. 2004, 2017.

[30] C. Olah, “Calculus on computational graphs: backpropagation,” 2015, http: //colah. github.io/posts/2015-08-Backprop/.